Journal of
*Marine Science
and Engineering*

Special Issue Reprint

Application of Artificial Intelligence in Maritime Transportation

Edited by
Xinqiang Chen, Dongfang Ma and Ryan Wen Liu

mdpi.com/journal/jmse

MDPI

Application of Artificial Intelligence in Maritime Transportation

Application of Artificial Intelligence in Maritime Transportation

Editors

Xinqiang Chen
Dongfang Ma
Ryan Wen Liu

MDPI

Basel • Beijing • Wuhan • Barcelona • Belgrade • Novi Sad • Cluj • Manchester

Editors

Xinqiang Chen Shanghai Maritime University Shanghai China	Dongfang Ma Zhejiang University Zhoushan China	Ryan Wen Liu Wuhan University of Technology Wuhan China

Editorial Office
MDPI
St. Alban-Anlage 66
4052 Basel, Switzerland

This is a reprint of articles from the Special Issue published online in the open access journal *Journal of Marine Science and Engineering* (ISSN 2077-1312) (available at: https://www.mdpi.com/journal/jmse/special_issues/H09F6V957P).

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

Lastname, A.A.; Lastname, B.B. Article Title. *Journal Name* **Year**, *Volume Number*, Page Range.

ISBN 978-3-7258-0655-3 (Hbk)
ISBN 978-3-7258-0656-0 (PDF)
doi.org/10.3390/books978-3-7258-0656-0

Contents

Journal of
Marine Science and Engineering

Editorial

Application of Artificial Intelligence in Maritime Transportation

Xinqiang Chen [1], Dongfang Ma [2] and Ryan Wen Liu [3,*]

[1] Institute of Logistics Science and Engineering, Shanghai Maritime University, Shanghai 201306, China; chenxinqiang@stu.shmtu.edu.cn
[2] Institute of Intelligent Transportation System, Zhejiang University, Hangzhou 310058, China; mdf2004@zju.edu.cn
[3] School of Navigation, Wuhan University of Technology, Wuhan 430063, China
* Correspondence: wenliu@whut.edu.cn

Maritime logistics and supply chain management have become more complicated due to economic globalization development. It is urgent to improve maritime transportation efficiency to enhance maritime logistics and supply chain management efficacy. Artificial intelligence (AI) technology has become more sophisticated via the support of varied real-world application scenarios, large-scale training samples, affordable, yet powerful, computational powers, etc. AI usage and its success in the maritime field has been significantly enhanced due to widely deployed sensors on ships, coastal buildings, etc. (i.e., the AI models can be trained with sufficient maritime data samples). In this way, the maritime and computer science communities have attempted to improve the maritime transportation efficiency by introducing varied cutting-edge AI techniques. The maritime transportation industry has anticipated more demanding AI methods for the purpose of maritime safety and environmental protection. In other words, the newly developed AI techniques can help avoid maritime traffic accidents and environ-mental pollution and improve the safety and greenness of navigation through the ship's autopilot, intelligent navigation, real-time monitoring, etc.

Artificial intelligence can perceive and predict maritime transportation situations by analyzing the ship's position, speed, and heading direction by integrating meteorology and sea current data [1,2]. These data can be used in ship automatic control and ship–port–vehicle cooperation-related activities. The ship automation control procedure can be used for autonomous driving, real-time navigation, automatic collision avoidance, automatous ship berthing, etc. Ships can be aware of potential traffic accidents with the help of computer vision and deep learning models, which can be further integrated to improve maritime traffic safety and efficiency [3]. Ship–shore–vehicle collaboration can achieve the efficient docking between ships and land vehicles using real-time data exchange and intelligent scheduling algorithms, which improves the cargo transportation efficiency and reduces congestion.

Ship trajectory optimization deserves the community's attentions by considering factors such as the ship's speed, fuel consumption, sailing time, etc. In this way, the crew can optimize their trajectory with less energy consumption, higher transportation efficiency, and on safer travelling routes [4]. The AI technique can be applied to maritime monitoring and management systems, which can realize the monitoring and early warning of maritime violations using varied maritime data sources (historical ship trajectory data, navigation data, and satellite imagery) [5]. Ship berthing and disembarking information, the port traffic condition, and the quay (yard) crane schedule can be further integrated to optimize the container terminal productivity. In this way, the cargo loading/unloading efficiency can be significantly improved, and the waiting time can be reduced as well [6]. AI techniques can also enable multi-ship collaboration to fulfill the task of cargo transportation via ship fleets. Based on the above-mentioned analysis, AI techniques can enhance the maritime transportation efficiency in a more intelligent, automatic, and environmentally friendly

Citation: Chen, X.; Ma, D.; Liu, R.W. Application of Artificial Intelligence in Maritime Transportation. *J. Mar. Sci. Eng.* **2024**, *12*, 439. https://doi.org/10.3390/jmse12030439

Received: 29 January 2024
Accepted: 6 February 2024
Published: 1 March 2024

manner. The primary goal of the Special Issue is to explore typical AI applications and solutions in the maritime transportation field, which can be described in terms of the following aspects.

Ship trajectory planning and optimization using automatic identification system (AIS) data has attracted attention in the maritime field. Sedaghat et al. (Contribution 1) proposed a novel system to monitor maritime traffic, which can be further used to predict ship positions in a real-time manner. Zhao et al. (Contribution 2) proposed an encoder–decoder-based deep learning model to predict long-range ship trajectories. Lee et al. (Contribution 3) introduced a Dijkstra-based model to efficiently find the shortest ship travelling path using country-wide AIS data. Zheng et al. (Contribution 4) identified spoofing ship trajectories using large-scale AIS data with an isolation forest based framework. Zhen et al. (Contribution 5) analyzed the varied influence of environment factors on maritime traffic safety, which included the current, water depth, traffic volume, etc. A-star related conventional machine learning models were introduced to optimize ship trajectory planning and optimization tasks (Contributions 6 and 7). Li et al. (Contribution 8) tried to explore spatial–temporal relationships between ports via the help of a graph neural network. More specifically, the proposed framework employed a graph attention network to identify the traffic patterns, which aimed to determine the potential, yet intrinsic, relationship between two neighboring feature dimensions. Chen et al. (Contribution 9) explored the ship maneuvering performance in polar waters by considering both the static and kinematic ship information. Arbabkhah et al. (Contribution 10) employed a traditional XGBoost model to predict the time of arrival to enhance the port operation productivity.

Ship detection, tracking, and identification using maritime surveillance images has also become a hot topic in the community to fulfill ship visual navigation and intelligent navigation needs. Chen et al. (Contribution 11) employed a contextual encoder to enhance the maritime image restoration performance, and a weighted bidirectional feature pyramid network was further proposed to accurately detect ships in rain and fog-interference video clips. Zhou et al. (Contribution 12) developed a novel multiple feature fusion-guided deep learning model to enhance the resolutions of maritime images captured under adverse weather conditions. The maritime community attempts to obtain ship kinematic information (i.e., speed and distance) using varied visual sensory data sources. Zhao et al. (Contribution 13) proposed a novel you only look once (YOLO)-based ship speed extraction model under hazy weather situations. The framework employed a lightweight convolutional neural network to suppress the haze interference from maritime images, and the YOLO V5 model was introduced to detect ships in the haze-free image sequences. The ship speeds were further exploited by mapping ship imaging displacement in the real world. In addition, Zhao et al. (Contribution 14) proposed an improved U-Net pixel segmentation model to identify the shoreline in a pixel-wise manner. Ye et al. (Contribution 15) proposed an enhanced attention mechanism based a YOLO model to implement a ship detection task in real time.

Attention has also been given to ship fleet management optimization, ship–port cooperation, ship energy consumption reduction, autonomous port management, etc. Cheng et al. (Contribution 16) developed a novel cooperative unmanned surface vehicle (USV) unmanned autonomous vehicle system for the purpose of enhancing the USV perception capability in an underwater environment. Chen et al. (Contribution 17) proposed an ensemble farmwork to simulate ship autonomous berthing and controlling with a linear quadratic regulator and a covariance matrix adaptation evolution module. The study aimed to tackle ship autonomous berthing challenges, which involved with ship route planning, speed controlling, etc. Yan et al. (Contribution 18) developed a novel framework to analyze maritime traffic safety in wind farm water areas using a complex network theory. Li et al. (Contribution 19) proposed a magnetic focusing-related model to quantify the ship main engine crankshaft angle using inductive angular displacement sensory data. Bai et al. (Contribution 20) proposed a ship-controlling algorithm by integrating a composite sliding mode observer and a modified feed-forward phase-locked loop. Yang et al.

(Contribution 21) tried to optimize a double-cantilevered rail crane schedule in a U-shaped automated container terminal.

Maritime transportation will emit low levels of carbon in the future, and artificial intelligence techniques will play an increasingly important role in the smart maritime shipping era. The Special Issue aimed to enhance the maritime transportation efficiency via artificial intelligence techniques, while typical maritime traffic situations were exploited. Overall, the AIS data are commonly used for intelligent navigation, and must attention is paid to suppressing the AIS data outliers, optimizing the ship travelling trajectories, ship speed control, etc. In addition, intelligent maritime traffic situation awareness was also exploited via the support of maritime monitoring videos. Ships' trajectories and speeds were accurately estimated from the maritime videos via cutting-edge computer vision models. Moreover, maritime traffic efficiency and safety were further investigated via the help of varied maritime data sources and AI techniques.

Funding: This work was jointly supported by National Natural Science Foundation of China (52331012, 52102397, 52071200, and 52201401).

Conflicts of Interest: The authors declare no conflicts of interest.

List of Contributions

1. Sedaghat, A.; Arbabkhah, H.; Jafari Kang, M.; Hamidi, M. Deep Learning Applications in Vessel Dead Reckoning to Deal with Missing Automatic Identification System Data. *J. Mar. Sci. Eng.* **2024**, *12*, 152. https://doi.org/10.3390/jmse12010152.
2. Zhao, L.; Zuo, Y.; Li, T.; Chen, C.L.P. Application of an Encoder–Decoder Model with Attention Mechanism for Trajectory Prediction Based on AIS Data: Case Studies from the Yangtze River of China and the Eastern Coast of the U.S. *J. Mar. Sci. Eng.* **2023**, *11*, 1530. https://doi.org/10.3390/jmse11081530.
3. Lee, J.-S.; Kim, T.-H.; Park, Y.-G. Maritime Transport Network in Korea: Spatial-Temporal Density and Path Planning. *J. Mar. Sci. Eng.* **2023**, *11*, 2364. https://doi.org/10.3390/jmse11122364.
4. Zheng, H.; Hu, Q.; Yang, C.; Mei, Q.; Wang, P.; Li, K. Identification of Spoofing Ships from Automatic Identification System Data via Trajectory Segmentation and Isolation Forest. *J. Mar. Sci. Eng.* **2023**, *11*, 1516. https://doi.org/10.3390/jmse11081516.
5. Zhen, R.; Gu, Q.; Shi, Z.; Suo, Y. An Improved A-Star Ship Path-Planning Algorithm Considering Current, Water Depth, and Traffic Separation Rules. *J. Mar. Sci. Eng.* **2023**, *11*, 1439. https://doi.org/10.3390/jmse11071439.
6. Hu, S.; Tian, S.; Zhao, J.; Shen, R. Path Planning of an Unmanned Surface Vessel Based on the Improved A-Star and Dynamic Window Method. *J. Mar. Sci. Eng.* **2023**, *11*, 1060. https://doi.org/10.3390/jmse11051060.
7. Ma, Y.; Li, B.; Huang, W.; Fan, Q. An Improved NSGA-II Based on Multi-Task Optimization for Multi-UAV Maritime Search and Rescue under Severe Weather. *J. Mar. Sci. Eng.* **2023**, *11*, 781. https://doi.org/10.3390/jmse11040781.
8. Li, Y.; Li, Z.; Mei, Q.; Wang, P.; Hu, W.; Wang, Z.; Xie, W.; Yang, Y.; Chen, Y. Research on Multi-Port Ship Traffic Prediction Method Based on Spatiotemporal Graph Neural Networks. *J. Mar. Sci. Eng.* **2023**, *11*, 1379. https://doi.org/10.3390/jmse11071379.
9. Chen, L.; Huang, C.; Wang, Y. A Study on the Correlation between Ship Movement Characteristics and Ice Conditions in Polar Waters. *J. Mar. Sci. Eng.* **2023**, *11*, 729. https://doi.org/10.3390/jmse11040729.
10. Arbabkhah, H.; Sedaghat, A.; Jafari Kang, M.; Hamidi, M. Automatic Identification System-Based Prediction of Tanker and Cargo Estimated Time of Arrival in Narrow Waterways. *J. Mar. Sci. Eng.* **2024**, *12*, 215. https://doi.org/10.3390/jmse12020215.
11. Chen, X.; Wei, C.; Xin, Z.; Zhao, J.; Xian, J. Ship Detection under Low-Visibility Weather Interference via an Ensemble Generative Adversarial Network. *J. Mar. Sci. Eng.* **2023**, *11*, 2065. https://doi.org/10.3390/jmse11112065.
12. Zhou, W.; Li, B.; Luo, G. Multi-Feature Fusion-Guided Low-Visibility Image Enhancement for Maritime Surveillance. *J. Mar. Sci. Eng.* **2023**, *11*, 1625. https://doi.org/10.3390/jmse11081625.

13. Zhou, Z.; Zhao, J.; Chen, X.; Chen, Y. A Ship Tracking and Speed Extraction Framework in Hazy Weather Based on Deep Learning. *J. Mar. Sci. Eng.* **2023**, *11*, 1353. https://doi.org/10.3390/jmse11071353.
14. Zhao, J.; Song, F.; Gong, G.; Wang, S. Improved UNet-Based Shoreline Detection Method in Real Time for Unmanned Surface Vehicle. *J. Mar. Sci. Eng.* **2023**, *11*, 1049. https://doi.org/10.3390/jmse11051049.
15. Ye, Y.; Zhen, R.; Shao, Z.; Pan, J.; Lin, Y. A Novel Intelligent Ship Detection Method Based on Attention Mechanism Feature Enhancement. *J. Mar. Sci. Eng.* **2023**, *11*, 625. https://doi.org/10.3390/jmse11030625.
16. Cheng, C.; Liu, D.; Du, J.-H.; Li, Y.-Z. Research on Visual Perception for Coordinated Air–Sea through a Cooperative USV-UAV System. *J. Mar. Sci. Eng.* **2023**, *11*, 1978. https://doi.org/10.3390/jmse11101978.
17. Chen, G.; Yin, J.; Yang, S. Ship Autonomous Berthing Simulation Based on Covariance Matrix Adaptation Evolution Strategy. *J. Mar. Sci. Eng.* **2023**, *11*, 1400. https://doi.org/10.3390/jmse11071400.
18. Yan, K.; Wang, Y.; Wang, W.; Qiao, C.; Chen, B.; Jia, L. A System-Theory and Complex Network-Fused Approach to Analyze Vessel–Wind Turbine Allisions in Offshore Wind Farm Waters. *J. Mar. Sci. Eng.* **2023**, *11*, 1306. https://doi.org/10.3390/jmse11071306.
19. Li, Z.; Wang, B.; Wang, X.; Zhang, C.; Meng, X. Modelling, Linearity Analysis and Optimization of an Inductive Angular Displacement Sensor Based on Magnetic Focusing in Ships. *J. Mar. Sci. Eng.* **2023**, *11*, 1028. https://doi.org/10.3390/jmse11051028.
20. Bai, H.; Yu, B.; Gu, W. Research on Position Sensorless Control of RDT Motor Based on Improved SMO with Continuous Hyperbolic Tangent Function and Improved Feedforward PLL. *J. Mar. Sci. Eng.* **2023**, *11*, 642. https://doi.org/10.3390/jmse11030642.
21. Yang, Y.; Sun, S.; Zhong, M.; Feng, J.; Wen, F.; Song, H. A Refined Collaborative Scheduling Method for Multi-Equipment at U-Shaped Automated Container Terminals Based on Rail Crane Process Optimization. *J. Mar. Sci. Eng.* **2023**, *11*, 605. https://doi.org/10.3390/jmse11030605.

References

1. Hua, R.; Yin, J.; Wang, S.; Han, Y.; Wang, X. Speed optimization for maximizing the ship's economic benefits considering the Carbon Intensity Indicator (CII). *Ocean Eng.* **2024**, *293*, 116712. [CrossRef]
2. Chen, X.; Wang, M.; Ling, J.; Wu, H.; Wu, B.; Li, C. Ship imaging trajectory extraction via an aggregated you only look once (YOLO) model. *Eng. Appl. Artif. Intell.* **2024**, *130*, 107742. [CrossRef]
3. Moreno, F.C.; Gonzalez, J.R.; Muro, J.S.; Maza, J.A.G. Relationship between human factors and a safe performance of vessel traffic service operators: A systematic qualitative-based review in maritime safety. *Saf. Sci.* **2022**, *155*, 105892. [CrossRef]
4. Karatuğ, Ç.; Tadros, M.; Ventura, M.; Soares, C.G. Decision support system for ship energy efficiency management based on an optimization model. *Energy* **2024**, *292*, 130318. [CrossRef]
5. Soner, O.; Kayisoglu, G.; Bolat, P.; Tam, K. Risk sensitivity analysis of AIS cyber security through maritime cyber regulatory frameworks. *Appl. Ocean Res.* **2024**, *142*, 103855. [CrossRef]
6. Khan, R.U.; Yin, J.; Mustafa, F.S.; Shi, W. Factor assessment of hazardous cargo ship berthing accidents using an ordered logit regression model. *Ocean Eng.* **2023**, *284*, 115211. [CrossRef]

Journal of
*Marine Science
and Engineering*

Article

A Refined Collaborative Scheduling Method for Multi-Equipment at U-Shaped Automated Container Terminals Based on Rail Crane Process Optimization

Yongsheng Yang [1,*], Shu Sun [1], Meisu Zhong [2], Junkai Feng [1], Furong Wen [3] and Haitao Song [3]

[1] Institute of Logistics Science and Engineering, Shanghai Maritime University, Shanghai 201306, China
[2] College of Transport & Communications, Shanghai Maritime University, Shanghai 201306, China
[3] Beibu Gulf Port Co., Ltd., Nanning 530000, China
* Correspondence: yangys_smu@126.com

Abstract: A U-shaped automated container terminal (ACT) has been proposed for the first time globally and has been adopted to construct the Beibu Gulf Port ACT. In this ACT layout, the double cantilevered rail crane (DCRC) simultaneously provides loading and unloading services for the external container trucks (ECTs) and the automatic guided vehicles (AGVs) entering the yard. The DCRC has a complex scheduling coupling relationship with the AGV and the ECT, and its mathematical model is extremely complex. There is an urgent need to study a practical collaborative scheduling optimization model and algorithm for the DCRC, the AGV, and the ECT. In this paper, we optimize the process flow of DCRCs to study the refined collaborative scheduling model of DCRCs, AGVs and ECTs in U-shaped ACTs. Firstly, we analyze the operation process of the DCRC and divide the 16 loading and unloading conditions of the DCRC into four operation modes for process optimization. Secondly, different variables and parameters are set for the DCRC's four operating modes, and a refined collaborative dispatching model for the DCRCs with AGVs and ECTs is proposed. Finally, a practical adaptive co-evolutionary genetic algorithm solves the model. Meanwhile, arithmetic examples verify the correctness and practicality of the model and algorithm. The experimental results show that the total running time of the DCRCs is the shortest in the U-shaped ACT when the number of quay cranes (QC) to DCRC and AGV ratios are 1:2 and 1:10, respectively. At the same time, the number of QCs and DCRCs has a more significant impact on the efficiency of the ACT than that of AGVs, and priority should be given to the allocation of QCs and DCRCs. The research results have essential guidance value for U-shaped ACTs under construction and enrich the theory and method of collaborative scheduling of U-shaped ACT equipment.

Keywords: U-shaped automated container terminal; double cantilever rail crane; refined collaborative scheduling; equipment ratio

Citation: Yang, Y.; Sun, S.; Zhong, M.; Feng, J.; Wen, F.; Song, H. A Refined Collaborative Scheduling Method for Multi-Equipment at U-Shaped Automated Container Terminals Based on Rail Crane Process Optimization. *J. Mar. Sci. Eng.* **2023**, *11*, 605. https://doi.org/10.3390/jmse11030605

Academic Editor: Luca Cavallaro

Received: 15 February 2023
Revised: 3 March 2023
Accepted: 8 March 2023
Published: 13 March 2023

1. Introduction

With their high efficiency, safety, and low dependence on manual labor, automated container terminals (ACTs) have become an inevitable trend in transforming the world's ports [1]. All the top 10 container terminals reported by Alphaliner in 2021 have ACT in operation or under construction [2]. In the construction of ACTs, a U-shaped ACT is proposed for the first time in the world and adopted in the construction of the ACT at Beibu Gulf Port, attracting widespread attention from academia and industry.

Regarding management, strategic, tactical and operational are three different levels of decision-making that affect a system's operational efficiency. To improve the efficiency of a terminal, the layout and handling technology are key issues that should be considered at the strategic level, influencing all other decisions [3]. As the terminal's second-largest source of carbon emissions, the layout design of the terminal should focus on the yard area [4], which points us in the direction of studying U-shaped ACTs. Currently, there are two types of yard layouts in ACTs: one is a layout where the blocks are parallel to the quay (hereafter called a parallel layout), and the other is a layout where the blocks are perpendicular to the quay (hereafter called a perpendicular layout). The loading and unloading scheme is divided into operating at the end of the block and the side of the block [5].

Loading and unloading in a perpendicular layout at the end of the block is more common in ACTs because of their relatively simple traffic control logic. When looking at the evolution of ACTs in the United States and abroad, traditional ACTs have used a perpendicular layout with end interactions. The representative real-world examples include the Euromax terminal in Rotterdam, the Yangshan Terminal in Shanghai, and the Altenwerder terminal in Hamburg. Some studies have suggested that the perpendicular layout requires fewer automatic guided vehicles (AGVs) than the parallel layout for the same single-side loading and unloading solution inside the blocks [6]. However, the perpendicular layout with an end loading and unloading scheme is less efficient because the yard crane (YC) travels longer distances to the end of the block for each task [7]. In general, there is no absolute advantage of the yard layout, and each layout has to find its own suitable loading and unloading solution [8]. Some scholars have noted the necessity of researching new layouts and loading and unloading schemes for yards [9].

Typically, neither the AGV nor the external container truck (ECT) enter the yard in established ACTs with the perpendicular layout and end interactions. The AGV is decoupled from the YC at the seaside end by AGV partners, and the ECT interacts with the YC at fixed positions at the landside end. According to the needs of the construction of ACTs with extended depths and to overcome the shortcomings of the perpendicular layout terminal with end interaction, Zhenhua Company has proposed a U-shaped ACT scheme. Beibu Gulf Port uses the scheme in the construction of its ACT [10].

However, the new scheme and equipment have created new problems. Firstly, the double cantilevered rail cranes (DCRC) provide loading and unloading services for both the ECT and the AGV. In scheduling models, there is a coupling relationship between the DCRC, the AGV and the ECT which makes the mathematical model more complex. There has yet to be a practical scheduling optimization model. Secondly, there is no container staging point in the block, and the DCRC directly couples vehicles on both sides. This puts higher requirements on collaborative scheduling between equipment, and the model requires more constraints. Thirdly, the DCRC can be loaded and unloaded at any bay in the block. The interaction points are tens or even hundreds of times higher than the end scheme, making the model more difficult to solve [11].

Therefore, the multi-equipment collaborative scheduling problem, with the integrated consideration of loading and unloading efficiency, and energy consumption as the optimization objective, is a unique scheduling problem for U-shaped ACTs. The scientific problem of multi-equipment collaborative scheduling and the multi-objective optimization of ACTs induced by the layout scheme design of U-shaped ACTs needs to be specifically studied based on the actual demand for U-shaped ACT construction, and taking into account its yard, the core aspect of the U-shaped ACT energy consumption and cost, and the coupling and complexity of the multi-equipment collaborative scheduling mathematical model. In this paper, we first select the collaborative scheduling problem of DCRCs, AGVs and ECTs in the U-shaped ACT, with loading and unloading efficiency as the optimization objective for research.

The contributions of this study are as follows. (1) A refined collaborative scheduling model for multi-equipment in U-shaped ACTs has been established for the first time. This problem is an urgent practical problem in the construction of U-shaped ACTs and an essential academic issue in management disciplines. It is a universal problem for ACTs and a unique problem for U-shaped ACTs. However, scholars have already researched the equipment refinement scheduling model and the port multi-equipment collaborative scheduling model. However, research on multi-equipment refined collaborative dispatching models for ACTs has yet to be conducted to date. This thesis firstly analyses the motion process of the DCRC during loading and unloading at the U-shaped ACT. The 16 loading and unloading situations encountered when the DCRC operates both sides of the AGV and the ECT are grouped into four modes to optimize the loading and unloading process of the U-shaped ACT. Then, a multi-equipment refinement and collaborative scheduling model for the U-shaped ACT is established, with the shortest total running time as the target, based on the four loading and unloading modes of the DCRC. (2) The optimal equipment ratio of the U-shaped ACT is derived through simulation experiments at different scales to meet the actual production requirements.

The following is the reminder of this paper: Section 2 provides a review of the corresponding references. Section 3 analyses the motion pattern of DCRCs and proposes a refined collaborative scheduling model for multi-equipment. Section 4 designs a suitable algorithm for the model. Section 5 conducts numerical experiments and performs equipment rationing analysis. Conclusions are given in Section 6.

2. Literature Review

In this section, we look at the existing research that is relevant to this study. This research can be divided into four groups: scheduling problems for YCs, collaborative scheduling problems for YCs and AGVs, collaborative scheduling problems for multi-equipment, and collaborative scheduling problems for multi-equipment in U-shaped ACTs.

2.1. Scheduling Problems for YCs

Among container handling equipment, YCs play an essential role in the production of container terminals. Academics have carried out long-term and extensive research on various aspects. Some scholars have studied the scheduling of tire cranes between flat-banked blocks, [12–17], a situation more often found in conventional terminals. Some have studied the scheduling of multiple YCs, taking into account interferences between YCs [18–24]. Some scholars have studied the YC's dispatching rules and travel paths. Ref. [25] compared the rules of the YC serving outbound collectors, such as first-come, first-served, one-way travel, and minimum processing time rules, to reduce the waiting time of outbound collectors in the yard. The results showed robust, high-level performance under the shortest processing time rule. Ref. [26] conducted a preliminary exploration of container handling theory and proposed a YC spreader loading and unloading route that balances the length of the travel route and the safety distance. The optimal travel path of the spreader was determined while determining the scheduling rules of the YC. Ref. [27] first proposed the refined scheduling of the YC. The study divided the operation cycle of the YC into the primary motion gantry movement time, the spreader's vertical movement (lifting/lowering) time, and the trolley movement time. Moreover, it pointed out that the expected cycle time of the YC based on the Chebyshev motion pattern is shorter than that of the YC based on matrix motion. This provided a theoretical basis for subsequent research into the scheduling of YCs based on the Chebyshev motion.

2.2. Collaborative Scheduling Problems for YCs and AGVs

ACTs are a complex overall system, and studying a single subsystem in isolation does not fit the reality of ACT operations. Without collaborative scheduling, improvements in one area may be lost due to inefficiencies in another. Collaborative scheduling holds great promise for increasing terminal throughput and achieving a high utilization of yard equipment [28]. The main factor that affects how well ACTs work is how well AGVs and YCs work together [29].

Some researchers are investigating the synergistic scheduling of AGVs and YCs in a vertical layout with end operations. Ref. [30] pointed out that previous studies have usually reduced operation time by increasing the amount of operating equipment but ignoring the additional cost and energy consumption caused by increasing the number of pieces of equipment. The paper considered the matching of equipment operation time and equipment quantity. It established a collaborative scheduling model for AGVs and YCs to minimize the operating equipment's total energy consumption. Ref. [31] proposed a new method to optimize the scheduling of YCs and AGVs in the YC relay mode, considering the buffer capacity constraint and the interference of dual YC operations.

The collaborative scheduling of internal vehicles and YCs has been investigated during loading and unloading processes. For example, Ref. [32] divided individual container tasks into “unloading,” “loading,” “receiving,” and “delivery.” Ref. [29] investigated the integrated scheduling problem of YCs and AGVs as a multi-robot collaborative scheduling problem and proposed a multi-commodity network traffic model with two sets of traffic balancing constraints. Ref. [33] used four heuristic algorithms for YCs and yard trucks scheduling. The results showed that the hybrid algorithm is more advantageous. For the first time, Ref. [34] investigated internal vehicle tasks on both sides of the block of YC operations in an ACT. It described the motion of the YC loading and unloading process. It again demonstrated the advantages of a Chebyshev motion-based YC control model for loading and unloading scenarios.

2.3. Collaborative Scheduling Problems for Multi-Equipment

None of the above studies involved the collaborative scheduling of three different types of handling equipment. As a result, planning decisions may be suboptimal, and efficiency improvements may not be as significant as with integrated scheduling methods. Of the studies conducted, many multi-equipment collaborative scheduling studies have been performed by adding the study of QCs to YCs and internal vehicles. These studies have been divided into two categories: loading and unloading at the end, and loading and unloading at the side.

In the end-loading solution, Ref. [35] considered the problem of overlapping operations with multiple QCs, dividing and modeling the YC into two cases, from the end buffer and from inside the block. It was concluded that, for the vertical end layout, the optimal block parameters required for both side loading and loading are the same. Ref. [36] also considered the integrated scheduling of the three to minimize the amount of equipment used and the time taken by the equipment to complete the task.

In the side-loading solution, Ref. [37] considered the uncertainties in the speed of operation of terminal equipment and developed a collaborative scheduling model for the QC, YC, and AGV. Ref. [38] was concerned with finding a trade-off between efficiency and energy consumption among the three. Ref. [39] used a double cycle to handle containers to minimize the number of empty yard truck loads.

The impact of ECTs on the terminal is additionally considered. This article used a leave queuing model to describe the co-loading of inbound trucks and ECTs by the YC on both sides, and modeled the optimization of truck booking quotas to minimize the waiting costs for both internal and external container trucks [40].

2.4. Collaborative Scheduling Problems for Multi-Equipment in U-Shaped ACTs

As mentioned above, multi-equipment collaborative scheduling methods mainly focus on the perpendicular layout of ACTs. For U-shaped ACTs, there has yet to be a mature multi-equipment collaborative scheduling optimization method. Beibu Gulf Port, which is building the world’s first U-shaped ACT, urgently needs a mature collaborative scheduling optimization method to solve the problem of collaborative scheduling of multiple pieces of equipment in ACTs and the optimization of energy consumption and efficiency in the yard.

The U-shaped ACT uses DCRCs to simultaneously load and unload ECTs and AGVs entering the yard. In contrast to the non-cantilever automatic rail-mounted gantry commonly used in ACTs, the DCRC has two cantilevers extending on each side. The spreader can service vehicles at the operating lane along the cantilever on either side without traveling to the end of the block to interact each time a new task is performed [10].

The U-shaped scheme has many advantages compared with the end-operation scheme. To begin with, the ECT runs along the U-shaped lanes and the AGV runs along the straight lanes. This method uses physical segregation to divert internal and external vehicles, solving the practical problem of simultaneously entering the yard operation [8,11].

Then, the DCRC can interact with vehicles on both sides at each bay without moving to the end of the block for each task to hand over the container, which can significantly reduce the travel distance of the DCRC. Furthermore, installing internal roads in the yard allows the terminal to accommodate more vehicles, while making the AGVs and ECTs more drivable and allowing them to be more evenly distributed in the terminal, thus reducing vehicle congestion. Finally, ECTs can exit along the U-shaped lanes in the same direction, reducing backup time and making it easier to recruit truck drivers. Refs. [41,42] addressed the environmental protection issues and related safety concerns in the ocean. Ref. [43] was conducted to simulate the layout of ACTs from efficiency, economic, and environmental perspectives. The results showed that the U-shaped scheme outperforms the terminal with the end scheme regarding operational efficiency and waiting time under the same conditions.

Our team has carried out some preliminary research on the multi-equipment collaborative scheduling optimization problem in U-shaped ACTs. However, the models and algorithms have yet to reach practicality. Ref. [10] established a hybrid scheduling model for YCs, AGVs and ECTs in U-shaped ACTs. A scheduling model architecture with the hierarchical abstraction of scheduling objects was proposed to refine the problem. However, the study required that only one ECT or AGV can exist in a lane, with AGVs and ECTs queuing at the entrance to the block before entering the block. This is different from the multi-vehicle approach in a U-shaped ACT and reduces the flexibility of vehicle scheduling. Ref. [8] investigated the integrated scheduling of QCs, AGVs and DCRCs. They considered conflict-free path planning for AGVs and proposed an integrated scheduling optimization model based on mixed integer planning. However, the study ignored the DCRC's operation process and operation time and modeled the ACT loading and unloading process in three parts, which revealed weaknesses in the model's holistic nature and collaboration between equipment. Ref. [11] proposed a decision tree learning method based on a heuristic Monte Carlo tree search algorithm. This study was based on the U-shaped ACT YC feature allowing it to operate on both sides of the block. Coordinated scheduling with the ECT was further considered based on the QC, YC and AGV. However, the paper assumed that there are only four loading and unloading points in each block. After entering the block, AGVs cannot stop anywhere. This does not break through the limitation of fixed loading and unloading points in the end loading and unloading mode, and it is a tremendous difference from how loading and unloading actually work in U-shaped ACTs, where it can happen at any berth.

Based on a review of the above literature, the following bottlenecks remain:

- As a device that interacts directly with the yard, the impact of the ECT on the overall scheduling of the terminal is indispensable; however, there are relatively few studies that consider the ECT as a factor for multi-equipment collaborative scheduling.
- Existing research mostly looks at the movement of the YC as a whole, and there is no research on the fine-grained collaborative scheduling of multi-equipment in U-shaped ACTs.
- U-shaped ACTs are a new loading and unloading scheme using new equipment, and there is no research on the equipment ratios of U-shaped ACTs.

In response to the above problems, this paper analyzes the motion process of DCRCs in U-shaped ACTs and summarizes the 16 working conditions of DCRCs into four modes. A multi-equipment refined collaborative scheduling model is established, and the validity of the model and algorithm is verified using an adaptive collaborative genetic algorithm (ACGA). In addition, the optimum number of equipment ratios required in the loading and unloading modes is determined through simulation experiments, solving the practical challenges of U-shaped ACTs.

3. Model Establishment

3.1. Process Optimization of DCRCs

The layout of the U-shaped ACT is shown in Figure 1. In the U-shaped scheme, four AGV lanes are set between every two blocks in the yard. The middle two are overtaking lanes, and the two near the block are operation lanes. Every other block is provided with three "U"-shaped ECT lanes. The middle lane is the overtaking lane, and the two lanes near the block are operation lanes. After loading and unloading the ECT, there is no need to turn around and the truck leaves the container area directly along the U-shaped lanes. There are two exit lanes to avoid congestion.

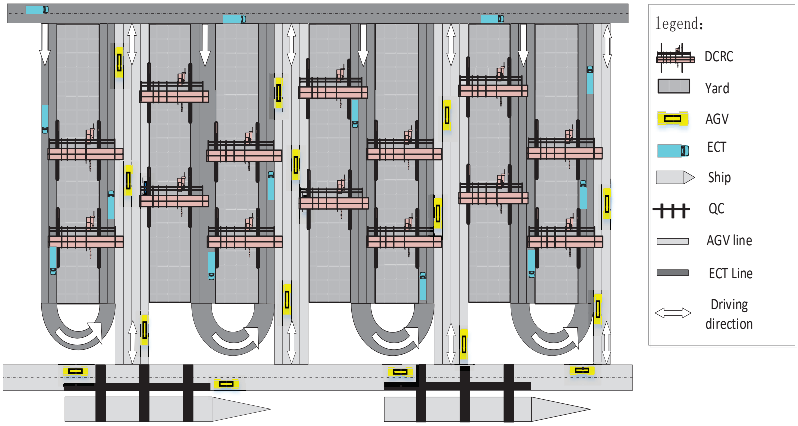


Figure 1. Layout of the U-shaped ACT.

To study the refined collaborative dispatching method of DCRCs with AGVs and ECTs based on the process optimization of DCRCs, this section is dedicated to optimizing the loading and unloading process of DCRCs to provide a basis for the refined collaborative dispatching of multiple devices.

In a U-shaped ACT, DCRCs can work on both sides at any bay with the ECT and the AGV. As the DCRC operation is continuous, different types of tasks in different states take different amounts of time to complete. This time needs to be accurately calculated for different working conditions. To maximize the advantages of U-shaped ACTs and to keep the gantry of DCRCs moving as little as possible, the operating process of the DCRC needs to be optimized. Considering the ECT approach, the DCRC has 16 working conditions in the U-shaped ACT loading and unloading process, as shown in Figure 2.

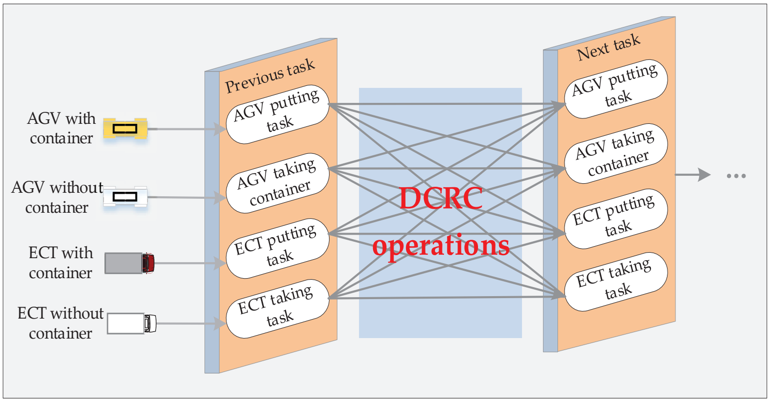


Figure 2. Schematic diagram of the working conditions of the DCRC.

To reduce the energy consumption of the DCRC movement and to allow the DCRC gantry to move as little as possible, the above 16 situations are divided into four modes: putting before putting (PP), putting before taking (PT), taking before putting (TP), and taking before taking (TT), as shown in Figure 3, of which the solid line shows the trajectory of the AGV task for the DCRC operation, and the dashed line shows the trajectory of the ECT task for the DCRC operation. The numbered circles indicate the order of movement for the DCRC gantry and its spreader.

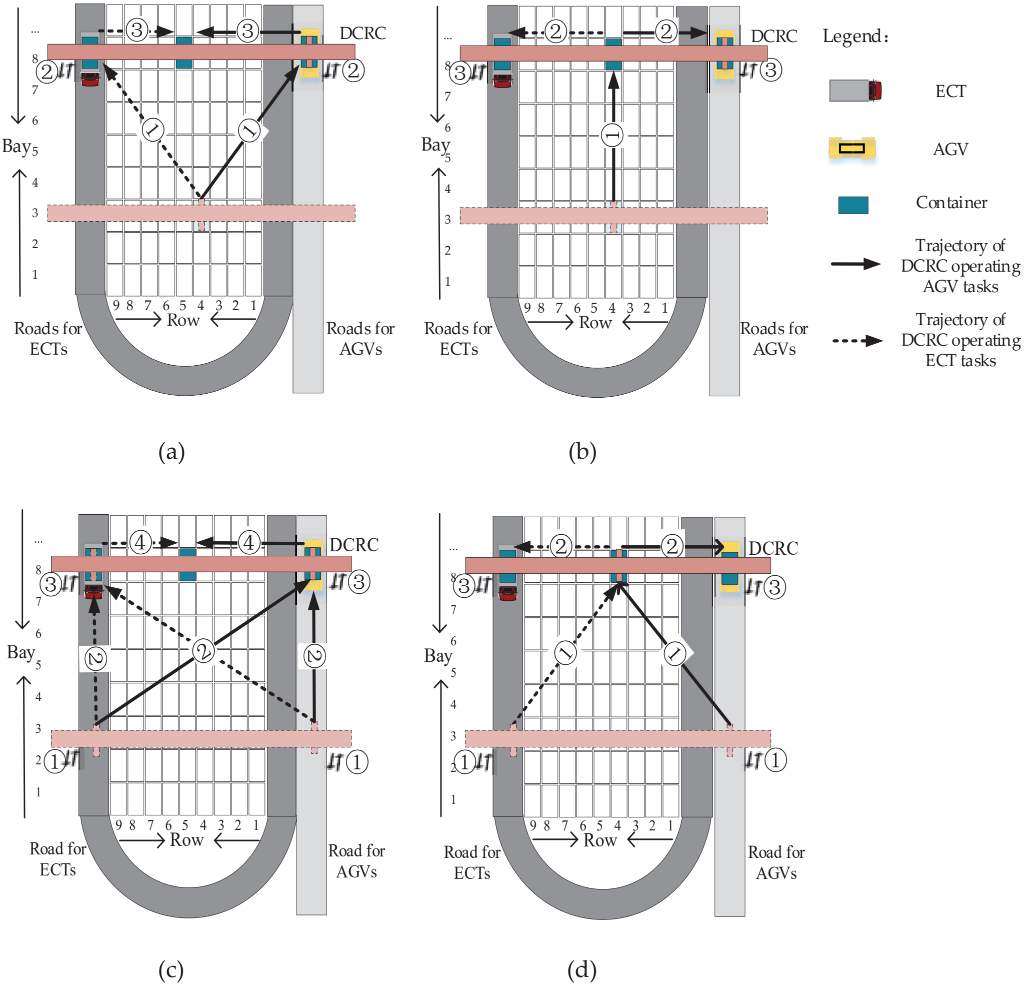


Figure 3. Operation process of the DCRC for handling different modes of tasks. (**a**) "PP" Mode; (**b**) "PT" Mode; (**c**) "TP" Mode; (**d**) "TT" Mode.

The "PP" mode means that one DCRC has worked on a putting task first, followed by the next putting task. At this time, the motion process of the DCRC is described as follows: (1) The gantry and spreader of the DCRC move from the target position of the previous putting task to the upper edge of the block of the target bay of the next task. (2) The spreader of the DCRC drops down to take the container from the AGV/ECT, completing the task for the AGV/ECT. (3) The spreader of DCRC rises to the upper edge of the block. (4) The spreader of DCRC moves horizontally to the corresponding stack position to put the container, and the DCRC putting task is completed. This mode includes four cases:

- The DCRC continuously handles two putting tasks of AGVs.
- The DCRC handles an AGV putting task after handling the putting task of an ECT.
- The DCRC handles an ECT putting task after handling the putting task of an AGV.
- The DCRC continuously handles two putting tasks of ECTs.

The “PT” mode means that one DCRC first handles a putting task and then handles a taking task. At this time, the motion process of the DCRC is described as follows: (1) The gantry and spreader of the DCRC move from the target position of the previous putting task to the target position of the next taking task. (2) The spreader takes the taking task and moves it to the upper edge of the block. (3) The DCRC’s spreader drops down to place the taking task on the AGV/ECT, and both the AGV/ECT and the DCRC are completed. This mode includes four cases:

- The DCRC handles an AGV taking task after handling the putting task of an AGV.
- The DCRC operates the taking task on the AGV after completing the putting task on the ECT.
- The DCRC operates the taking task on the ECT after completing the putting task on the AGV.
- The DCRC handles an ECT taking task after handling the putting task of an ECT.

The “TP” mode means that one DCRC first handles a taking task and then handles a putting task. At this time, the motion process of the DCRC is described as follows: (1) The DCRC’s spreader ascends from the previous taking task vehicle to the block top. (2) The DCRC’s gantry and spreader move to the upper edge of the block on the side of the next putting task. (3) The spreader of the DCRC drops down to pick up the container from the AGV/ECT, and the AGV/ECT completes its task. (4) The spreader of DCRC rises to the upper edge of the block. (5) The DCRC’s spreader traverse moves to the target stack and places the task in the corresponding bay, at which point the taking task of the DCRC is completed. This model includes four cases:

- The DCRC handles an AGV putting task after handling the taking task of an AGV.
- The DCRC operates the putting task on the AGV after completing the taking task on the ECT.
- The DCRC operates the putting on the ECT after completing the taking task on the AGV.
- The DCRC handles an ECT putting task after handling the taking task of an ECT.

The “TT” mode means that one DCRC has worked on a taking task first, followed by the next taking task. At this time, the motion process of the DCRC is described as follows: (1) The spreader of the DCRC moves up from the interaction point of the previous taking task to the upper edge of the block. (2) The gantry and spreader of the DCRC move to the target position for the next taking task. (3) The spreader picks up the taking task and moves it to the edge of the block on the taking task side. (4) The DCRC’s spreader drops down to place the taking task on the AGV/ECT, and the task of the AGV/ECT and the DCRC is completed. This model includes four cases:

- The DCRC continuously handles two taking tasks of AGVs.
- The DCRC handles an AGV taking task after handling the taking task of an ECT.
- The DCRC handles an ECT taking task after handling the taking task of an AGV.
- The DCRC continuously handles two taking tasks of ECTs.

From the above description, the sequence of operations and target container position for unloading containers, as well as the target container position and reach time for the ECT, are known. However, the DCRC’s subsequent operation and which AGV will go to which QC to operate which container are unknown. Scheduling the AGVs to do the different container tasks and the DCRCs to handle the right tasks at the right time so that the total running time of all the DCRCs is as short as possible will make the U-shaped ACT more efficient. Consequently, we propose a fine-grained collaborative scheduling model for multiple devices that considers the entry of ECTs into the yard.

3.2. Assumptions

(1) All the containers discussed are 40-foot standard containers.
(2) The arrival time of the ECT, the target position of the unloading containers, and the position of the loading containers are known.
(3) The motion of the DCRC is based on the Chebyshev's motion without considering the interference between them.
(4) The AGVs, ECTs and DCRCs can work on only one task simultaneously.
(5) The problem of container turnover is not considered.
(6) The traveling speeds of the AGV and the ECT and the moving speeds of the DCRC gantry and spreader are fixed values.
(7) The time for the spreader to travel vertically between the top and bottom of the block is a fixed value.
(8) The operation time of the AGVs under the QC, the time required for the spreader to grasp the container from the AGVs and ECTs, and the time of the spreader putting at the target position are not counted.

3.3. Notations

(1) Parameters

W	Set of ECT tasks, indexed by $w \in (1, 2, 3, \cdots, W)$
A	Set of AGV tasks, indexed by $a \in (1, 2, 3, \cdots, A)$
A_L	Set of AGV taking tasks, indexed by $A_{Li} \in (1, 2, 3, \cdots, A_L)$
W_L	Set of ECT taking tasks, indexed by $W_{Li} \in (1, 2, 3, \cdots, W_L)$
I	Set of AGVs, indexed by $i \in (1, 2, 3, \cdots, I)$
C	Set of blocks in yards, indexed by $c, d \in (1, 2, 3, \cdots, C)$
E	Set of bays in a block, indexed by $e, s \in (1, 2, 3, \cdots, E)$
F	Set of rows in a block, indexed by $f, z \in (1, 2, 3, \cdots, F)$
c_A	Set of tasks in the block c, indexed by $\mathrm{c}_p, \mathrm{c}_q \in (\mathrm{c}_1, \mathrm{c}_2, \mathrm{c}_3, \cdots, \mathrm{c}_A)$
e_{c_q}	Bay number of the qth task in block c
f_{c_q}	Row number of the qth task in block c
v_1	Speed of the AGV
v_2	Speed of the DCRC's spreader
v_3	Speed of the DCRC's gantry
M	A very large positive number
J	Length of a block
K	Width of a block
BL	Length of a bay
EW	Length of a row
TQ_c	Time matrix of AGVs traveling between the QC and block c
T_{cd}	Time matrix of AGVs traveling between the block c and block d
Tw_c	Arrival time of the wth ECT in block c
TT	Time required for DCRC's spreader to travel vertically

(2) Decision variables

$ti_1^{c_q}$	The moment when the AGV receives the qth task of block c at the QC
$ti_2^{c_q}$	The moment when the AGV transports the qth task of block c to the end of the block
$ti_3^{c_q}$	The moment when the AGV arrives at the designated bay of the qth task of block c
$ti_{end}^{c_q}$	The moment when the AGV finishes the qth task in block c
$td_1^{c_q}$	The moment when the spreader moves across to the edge of the block
$td_2^{c_q}$	The moment the spreader drops to the vehicle interaction position
$td_3^{c_q}$	The moment the spreader moves up to the edge of the block
$td_4^{c_q}$	The moment the spreader moves across to the target position
$td_{end}^{c_q}$	The moment when the DCRC finish the qth task in block c
$tid_{inter}^{c_q}$	The moment of interaction of the AGV with the spreader for the qth task in block c
λ_{c_q}	When the qth task of block c is an AGV task $\lambda_{c_q} = 1$, otherwise $\lambda_{c_q} = 0$
π_{c_q}	When the qth task of block c is a putting task $\pi_{c_q} = 1$, otherwise $\pi_{c_q} = 0$

$\theta_{d_p}^{c_q}$	When the AGV finishes processing the *qth* putting task in block *c* and goes to process the *qth* taking task in block *d*, $\theta_{d_p}^{c_q} = 1$; otherwise, $\theta_{d_p}^{c_q} = 0$
$\partial_{d_p}^{c_q}$	When the AGV finishes processing the *qth* taking task in block *c* and goes to process the *qth* putting task in block *d*, $\partial_{d_p}^{c_q} = 1$; otherwise, $\partial_{d_p}^{c_q} = 0$
$\varepsilon_i^{c_{pq}}$	When the AGV *i* operates the *qth* putting task in block *c* and then continues to operate the *qth* taking task in this block, $\varepsilon_i^{c_{pq}} = 1$;; otherwise, $\varepsilon_i^{c_{pq}} = 0$
$\overset{i}{\sigma}_{c_q}$	When AGV *i* performs the task of the *qth* container in block *c*, $\overset{i}{\sigma}_{c_q} = 1$; otherwise, $\overset{i}{\sigma}_{c_q} = 0$
$\phi_{A_{Li}}^{e_{c_q},f_{c_q}}$	When the task in block *c* of bay *b*, row *r* is the *ith* taking task of AGV, $\phi_{A_{Li}}^{e_{c_q},f_{c_q}} = 1$; otherwise, $\phi_{A_{Li}}^{e_{c_q},f_{c_q}} = 0$
$\Omega_{W_{Li}}^{e_{c_q},f_{c_q}}$	When the task in block *c* of bay *b*, row *r* is the *ith* taking task of ECT, $\Omega_{W_{Li}}^{e_{c_q},f_{c_q}} = 1$; otherwise, $\Omega_{W_{Li}}^{e_{c_q},f_{c_q}} = 0$.

3.4. Model

To solve the collaborative scheduling problem of DCRCs and AGVs, this paper proposes a mixed-integer programming model with the goal of achieving the minimum total running time for all DCRCs.

$$Z = \sum_{c_q \in c_A, c \in C} \max\left\{ t_{end}^{c_q} \right\} \tag{1}$$

Equation (1) is the objective function that minimizes the sum of the total occupied time of all DCRCs.

$$\sum_{c,d \in C} \sum_{\substack{c_q \in c_A \\ d_p \in d}} \theta_{d_p}^{c_q} = 1 \quad \forall i \in I \tag{2}$$

$$\sum_{c,d \in C} \sum_{\substack{c_q \in c \\ d_p \in d}} \partial_{d_p}^{c_q} = 1 \quad \forall i \in I \tag{3}$$

$$\sum_{c \in C} \sum_{c_q \in c_A} \overset{i}{\sigma}_{c_q} = 1 \quad \forall i \in I \tag{4}$$

$$\sum_{i \in I} \overset{i}{\sigma}_{c_q} = 1 \quad \forall c \in C, c_q \in c_A \tag{5}$$

$$\sum_{c \in C} \sum_{c_q \in c_A} \phi_{A_{Li}}^{e_{c_q},f_{c_q}} = 1 \quad \forall A_{Li} \in A_L \tag{6}$$

$$\sum_{c \in C} \sum_{c_q \in c_A} \Omega_{W_{Li}}^{e_{c_q},f_{c_q}} = 1 \quad \forall W_{Li} \in W_L \tag{7}$$

Constraints (2)–(7) ensure the uniqueness of the task for the AGV. Constraint (2) provides for the AGV performing a taking task after completing a putting task. Constraint (3) requires that the AGV performs a putting task after completing a taking task. Constraints (2) and (3) guarantee the integrity of the loading and unloading processes. Constraint (4) assures that each task is operated by one and only one AGV. Constraint (5) ensures that each AGV handles only one task simultaneously. Constraint (6) guarantees that each AGV taking task has a unique taking sequence corresponding to it. Constraint (7) assures that each ECT taking task has a unique taking sequence corresponding to it.

$$ti_1^{c_q} + TQ_c - ti_2^{c_q} + \left(1 - \pi_{c_q}\right) \cdot M \le 0 \quad \begin{array}{l} \forall c \in C, c_q \in c_A \\ \forall i \in I \end{array} \tag{8}$$

$$ti_2^{c_q} + \frac{BL \cdot e_{c_q}}{v_1} - ti_3^{c_q} + \left(1 - \pi_{c_q}\right) \cdot M \le 0 \quad \begin{array}{l} \forall c \in C, c_q \in c_A \\ \forall i \in I \end{array} \tag{9}$$

$$ti_{end}^{c_q} + \frac{\left(e_{c_q} - e_{c_p}\right) \cdot BL}{v_1} - ti_3^{c_q} + \pi_{c_p} \cdot M + \left(1 - \varepsilon_i^{c_{pq}}\right) \cdot M \le 0 \quad \begin{array}{l} \forall c \in C, c_q, c_p \in c_A \\ \forall i \in I \end{array} \tag{10}$$

$$ti_{end}^{c_q} + \frac{J + T_{cd} + e_{c_q} \cdot BL}{v_1} - ti_3^{d_p} + \pi_{d_p} \cdot M + \varepsilon_{c_{pq}} \cdot M \le 0 \quad \begin{array}{l} \forall c, d \in C, c_q \in c_A \\ \forall d_p \in d_A, i \in I \end{array} \tag{11}$$

Constraints (8)–(11) represent the time relationship at the horizontal transport link when the AGV releases the putting task. Constraint (8) indicates the relationship between the moment when the AGV receives the *qth* task from QC to the front end of block *c*. Constraint (9) represents the moment the AGV transports the *qth* task of block *c* to the front of the block, concerning the moment the task is delivered to the target bay *e*. Constraints (10) and (11) indicate the relationship between the moment the AGV completes the previous putting task and the moment it starts the next taking task. Constraint (10) represents the time relationship between the AGV going to the same block to take the container after it has completed the putting task. Constraint (11) represents the time relationship between the AGV going to a different block to take a container after it has completed the putting task.

$$tid_{inter}^{c_q} + \frac{BL \cdot e_{c_q}}{v_1} - ti_2^{c_q} + \left(1 - \pi_{c_q}\right) \cdot M \le 0 \quad \begin{array}{l} \forall c \in C, i \in I \\ \forall c_q \in c_A \end{array} \tag{12}$$

$$ti_3^{c_q} + TQ_c - ti_1^{c_q} + \left(1 - \pi_{c_q}\right) \cdot M \le 0 \quad \begin{array}{l} \forall c \in C, i \in I \\ \forall c_q \in c_A \end{array} \tag{13}$$

$$ti_1^{c_q} - ti_{end}^{c_q} = 0 \quad \begin{array}{l} \forall c \in C, i \in I \\ \forall c_q \in c_A \end{array} \tag{14}$$

$$ti_{end}^{c_q} + \left(1 - \pi_{c_q}\right) \cdot M - ti_1^{d_p} \cdot \left(\pi_{d_p} - 1\right) \le 0 \quad \begin{array}{l} \forall c, d \in C, i \in I \\ \forall c_q \in c_A, d_p \in d_A \end{array} \tag{15}$$

Constraints (12)–(14) indicate the time relationship between the horizontal transport links when the AGV performs the taking task. Constraint (12) indicates the time relationship in which the AGV runs the *qth* task of block c from the block bay position to the front of the block. Constraint (13) means the relationship between the moment when the AGV runs the *qth* task of block *c* from the front of the block to the QC. Constraint (14) describes the moment at which the AGV has completed its operation, when it reaches the QC with the *qth* task of block *c* loading onto the vessel. Constraint (15) concerns the time relationship between when the AGV completes the *qth* taking task in the previous block *e* and when it starts the *pth* putting task in block *d*.

$$\left[td_{end}^{c_{q-1}} + \max\left(\begin{array}{l} \frac{BL \cdot \left|e_{c_q} - e_{c_{q-1}}\right|}{v_2}, \\ \frac{EW \cdot f_{c_{q-1}} \cdot \lambda_{ct_q} + \left(K - EW \cdot e_{c_{q-1}}\right) \cdot \left(1 - \lambda_{ct_q}\right)}{v_3} \end{array} \right) - td_1^{c_q}\right] \cdot \pi_{c_{q-1}} \cdot \pi_{c_q} \le 0 \quad \begin{array}{l} \forall c \in C, \\ \forall c_{q-1}, c_q \in c_A \end{array} \tag{16}$$

$$\left(td_1^{c_q} + TT - td_2^{c_q}\right) \cdot \pi_{c_{q-1}} \cdot \pi_{c_q} \le 0 \quad \begin{array}{l} \forall c \in C, \\ \forall c_{q-1}, c_q \in c_A \end{array} \tag{17}$$

$$\left[tid_{inter}^{c_q} - \max\left(ti_3^{c_q} \cdot \lambda_{c_q}, Tw_c \cdot \left(1 - \lambda_{c_q}\right), td_2^{c_q}\right)\right] \cdot \pi_{c_{q-1}} \cdot \pi_{c_q} = 0 \quad \begin{array}{l} \forall c \in C \\ \forall i \in I, w \in W \\ \forall c_{q-1}, c_q \in c_A \end{array} \tag{18}$$

$$\left(ti_{end}^{c_q} - tid_{inter}^{c_q}\right) \cdot \lambda_{c_q} \cdot \pi_{c_{q-1}} \cdot \pi_{c_q} = 0 \quad \begin{array}{l} \forall c \in C, i \in I \\ \forall c_{q-1}, c_q \in c_A \end{array} \tag{19}$$

$$\left(tid_{inter}^{c_q} + TT - td_3^{c_q}\right) \cdot \pi_{c_{q-1}} \cdot \pi_{c_q} \le 0 \quad \begin{array}{l} \forall c \in C, i \in I \\ \forall c_{q-1}, c_q \in c_A \end{array} \tag{20}$$

$$\left[td_3^{ct_q} + \frac{EW \cdot f_{c_q} \cdot \lambda_{c_q} + \left(K - EW \cdot f_{c_q}\right) \cdot \left(1 - \lambda_{c_q}\right)}{v_3} - td_4^{c_q}\right] \cdot \pi_{c_{q-1}} \cdot \pi_{c_q} \leq 0 \quad \begin{array}{l} \forall c \in C, \\ \forall c_{q-1}, c_q \in c_A \end{array} \tag{21}$$

$$\left(td_{end}^{c_q} - td_4^{c_q}\right) \cdot \pi_{c_{q-1}} \cdot \pi_{c_q} = 0 \quad \begin{array}{l} \forall c \in C, \\ \forall c_{q-1}, c_q \in c_A \end{array} \tag{22}$$

Constraints (16)–(22) indicate the time relationship when the DCRC is operating in "PP" mode. Constraint (16) represents the time relationship between the end of the previous putting task of the DCRC and the spreader moving across to the edge of the block where the current putting task is positioned. Constraint (17) describes the time taken for the spreader to descend from above the block to the interaction point. Constraint (18) shows that the moment of interaction is the maximum of the moment when the spreader drops to the interaction point and the arrival time of the AGV or the ECT. Constraint (19) states that the DCRC and the AGV will interact when the AGV has finished its block delivery putting task. Constraint (20) means the moment when the spreader returns to the edge of the block after lifting the container. Constraint (21) represents the time relationship between the spreader running from the edge of the block on both sides to the specified stack. Constraint (22) means the moment when the DCRC has completed the current putting task.

$$td_{end}^{c_{q-1}} + \max\left(\frac{BL \cdot \left|e_{c_q} - e_{c_{q-1}}\right|}{v_2}, \frac{EW \cdot \left|f_{c_q} - f_{c_{q-1}}\right|}{v_3}\right) - td_4^{c_q} \cdot \pi_{c_{q-1}} \cdot \left(1 - \pi_{c_q}\right) \leq 0 \quad \begin{array}{l} \forall c \in C, \\ \forall c_{q-1}, c_q \in c_A \end{array} \tag{23}$$

$$\left(td_4^{ct_q} + \frac{BL \cdot f_{c_q} \cdot \lambda_{c_q} + \left(K - BL \cdot f_{c_q}\right) \cdot \left(1 - \lambda_{c_q}\right)}{v_3} - td_1^{ct_q}\right) \cdot \pi_{c_{q-1}} \cdot \left(1 - \pi_{c_q}\right) \leq 0 \quad \begin{array}{l} \forall c \in C, \\ \forall c_{q-1}, c_q \in c_A \end{array} \tag{24}$$

$$\left(td_1^{c_q} + TT - td_2^{c_q}\right) \cdot \pi_{c_{q-1}} \cdot \left(1 - \pi_{c_q}\right) \leq 0 \quad \begin{array}{l} \forall c \in C, \\ \forall c_{q-1}, c_q \in c_A \end{array} \tag{25}$$

$$tid_{inter}^{c_q} - \max\left(td_2^{c_q}, Tw_c\right) \cdot \lambda_{c_q} \cdot \pi_{c_{q-1}} \cdot \left(1 - \pi_{c_q}\right) = 0 \quad \begin{array}{l} \forall c \in C \\ \forall i \in I, w \in W \\ \forall c_{q-1}, c_q \in c_A \end{array} \tag{26}$$

$$td_{end}^{c_q} - \max\left(td_2^{c_q}, Tw_c\right) \cdot \pi_{c_{q-1}} \cdot \left(1 - \pi_{c_q}\right) = 0 \quad \begin{array}{l} \forall c \in C \\ \forall i \in I, w \in W \\ \forall c_{q-1}, c_q \in c_A \end{array} \tag{27}$$

Constraints (23)–(27) indicate the time relationship when the DCRC is operating in "PT" mode. Constraint (23) represents the time relationship between the previous putting task ending and the spreader reaching the position of the current taking task. Constraint (24) represents the time relationship between the spreader picking up the putting task and moving across to the edge of the block where the vehicle is positioned. Constraint (25) represents the moment when the spreader reaches the edge of the block and drops down to the AGV or the ECT interaction position. Constraint (26) indicates that if the task is an AGV taking task, the moment of interaction between the AGV and the DCRC is the maximum of the moment when the spreader drops to the point of interaction with the AGV and the moment of arrival of the AGV. Constraint (27) indicates that the end of the taking task of the AGV/ECT and the DCRC is the maximum of the moment when the spreader reaches the point of interaction and the moment of arrival of the AGV or the ECT.

$$td_{end}^{c_{q-1}} + TT - td_3^{c_q} \cdot \left(1 - \pi_{c_{q-1}}\right) \cdot \pi_{c_q} \leq 0 \quad \begin{array}{l} \forall c \in C, \\ \forall c_{q-1}, c_q \in c_A \end{array} \tag{28}$$

$$\left[td_3^{c_q}+\max\left(\frac{EW\cdot\left|e_{c_q}-e_{c_{q-1}}\right|}{v_2},\frac{K}{v_3}\cdot\left|\lambda_{c_{q-1}}-\lambda_{c_q}\right|\right)-td_1^{c_q}\right]\cdot\left(1-\pi_{c_{q-1}}\right)\cdot\pi_{c_q}\le 0 \quad \begin{array}{l}\forall c\in C,\\ \forall c_{q-1},c_q\in c_A\end{array} \tag{29}$$

$$\left(td_1^{c_q}+TT-td_2^{c_q}\right)\cdot\left(1-\pi_{c_{q-1}}\right)\cdot\pi_{c_q}\le 0 \quad \begin{array}{l}\forall c\in C,\\ \forall c_{q-1},c_q\in c_A\end{array} \tag{30}$$

$$\left[tid_{inter}^{c_q}-\max\left(ti_3^{c_q}\cdot\lambda_{c_q},Tw_c\cdot\left(1-\lambda_{c_q}\right),td_2^{c_q}\right)\right]\cdot\left(1-\pi_{c_{q-1}}\right)\cdot\pi_{c_q}=0 \quad \begin{array}{l}\forall c\in C\\ \forall i\in I,w\in W\\ \forall c_{q-1},c_q\in c_A\end{array} \tag{31}$$

$$\left(ti_{end}^{c_q}-tid_{inter}^{c_q}\right)\cdot\lambda_{c_q}\cdot\left(1-\pi_{c_{q-1}}\right)\cdot\pi_{c_q}=0 \quad \begin{array}{l}\forall c\in C,i\in I\\ \forall c_{q-1},c_q\in c_A\end{array} \tag{32}$$

$$\left(tid_{inter}^{c_q}+TT-td_3^{c_q}\right)\cdot\left(1-\pi_{c_{q-1}}\right)\cdot\pi_{c_q}\le 0 \quad \begin{array}{l}\forall c\in C,i\in I\\ \forall c_{q-1},c_q\in c_A\end{array} \tag{33}$$

$$\left[td_3^{ct_q}+\frac{EW\cdot f_{c_q}\cdot\lambda_{c_q}+\left(K-EW\cdot f_{c_q}\right)\cdot\left(1-\lambda_{c_q}\right)}{v_3}-td_4^{c_q}\right]\cdot\left(1-\pi_{c_{q-1}}\right)\cdot\pi_{c_q}\le 0 \quad \begin{array}{l}\forall c\in C,\\ \forall c_{q-1},c_q\in c_A\end{array} \tag{34}$$

$$\left(td_{end}^{c_q}-td_4^{c_q}\right)\cdot\left(1-\pi_{c_{q-1}}\right)\cdot\pi_{c_q}=0 \quad \begin{array}{l}\forall c\in C,\\ \forall c_{q-1},c_q\in c_A\end{array} \tag{35}$$

Constraints (28)–(35) indicate the time relationship when the DCRC is operating in “TP” mode. Constraint (28) means the moment the spreader is lifted above the block after completing the previous putting task. Constraint (29) indicates the time relationship after the spreader has been lifted and the spreader reaches the corresponding bay at the next task’s block side. Constraint (30) represents the time relationship between the spreader leaving the block’s edge and interacting with the vehicle. Constraint (31) indicates that the interaction time is the maximum of the moment when the spreader sags to the interaction point and the arrival time of the AGV or the ECT. Constraint (32) indicates that, if the task is an AGV putting task, the moment of interaction between the DCRC and the AGV is when the AGV discharge task is completed. Constraint (33) represents the moment when the spreader returns to the edge of the block after picking up the container. Constraint (34) represents the time relationship when the spreader runs from the edges on the sides of the block to the designated stack. Constraint (35) indicates the moment when the DCRC has completed its putting task.

$$\left(td_{end}^{c_{q-1}}+TT-td_3^{c_q}\right)\cdot\left(1-\pi_{c_{q-1}}\right)\cdot\left(1-\pi_{c_q}\right)\le 0 \quad \begin{array}{l}\forall c\in C,\\ \forall c_{q-1},c_q\in c_A\end{array} \tag{36}$$

$$\begin{array}{c}td_3^{c_q}+\max\left(\begin{array}{c}\frac{BL\cdot\left|e_{c_q}-e_{c_{q-1}}\right|}{v_2},\\ \frac{EW\cdot f_{c_q}\cdot\lambda_{c_{q-1}}+\left(K-EW\cdot f_{c_q}\right)\cdot\left(1-\lambda_{c_{q-1}}\right)}{v_3}\end{array}\right)-\\ td_4^{ct_q}\cdot\left(1-\pi_{c_{q-1}}\right)\cdot\left(1-\pi_{c_q}\right)\le 0\end{array} \quad \begin{array}{l}\forall c\in C,\\ \forall c_{q-1},c_q\in c_A\end{array} \tag{37}$$

$$\left(td_4^{c_q}+\frac{EW\cdot f_{c_q}\cdot\lambda_{c_q}+\left(K-EW\cdot f_{c_q}\right)\cdot\left(1-\lambda_{c_q}\right)}{v_3}-td_1^{c_q}\right)\cdot\left(1-\pi_{c_{q-1}}\right)\cdot\left(1-\pi_{c_q}\right)\le 0 \quad \begin{array}{l}\forall c\in C,\\ \forall c_{q-1},c_q\in c_A\end{array} \tag{38}$$

$$\left(td_1^{c_q}+TT-td_2^{c_q}\right)\cdot\left(1-\pi_{c_{q-1}}\right)\cdot\left(1-\pi_{c_q}\right)\le 0 \quad \begin{array}{l}\forall c\in C,\\ \forall c_{q-1},c_q\in c_A\end{array} \tag{39}$$

$$tid_{inter}^{c_q} - \max\left(td_2^{c_q}, Tw_c\right) \cdot \lambda_{c_q} \cdot \left(1 - \pi_{c_{q-1}}\right) \cdot \left(1 - \pi_{c_q}\right) = 0 \quad \begin{array}{l} \forall c \in C \\ \forall i \in I, w \in W \\ \forall c_{q-1}, c_q \in c_A \end{array} \tag{40}$$

$$td_{end}^{ct_q} - \max\left(td_2^{c_q}, Tw_c\right) \cdot \left(1 - \pi_{c_{q-1}}\right) \cdot \left(1 - \pi_{c_q}\right) = 0 \quad \begin{array}{l} \forall c \in C, w \in W \\ \forall c_{q-1}, c_q \in c_A \end{array} \tag{41}$$

Constraints (36)–(41) indicate the time relationship when the DCRC is operating in "TT" mode. Constraint (36) represents the relationship between the moment when the previous taking task ends and the moment when the spreader moves up to the edge of the block. Constraint (37) represents the time relationship between the arrival of the spreader from the edge of the block of the previous taking task to the container position to be taken. Constraint (38) represents the time relationship between the moment when the spreader takes the taking task to be lifted and the moment when it moves across to the edge of the block for the taking task to be lifted. Constraint (39) shows when the spreader reaches the edge of the block and when it drops down to the position where the vehicle can interact with it. Constraint (40) indicates that, if the task is an AGV taking task, this interaction moment is the maximum of the moment when the spreader is dropped at the interaction point of the AGV and the arrival moment of the AGV. Constraint (41) represents that the greater time between when the spreader drops to the point of interaction and the time of arrival of the AGV/ECT is the end time of the DCRC task.

$$ti_1^{c_q}, ti_2^{c_q}, ti_3^{c_q}, ti_{end}^{c_q}, td_1^{c_q}, td_2^{c_q}, td_3^{c_q}, td_4^{c_q}, td_{end}^{c_q}, tid_{inter}^{c_q} \geq 0 \quad \forall c \in C, c_q \in c_A \tag{42}$$

Constraint (42) defines the range of parameters of the decision variables.

4. Proposed Algorithm

As the logic of the model studied in this paper is more complex and there are many optional loading and unloading points, the chosen algorithm needs to be practical. Genetic algorithms (GA) have good global search capabilities and excellent robustness, and several scholars have verified the reliability of GA for equipment scheduling problems [36,44,45].

Because each bay of the block in the U-shaped scheme can perform loading and unloading tasks, the available optional point is dozens of times that of the end operation scheme. To better fit the actual operation of the port, the refined collaborative scheduling model of the U-shaped ACT in the loading and unloading process established in this paper considers more operation modes and more complex logical relationships. This paper proposes the adaptive co-evolutionary genetic algorithm (ACGA) to solve the model better. The ACGA uses the idea of two-population coevolution to improve population diversity, and improves the convergence speed of the algorithm through adaptive dynamic adjustment of parameters. This paper uses two rules as stop criteria to ensure reasonable calculation time and accuracy of results: when the optimal fitness value (OFV) remains unchanged for several generations or the number of iterations reaches the maximum algebra, the algorithm stops. The flow of the ACGA is shown in Figure 4.

4.1. Chromosome Coding

The algorithm uses an integer coding approach where one chromosome represents the sequence of operations for a set of AGVs, i.e., a candidate solution. In the above hypothetical model, the target container position and sequence of operations for each container are known, and the ECT's type of operation and arrival time are known. The order in which AGV will serve the QC and the type of task for the next DCRC operation are unknown. The task sequence of the AGVs is scheduled to determine the type of operation of the DCRC in conjunction with the arrival time of the ECTs.

To distinguish between the AGV task, the ECT task, and the DCRC task, this paper is coded with a multi-layer chromosome in the form of a task assignment. The initial chromosome is shown in Figure 5. It is assumed that there are two DCRCs and four AGVs. Eight AGV-putting tasks are denoted by the numbers 0–7. Eight AGV-taking tasks are denoted by the numbers 8–15. Four ECT-putting tasks are denoted by the numbers 16–19. Four ECT-taking tasks are denoted by the numbers 20–23. All odd-numbered tasks are those of DCRC 1, and even-numbered tasks are those of DCRC 2. The green, purple, orange, and grey genes indicate the tasks of AGV 1, AGV 2, AGV 3, and AGV 4, respectively.

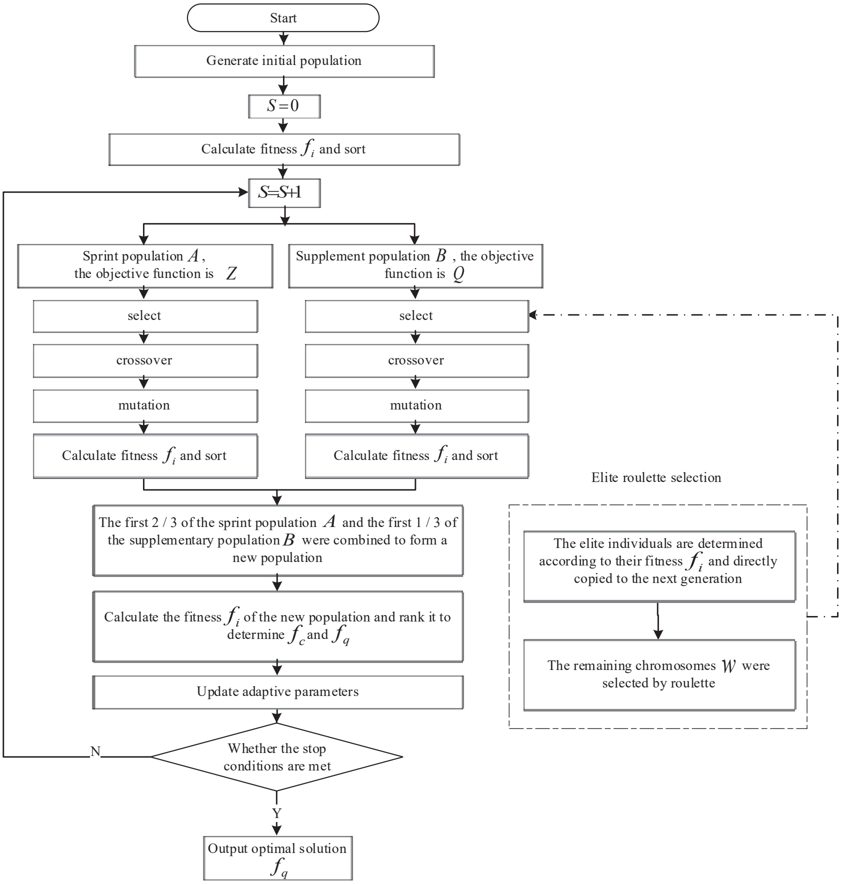
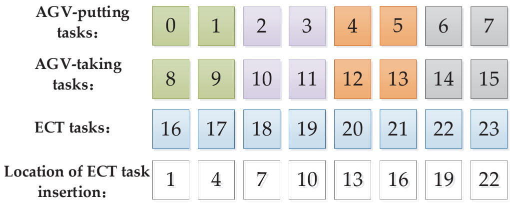


Figure 4. ACGA flow.



AGV-putting tasks:	0	1	2	3	4	5	6	7
AGV-taking tasks:	8	9	10	11	12	13	14	15
ECT tasks:	16	17	18	19	20	21	22	23
Location of ECT task insertion:	1	4	7	10	13	16	19	22

Figure 5. Initial chromosome.

As shown in Figure 5, the first chromosome layer is for AGV-putting tasks. The second chromosome layer is for AGV-taking tasks. Since the AGV always performs a putting task and then goes to a taking task, the second chromosome layer is inserted sequentially after the genes in the first chromosome layer. The third chromosome layer is for ECT tasks. The position in the chromosome of the ECT task at the corresponding position in the third layer is indicated by the fourth layer of chromosomes. The final obtained initial chromosome corresponds to the sequence of jobs, as shown in Figure 6. The task assignments of the AGV and DCRC for this chromosome are shown in Figure 7.

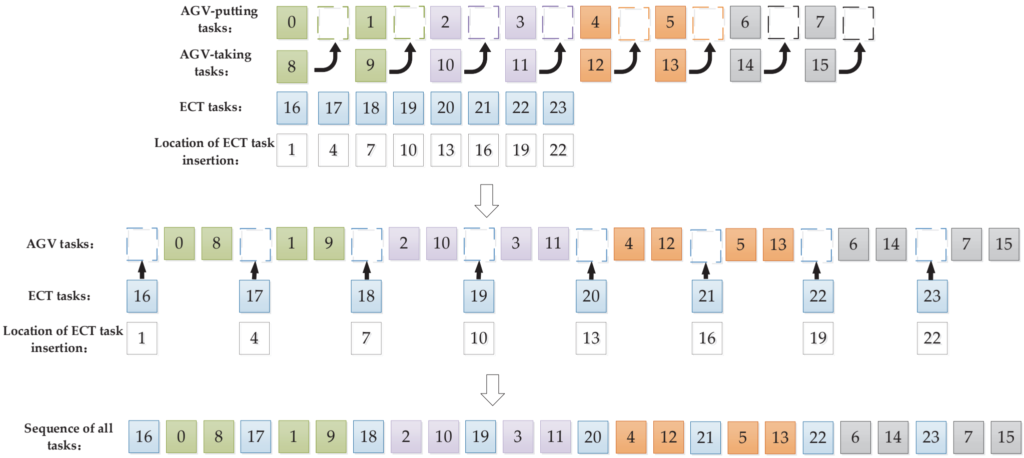


Figure 6. Sequence of tasks represented by the initial chromosome.

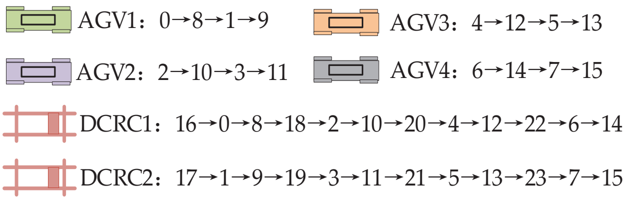


Figure 7. AGV and DCRC task assignments.

4.2. Coevolution

Coevolution was first proposed by Ehrlich and Raven. In this process, different populations have different evolutionary goals, and they can complement each other with information, have strong global search ability, and overcome the phenomenon of a single population and premature convergence in the later stage of the GA [46–48].

The first illustration of the degree of variation in the task order is given by assuming that there are V AGVs and Y containers. A population has X task orders, and the degree of variation between task orders x_b and x_m is:

$$Q = \frac{1}{Y}\sum_{y=1}^{Y}\lambda \qquad if\begin{cases} x_y^b = x_y^m, \lambda = 0 \\ x_y^b \neq x_y^m, \lambda = 1 \end{cases}, x_y^b, x_y^m \in V \tag{43}$$

x_y^b denotes the yth task sequence number of task sequence b, resulting in the shortest total running time of the current DCRCs. x_y^m is the yth task sequence number of the mth task sequence. The larger the value of Q, the greater the degree of difference between the total running time of the DCRC represented by that task sequence and the current shortest total running time.

After generating the initial task order population, as coded above, this task order population is divided into a sprint population and a supplementary population. The objective function for the sprint population is $Z = \sum\limits_{a \in A}\sum\limits_{w \in W}\min(Z_1 + Z_2)$, i.e., the shortest running time for all DCRCs, to guide the population to converge as soon as possible. The objective function of the supplementary population is $Q * Z$, which aims to maintain population diversity, complement the primary population, and avoid premature maturation of the algorithm. When the two task sequential populations have completed the crossover and variation operations, respectively, they are ranked by the shortest operation time of the respective task populations. The top 2/3 of the sprint population and the top 1/3 of the complementary population are selected from lowest to highest to form a new task population. The two populations are independent of each other and can be made computationally more efficient by using parallel computing methods during the calculation.

4.3. Adaptive Crossover and Mutation

The probability of changing the order of tasks determines the richness of the task population. By adaptively changing the probability of crossover and mutation, faster and more accurate convergence to the shortest total running time of a DCRC can be achieved. For example, the probability of crossover and mutation becomes larger when the degree of variation in the minimum total running time of the task order in the task population is low. When the degree of variation is high, the probability of crossover and mutation is low. The adaptive probability adjustment expression for crossover and variation is:

$$if\arcsin\left(\frac{fave}{f\max}\right) < \pi/6\begin{cases} P_c = k_1\dfrac{\arcsin\left(\frac{fave}{f\max}\right)}{\pi/2} \\ P_m = k_2\left(1-\dfrac{\arcsin\left(\frac{fave}{f\max}\right)}{\pi/2}\right) \end{cases} \tag{44}$$

$$if\arcsin\left(\frac{fave}{f\max}\right) \geq \pi/6\begin{cases} P_c = k_1\left(1-\dfrac{\arcsin\left(\frac{fave}{f\max}\right)}{\pi/2}\right) \\ P_m = k_2\dfrac{\arcsin\left(\frac{fave}{f\max}\right)}{\pi/2} \end{cases} \tag{45}$$

p_c and p_m are the crossover probability and mutation probability; k_1 and k_2 are random numbers between [0, 1]; and f_{max} and f_{avg} are the maximum and average values of the DCRC's shortest operating times in the task population, respectively.

4.3.1. Cross the Task Sequence

The crossover means that the parent task order to be crossed is selected according to the crossover probability. The crossover probabilities for each layer are obtained by adaptive adjustment. A partial crossover is applied to the selected task order. Two crossover points are randomly selected for each chromosome layer, which form a crossover fragment and are crossed in the same layer. The crossover process is shown in Figure 8.

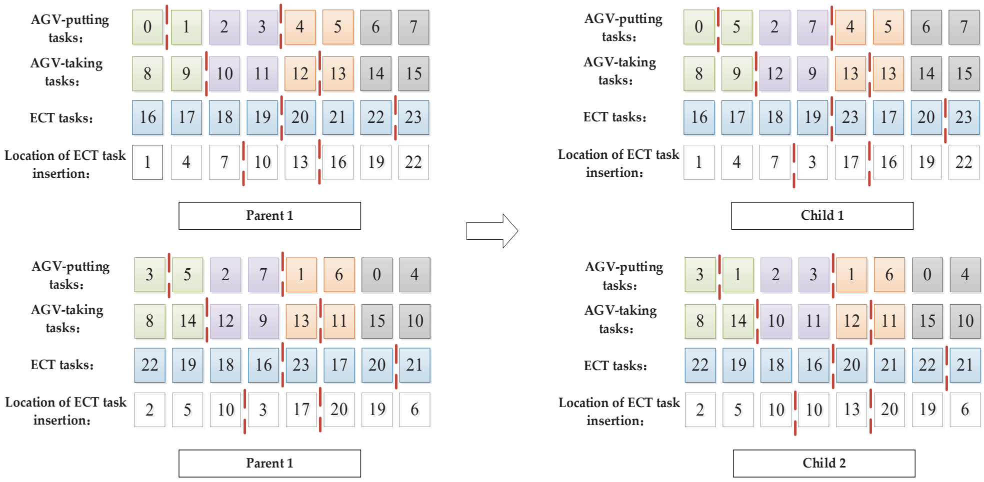


Figure 8. Cross the task sequence.

]

4.3.2. Mutate the Task Sequence

The mutation means that the parent task order to be crossed is selected according to the mutation probability. The mutation probabilities for each stratum are obtained by adaptive adjustment. One to three layers of chromosomes were randomly selected for one task, and the fourth layers of chromosomes were randomly selected for two tasks. The selected tasks were all mutated within the allowed range. The mutation process is shown in Figure 9.

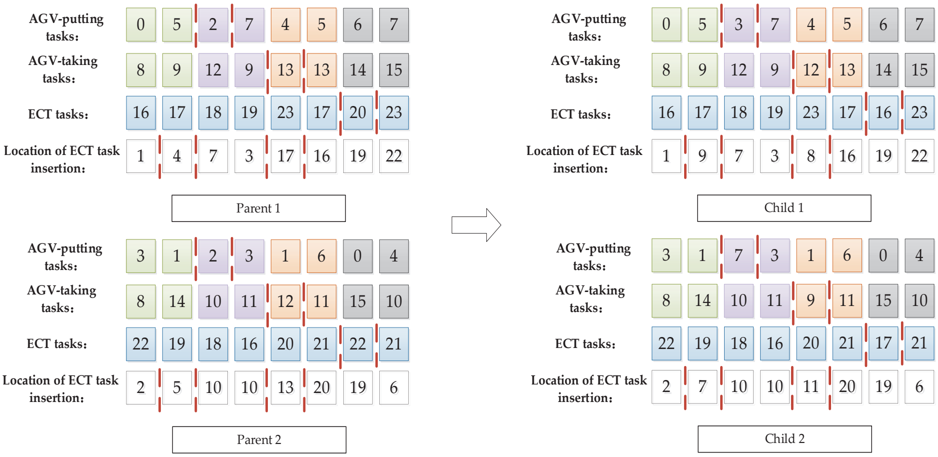


Figure 9. Mutate the task sequence.

4.4. Repair the Task Sequence

As the new task sequence obtained after crossover or mutation may have duplicated or missing tasks, the repair operation checks the sequence of tasks obtained after crossover and mutation, eliminating duplicate tasks and completing missing tasks. The repair process is shown in Figure 10.

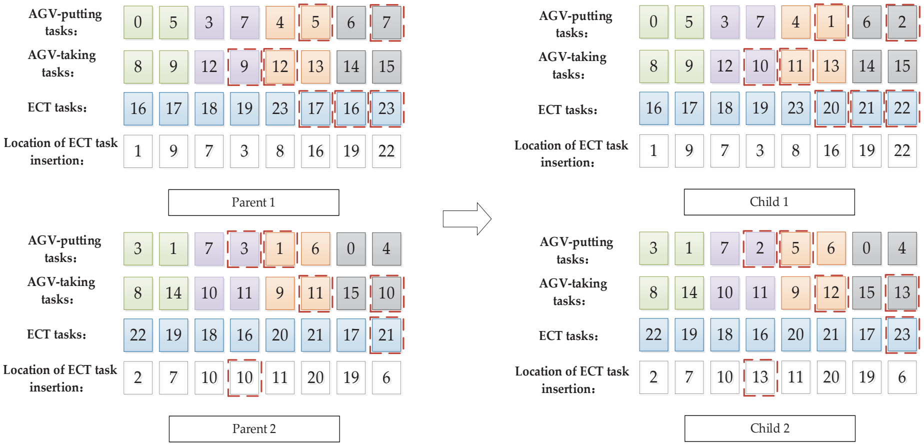


Figure 10. Repair the task sequence.

5. Simulation Experiments and Analysis

This chapter first compares the proposed ACGA algorithm with the adaptive genetic algorithms (AGA) [49] and the GA [36] through a small-scale arithmetic example to demonstrate the correctness and practicality of the AA algorithm for solving the present model.

Secondly, a sensitivity analysis of the critical factors of the problem is carried out in solving the large-scale arithmetic case to investigate the reasonable allocation of the amount of equipment under the U-shaped ACT.

5.1. Parameter Settings

(1) Define 4–500 container tasks as small-scale problems and problems above 500 containers as large-scale problems.
(2) The number of QCs is one to three, the number of AGVs is two to fifty, the number of blocks is two to ten, and the number of ECTs reaches twenty every hour and the arrival times obey a uniform distribution.
(3) The block and position to be taken or put and the operation type of the ECT are all randomly generated.
(4) Through multiple experiments by controlling variables, in this paper the maximum number of iterations is set to 20–800 (depending on the number of tasks), the population size is 80, and the initial probabilities of crossover and mutation are set to 0.7 and 0.3, respectively.
(5) The experiments are implemented in MATLAB 2016a, and all simulations are performed on a computer with Intel Core TM i5 CPU 2.1 GHz and 64 GB RAM under a Windows operating system.
(6) In the U-shaped ACT, the width of the horizontal transport area is 80 m. There are 99 bays and nine rows in a block. The containers are 12 m long and 2.5 m wide. The DCRC's gantry and spreader move at a speed of 1 m/s, and the AGV travels at 4m/s. The specific parameters of the ACT are shown in Table 1.

Table 1. ACT parameters.

Parameters	**Numerical Values**
Width of horizontal transportation area	80 m
Bay number in block	80
Row number in block	9
Container length	12 m
Container width	2.5 m
Moving speed of DCRC gantry	1 m/s
Moving speed of DCRC spreader	1 m/s
Driving speed of AGV	4 m/s

5.2. Results and Algorithm Comparison for Small-Sized Problems

This section examines the feasibility of the model and algorithm in small-scale experiments. Different example sizes are compared, and the performance of the proposed ACGA is analyzed in terms of the OFV, i.e., the total running time of all DCRCs and the required computation time (CPUT).

Table 2 records the results of the ACGA, GA, and AGA for the case of one QC, varying the number of containers from 20 to 50 and the number of AGVs from two to ten, respectively. Where "Containers" indicate the number of containers at the QC for each test, "AGVs/DCRCs" indicates the number of AGVs and DCRCs used, "CPUT(s)" indicates the time taken for the calculation, and "OFV(s)" indicates the shortest running time of the DCRC for each calculation. The experimental results in Table 2 show that, in terms of CPUT, all three algorithms increase as the problem size increases. The ACGA takes slightly more time than the AGA and is significantly better than the GA. When there were 50 container tasks, the difference between the ACGA and the AGA was about 5%, and the performance improvement with the GA was about 12%. In terms of minimum total running time, the GA, the AGA and the ACGA increase with the number of containers. However, the advantage of the ACGA is demonstrated. The minimum total runtime with the ACGA improves by approximately 15% compared to the GA and the AGA.

Table 2. Comparison table of small-scale problems.

No.	Containers	AGVs/DCRCs	ACGA		AGA				GA			
			CPUT (s)	OFV (s)	CPUT (s)	OFV (s)	Gap Rate (%)		CPUT (s)	OFV (s)	Gap Rate (%)	
							CPUT	OFV			CPUT	OFV
1	20	2/2	1.28	1129	1.27	1287	−0.72	12.28	1.29	1343	0.78	15.96
2	20	2/4	1.58	2918	1.57	3383	−0.60	13.75	1.54	3502	−2.60	16.66
3	20	4/2	1.69	1031	1.66	1199	−1.59	13.99	1.73	1316	2.31	21.67
4	30	5/5	2.35	4698	2.27	5623	−3.55	16.45	2.49	5546	5.62	15.30
5	30	10/5	2.63	4204	2.51	5047	−4.70	16.70	2.69	5206	5.73	19.25
6	30	2/2	1.74	2025	1.68	2376	−3.47	14.78	1.9	2433	8.42	16.75
7	40	4/2	2.13	3305	2.04	3942	−4.20	16.16	2.36	3836	9.75	13.85
8	40	4/4	2.74	5246	2.62	6317	−4.44	16.95	3.11	6347	11.90	17.35
9	50	8/6	3.49	9315	3.32	10,969	−5.18	15.08	3.88	11,115	10.05	16.19
10	50	9/5	2.90	7324	2.75	8658	−5.27	15.41	3.31	8795	12.39	16.70

Figure 11 shows the convergence effect of Example 10. The blue line shows the convergence curve of the GA, the green line shows the convergence curve of the AGA, and the orange line shows the convergence curve of the ACGA. As can be seen from the graph, the ACGA and the AGA converge at around 50 generations, both converging faster than the GA, at around 70 generations. At the same time, the ACGA has a significantly lower DCRC total running time and better solution results. The experimental results show that the ACGA can find near-optimal solutions in a short time and has better global search capability. Therefore, the ACGA is used next in solving large-scale algorithms.

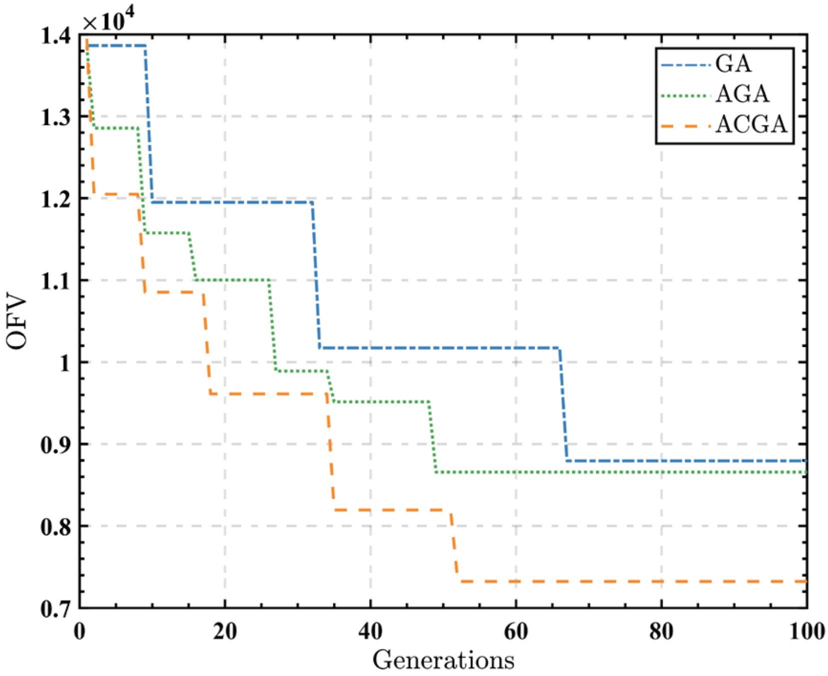


Figure 11. Comparison of algorithm results.

5.3. Results and Analysis for Large-Sized Problems

This section examines the impact of different equipment ratios on a large-scale task. Tables 3–5 record the results of varying the number of AGVs from five to fifty and the number of DCRCs from two to ten when using one, two, and three QCs, respectively. Figures 12–17 are all plotted from the data in Tables 3–5.

Table 3. Calculation example of 1 QC.

DCRCs-AGVs	OFV(s)	CPUT(s)	DCRCs-AGVs	OFV(s)	CPUT(s)	DCRCs-AGVs	OFV(s)	CPUT(s)	DCRCs-AGVs	OFV(s)	CPUT(s)	DCRCs-AGVs	OFV(s)	CPUT(s)
2–5	81,437	10.61	3–45	77,369	17.76	5–35	83,845	18.88	7–25	98,135	22.45	9–15	117,514	22.86
2–10	70,641	14.08	3–50	77,372	18.98	5–40	84,128	19.18	7–30	94,163	24.08	9–20	113,673	23.37
2–15	71,087	15.31	4–5	89,359	14.08	5–45	84,088	19.59	7–35	90,694	24.59	9–25	108,831	24.49
2–20	71,196	16.84	4–10	85,068	14.18	5–50	84,019	20.92	7–40	91,394	24.69	9–30	105,287	27.35
2–25	70,967	15.51	4–15	81,336	15.20	6–5	102,691	20.20	7–45	91,215	25.71	9–35	104,370	29.08
2–30	71,095	16.53	4–20	79,421	16.12	6–10	99,994	21.43	7–50	91,271	27.14	9–40	101,227	29.18
2–35	70,959	17.24	4–25	79,772	13.47	6–15	96,735	19.90	8–5	115,317	20.51	9–45	98,227	30.10
2–40	71,173	17.86	4–30	79,922	15.71	6–20	93,679	21.02	8–10	111,492	22.65	9–50	98,800	31.12
2–45	70,193	17.45	4–35	79,944	15.61	6–25	90,243	21.53	8–15	108,638	24.29	10–5	129,741	25.10
2–50	71,400	18.37	4–40	80,160	17.45	6–30	87,068	24.18	8–20	106,512	23.57	10–10	126,531	28.37
3–5	84,375	13.16	4–45	80,827	18.57	6–35	87,141	22.45	8–25	103,965	24.18	10–15	123,334	29.69
3–10	80,296	14.18	4–50	80,172	19.59	6–40	87,155	22.04	8–30	99,803	26.43	10–20	120,691	26.94
3–15	76,367	13.57	5–5	98,897	20.31	6–45	87,927	22.65	8–35	97,010	25.41	10–25	117,998	30.31
3–20	77,177	15.00	5–10	95,671	19.08	6–50	88,119	23.47	8–40	94,945	25.10	10–30	115,575	25.82
3–25	77,293	13.57	5–15	90,630	21.73	7–5	112,536	20.31	8–45	94,976	26.63	10–35	111,534	28.88
3–30	77,331	16.12	5–20	87,074	18.27	7–10	109,049	20.31	8–50	94,862	29.29	10–40	107,767	28.57
3–35	76,464	16.73	5–25	83,949	17.86	7–15	105,849	24.49	9–5	122,217	21.02	10–45	104,866	29.39
3–40	76,553	17.24	5–30	84,268	18.27	7–20	101,038	23.06	9–10	120,147	21.94	10–50	101,849	30.20

Table 4. Calculation example of 2 QCs.

DCRCs-AGVs	OFV(s)	CPUT(s)	DCRCs-AGVs	OFV(s)	CPUT(s)	DCRCs-AGVs	OFV(s)	CPUT(s)	DCRCs-AGVs	OFV(s)	CPUT(s)	DCRCs-AGVs	OFV(s)	CPUT(s)
2–5	77,897	12.37	3–45	57,782	20.39	5–35	51,217	20.26	7–25	68,291	20.26	9–15	96,756	24.08
2–10	61,974	13.03	3–50	57,571	21.58	5–40	51,862	19.74	7–30	65,047	20.00	9–20	94,556	26.84
2–15	62,381	13.82	4–5	72,956	13.95	5–45	51,668	20.79	7–35	62,878	21.84	9–25	90,110	28.42
2–20	62,308	12.89	4–10	66,817	13.95	5–50	51,448	21.05	7–40	62,915	23.03	9–30	87,274	29.21
2–25	62,543	14.08	4–15	58,604	15.53	6–5	62,632	17.11	7–45	62,505	24.87	9–35	83,224	30.26
2–30	61,981	15.13	4–20	50,990	14.21	6–10	60,355	18.95	7–50	61,760	25.79	9–40	80,114	30.39
2–35	62,357	14.21	4–25	51,098	15.00	6–15	59,435	19.61	8–5	99,272	21.45	9–45	76,563	31.97
2–40	63,049	15.00	4–30	50,629	13.95	6–20	57,463	19.61	8–10	96,875	23.16	9–50	76,901	33.42
2–45	62,654	15.79	4–35	50,321	16.71	6–25	56,849	18.16	8–15	92,156	23.16	10–5	117,498	24.08
2–50	62,688	16.58	4–40	50,574	16.32	6–30	55,626	19.61	8–20	89,666	25.66	10–10	113,019	26.18
3–5	73,847	13.55	4–45	50,639	17.89	6–35	56,032	19.87	8–25	83,810	24.21	10–15	108,703	27.24
3–10	63,594	13.82	4–50	51,760	19.08	6–40	56,039	20.13	8–30	79,476	26.32	10–20	105,409	28.16
3–15	57,578	13.03	5–5	61,748	16.58	6–45	55,792	21.32	8–35	74,576	27.63	10–25	102,564	29.47
3–20	58,102	14.21	5–10	58,475	16.84	6–50	55,909	22.63	8–40	70,128	28.29	10–30	98,156	32.50
3–25	57,743	15.66	5–15	55,784	17.89	7–5	85,278	18.03	8–45	70,641	30.66	10–35	95,838	33.95
3–30	57,673	17.37	5–20	53,014	17.63	7–10	81,612	18.42	8–50	69,849	32.63	10–40	92,097	34.34
3–35	57,898	19.08	5–25	51,355	17.63	7–15	77,056	19.47	9–5	103,102	23.16	10–45	88,525	34.87
3–40	57,602	20.92	5–30	51,971	18.55	7–20	72,028	18.55	9–10	99,831	23.68	10–50	84,330	36.05

Table 5. Calculation example of 3 QCs.

DCRCs-AGVs	OFV(s)	CPUT(s)	DCRCs-AGVs	OFV(s)	CPUT(s)	DCRCs-AGVs	OFV(s)	CPUT(s)	DCRCs-AGVs	OFV(s)	CPUT(s)	DCRCs-AGVs	OFV(s)	CPUT(s)
2–5	58,467	18.57	3–45	51,769	31.53	5–35	40,283	27.76	7–25	47,874	31.23	9–15	70,497	37.36
2–10	54,852	19.80	3–50	52,043	33.67	5–40	40,579	28.98	7–30	44,406	30.88	9–20	67,450	39.93
2–15	54,966	26.33	4–5	56,334	20.61	5–45	40,051	31.53	7–35	42,727	36.20	9–25	63,971	44.37
2–20	54,914	20.92	4–10	54,639	21.53	5–50	41,128	34.69	7–40	43,246	39.04	9–30	59,847	49.51
2–25	54,827	19.08	4–15	50,224	28.88	6–5	49,512	31.33	7–45	43,264	39.75	9–35	56,566	50.40
2–30	54,869	21.94	4–20	46,184	22.76	6–10	45,777	30.51	7–50	42,654	39.31	9–40	52,259	46.41
2–35	54,773	20.92	4–25	46,922	25.10	6–15	41,263	23.78	8–5	69,864	33.19	9–45	49,608	52.35
2–40	54,828	25.00	4–30	47,520	20.00	6–20	39,835	33.78	8–10	65,199	35.30	9–50	49,713	52.00
2–45	54,883	26.73	4–35	46,011	26.12	6–25	38,032	34.59	8–15	63,669	36.11	10–5	79,906	38.65
2–50	54,796	29.18	4–40	45,934	25.41	6–30	37,734	35.92	8–20	61,748	35.94	10–10	77,083	41.42
3–5	57,723	18.67	4–45	46,639	27.65	6–35	37,932	38.57	8–25	57,940	41.88	10–15	74,329	41.74
3–10	53,654	19.90	4–50	46,007	30.82	6–40	38,365	39.29	8–30	53,056	41.17	10–20	71,833	43.76
3–15	51,246	24.29	5–5	50,231	25.41	6–45	37,486	34.08	8–35	50,825	43.48	10–25	68,519	45.54
3–20	51,303	20.31	5–10	48,678	20.71	6–50	37,593	38.27	8–40	47,064	44.19	10–30	64,965	48.87
3–25	51,271	28.78	5–15	44,377	36.53	7–5	62,884	27.60	8–45	47,748	39.75	10–35	60,289	51.80
3–30	50,993	25.61	5–20	41,747	27.55	7–10	59,611	28.66	8–50	47,649	40.46	10–40	58,568	55.05
3–35	51,096	24.80	5–25	39,495	37.96	7–15	55,589	32.83	9–5	75,614	35.47	10–45	54,050	56.22
3–40	51,244	27.45	5–30	40,036	29.29	7–20	51,165	29.90	9–10	72,052	35.76	10–50	51,630	59.09

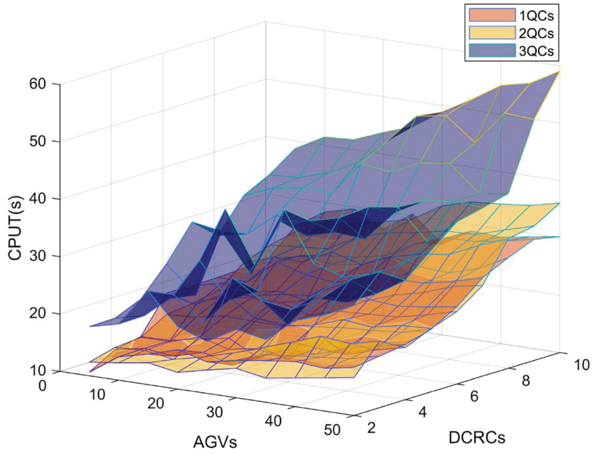


Figure 12. CPUT(s).

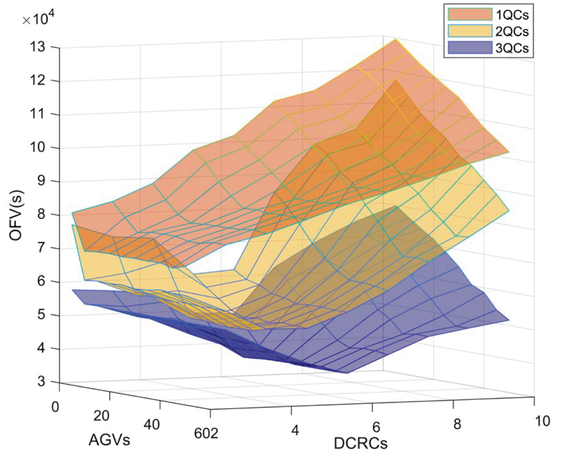


Figure 13. OFV(s).

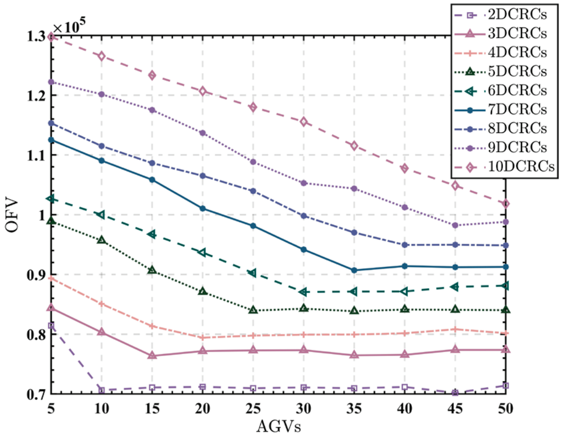


Figure 14. OFV at 1 QC.

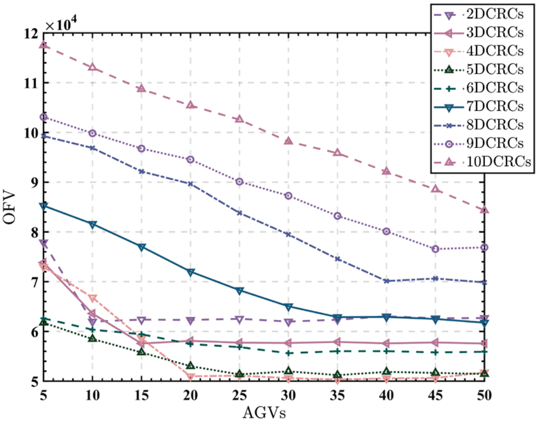


Figure 15. OFV at 2 QCs.

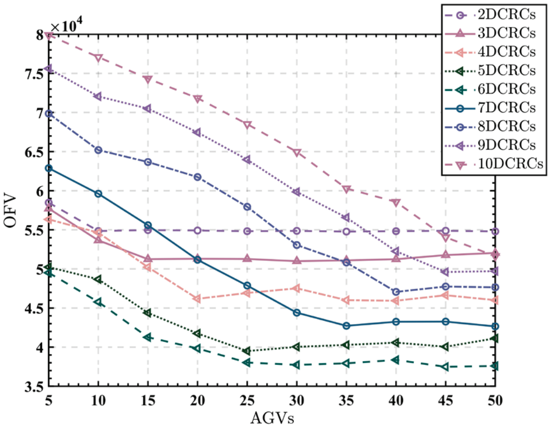


Figure 16. OFV at 3 QCs.

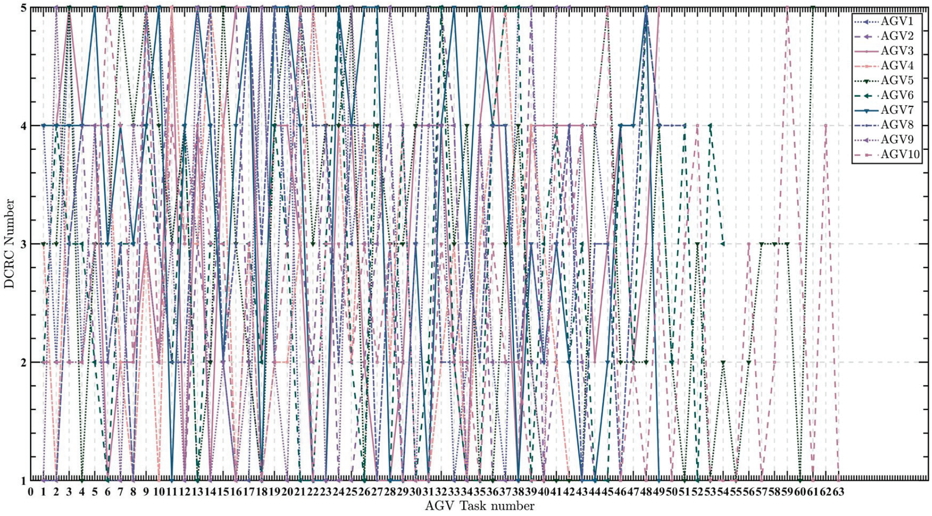


Figure 17. AGV job sequence for 500 container tasks.

Figure 12 is plotted from the CPUT(s) in Tables 3–5. The aim is to analyze the effect of the number of QCs on the required calculation time. As can be seen from the figure, the images are divided into three layers corresponding to the computation time using one to three QCs, respectively. The computation time for the case using one QC is distributed at the bottom of the image and ranges from 10.61 s to 30.31 s. The computation time for the case using three QCs is distributed at the top of the graph and ranges from 18.57 s to 59.09 s. This means that the complexity of the case and the computation time increase with the number of QCs.

Figure 13 is plotted from the OFV(s) in Tables 3–5. The aim is to analyze the effect of the number of QCs on the total running time of all DCRCs. It can be seen that the image is divided into three layers according to the number of QCs. The uppermost layer is the total DCRC running time for one QC, ranging from 70,193 to 129,741 s. The bottom layer is the total DCRC running time for three QCs, which ranges from 39,495 s to 72,052 s. It can be concluded that the higher the number of QCs, the lower the total running time of the DCRC.

Figures 14–16 show the effect of varying the number of DCRCs and AGVs on the total running time of the DCRC at different numbers of QCs. The horizontal coordinate is the number of AGVs, and the vertical coordinate is the OFV. Lines with different colors correspond to the use of different numbers of DCRCs.

Allocating the correct number of DCRCs for the vessels is one of the issues that must be considered by the terminal staff, which will directly affect the overall operating costs and efficiency. As shown in Figure 14, the total running time using two DCRCs is located at the bottom of the diagram when one QC is used. As the number of DCRCs increases, their total running time gradually increases until it reaches a maximum with 10 DCRCs. The case of using two QCs is shown in Figure 15. When the number of DCRCs is increased, the total running time of the DCRCs first decreases—the line corresponding to using four DCRCs is at the bottom of Figure 15, where the result is optimal. Then, as the number of DCRCs increases, the total running time of the DCRC increases rather than decreases. The same situation occurs when using three QCs, as shown in Figure 16. The total running time of the DCRC first decreases as the number of DCRCs increases, until it reaches a minimum when six DCRCs are used, and then gradually increases. It is worth noting that the shortest total running time of the double cantilever rail cranes is achieved when the number of DCRCs and QCs is 2:1 rather than more.

The AGV is an essential part of the transport chain on the quay. If the number of AGVs is too small, it cannot meet the working needs of the horizontal transportation of the quay, and if the number of AGVs is too large, it will cause a problem of wasted resources. As can be seen from Figures 14–16, the total running time of the DCRC gradually decreases as the number of AGVs increases, but the two are not exactly positively correlated. When the number of QCs is fixed, the minimum total running time of the DCRC is always generated around a specific number of AGVs and then plateaus. For example, when the number of QCs is one, the shortest total running time of the DCRC is generated around the AGV number of 10. When the number of QCs is two, the shortest total running time of the DCRC is generated at around 20 for the AGV. When the number of QCs is three, the shortest total running time of the DCRC is generated at around 30 for the AGV. It can be seen that the number of AGVs and QCs should be kept at around 10:1, when the total running time of the DCRC is the shortest and the number of AGVs required is the least.

Figure 17 shows the optimal scheduling scheme for 500 container tasks using one QC, ten AGVs, and five DCRCs. The horizontal axis represents the tasks of the AGVs, and the vertical axis represents the corresponding DCRCs. This figure is a concrete manifestation of the case study in our paper, which visually demonstrates the tasks of each device and further proves the feasibility of our model and algorithm.

The above analysis shows that the calculation time required for scheduling increases with the number of tasks, QCs and DCRCs. The total running time of the DCRC generally decreases as the number of QCs, DCRCs, and AGVs increases. However, when the number of QCs is fixed, the number of DCRCs and AGVs is not as high as it could be. When the number of both reaches a certain level, adding more equipment will not effectively reduce the total running time of the DCRC but may increase it. According to the above analysis, we can conclude that, in the U-shaped ACT loading and unloading process, a QC equipped with two DCRCs and ten AGVs will have a total running time of DCRCs. At the same time, the impact of the change in the number of QCs and DCRCs on the total running time of the DCRCs is greater than the impact of the change in the number of AGVs. Therefore, the allocation of resources for QCs and DCRCs should be considered first in the actual operation of U-shaped ACTs.

6. Conclusions

To improve the operational efficiency of the U-shaped ACT and to fit its actual operational characteristics, this paper first optimizes the working process of U-shaped ACTs. In U-shaped ACTs, the movement of the DCRC during loading and unloading is described in more detail by grouping the 16 working conditions into four operating modes based on how the DCRC works. Secondly, to minimize the total running time of all DCRCs, a refined collaborative dispatching model for AGVs and DCRCs, considering the entry of ECTs into the yard, was developed based on these four operating modes. This improves the accuracy of the modeling. The model is then solved using the ACGA to verify the model's correctness and the algorithm's practicality. Finally, experimental simulations were carried out to derive the optimum ratio of each equipment in different environments. The results show that, in the U-shaped ACT, the optimal ratio of the QC to the DCRC and the AGV is 1:2 and 1:10, respectively, where the total running time of DCRCs is the shortest. Simultaneously, the number of QCs and DCRCs has a more significant impact on terminal efficiency than the AGVs, and priority should be given to the allocation of both. This addresses the realities of U-shaped ACTs, and provides a reference for the construction.

However, the problem of multi-device scheduling in U-shaped ACTs is a highly complex one, and this study still has several shortcomings that need to be addressed in future research to improve and refine its findings.

Firstly, this study focused solely on efficiency-oriented collaborative scheduling, neglecting to consider energy consumption issues. However, energy consumption plays a vital role in practical applications, and it is crucial to consider optimizing device collaborative scheduling to strike a balance between economic and environmental factors.

Secondly, this study did not include the QC, which is another critical device with complex coupling relationships with AGVs. Future research should, therefore, incorporate the QC into the scheduling to achieve more comprehensive and refined results.

Thirdly, the mutual interference between multiple YCs has not been thoroughly investigated, which may significantly affect the overall efficiency of the ACT. As such, this area should be explored in more depth to achieve more efficient and stable ACT operations.

Furthermore, while this study primarily focuses on scheduling issues, it fails to address the path planning of AGVs, which is a vital component that requires the comprehensive consideration of various factors, such as time, distance, vehicle speed, and safety, to achieve optimal results.

Lastly, the study did not consider the charging issue of AGVs, which is essential in practical operations. To achieve a long-term and efficient operation of AGVs, the charging issue should also be considered while optimizing scheduling. It is, therefore, suggested that future research should include these factors in their modeling efforts to develop more realistic and effective solutions.

Author Contributions: Conceptualization, resources, review and editing, supervision, funding acquisition, project administration, Y.Y.; methodology, software, validation, formal analysis, writing, S.S.; validation, J.F.; review and editing, M.Z.; resources, F.W. and H.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (Grant No. 61540045), Scientific Research Program of Shanghai Science and Technology Commission (Grant No. 19595810700).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are included within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Yang, Y.; Zhong, M.; Dessouky, Y.; Postolache, O. An integrated scheduling method for AGV routing in automated container terminals. *Comput. Ind. Eng.* **2018**, *126*, 482–493. [CrossRef]
2. Yu, H.; Huang, M.; He, J.; Tan, C. The clustering strategy for stacks allocation in automated container terminals. *Marit. Policy Manag.* **2022**, 1–16. [CrossRef]
3. Bierwirth, C.; Meisel, F. A survey of berth allocation and quay crane scheduling problems in container terminals. *Eur. J. Oper. Res.* **2010**, *202*, 615–627. [CrossRef]
4. Wang, N.; Chang, D.; Shi, X.; Yuan, J.; Gao, Y. Analysis and design of typical automated container terminals layout considering carbon emissions. *Sustain* **2019**, *11*, 2957. [CrossRef]
5. Lee, B.K.; Lee, L.H.; Chew, E.P. Analysis on high throughput layout of container yards. *Int. J. Prod. Res.* **2018**, *56*, 5345–5364. [CrossRef]
6. Liu, C.I.; Jula, H.; Vukadinovic, K.; Ioannou, P. Automated guided vehicle system for two container yard layouts. *Transp. Res. Part C Emerg. Technol.* **2004**, *12*, 349–368. [CrossRef]
7. Roy, D.; Gupta, A.; Parhi, S.; De Koster, M.B.M. Optimal Stack Layout in a Sea Container Terminal with Automated Lifting Vehicles. *SSRN Electron. J.* **2017**, *55*, 3747–3765. [CrossRef]
8. Xu, B.; Jie, D.; Li, J.; Yang, Y.; Wen, F.; Song, H. Integrated scheduling optimization of U-shaped automated container terminal under loading and unloading mode. *Comput. Ind. Eng.* **2021**, *162*, 107695. [CrossRef]
9. Gharehgozli, A.; Zaerpour, N.; de Koster, R. Container terminal layout design: Transition and future. *Marit. Econ. Logist.* **2020**, *22*, 610–639. [CrossRef]
10. Li, J.; Yang, J.; Xu, B.; Yang, Y.; Wen, F.; Song, H. Hybrid scheduling for multi-equipment at u-shape trafficked automated terminal based on chaos particle swarm optimization. *J. Mar. Sci. Eng.* **2021**, *9*, 1080. [CrossRef]
11. Niu, Y.; Yu, F.; Yao, H.; Yang, Y. Multi-equipment coordinated scheduling strategy of U-shaped automated container terminal considering energy consumption. *Comput. Ind. Eng.* **2022**, *174*, 108804. [CrossRef]
12. He, J.; Chang, D.; Mi, W.; Yan, W. A hybrid parallel genetic algorithm for yard crane scheduling. *Transp. Res. Part E Logist. Transp. Rev.* **2010**, *46*, 136–155. [CrossRef]
13. Chang, D.; Jiang, Z.; Yan, W.; He, J. Developing a dynamic rolling-horizon decision strategy for yard crane scheduling. *Adv. Eng. Inform.* **2011**, *25*, 485–494. [CrossRef]
14. He, J.; Huang, Y.; Yan, W. Yard crane scheduling in a container terminal for the trade-off between efficiency and energy consumption. *Adv. Eng. Inform.* **2015**, *29*, 59–75. [CrossRef]
15. Tan, C.; He, J.; Yu, H.; Tang, Y. Modelling the problem of yard crane deployment for resource-limited container terminal. *Int. J. Ind. Syst. Eng.* **2019**, *32*, 137–169. [CrossRef]
16. He, J.; Tan, C.; Zhang, Y. Yard crane scheduling problem in a container terminal considering risk caused by uncertainty. *Adv. Eng. Inform.* **2019**, *39*, 14–24. [CrossRef]
17. Liu, W.; Zhu, X.; Wang, L.; Yan, B.; Zhang, X. Optimization Approach for Yard Crane Scheduling Problem with Uncertain Parameters in Container Terminals. *J. Adv. Transp.* **2021**, *2021*, 1–15. [CrossRef]
18. Ng, W.C. Crane scheduling in container yards with inter-crane interference. *Eur. J. Oper. Res.* **2005**, *164*, 64–78. [CrossRef]
19. Wu, Y.; Li, W.; Petering, M.E.; Goh, M.; Souza, R.D. Scheduling multiple yard cranes with crane interference and safety distance requirement. *Transp. Sci.* **2015**, *49*, 990–1005. [CrossRef]
20. Zheng, F.; Man, X.; Chu, F.; Liu, M.; Chu, C. Two yard crane scheduling with dynamic processing time and interference. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 3775–3784. [CrossRef]
21. Fatemi Ghomi, M.S.; Javanshir, H.; Ganji, S.S.; Fatemi Ghomi, S.M.T. Optimisation of yard crane scheduling considering velocity coefficient and preventive maintenance. *Int. J. Shipp. Transp. Logist.* **2014**, *6*, 88–108. [CrossRef]
22. Nossack, J.; Briskorn, D.; Pesch, E. Container dispatching and conflict-free yard crane routing in an automated container terminal. *Transp. Sci.* **2018**, *52*, 1059–1076. [CrossRef]
23. Eilken, A. A decomposition-based approach to the scheduling of identical automated yard cranes at container terminals. *J. Sched.* **2019**, *22*, 517–541. [CrossRef]
24. Kim, K.H.; Lee, K.M.; Hwang, H. A study on multi-ASC scheduling method of automated container terminals based on graph theory. *Comput. Ind. Eng.* **2019**, *129*, 404–416. [CrossRef]

25. Kim, K.H.; Lee, K.M.; Hwang, H. Sequencing delivery and receiving operations for yard cranes in port container terminals. *Int. J. Prod. Econ.* **2003**, *84*, 283–292. [CrossRef]
26. Huang, Y.; Liang, C.; Yang, Y. The optimum route problem by genetic algorithm for loading/unloading of yard crane. *Comput. Ind. Eng.* **2009**, *56*, 993–1001. [CrossRef]
27. Lee, B.K.; Kim, K.H. Comparison and evaluation of various cycle-time models for yard cranes in container terminals. *Int. J. Prod. Econ.* **2010**, *126*, 350–360. [CrossRef]
28. Bhatnagar, R.; Chandra, P.; Goyal, S.K. Models for multi-plant coordination. *Eur. J. Oper. Res.* **1993**, *67*, 141–160. [CrossRef]
29. Chen, X.; He, S.; Zhang, Y.; Tong, L.C.; Shang, P.; Zhou, X. Yard crane and AGV scheduling in automated container terminal: A multi-robot task allocation framework. *Transp. Res. Part C Emerg. Technol.* **2020**, *114*, 241–271. [CrossRef]
30. Zhao, Q.; Ji, S.; Zhao, W.; De, X. A Multilayer Genetic Algorithm for Automated Guided Vehicles and Dual Automated Yard Cranes Coordinated Scheduling. *Math. Probl. Eng.* **2020**, *2020*, 1–13. [CrossRef]
31. Zhang, Q.; Hu, W.; Duan, J.; Qin, J. Cooperative Scheduling of AGV and ASC in Automation Container Terminal Relay Operation Mode. *Math. Probl. Eng.* **2021**, *2021*, 1–18. [CrossRef]
32. Yang, X.M.; Jiang, X.J. Yard Crane Scheduling in the Ground Trolley-Based Automated Container Terminal. *Asia-Pac. J. Oper. Res.* **2020**, *37*, 2050007. [CrossRef]
33. Hsu, H.P.; Tai, H.H.; Wang, C.N.; Chou, C.C. Scheduling of collaborative operations of yard cranes and yard trucks for export containers using hybrid approaches. *Adv. Eng. Inform.* **2021**, *48*, 101292. [CrossRef]
34. Zhou, C.; Lee, B.K.; Li, H. Integrated optimization on yard crane scheduling and vehicle positioning at container yards. *Transp. Res. Part E Logist. Transp. Rev.* **2020**, *138*, 101966. [CrossRef]
35. Roy, D.; de Koster, R. Stochastic modeling of unloading and loading operations at a container terminal using automated lifting vehicles. *Eur. J. Oper. Res.* **2018**, *266*, 895–910. [CrossRef]
36. Li, H.; Peng, J.; Wang, X.; Wan, J. Integrated Resource Assignment and Scheduling Optimization with Limited Critical Equipment Constraints at an Automated Container Terminal. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 7607–7618. [CrossRef]
37. Lu, Y.; Le, M. The integrated optimization of container terminal scheduling with uncertain factors. *Comput. Ind. Eng.* **2014**, *75*, 209–216. [CrossRef]
38. He, J.; Huang, Y.; Yan, W.; Wang, S. Integrated internal truck, yard crane and quay crane scheduling in a container terminal considering energy consumption. *Expert Syst. Appl.* **2015**, *42*, 2464–2487. [CrossRef]
39. Ahmed, E.; El-Abbasy, M.S.; Zayed, T.; Alfalah, G.; Alkass, S. Synchronized scheduling model for container terminals using simulated double-cycling strategy. *Comput. Ind. Eng.* **2021**, *154*, 107118. [CrossRef]
40. Zhang, X.; Zeng, Q.; Yang, Z. Optimization of truck appointments in container terminals. *Marit. Econ. Logist.* **2019**, *21*, 125–145. [CrossRef]
41. Chen, X.; Wu, X.; Prasad, D.K.; Wu, B.; Postolache, O.; Yang, Y. Pixel-Wise Ship Identification From Maritime Images via a Semantic Segmentation Model. *IEEE Sens. J.* **2022**, *22*, 18180–18191. [CrossRef]
42. Chen, X.; Liu, S.; Liu, R.W.; Wu, H.; Han, B.; Zhao, J. Quantifying Arctic oil spilling event risk by integrating an analytic network process and a fuzzy comprehensive evaluation model. *Ocean Coast. Manag.* **2022**, *228*, 106326. [CrossRef]
43. Li, X.; Peng, Y.; Huang, J.; Wang, W.; Song, X. Simulation study on terminal layout in automated container terminals from efficiency, economic and environment perspectives. *Ocean Coast. Manag.* **2021**, *213*, 105882. [CrossRef]
44. Lau, H.Y.; Zhao, Y. Integrated scheduling of handling equipment at automated container terminals. *Int. J. Prod. Econ.* **2008**, *112*, 665–682. [CrossRef]
45. Kaveshgar, N.; Huynh, N.; Rahimian, S.K. An efficient genetic algorithm for solving the quay crane scheduling problem. *Expert Syst. Appl.* **2012**, *39*, 13108–13117. [CrossRef]
46. Deng, Z.; Fan, J.; Shi, Y.; Shen, W. A Coevolutionary algorithm for cooperative platoon formation of connected and automated vehicles. *IEEE Trans. Veh. Technol.* **2022**, *71*, 12461–12474. [CrossRef]
47. Valenzuela-Alcaraz, V.M.; Cosio-Leon, M.A.; Romero-Ocaño, A.D.; Brizuela, C.A. A cooperative coevolutionary algorithm approach to the no-wait job shop scheduling problem. *Expert Syst. Appl.* **2022**, *194*, 116498. [CrossRef]
48. Zhang, X.; Ma, Z.; Ding, B.; Fang, W.; Qian, P. A coevolutionary algorithm based on the auxiliary population for constrained large-scale multi-objective supply chain network. *Math. Biosci. Eng.* **2022**, *19*, 271–286. [CrossRef] [PubMed]
49. So, C.; Ho, I.M.; Chae, J.S.; Hong, K.H. PWR core loading pattern optimization with adaptive genetic algorithm. *Ann. Nucl. energy* **2021**, *159*, 108331. [CrossRef]

Journal of
*Marine Science
and Engineering*

MDPI

Article

A Novel Intelligent Ship Detection Method Based on Attention Mechanism Feature Enhancement

Yingdong Ye, Rong Zhen *, Zheping Shao, Jiacai Pan and Yubing Lin

Navigation College, Jimei University, Xiamen 361021, China
* Correspondence: zrandsea@163.com

Abstract: The intelligent perception ability of the close-range navigation environment is the basis of autonomous decision-making and control of unmanned ships. In order to realize real-time perception of the close-range environment of unmanned ships, an enhanced attention mechanism YOLOv4 (EA-YOLOv4) algorithm is proposed. First of all, on the basis of YOLOv4, the convolutional block attention module (CBAM) is used to search for features in channel and space dimensions, respectively, to improve the model's feature perception of ship targets. Then, the improved-efficient intersection over union (EIoU) loss function is used to replace the complete intersection over union (CIoU) loss function of the YOLOv4 algorithm to improve the algorithm's perception of ships of different sizes. Finally, in the post-processing of algorithm prediction, soft non-maximum suppression (Soft-NMS) is used to replace the non-maximum suppression (NMS) of YOLOv4 to reduce the missed detection of overlapping ships without affecting the efficiency. The proposed method is verified on the large data set SeaShips, and the average accuracy rate of $\mathrm{mAP}^{0.5\text{–}0.95}$ reaches 72.5%, which is 10.7% higher than the original network YOLOv4, and the FPS is 38 frames/s, which effectively improves the ship detection accuracy while ensuring real-time performance.

Keywords: water transportation; target detection; unmanned ship; deep learning; attention mechanism

Citation: Ye, Y.; Zhen, R.; Shao, Z.; Pan, J.; Lin, Y. A Novel Intelligent Ship Detection Method Based on Attention Mechanism Feature Enhancement. *J. Mar. Sci. Eng.* **2023**, *11*, 625. https://doi.org/10.3390/jmse11030625

Academic Editor: Claudio Ferrari

Received: 23 February 2023
Revised: 12 March 2023
Accepted: 14 March 2023
Published: 16 March 2023

1. Introduction

With the development of artificial intelligence and unmanned driving technology, unmanned ships have become an important research field in intelligent maritime transportation. The development of unmanned ships includes four stages: perception, understanding, decision-making, and control [1–4], among which perception is the basis for autonomous decision-making and control. Traditional environmental perception methods are affected by timeliness and cannot meet the requirements of real-time and accuracy of unmanned ships. Constructing a visible light detection system composed of a panoramic vision system can enable real-time and intuitive monitoring of dynamic water environments [5–7]. Furthermore, using computer vision and deep learning methods to detect and track ships in the system can effectively improve the efficiency of unmanned ship environmental perception. Therefore, the panoramic vision-based visible light ship detection method has received wide attention from scholars [3].

Currently, ship target detection under visible light can be divided into two categories: traditional feature-based methods and deep learning-based methods using convolutional neural networks. Traditional methods mostly rely on modeling based on target shape, background saliency contrast, and other factors. The algorithm depends on given parameters that satisfy specific regional conditions, which limits its accuracy and generalization ability in complex scenes [8]. In recent years, with the development of deep learning technology, many scholars have begun to use deep learning frameworks to recognize maritime targets. Deep learning detection methods can be classified into two categories: one-stage and two-stage methods [9–11]. Two-stage methods divide the detection process into two stages: first, generating a series of regions where targets may exist, and then searching and identifying targets within these regions. For example, Liu et al. [12] modified the region extraction network in Fast-RCNN [13] to ResNet-101 [14] and combined the global and local features of proposed regions using the multi-region feature fusion module to improve detection performance of ships in different directions. Yu et al. [15] improved the feature weighting method in Faster-RCNN to provide more suitable feature fusion for the background differentiation of difficult ships in busy waters. In the far shore scene with low discrimination between complex background and ship target, ship target can be recognized. Two-stage methods have high accuracy in detecting ships, but these methods take longer time and cannot achieve real-time detection. Therefore, to improve detection speed, many scholars use one-stage methods to improve ship detection efficiency. These methods use regression analysis principles to obtain the position and category of the target in a single detection. For example, Liu et al. [16] redesigned ship anchor box size based on YOLOv3 [17], introduced soft non-maximum suppression, and reconstructed mixed loss functions to improve the network's learning and expression ability for ship features. Hong et al. [18] used a residual network instead of continuous convolution operation in YOLOv4 to solve the problem of network degradation and gradient disappearance, and established a nonlinear target tracking model based on UKF method, which improved the accuracy of ship detection.

Although the deep learning method has made some progress, it still cannot solve the following problems well in practical use, such as: (1) distant ship targets are smaller and are often obstructed, (2) the background areas typically contain floating objects, shorelines, and other interferences, leading to occurrences of false positives and false negatives, and (3) video images captured under fixed monitoring can be easily affected by adverse weather conditions, such as heavy fog, rain, or snow, further increasing the difficulty of ship detection. To effectively address the aforementioned problems and further improve the accuracy of ship detection using deep learning algorithms, this paper proposes a ship detection algorithm EA-YOLOv4 based on attention mechanism feature enhancement. The main contributions of the proposed algorithm are as follows:

(1) In order to improve the accuracy of ship type identification and the recall of ships in bad weather, a CBAM module is embedded in YOLOv4 structure. The spatial attention mechanism and channel attention mechanism in this module change the search weight of YOLOv4, making the network structure focus on the unique characteristics and effective channels of ships.
(2) In terms of loss function, EIoU is used to replace the CIoU of YOLOv4. CIoU performs well in general target detection, but ship targets have relatively fixed aspect ratio characteristics. The use of EIoU can better identify the ship position, speed up the algorithm fitting, and improve the ship positioning ability.
(3) Overlapping of ship targets at sea is common. In order to improve the detection ability of occluded targets, Soft-NMS is used to replace the NMS of YOLOv4 to post-process the algorithm output, improve the network's attention to overlapping targets, and further improve the ship detection performance while ensuring the solution efficiency.

The rest of this article is organized as follows. In Section 2, the algorithm EA-YOLOv4 in this paper is proposed, including the CBAM to change the network structure, the EIoU loss function replacement and the improved Soft-NMS. In Section 3, all the improved algorithms are ablated, and the best set of results are selected as the algorithm EA-YOLOv4 in this paper. EA-YOLOv4 is compared with other similar algorithms, and the advantages of the algorithm are analyzed. Section 4 presents the conclusions and future research directions.

2. EA-YOLOv4 Algorithm

In this section, we will provide a detailed introduction to the algorithm from four aspects: algorithm overview, multi-dimensional attention mechanism feature enhancement extraction, loss function improvement, and non-maximum suppression improvement.

2.1. Algorithm Overview

The algorithm proposed in this paper is illustrated in Figure 1, with YOLOv4 [19] selected as the backbone network. To enhance the network's ability to perceive the scale features of ships of different sizes, a spatial attention mechanism is adopted to increase the network's focus on ship scale features. In order to avoid the network ignoring different dimensional channel information, a combination of channel attention mechanisms is inserted to form the CBAM [20] structure, obtaining $f_3 - f_5$ as the feature representation. Next, in order to improve the detection position optimization problem of traditional CIoU [21] loss function, which only focuses on aspect ratio, an EIoU [22] loss function is employed to increase the network's attention to ship size and enhance its perception of ships of different sizes. Finally, to enhance the detection ability of overlapping targets, an improved Soft-NMS [23] is used to perform secondary screening on the output results, while ensuring detection speed and accuracy. The CBAM, Soft-NMS, and EIoU loss will be discussed in detail in subsequent sections.

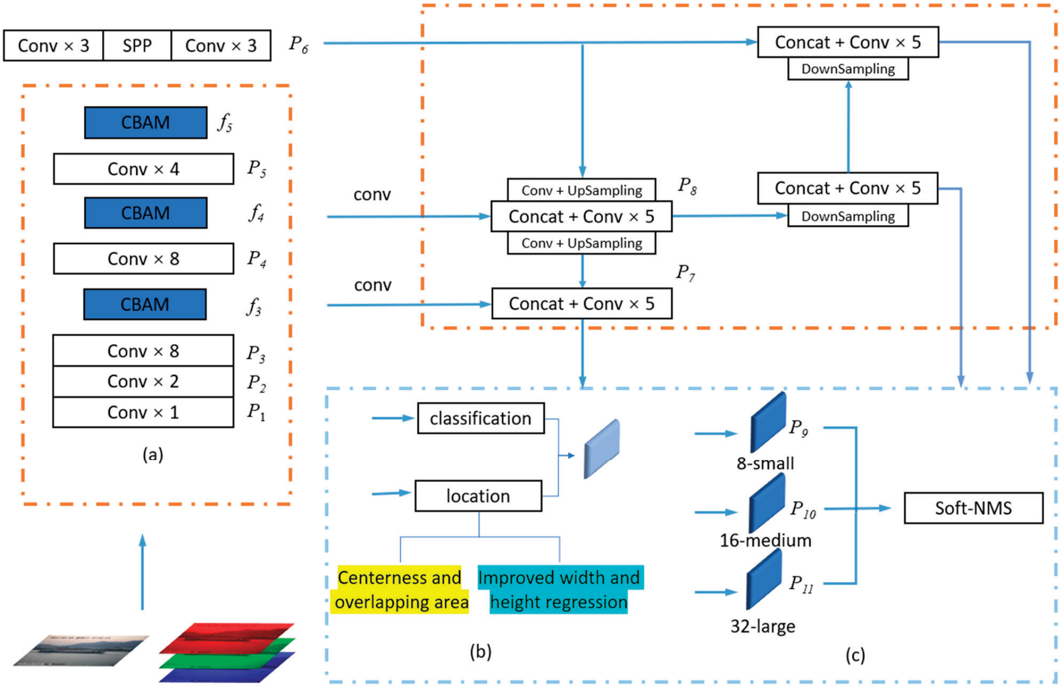


Figure 1. Algorithm structure diagram of EA-YOLOv4; (**a**) improvement of feature extraction network; (**b**) improvement of loss function; (**c**) improvement of loss function.

2.2. Multi-Dimensional Attention Mechanisms and Their Feature Enhancement

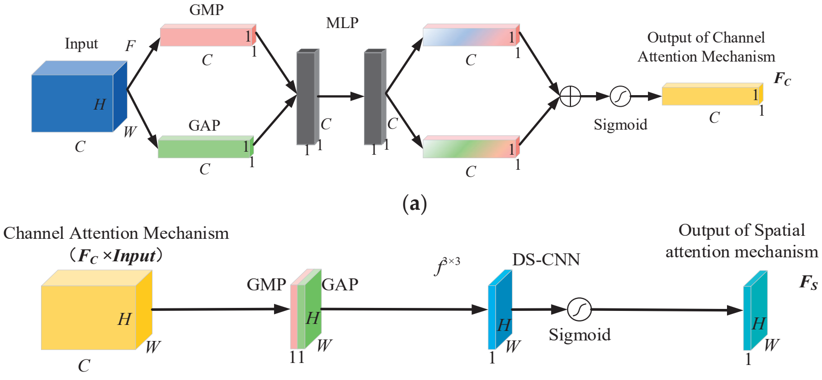
To enhance the feature representation capability of ships in complex backgrounds and improve their salience, this paper inserts attention mechanisms for feature selection in the network skeleton. Hu et al. [24] proposed using channel attention mechanisms to strengthen network feature extraction capability, but using only channel attention mechanisms ignores the spatial semantic information of ships. Since different ships have obvious differences in spatial representation, searching for ship spatial information can help the algorithm to quickly and accurately determine the ship type. M et al. [25] introduced a new learnable module, spatial transformer, which enables the neural network to actively transform the feature map according to the feature map itself, but this algorithm did not consider the semantic information of feature map channels, which resulted in too many computing resources being occupied by irrelevant channels that negatively affected detection results. The above research shows that the use of both spatial and feature attention mechanisms can enhance the algorithm's recognition capability. To comprehensively consider ship spatial information and feature map channel information, this paper introduces CBAM as an embedded module of multi-dimensional attention mechanisms to improve the feature extraction network's extraction capability.

As shown in Figure 2, the schematic diagram of CBAM includes a channel attention mechanism in Figure 2a and a spatial attention mechanism in Figure 2b. The CBAM attention mechanism combines the advantages of channel and spatial attention mechanisms. A CBAM unit takes any tensor $X = [x_1, x_2, \ldots, x_c] \in R^{C\times H\times W}$ as input and outputs a tensor $Y = [y_1, y_2, \ldots, y_c] \in R^{C\times H\times W}$ of the same size. In order to make CBAM focus on both channel and spatial information, channel attention mechanism is first implemented on the input feature map as follows:

$$F_c = \sigma\{MLP[GAP(X)] + MLP[GMP(X)]\} \tag{1}$$

where X is the input feature map, GAP is the global average pooling, GMP is the global maximum pooling, MLP is the multi-layer perceptron, and σ is the activation function defined as:

$$\sigma = \frac{1}{1+e^{-x}} \tag{2}$$



(**a**)

(**b**)

Figure 2. Multi-dimensional attention mechanism CBAM; (**a**) channel attention mechanism; (**b**) spatial attention mechanism.

After the channel attention, a vector $F_c = (\alpha_1, \alpha_2, \ldots, \alpha_c) \in R^{C\times 1\times 1}$ of dimension C is obtained, where $\alpha_i, i = 1, 2, \ldots, c$ represents the allocation weights of different channels in the original feature map. When the detection platform encounters situations such as heavy fog or water surface reflection, attention tracking on channels helps the algorithm focus on effective information and reduce the impact of noise on recognition results.

The spatial attention mechanism is added after the channel attention, as shown in Figure 2b, and is implemented as follows:

$$F_s = \sigma\left\{ f^{3\times 3}\left[\begin{array}{c} GAP(Y_1) \\ GMP(Y_1) \end{array} \right] \right\} \tag{3}$$

where Y_1 comes from channel attention mechanism output Y_c dot product with the original feature map X, which reflects the channel region of interest in the original feature map.

$f^{3\times3}$ is the convolutional layer with 3×3 kernel size, and F_s is the weight matrix obtained after the spatial attention mechanism, whose expression is:

$$F_s = \begin{pmatrix} \beta_{1,1} & \beta_{1,2} & \cdots & \beta_{1,w} \\ \beta_{2,1} & \beta_{2,2} & \cdots & \beta_{2,w} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{H,1} & \beta_{H,2} & \cdots & \beta_{H,W} \end{pmatrix} \tag{4}$$

where $\beta_{i,j}, i = 1,2,\ldots,H; j = 1,2,\ldots,W$ represents the allocation weights of spatial coordinates, which will be adaptively adjusted according to the search target. Finally, the matrix is multiplied by Y_1 to obtain the final output $Y = [y_1, y_2, \ldots, y_c] \in R^{C\times H\times W}$. The output can simultaneously resample the spatial structure information of ships on the basis of focusing on channel information.

2.3. Optimizing the Loss Function

The loss function affects the convergence speed of the model and the fitting performance of the evaluated algorithm. In the original YOLOv4 network, the bounding box regression loss is calculated using the CIoU function, as shown in Equation (5):

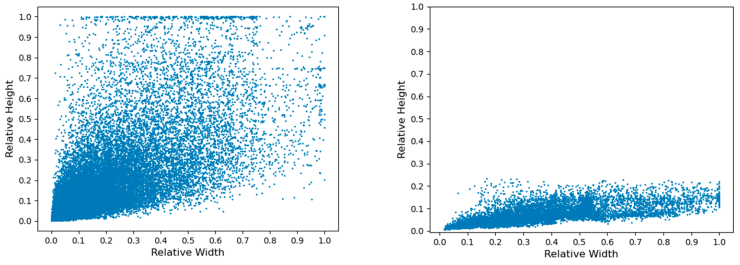
$$L_{CIoU} = 1 - CIoU = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha\frac{4}{\pi^2}(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h})^2 \tag{5}$$

In the equation, b, b^{gt} denotes the predicted box and the reference box, where the reference box is the true target used to guide the model learning. IoU represents the intersection over union, which reflects the overlapping area between the two boxes. $\rho^2(b, b^{gt})$ represents the Euclidean distance between the two boxes.c is the diagonal distance between the minimum enclosing region containing the two boxes. α is a correction parameter. $\frac{w^{gt}}{h^{gt}}$ and $\frac{w}{h}$ represent the aspect ratios of the reference boxes and predicted boxes, respectively.

On the basis of IoU, CIoU adds center point distance detection and predicted box width and height detection, making the relative position between the predicted box and the reference box more accurate. However, compared with conventional detection tasks, ship detection in complex marine environments involves many small targets, and the shapes of marine targets are relatively fixed. Therefore, the aspect ratio used by CIoU is not suitable for the bounding box regression loss in ship target detection. In order to analyze the differences between ship targets and traditional targets and optimize the loss function, this paper compares the objects in the CoCo2017test (CoCo) dataset [26] and the SeaShips dataset [27]. The data is processed using linear normalization, and the results are shown in Figure 3. The results show that the aspect ratio distribution of the objects in the CoCo dataset is relatively balanced, while the aspect ratio distribution of the objects in the SeaShips dataset is close to a linear function, and the aspect ratios of the SeaShips dataset objects are relatively close. Therefore, the aspect ratio used by CIoU cannot accurately locate the ship's position. To improve this, this paper introduces EIoU, and its formula is given by Equation (6):

$$L_{EIoU} = 1 - EIoU = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \frac{\rho^2(w, w^{gt})}{C_w^2} + \frac{\rho^2(h, h^{gt})}{C_h^2} \tag{6}$$

where b, b^{gt} denotes the predicted box and the reference box, $\rho(w, w^{gt})$ is the difference in width between the two boxes, $\rho(h, h^{gt})$ is the difference in height between the two boxes, and C_w, C_h is the width and height of the minimum outer box that covers the two boxes. When the EIoU value is larger, it means that the predicted box is closer to the position of the reference box, and the network loss is smaller. During network training, the network parameters are adjusted based on the overlap area, center point, and width and height values to make the network converge faster.



(**a**) (**b**)

Figure 3. Relative shape of targets under different data sets; (**a**) CoCo dataset target box relative value; (**b**) relative value of SeaShips dataset target box.

2.4. Improvement of Soft-Non-Maximum Suppression

In object detection algorithms, NMS is used to filter out overlapping detection boxes for the same object. The specific processing method is as follows:

$$s_i = \begin{cases} s_i & IoU(M, b_i) < N_t \\ 0 & IoU(M, b_i) > N_t \end{cases} \tag{7}$$

where M is the box with the highest score, IoU represents the degree of overlap between the highest-scoring box and other candidate boxes, and N_t is the threshold. NMS removes detection boxes within the threshold of the highest-scoring box, but the problem is that it will force high-scoring detection boxes adjacent to the target box to be removed, resulting in missed detection of occluded ship targets with high confidence. In order to balance the problem of repeated detection of targets and missed detection of occluded ship targets, Soft-NMS is used instead of traditional NMS for post-processing. The Soft-NMS processing method is shown in the equation below:

$$s_i = \begin{cases} s_i & IoU(M, b_i) < N_t \\ s_i[1 - IoU(M, b_i)] & IoU(M, b_i) > N_t \end{cases} \tag{8}$$

Soft-NMS simultaneously considers the score and overlap degree, and sets a penalty term for the occluded ship targets with higher scores to avoid missed detection while also ensuring that the same target is not detected repeatedly. As shown in Figure 4a, the two ships to be detected are a cargo ship and an occluded target ore ship, and the IoU is about 0.45. When using traditional NMS algorithm, the cargo ship and ore ship detection boxes are too close, causing the algorithm to consider the detected cargo ship and ore ship as the same target, thus directly ignoring the cargo ship. Figure 4b uses Soft-NMS to process the same target, and when the IoU of the cargo ship and ore ship is greater than given threshold, the score of the smaller cargo ship is reduced by such IoU. At this time, the algorithm considers the probability of the existence of a cargo ship here to be 0.53, which meet the set threshold and the occluded target is detected.

(**a**) (**b**)

Figure 4. Processing results of overlapping targets by NMS and Soft-NMS; (**a**) NMS processing mode; (**b**) Soft-NMS processing mode.

3. Experiment and Analysis

3.1. Experimental Platform and Dataset

The experimental environment in this article is based on the Windows platform, with an i7-10700F CPU, 32GB of memory, and an NVIDIA® GeForce® RTX 2070 Super GPU processor with 8GB of video memory. The experimental framework is built using the Python programming language. The deep learning development environment includes Python 3.7.8, PyCharm2019, Anaconda3.4.1, TensorFlow-GPU 2.3.0, CUDA 10.1.234, and cuDNN 7.6.5.

The SeaShips dataset was used to validate the ship detection algorithm. All images in the dataset come from approximately 1080 real video clips, including 7000 ship images of six categories. The six categories of images are ore carrier (OC), fishing boats (FB), container ships (CS), bulk carrier (BC), general cargo ship (GCS), and passenger ships (PS). The notable challenges include small targets, high overlap between coastline and ships, changes in brightness, and ship occlusion. The data set is randomly divided into training set, verification set and test set according to 2:1:1 ratio. That is, 3500 pictures in the training set, 1750 pictures in the verification set and 1750 pictures in the test set. The validation set is used to test the model training results. When the loss of the model on the verification set does not decrease for 20 consecutive rounds, the optimal solution is considered to be reached. The distribution of ship numbers in each set is shown in Figure 5. The proportion of the three datasets is consistent across different ship targets, ensuring that the algorithm has good robustness.

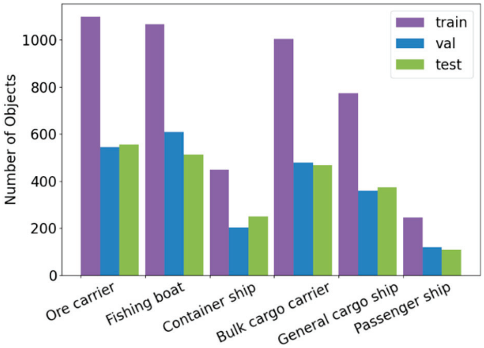


Figure 5. Distribution of six categories of ships in training set, verification set and test set.

3.2. Hyperparameters and Evaluation Metrics

The hyperparameters set for the experiment are shown in Table 1, where SGD refers to stochastic gradient descent. Before training, various data augmentation strategies were employed, such as horizontal rotation, mosaic, image cropping, and adjustments to brightness and contrast, to expand the training set. During training, all epochs were saved, and the epoch with the best performance on the validation set was selected as the test model for the current model. In the testing phase, the target confidence threshold and Soft-NMS threshold were both set to 0.5. The maximum training epoch is set to 300.

Table 1. Experiment parameter setting.

Hyperparameter	Value
Epoch	300
Batch size	4
Optimizer	SGD
First learning rate	1×10^{-4}
Last learning rate	1×10^{-5}
Learning rate decay rate	0.1
Weight decay	5×10^{-4}

To fully verify the ship detection capability of the model in complex scenes, the mean average precision (mAP) was used to evaluate the algorithm, which reflects the mean of AP under different IoU settings. At the same time, the frames per second (FPS) was used to measure the detection speed of the algorithm, which indicates the number of images processed per second. The AP can be expressed as:

$$mAP = \frac{1}{N}\frac{1}{10}\sum_{c=1}^{N}\sum_{IoU=0.5}^{0.95} AP_c^{IOU} \tag{9}$$

where N is the number of ship categories, AP_c^{IOU} is the AP for a given IoU threshold for a specific ship category, and mAP is the mean AP of the algorithm at 0.5–0.95 IoU thresholds, reflecting the recognition performance of the algorithm under different configurations. Specifically, the AP for a specific category can be expressed as:

$$AP = \int_0^1 p(r)dr \tag{10}$$

where $p(r)$ is the precision-recall curve.

3.3. Experiment

3.3.1. Ablation Study

In order to verify the best location for adding CBAM, this paper embedded CBAM in the backbone, FPN feature pyramid, and head, respectively, and the three embedding methods were represented by C1, C2, and C3. C1, C2, and C3 were used for recognition of six types of ships, and the results are shown in Table 2.

Table 2. Test results of three different embedding methods.

Algorithm	**AP$^{0.5–0.95}$/%**						**mAP$^{0.5–0.95}$/%**	**FPS**
	OC	**FB**	**CS**	**BC**	**GCS**	**PS**		
YOLOv4	58.7	56.9	67.6	60.7	64.7	62.0	61.8	45
C1	66.1	62.7	71.1	66.8	70.7	67.6	67.5	40
C2	64.3	62.1	72.4	67.3	69.4	66.3	66.9	42
C3	61.5	60.8	70.3	66.0	68.9	65.7	65.5	43

As can be seen from the results, the addition of CBAM structure in the backbone, FPN, or head has little effect on the overall recognition speed of the algorithm. The C1 algorithm, which adds CBAM to the backbone, improves the recognition accuracy due to its use of shallow features. Among them, the highest detection accuracy was achieved for ore ships, fishing boats, general cargo ships, and passenger ships. The C2 algorithm, which adds CBAM to the FPN, performed well in recognizing container ships and bulk cargo ships. The C3 algorithm, which adds CBAM to the head, did not achieve ideal results because the high-dimensional semantic information of the head feature map was disrupted by the addition of CBAM, which destroyed the original network representation of high-dimensional semantic information. Overall, the C1 algorithm performed the best with the highest average precision of 67.5, which was 5.7 percentage points higher than the original algorithm without CBAM, and the FPS was only 5 percentage points lower than the original algorithm.

In order to further investigate the differences in target representation resulting from adding the CBAM algorithm in different positions, the gradient-weighted class activation mapping (grad-CAM) method [28] was used to analyze three different attention embedding methods. The images were related to three common challenges in detection: foggy weather, small targets, and occluded ships, as shown in Figure 6. The heat map shows the algorithm's focus region on the image. By comparing the visualized feature maps of the improved networks, it can be observed that: (1) C1 has stronger recognition ability for small targets. In the scene of recognizing small targets, C1 has a larger interested area for small targets, and a higher response to the main target area, indicating that C1 can recognize smaller targets well. (2) C1 enhances recognition ability in foggy weather. Under the foggy conditions, C1's response to the target region is more concentrated, while C2, C3 and the original network have a large amount of dispersed semantics, which means that the channel attention mechanism used in the backbone network avoids the image noise generated by foggy weather, enabling the algorithm to focus on ship features. (3) C1 has stronger recognition ability for overlapping targets. In the highly overlapping target recognition scene, C2, C3, and the original network have severe feature confusion, while C1 can accurately identify two different ship categories. This indicates that C1 has a higher focus on the target contour around it, and retains a large amount of ship's original spatial semantic information in the low-level feature map of the backbone, which enables the spatial attention mechanism to recognize and filter this information, improving the network's recognition accuracy for different categories of ships.

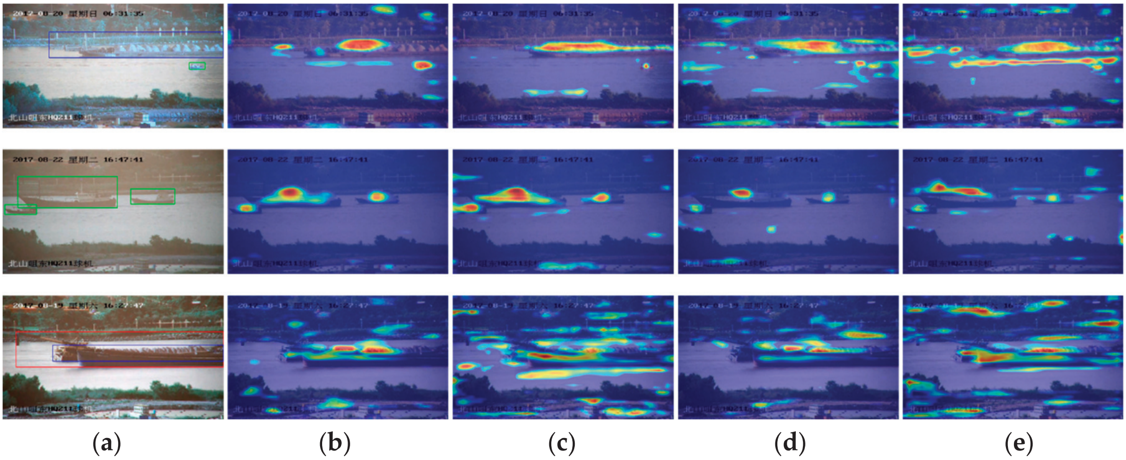


Figure 6. Grad-CAM renderings under three conditions; (**a**) original drawing; (**b**) C1; (**c**) C2; (**d**) C3; (**e**) not embedded.

Therefore, in different situations, the C1 algorithm with CBAM added to the backbone can improve the network's recognition ability to a certain extent. In the subsequent comparative experiments, this paper uses C1 as the basic framework, and tests the performance with the addition of Soft-NMS and the use of the improved EIoU loss function, as shown in Table 3. Comparing the two improved algorithms, the Soft-NMS algorithm (C1-1) improves the overall mAP by 2.5 percentage points, mainly due to the presence of many overlapping targets in the navigational ship images. The use of the EIoU improvement algorithm (C1-2) increases the overall mAP by 3.5 percentage points, indicating that correcting the aspect ratio of the target box separately can effectively determine the ship's position and improve the ship recognition accuracy. Therefore, this paper combines the two (C1-3) to improve the algorithm, and the final mAP reaches 72.5, which is an improvement of 10.7 percentage points compared to the original algorithm.

Table 3. Ablation experiment.

Algorithm	Soft-NMS	EIoU	$AP^{0.5–0.95}$/%	FPS
C1			67.5	40
C1-1	√		70.0	40
C1-2		√	70.7	38
C1-3 (this paper)	√	√	72.5	38

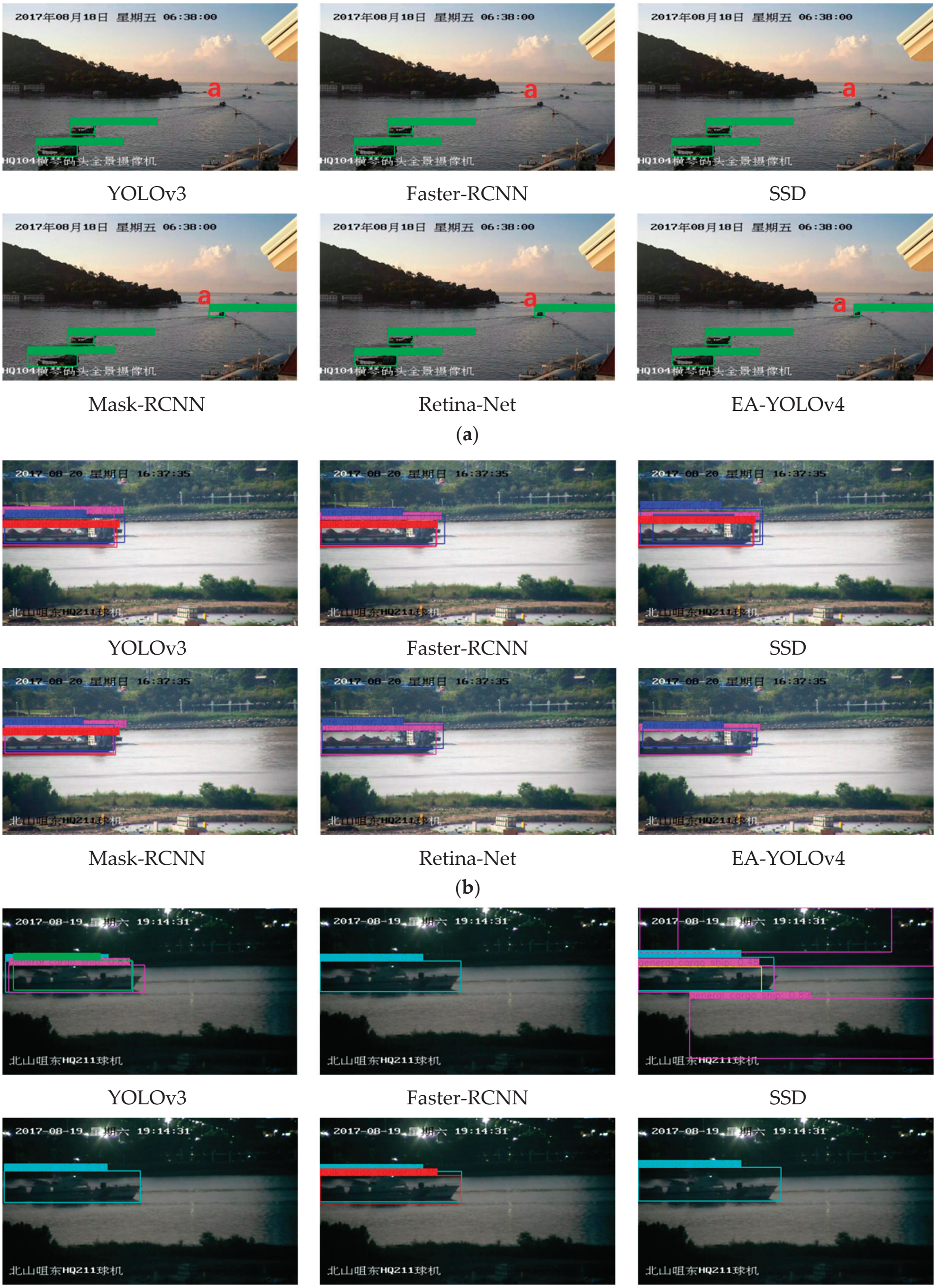
3.3.2. Comparative Experiments

To further demonstrate the superiority of the proposed algorithm, this paper conducted comparative experiments between the deep learning algorithms commonly used in object detection and EA-YOLOv4, including YOLOv3, SSD [29], Retina-Net [30] for one-stage methods, and Faster RCNN, Mask-RCNN [31] for two-stage methods. The experimental results are shown in Table 4. It shows that the lowest mAP came from YOLOv3, which is only 57.3. Faster-RCNN has an AP of 70.2% for container ships, but only 55.7% for fishing boats. The highest FPS came from SSD, which is 78, and the corresponding AP for passenger ships is 70.4%, which is only 1.2% lower than that of proposed algorithm, but its detection performance for other ships is unacceptable. Mask-RCNN has a relatively average detection performance for six types of ships. The overall performance of the proposed algorithm is better, and the detection performance for container ships, general cargo ships, and passenger ships is more prominent, with AP values of 75.7, 74.8, and 72.6, respectively. At the same time, the FPS of the proposed algorithm is 38, which can meet the real-time detection requirements for maritime ship targets.

Table 4. Comparison results between EA-YOLOv4 and other mainstream algorithms.

Algorithm	$AP^{0.5–0.95}$/%						$mAP^{0.5–0.95}$/%	FPS
	OC	FB	CS	BC	GCS	PS		
YOLOv3	52.7	57.1	59.6	51.8	60	62.3	57.3	37
SSD	63.5	60.7	68.5	61.4	61.8	70.4	64.4	78
RetinaNet	63.4	63.5	70.1	66.5	64.9	67.3	66.0	11
Faster-RCNN	64.1	55.7	70.2	57.8	63.7	61	62.1	5
Mask-RCNN	64.5	60.8	69.3	67.0	68.8	67.5	66.3	7
EA-YOLOv4 (this paper)	71.7	68.1	75.7	71.9	74.8	72.6	72.5	38

The detection results of different algorithms under various conditions are shown in Figure 7. It can be found that the scale of the ship in Figure 7a is small. YOLOv3, Faster-RCNN, and SSD cannot capture the features of small fishing boats well. EA-YOLOv4 improves the loss algorithm by using EIoU, which allows it to pay attention to the structural characteristics of small targets, thus improving the recognition ability for small targets. In Figure 7b, there are overlapping targets, namely a bulk carrier and a cargo ship. YOLOv3, Faster-RCNN, SSD, and Mask-RCNN all have prediction errors due to the unique shapes of bulk carriers and cargo ships. The CBAM mechanism embedded in EA-YOLOv4 has good feature representation capability for ship spatial features, and the introduced Soft-NMS mechanism prevents overlapping targets from being incorrectly removed. Therefore, EA-YOLOv4 has more accurate localization of overlapping ships. In Figure 7c, the scene has low brightness. YOLOv3, SSD, and Retina-Net in the one-stage methods cannot extract ship targets well. However, the CBAM mechanism of EA-YOLOv4 retains the high-value channels that contain ship information, enabling it to accurately search for ship targets in low brightness environments, achieving ship detection under low light conditions.



YOLOv3 Faster-RCNN SSD

Mask-RCNN Retina-Net EA-YOLOv4

(**a**)

YOLOv3 Faster-RCNN SSD

Mask-RCNN Retina-Net EA-YOLOv4

(**b**)

YOLOv3 Faster-RCNN SSD

Mask-RCNN Retina-Net EA-YOLOv4

(**c**)

Figure 7. Ship detection results of different algorithms; (**a**) small size target scene; (**b**) multi-target overlapping scene (ore ship and general cargo ship); (**c**) underlit scene.

4. Conclusions

To improve the real-time perception capability of unmanned ships in close-range environments and further enhance the positioning accuracy and recognition accuracy of ship detection algorithms, this paper proposes an EA-YOLOv4 algorithm based on YOLOv4. By using the multidimensional attention mechanism CBAM to improve the algorithm network framework, the extraction of shape features and suppression of background interference are achieved. The improved EIoU loss function is used to enhance the algorithm's perception ability for ships of different scales, accelerate network convergence, and enhance the algorithm's detection ability for small targets. Soft-NMS is used for post-processing of detection results to find missed targets from overlapping ships. The experiment evaluates the influence of all improvements on EA-YOLOv4 in terms of the MSCOCO $AP^{0.5–0.95}$ index and frame rate, and compares it with other similar algorithms. The results show that EA-YOLOv4 performs best among all algorithms. Without significantly affecting the detection speed, EA-YOLOv4 has improved the accuracy of the identification of six types of ships in the real-time monitoring of the nearshore, and can be used as the technical basis for unmanned ship detection and environmental identification. At present, due to the limitation of data sets, this study is limited to ship detection in some severe weather. In the future, we will collect optical ship images under more conditions, increase the types of sea targets that can be detected, and continue to optimize the model, while ensuring the detection speed of the algorithm can realize the full target perception of the unmanned ship navigation environment.

Author Contributions: Conceptualization, Y.Y., R.Z. and J.P.; data curation, Y.Y. and R.Z.; investigation, R.Z., Y.Y. and Y.L.; writing—original draft, Y.Y., R.Z. and J.P.; methodology, R.Z., Y.Y. and Z.S.; writing—review and editing, Y.Y., R.Z. and J.P.; supervision, R.Z. and Z.S.; visualization, Y.Y. and Y.L. funding acquisition, R.Z. and Z.S. All authors have read and agreed to the published version of the manuscript.

Funding: The research described in this paper is supported by the National Natural Science Foundation of China (No. 52001134) and Fuzhou-Xiamen-Quanzhou Independent Innovation Region Cooperated Special Foundation (No: 3502ZCQXT2021007).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Yan, X.P.; Wang, S.W.; Ma, F. Review and prospect for intelligent cargo ships. *Chin. J. Ship Res.* **2021**, *16*, 1–6.
2. Xiao, G.N.; Cui, W.Y. Evolutionary game between government and shipping companies based on shipping cycle and carbon quota. *Front. Mar. Sci.* **2023**, *10*, 1132174. [CrossRef]
3. Chen, X.Q.; Wu, S.B.; Shi, C.J.; Huang, Y.G.; Yang, Y.S.; Ke, R.M.; Zhao, J.S. Sensing Data Supported Traffic Flow Prediction via Denoising Schemes and ANN: A Comparison. *IEEE Sens. J.* **2020**, *20*, 14317–14328. [CrossRef]
4. Liu, Z.; Zhang, B.Y.; Zhang, M.Y.; Wang, H.L.; Fu, X.J. A quantitative method for the analysis of ship collision risk using AIS data. *Ocean Eng.* **2023**, *272*, 113906. [CrossRef]
5. Chen, X.Q.; Liu, S.H.; Liu, R.W.; Wu, H.F.; Han, B.; Zhao, J.S. Quantifying Arctic oil spilling event risk by integrating an analytic network process and a fuzzy comprehensive evaluation model. *Ocean Coast. Manag.* **2022**, *228*, 106326. [CrossRef]
6. Zhen, R.; Shi, Z.Q.; Liu, J.L.; Shao, Z.P. A novel arena-based regional collision risk assessment method of multi-ship encounter situation in complex waters. *Ocean Eng.* **2022**, *246*, 110531. [CrossRef]
7. Zhen, R.; Shi, Z.Q.; Shao, Z.P.; Liu, J.L. A novel regional collision risk assessment method considering aggregation density under multi-ship encounter situations. *J. Navig.* **2022**, *75*, 76–94. [CrossRef]
8. Liu, D.; Zhang, Y.; Zhao, Y.; Shi, Z.G.; Zhang, J.H.; Zhang, Y. Multi-Scale Inshore Ship Detection Based on Feature Re-Focusing Network. *Acta Opt. Sinica.* **2021**, *41*, 137–149.
9. Shao, Z.F.; Wang, L.G.; Wang, Z.Y.; Du, W.; Wu, W.J. Saliency-aware convolution neural network for ship detection in sur-veillance video. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *30*, 781–794. [CrossRef]

10. Zhen, R.; Ye, Y.; Chen, X.; Xu, L. A Novel Intelligent Detection Algorithm of Aids to Navigation Based on Improved YOLOv4. *J. Mar. Sci. Eng.* **2023**, *11*, 452. [CrossRef]
11. LA, T. YOLOv4-5D: An enhancement of YOLOv4 for autonomous driving. *Towards Data Sci.* **2021**. Available online: https://towardsdatascience.com/yolov4-5d-an-enhancement-of-yolov4-for-autonomous-driving-2827a566be4a (accessed on 3 February 2023).
12. Liu, Q.W.; Xiang, X.Q.; Yang, Z.; Hu, Y.; Hong, Y.M. Arbitrary Direction Ship Detection in Remote-Sensing Images Based on Multitask Learning and Multiregion Feature Fusion. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 1553–1564. [CrossRef]
13. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 11–18 December 2015; pp. 1440–1448.
14. He, K.M.; Zhang, X.Y.; Ren, S.Q.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 27–30 June 2016; pp. 770–778.
15. Yu, M.; Han, S.J.; Wang, T.F.; Wang, H.Y. An approach to accurate ship image recognition in a complex maritime transportation environment. *J. Marine Sci. Eng.* **2022**, *10*, 1903. [CrossRef]
16. Liu, R.W.; Yuan, W.; Chen, X.; Lu, Y. An enhanced CNN-enabled learning method for promoting ship detection in maritime surveillance system. *Ocean Eng.* **2021**, *235*, 109435. [CrossRef]
17. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804. 02767.
18. Hong, X.; Cui, B.; Chen, W.; Rao, Y.; Chen, Y. Research on Multi-Ship Target Detection and Tracking Method Based on Camera in Complex Scenes. *J. Mar. Sci. Eng.* **2022**, *10*, 978. [CrossRef]
19. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
20. Woo, S.H.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; Volume 11211, pp. 3–19.
21. Zheng, Z.H.; Wang, P.; Liu, W.; Li, J.Z.; Ye, R.G.; Ren, D.W. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12993–13000.
22. Zhang, Y.-F.; Ren, W.Q.; Zhang, Z.; Jia, Z.; Wang, L.; Tan, T.N. Focal and Efficient IOU Loss for Accurate Bounding Box Regression. *Neurocomputing* **2022**, *506*, 146–157. [CrossRef]
23. Bodia, N.; Singh, B.; Chellappa, R.; Davis, L.S. Soft-NMS-improving object detection with one line of Code. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 5562–5570.
24. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks (CVPR). In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
25. Jaderberg, M.; Simonyan, K.; Zisserman, A.; Kavukcuoglu, K. Spatial Transformer Networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; Volume 28.
26. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollar, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. In Proceedings of the European Conference on Computer Vision (ECCV), Zurich, Switzerland, 6–12 September 2014; Volume 8693, pp. 740–755.
27. Shao, Z.; Wu, W.; Wang, Z.; Du, W.; Li, C. SeaShips: A Large-Scale Precisely Annotated Dataset for Ship Detection. *IEEE Trans. Multimedia* **2018**, *20*, 2593–2604. [CrossRef]
28. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *Int. J. Comput. Vis.* **2020**, *128*, 336–359. [CrossRef]
29. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *Lecture Notes in Computer Science: Computer Vision—ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer: Cham, Switzerland, 2016; Volume 9905, pp. 21–37.
30. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.M.; Dollar, P. Focal Loss for Dense Object Detection. In Proceedings of the International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2999–3007.
31. He, K.M.; Gkioxari, G.; Dollar, P.; Girshick, R. Mask R-CNN. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 386–397. [CrossRef] [PubMed]

Journal of
*Marine Science
and Engineering*

MDPI

Article

Research on Position Sensorless Control of RDT Motor Based on Improved SMO with Continuous Hyperbolic Tangent Function and Improved Feedforward PLL

Hongfen Bai [1,2,*], Bo Yu [1] and Wei Gu [1]

[1] Key Laboratory of Transport Industry of Marine Technology and Control Engineering, Shanghai Maritime University, Shanghai 201306, China
[2] School of Transportation and Logistics Engineering, Wuhan University of Technology, Wuhan 430063, China
* Correspondence: hfbai@shmtu.edu.cn

Abstract: With the increasing use of electric propulsion ships, the emergence of the shaftless rim-driven thruster (RDT) as a revolutionary integrated motor thruster is gradually becoming an important development direction for green ships. The shaftless structure of RDTs leads to their dependence on position sensorless control techniques. In this study, a novel control algorithm using a composite sliding mode observer (SMO) with a modified feed-forward phase-locked loop (PLL) is presented for achieving high accuracy position and speed control of shaftless RDT motors. The deviation between the observed and actual currents is exploited to develop a current SMO to extract back electromotive force (back-EMF) errors. On this basis, a back-EMF observer is established to achieve accurate estimation of the back-EMF. The basic structure of the PLL was modified and incorporates a speed feedforward mechanism, which enhances the performance of rotor position estimation and facilitates bidirectional rotation. The stability of the algorithm has been verified in Matlab/Simulink for a range of steady-state, dynamic, and ship propeller loading conditions. Remarkably, the control algorithm boasts an impressive adjustment time of approximately 0.006 s and its position estimation error may be as low as 0.03 rad. Simulation results highlight the performance of the algorithm to achieve bidirectional rotation, while exhibiting fast convergence, minimal vibration, exceptional control accuracy, and robustness.

Keywords: rim-driven thruster (RDT); electric power propulsion ship system; permanent magnet synchronous motor (PMSM); position-sensorless control; sliding mode observer (SMO); phase-locked loop (PLL)

Citation: Bai, H.; Yu, B.; Gu, W. Research on Position Sensorless Control of RDT Motor Based on Improved SMO with Continuous Hyperbolic Tangent Function and Improved Feedforward PLL. *J. Mar. Sci. Eng.* **2023**, *11*, 642. https://doi.org/10.3390/jmse11030642

Academic Editor: Rosemary Norman

Received: 9 February 2023
Revised: 7 March 2023
Accepted: 15 March 2023
Published: 17 March 2023

1. Introduction

Electric propulsion ships [1], underwater robots [2], unmanned surface vehicles [3,4], and other navigational vehicles [5] have undergone significant development in recent years, imposing more demands on propulsion systems. Electric propulsion systems are phasing out thermal propulsion systems due to their better control performance and higher control efficiency [6]. With the development of electric propulsion systems, integrated motors are increasingly being applied in a wide range of applications [7]. The shaftless rim-driven thruster (RDT) [8] is a revolutionary integrated motor thruster that merges propeller blades and motors to conserve cabin space, increase power density, motor efficiency, and hydrodynamic efficiency, as well as reduce system installation and manufacturing costs [9]. The introduction of the shaftless RDTs brings the electric propulsion ship systems more prominent features such as integration, electrification, intelligence, and flexibility [10]. As a typical complicated electromechanical system, the selection and control of the propulsion motor in the shaftless RDT are the most important considerations. Due to its great power density and efficiency, the permanent magnet synchronous motor (PMSM) is widely utilized in the field of propulsion motors [11]. The field-oriented control (FOC) [12] method of a PMSM requires precise real-time rotor position and speed information, which is commonly measured by mechanical position sensors installed on the rotor shaft, including shaft encoders, resolvers, and Hall sensors [13]. Because RDTs should operate in a harsh marine environment with high temperature, high humidity, and high salt for an extended period of time, shaftless RDTs cannot be equipped with conventional mechanical position sensors [14]. Simultaneously, removing the physical sensors from the control system can significantly enhance the system's dependability and minimize motor production costs and size [15]. Consequently, it is necessary to develop the control algorithm of PMSM without a position sensor.

There are two types of PMSM position sensorless control strategies [16]: Back electromotive force (EMF) based mid-to-high speed method and saliency-tracking based zero-to-low speed approach. The motor dynamic model is utilized to obtain the back-EMF [17], which provides information on the rotor position and speed. The Kalman filter method [18], model adaptation method [19], Luenberger state observer [20], and sliding mode observer (SMO) [21] are the typical back-EMF-based methods.

The SMO approach has been widely adopted for the sensorless control of PMSM, owing to its straightforward construction, robustness against disturbances, and insensitivity to parameter variations [22]. However, the inherent switching characteristic of SMO gives rise to control discontinuity within the system, leading to undesirable chattering phenomena that compromise control precision and pose a risk to device integrity. The conventional SMO methods utilize the sign function as the sliding mode switching function [23], which exacerbates the problem of chattering due to the discontinuous nature of the sign function. Moreover, the back-EMF estimation by conventional SMO approaches is often plagued by high-frequency noise. As a result, one or more low-pass filters (LPF) are typically employed to extract the back-EMF signal. However, the use of LPFs introduces phase delays, necessitating position compensation based on the actual corner frequency and LPF cutoff frequency [24]. In response to these challenges, various research endeavors have been undertaken to address these issues and enhance the performance of the control system. In Ref. [25], a position-sensorless control method for PMSM was proposed, which utilized a SMO based on the variable boundary layer sigmoid function. The width of the boundary layer was adjusted based on the velocity to effectively address the chattering issue. However, this method used a LPF, which resulted in a phase shift of the system state and reduced the estimation accuracy. To solve the phase delay problem. In Ref. [26], a cascade low-pass filter (LPF) with a variable cut-off frequency is proposed to mitigate chattering. In addition, a variable phase delay compensation is applied to counterbalance the phase shift introduced by the LPF across the speed range. Although the LPF with delay compensation strategy has demonstrated its effectiveness in improving the performance of PMSMs, its topological complexity poses challenges for practical implementation. In Ref. [27], the authors proposed a new approach for rotor position estimation in permanent magnet synchronous motors (PMSMs) using a hyperbolic function-based SMO. Unlike conventional SMOs, the proposed approach eliminates the need for a low-pass filter (LPF) and angle compensator. The previous study proposed a method to reduce chattering in sliding mode observers (SMOs) by utilizing a continuous switching function to construct a transition boundary layer and generate a quasi-SMO. However, the thickness of the boundary layer had a significant impact on the performance of the SMO. A narrow boundary layer range improved the approach speed of the sliding mode motion, but introduced chattering and harmonic distortion in the back-EMF. Conversely, a wide boundary layer reduced chattering but weakened the robustness of the control system by decreasing the approximation speed. Thus, regulating the width of the boundary layer appropriately based on the continuous switching function was necessary to enhance the performance of the control system while minimizing the chattering.

Sensorless control algorithms traditionally employ the inverse tangent function to extract rotor position information from the back-EMF for motor speed calculation. However, the rotor position estimation approach that utilizes the inverse tangent function introduces high-frequency harmonics into the division operation, leading to an amplification of high-frequency harmonics that can cause significant errors in rotor position estimation. In Ref. [28], a current pre-compensation scheme based on a double sampling strategy with one switching cycle was proposed to improve the accuracy of position estimation and motor control performance by compensating for the computational delay. However, this technique significantly increased the computing complexity of the control system. To address this issue, many researchers have proposed using phase-locked loop (PLL) to extract rotor position and velocity information [29], which provides a simpler and more efficient alternative for sensorless control. PLL usually consists of three parts: phase detector (PD), loop filter (LF), and voltage-controlled oscillator (VCO). The PLL described in the preceding work was referred to the traditional PLL. The conventional phase-locked loop (PLL) has been shown to enhance the reliability of rotor position estimation. However, it is only appropriate for estimating position information when the motor moves in one direction with uniform parameters. Upon motor reversal, a position deviation of 180° arises, causing imprecise estimation of the system position. In Ref. [13], the authors introduced a position estimation technique for the sensorless control of a PMSM that utilized a PLL structure with a tangent function. This enhanced PLL provided reliable rotor position estimation during both forward and backward rotation, catering to the frequent forward and reverse RDT rotation requirements. However, due to its reliance on a tangent function, the technique was vulnerable to noise interference, leading to imprecise location estimation. An often overlooked problem with conventional phase-locked loops (PLLs) is their inability to quickly track rotor position during acceleration and deceleration. Additionally, maintaining a zero steady-state error is challenging, and sudden changes in velocity can result in significant position estimation errors, leading to tracking failures.

To address the aforementioned issues, a position-sensorless algorithm for PMSM is proposed in this study, based on a composite SMO and a feedforward PLL. The composite SMO comprises a modified current SMO and a back-EMF observer. The modified current SMO adopts a continuous hyperbolic tangent function to replace the discontinuous sign function, which reduces the SMO chatter. This SMO extracts the counter-electromotive force error, which is much smaller in amplitude than the counter-electromotive force, so a smaller SMO gain factor can be set to further weaken the chattering. The back-EMF observer is constructed based on the back-EMF model, which replaces the traditional LPF and avoids phase delay while greatly improving the observation accuracy of the back-EMF. The proposed PLL with rotor speed feedforward compensation enhances the tracking stability and position extraction accuracy, while maintaining the reliability of position extraction when the motor rotates in both directions. The stability of the composite SMO and PLL is verified through simulations under steady-state, dynamic, ship propeller load, and external random disturbance conditions. The experimental results demonstrate that the algorithm features fast convergence, small jitter, high control accuracy, and strong robustness compared with conventional algorithms. There are two main highlights in this paper:

(1) This study addresses SMO chattering by replacing the sign function with a continuous hyperbolic tangent function and selecting an appropriate boundary layer width. An observer based on the back-EMF model is developed to eliminate the LPF, reduce phase delay, and enhance the back-EMF signal estimation precision;
(2) Through optimization of the traditional PLL structure and the addition of feedforward compensation, this study has successfully realized position extraction during bi-directional rotation of the motor, while significantly improving the accuracy of position extraction during acceleration and deceleration.

The remainder of this paper is organized as follows: In Section 2, the mathematical model of PMSM is presented. Section 3 proposes the improved SMO and counter-electromotive force observer, and their stability is proven. Section 4 analyzes the limitations of the conventional PLL and presents the design of the improved feedforward PLL, along with a discussion of its stability. Simulation results under different conditions are presented in Section 5 to demonstrate the superiority and reliability of the proposed algorithm. Finally, Section 6 concludes the paper.

2. PMSM Mathematical Model

In the $d-q$ coordinate system, the mathematical model of the surface-mounted PMSM is shown in Equation (1), in which iron saturation, flux leakage, eddy current, and hysteresis losses can be ignored.

$$\begin{cases} u_d = R_s i_d + L_s \dfrac{di_d}{dt} - \omega_e L_s i_q \\ u_q = R_s i_q + L_s \dfrac{di_q}{dt} + \omega_e L_s i_d + \omega_e \psi_f \end{cases} \tag{1}$$

where u_d, u_q, i_d, i_q represent the voltage and current in the corresponding coordinate system, respectively. R_s, L_s represent the stator resistance and stator inductance, respectively. ψ_f represents the permanent magnet flux linkage, and ω_e represents the electrical angular velocity. Equation (1) can be converted to a stationary $\alpha-\beta$ coordinate system using the coordinate transformation as follows.

$$\begin{cases} u_\alpha = R_s i_\alpha + L_s \dfrac{di_\alpha}{dt} + e_\alpha \\ u_\beta = R_s i_\beta + L_s \dfrac{di_\beta}{dt} + e_\beta \end{cases} \tag{2}$$

where u_α, u_β, i_α, i_β, e_α, e_β represent the voltage, current, and back-EMF in the corresponding coordinate system, respectively. The back-EMF is described as

$$\begin{cases} e_\alpha = -\psi_f \omega_e \sin(\theta_e) \\ e_\beta = \psi_f \omega_e \cos(\theta_e) \end{cases} \tag{3}$$

where θ_e is the electrical angle and the ω_e can be expressed as

$$\omega_e = \frac{1}{\psi_f}\sqrt{(e_\alpha^2 + e_\beta^2)} \tag{4}$$

From Equation (3), it can be seen that the back-EMF contains information regarding the position and angular velocity of the motor rotor. The exact observation of the motor back-EMF is, thus, the foundation of the position-sensorless control algorithm.

3. SMO Design for PMSM Rotor Position Estimation

In this paper, based on the mathematical model of PMSM, an improved SMO is constructed based on the error between the actual current i_α, i_β and the observed current $\hat{i}_\alpha$, $\hat{i}_\beta$ to obtain the back-EMF errors $\tilde{e}_\alpha$, $\tilde{e}_\beta$. Next, the back-EMF observer is constructed based on the back-EMF model, and the back-EMF $\hat{e}_\alpha$, $\hat{e}_\beta$ is the output.

According to Equation (2), the expression for the current can be written as

$$\begin{cases} \dfrac{di_\alpha}{dt} = \dfrac{u_\alpha - R_s i_\alpha - e_\alpha}{L_s} \\ \dfrac{di_\beta}{dt} = \dfrac{u_\beta - R_s i_\beta - e_\beta}{L_s} \end{cases} \tag{5}$$

Define $\tilde{i}_\alpha$, $\tilde{i}_\beta$ as the current estimation error of SMO with the following expression:

$$\begin{cases} \tilde{i}_\alpha = \hat{i}_\alpha - i_\alpha \\ \tilde{i}_\beta = \hat{i}_\beta - i_\beta \end{cases} \tag{6}$$

where $\hat{i}_\alpha$, $\hat{i}_\beta$ are SMO estimated currents. Based on the theory of the sliding mode variable structure, the sliding surface is chosen as

$$s(\tilde{i}_s) = \begin{bmatrix} s_\alpha(\tilde{i}_\alpha) \\ s_\beta(\tilde{i}_\beta) \end{bmatrix} = \tilde{i}_s + \mu \int_0^t \tilde{i}_s d\tau \tag{7}$$

where $\tilde{i}_s = \begin{bmatrix} \tilde{i}_\alpha & \tilde{i}_\beta \end{bmatrix}^{\mathrm{T}}$, μ is a constant and satisfies $0 < \mu < R_s/L_s$.

According to the mathematical model of PMSM, the SMO is established as follows:

$$\begin{cases} \dfrac{d\hat{i}_\alpha}{dt} = \dfrac{u_\alpha - R_s\hat{i}_\alpha - \hat{e}_\alpha - \lambda \cdot F(S_\alpha(\tilde{i}_\alpha))}{L_s} \\ \dfrac{d\hat{i}_\beta}{dt} = \dfrac{u_\beta - R_s\hat{i}_\beta - \hat{e}_\beta - \lambda \cdot F(S_\beta(\tilde{i}_\beta))}{L_s} \end{cases} \tag{8}$$

where $F(\cdot)$ is the sliding mode switching function and λ is the sliding mode gain.

Due to the discontinuous character of the sign function, the back-EMF estimation generates significant chattering. In order to reduce the chattering phenomenon of SMO, the continuous hyperbolic tangent function is used instead of the sign function. The images of the sign function and hyperbolic tangent function are shown in Figure 1. The hyperbolic tangent function is demonstrated below:

$$F(s) = \frac{e^{hs} - e^{-hs}}{e^{hs} + e^{-hs}} \tag{9}$$

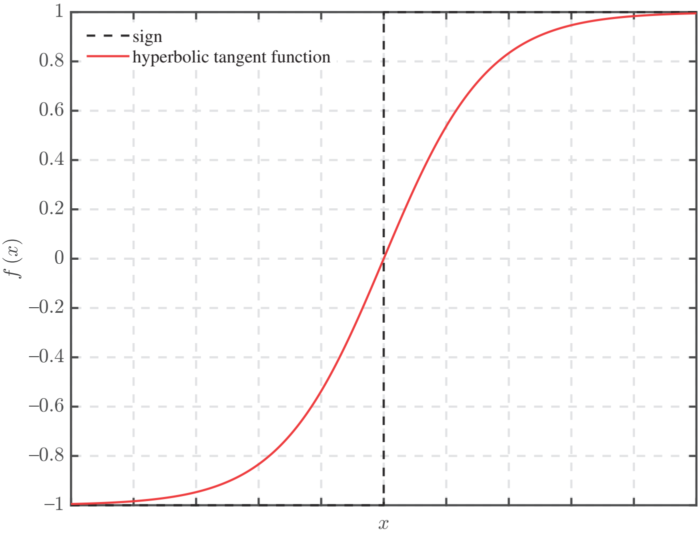


Figure 1. Sign function and hyperbolic tangent function.

Figure 2 depicts the hyperbolic tangent function with different values of h, which shows that the boundary layer width d is inversely proportional to the coefficient h. The magnitude of h is related to SMO chattering. The chattering of the state variable trajectory in SMO reduces as h decreases. When it increases, the approximation speed of the state variable improves, but if it is too large, SMO robustness is compromised. Therefore, the balance between the robustness of SMO and suppression of chattering needs to be considered when choosing h.

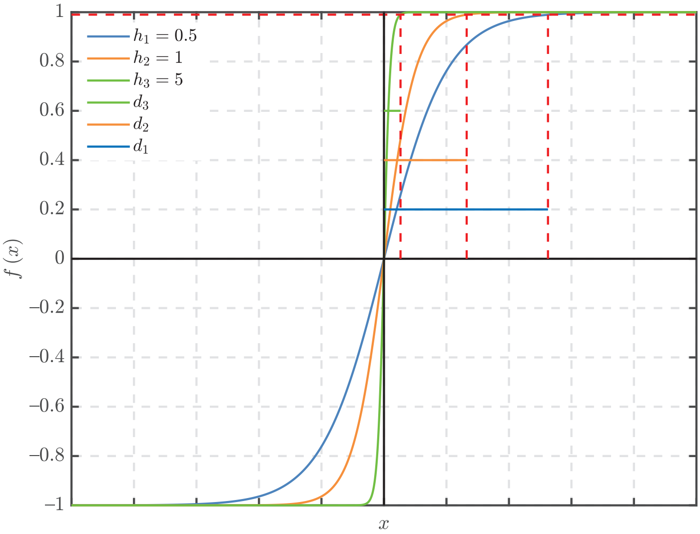


Figure 2. Hyperbolic tangent function for different h.

The current observation error equation can be obtained by subtracting Equation (5) from Equation (8),

$$\begin{cases} \dfrac{d\tilde{i}_\alpha}{dt} = \dfrac{-R_s\tilde{i}_\alpha - \tilde{e}_\alpha - \lambda \cdot F(S_\alpha(\tilde{i}_\alpha))}{L_s} \\ \dfrac{d\tilde{i}_\beta}{dt} = \dfrac{-R_s\tilde{i}_\beta - \tilde{e}_\beta - \lambda \cdot F(S_\beta(\tilde{i}_\beta))}{L_s} \end{cases} \tag{10}$$

According to the sliding mode equivalence principle, we may derive:

$$\begin{cases} \tilde{e}_\alpha = -\lambda \cdot F(S_\alpha(\tilde{i}_\alpha)) + (\mu L_s - R_s)\tilde{i}_\alpha \\ \tilde{e}_\beta = -\lambda \cdot F(S_\beta(\tilde{i}_\beta)) + (\mu L_s - R_s)\tilde{i}_\beta \end{cases} \tag{11}$$

Theorem 1. *Consider the closed-loop system, which consists of Equation* (2) *with environmental disturbances. Assume that Equation* (8)*, Equation* (21)*, and PLL are stable. When the system parameters are set to satisfy* $0 < \mu < R_s/L_s$ *and* $\lambda > \max(|\tilde{e}_\alpha|, |\tilde{e}_\beta|)$*, the current observation error, back-EMF estimation error, position estimation error, and velocity estimation error converge asymptotically to zero.*

Proof of Theorem 1. The Lyapunov function is chosen as follows,

$$V(\tilde{i}_s) = \frac{1}{2}S(\tilde{i}_s)^T S(\tilde{i}_s) \tag{12}$$

The derivative of the above equation yields,

$$\begin{aligned} \dot{V}(\tilde{i}_s) &= S(\tilde{i}_s)^T\dot{S}(\tilde{i}_s) \\ &= S_\alpha(\tilde{i}_\alpha)\dot{S}_\alpha(\tilde{i}_\alpha) + S_\beta(\tilde{i}_\beta)\dot{S}_\beta(\tilde{i}_\beta) \\ &= S_\alpha(\tilde{i}_\alpha)[\{\dot{\tilde{i}}_\alpha + \mu\tilde{i}_\alpha] + S_\beta(\tilde{i}_\beta)[\{\dot{\tilde{i}}_\beta + \mu\tilde{i}_\beta] \\ &= \dot{V}_\alpha(\tilde{i}_\alpha) + \dot{V}_\beta(\tilde{i}_\beta) \end{aligned} \tag{13}$$

where $\dot{V}_\alpha(\tilde{i}_\alpha) = S_\alpha(\tilde{i}_\alpha)[\dot{\tilde{i}}_\alpha + \mu\tilde{i}_\alpha]$ and $\dot{V}_\beta(\tilde{i}_\beta) = S_\beta(\tilde{i}_\beta)[\dot{\tilde{i}}_\beta + \mu\tilde{i}_\beta]$.

According to $\dot{V}_\alpha(\tilde{i}_\alpha)$ and Equation (10) it can be derived that

$$\begin{aligned}\dot{V}_\alpha(\tilde{i}_\alpha) &= S_\alpha(\tilde{i}_\alpha)[\dot{\tilde{i}}_\alpha + \mu\tilde{i}_\alpha] \\ &= S_\alpha(\tilde{i}_\alpha)[-\frac{R_s}{L_s}\tilde{i}_\alpha - \frac{1}{L_s}\tilde{e}_\alpha - \frac{1}{L_s}\lambda \cdot F(S_\alpha(\tilde{i}_\alpha)) + \mu\tilde{i}_\alpha] \\ &= (\mu - \frac{R_s}{L_s})\tilde{i}_\alpha S_\alpha(\tilde{i}_\alpha) - \frac{S_\alpha(\tilde{i}_\alpha)}{L_s}(\tilde{e}_\alpha + \lambda \cdot F(S_\alpha(\tilde{i}_\alpha)))\end{aligned} \tag{14}$$

According to Equation (7) and $0 < \mu < R_s/L_s$ we can get the first term of the above equation as

$$(\mu - \frac{R_s}{L_s})\tilde{i}_\alpha S_\alpha(\tilde{i}_\alpha) = (\mu - \frac{R_s}{L_s})(\tilde{i}_\alpha^2 + \tilde{i}_\alpha\mu\int_0^t \tilde{i}_\alpha d\tau) \le 0 \tag{15}$$

When $\lambda > |\tilde{e}_\alpha|$, the second term of Equation (14) satisfies

$$-\frac{S_\alpha(\tilde{i}_\alpha)}{L_s}(\tilde{e}_\alpha + \lambda \cdot F(S_\alpha(\tilde{i}_\alpha))) < 0 \tag{16}$$

In accordance with the aforementioned derivation, we may obtain

$$\dot{V}_\alpha(\tilde{i}_\alpha) = (\mu - \frac{R_s}{L_s})\tilde{i}_\alpha S_\alpha(\tilde{i}_\alpha) - \frac{S_\alpha(\tilde{i}_\alpha)}{L_s}(\tilde{e}_\alpha + \lambda \cdot F(S_\alpha(\tilde{i}_\alpha))) < 0 \tag{17}$$

Likewise, when $\lambda > |\tilde{e}_\beta|$, we may derive

$$\dot{V}_\beta(\tilde{i}_\beta) = (\mu - \frac{R_s}{L_s})\tilde{i}_\beta S_\beta(\tilde{i}_\beta) - \frac{S_\beta(\tilde{i}_\beta)}{L_s}(\tilde{e}_\beta + \lambda \cdot F(S_\beta(\tilde{i}_\beta))) < 0 \tag{18}$$

Thus, when $\lambda > \max(|\tilde{e}_\alpha|, |\tilde{e}_\beta|)$, we obtain

$$\dot{V}(\tilde{i}_s) = \dot{V}_\alpha(\tilde{i}_\alpha) + \dot{V}_\beta(\tilde{i}_\beta) < 0 \tag{19}$$

This shows that the sliding mode surface as depicted in Equation (7) is reachable, therefore the SMO is stable and can accurately obtain the back-EMF error as described in Equation (11). The conventional SMO sliding mode gain needs to satisfy $\lambda > \max(|e_\alpha|, |e_\beta|)$, while the improved SMO only needs to satisfy $\lambda > \max(|\tilde{e}_\alpha|, |\tilde{e}_\beta|)$. Obviously, the back-EMF error is smaller than the back-EMF. Therefore, the sliding mode gain of the improved SMO setting is significantly smaller than that of the conventional SMO, which can weaken the sliding mode chattering. □

The aforementioned SMO is capable of generating an equivalent back-EMF error signal, but this signal still contains a high-frequency component and cannot be utilized directly to extract rotor position information. Most conventional methods use LPF for filtering, but this introduces a phase delay, which impacts the subsequent extraction of rotor position information. This work employs the back-EMF model to construct the observer, extract the back-EMF signal, and then use PLL to estimate the rotor position and speed. Since the change rate of motor speed is significantly smaller than that of the stator current, assuming $\dot{\omega}_e = 0$, the back-EMF model of PMSM can be expressed as

$$\begin{cases}\frac{de_\alpha}{dt} = -\psi_f\dot{\omega}_e\sin\theta_e - \psi_f\omega_e\cos\theta_e\frac{d\theta_e}{dt} = -\omega_e e_\beta \\ \frac{de_\beta}{dt} = \psi_f\dot{\omega}_e\cos\theta_e - \psi_f\omega_e\sin\theta_e\frac{d\theta_e}{dt} = \omega_e e_\alpha\end{cases} \tag{20}$$

Based on the above equation, the back-EMF observer is established as

$$\begin{cases} \dfrac{d\hat{e}_\alpha}{dt} = -\hat{\omega}_e\hat{e}_\beta - m\tilde{e}_\alpha \\ \dfrac{d\hat{e}_\beta}{dt} = \hat{\omega}_e\hat{e}_\alpha - m\tilde{e}_\beta \\ \dfrac{d\hat{\omega}_e}{dt} = \tilde{e}_\alpha\hat{e}_\beta - \tilde{e}_\beta\hat{e}_\alpha \end{cases} \tag{21}$$

where m is the observer gain, and $m > 0$. The error equation can be obtained by subtracting Equation (20) from Equation (21),

$$\begin{cases} \dfrac{d\tilde{e}_\alpha}{dt} = -\tilde{\omega}_e\hat{e}_\beta - \omega_e\tilde{e}_\beta - m\tilde{e}_\alpha \\ \dfrac{d\tilde{e}_\beta}{dt} = \tilde{\omega}_e\hat{e}_\alpha + \omega_e\tilde{e}_\alpha - m\tilde{e}_\beta \\ \dfrac{d\tilde{\omega}_e}{dt} = \tilde{e}_\alpha\hat{e}_\beta - \tilde{e}_\beta\hat{e}_\alpha \end{cases} \tag{22}$$

Theorem 2. *Considering that the change rate of motorspeed is significantly smaller than that of the stator current, assume that $\dot{\omega}_e = 0$. When the back-EMF observer is stable,the back-EMF error converges to zero and the back-EMF observation can be obtained accurately.*

Proof of Theorem 2. To verify the stability of the back-EMF observer, the Lyapunov function is chosen as

$$V = \frac{\tilde{e}_\alpha^2 + \tilde{e}_\beta^2 + \tilde{\omega}_e^2}{2} \tag{23}$$

Derivation of the above equation yields

$$\begin{aligned} \dot{V} &= \tilde{e}_\alpha\dot{\tilde{e}}_\alpha + \tilde{e}_\beta\dot{\tilde{e}}_\beta + \tilde{\omega}_e\dot{\tilde{\omega}}_e \\ &= \tilde{e}_\alpha(-\tilde{\omega}_e\hat{e}_\beta - \omega_e\tilde{e}_\beta - m\tilde{e}_\alpha) + \tilde{e}_\beta(\tilde{\omega}_e\hat{e}_\alpha + \omega_e\tilde{e}_\alpha - m\tilde{e}_\beta) + \tilde{\omega}_e(\tilde{e}_\alpha\hat{e}_\beta - \tilde{e}_\beta\hat{e}_\alpha) \\ &= -m(\tilde{e}_\alpha^2 + \tilde{e}_\beta^2) \\ &\leq 0 \end{aligned} \tag{24}$$

According to the above proof, the back-EMF observer is asymptotically stable, and its stability is solely dependent on the motor parameters and not external disturbances. □

4. Rotor Position and Speed Extraction

4.1. Traditional PLL Analysis

The PLL is significantly more stable than the arctangent function for extracting rotor position information from the back-EMF, whose structure and equivalent model are depicted in Figures 3 and 4, respectively. However, in the conventional PLL, once the controller parameters are determined, the motor only runs in a single direction and the position information cannot be estimated accurately if the parameters are not modified when the motor reverses in the other direction. In addition, when the motor is accelerating or decelerating, the estimation error is significant. In Figures 3 and 4, θ_e, $\hat{\theta}_e$ mean the rotor electrical angle and the estimated value of the electrical angle, respectively, Δe indicates the electrical angle estimation error, and k_p, k_i denote the proportional coefficient and integral coefficient of the PI in the PLL, respectively.

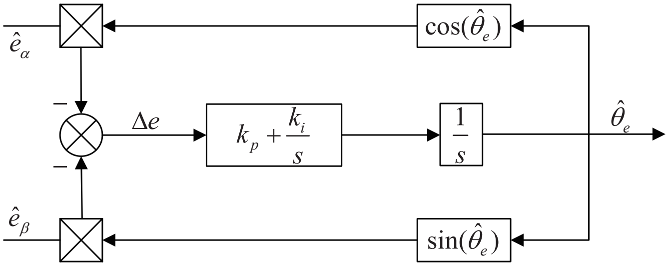


Figure 3. Structure of conventional PLL.

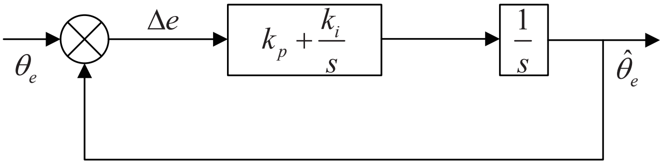


Figure 4. Traditional PLL equivalent model.

When the motor is rotating in the forward direction, the conventional PLL dynamics equation is as follows.

$$\begin{cases} \dfrac{d\tilde{\theta}}{dt} = \tilde{\omega} \\ \dfrac{d\tilde{\omega}}{dt} = -k_p \cos(\tilde{\theta})\tilde{\omega} - k_I \sin(\tilde{\theta}) \end{cases} \tag{25}$$

where $\tilde{\omega} = \omega - \hat{\omega}$ is the electrical angular velocity estimation error and k_p, k_I are positive. The phase plane of the system is shown in Figure 5, where $(\pm\pi, 0)$ is the saddle point, and $(0, 0)$ is the stable focus. The trajectory in the phase plane of the system would converge to the origin, i.e., the position and velocity estimation errors would converge to zero, and the system would be able to precisely estimate the rotor position.

When the motor is reversed, the back-EMF of the motor would change from Equation (3) to Equation (20).

$$\begin{cases} e_\alpha = \psi_f \omega_e \sin(\theta_e) \\ e_\beta = -\psi_f \omega_e \cos(\theta_e) \end{cases} \tag{26}$$

At this stage, the dynamics equation of a conventional PLL is as follows:

$$\begin{cases} \dfrac{d\tilde{\theta}}{dt} = \tilde{\omega} \\ \dfrac{d\tilde{\omega}}{dt} = k_p \cos(\tilde{\theta})\tilde{\omega} + k_I \sin(\tilde{\theta}) \end{cases} \tag{27}$$

The phase plane diagram of the system is shown in Figure 6, where $(\pm\pi, 0)$ is the stable focus, and $(0, 0)$ is the saddle point. Therefore, the system is stable at $(\pm\pi, 0)$. In such a case, the PLL would not converge the rotor position error to zero, but rather generate a π deviation. When the motor is reversed, the conventional PLL would be unable to reliably detect the rotor position.

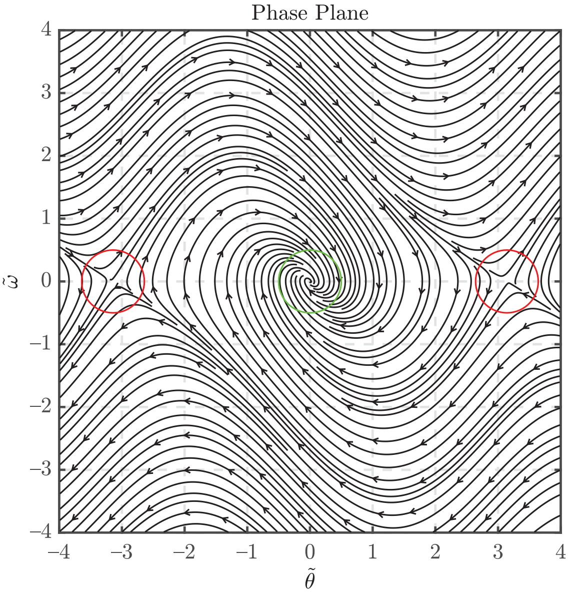


Figure 5. Traditional PLL forward phase plane.

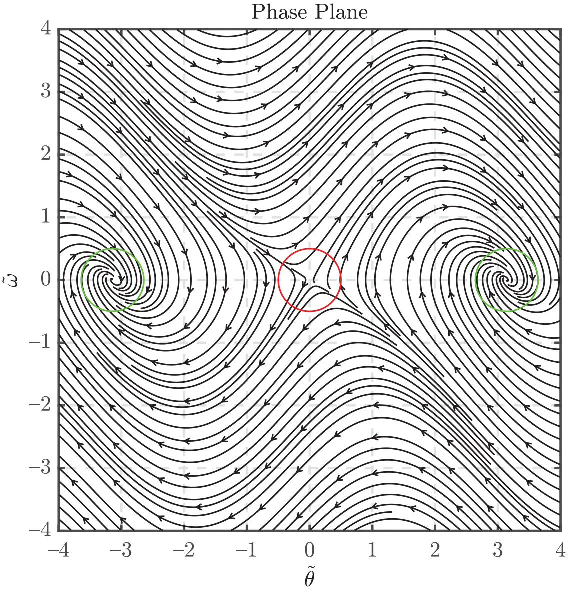


Figure 6. Traditional PLL reversed phase plane.

According to Figure 4, the error closed-loop transfer function can be obtained as

$$\Phi_{ess0}(s) = \frac{s^2}{s^2 + k_p s + k_I} \tag{28}$$

When the motor is accelerated or decelerated, the input is acceleration equivalent to the angle. Assuming that the aforementioned transfer function is provided with a unit acceleration input, the steady-state error can be calculated as follows:

$$\begin{aligned} e_{ss0} &= \lim_{s\to 0} s\cdot \Phi_{ess0}(s)\cdot \frac{1}{s^3} \\ &= \lim_{s\to 0}\frac{1}{s^2+k_p s+k_I} \\ &= \frac{1}{K_I} \end{aligned} \tag{29}$$

It is known that the steady-state error of the system is not 0 at this time, and the steady-state error of the system would be reduced when increasing the integral coefficient k_I. However, increasing the integral coefficient k_I may cause the stability of the system degradation. Therefore, under the traditional PLL, the rotor position error cannot converge to 0.

4.2. Improved PLL Analysis

Based on the deficiencies of the traditional PLL outlined in the previous section, this section will provide an improved PLL and demonstrate its stability. The structure diagram of the improved PLL is shown in Figure 7. The relationship between Δe and θ_e, $\hat{\theta}_e$ is as follows.

$$\begin{aligned} \Delta e &= -[\frac{1}{2}(e_\alpha^2+e_\beta^2)\sin(2\hat{\theta}_e)] - [e_\alpha e_\beta\cos(2\hat{\theta}_e)] \\ &= -\frac{1}{2}\psi_f^2\omega_e^2\cos(2\theta_e)\sin(2\hat{\theta}_e) + \frac{1}{2}\psi_f^2\omega_e^2\sin(2\theta_e)\cos(2\hat{\theta}_e) \\ &= \frac{1}{2}\psi_f^2\omega_e^2\sin 2(\theta_e-\hat{\theta}_e) \propto \sin(2\tilde{\theta}_e) \end{aligned} \tag{30}$$

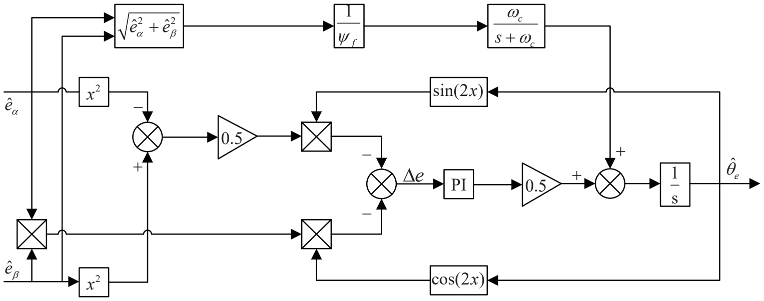


Figure 7. Structure of the improved PLL.

When the PLL is steadily tracking the rotor position, $\tilde{\theta}_e \to 0$, $\hat{\theta}_e \approx \theta_e$.

Combining Equations (3), (26), and (30), when the improved PLL is used, its dynamic equation is as follows whether the motor is in forward or reverse rotation.

$$\begin{cases} \dfrac{d\tilde{\theta}}{dt} = \tilde{\omega} \\ \dfrac{d\tilde{\omega}}{dt} = \dfrac{1}{2}[-k_p\cos(2\tilde{\theta})\cdot 2\tilde{\omega} - k_I\sin(2\tilde{\theta})] \end{cases} \tag{31}$$

The phase plane diagram of the system is shown in Figure 8, where $(\pm\pi, 0)$, $(0, 0)$, $(\pm 0.5\pi, 0)$ are the five balance points of the system, $(\pm 0.5\pi, 0)$ are the saddle points, and $(\pm\pi, 0)$, $(0, 0)$ are the stable focus. The trajectory in the phase plane of the system would approach the origin. That is, the position and speed estimation errors would converge to zero. At this time, $\hat{\theta}_e \approx \theta_e$, the system can accurately estimate the position and speed of the rotor. The improved PLL open-loop transfer function is

$$G_1(s) = \frac{(k_p + \omega_c)s^2 + (k_I + k_p\omega_c)s + k_I\omega_c}{s^2(s + \omega_c)} \tag{32}$$

where ω_c is the cut-off frequency of the LPF in the feedforward channel. By Equation (26), the error closed-loop transfer function of the rotor position can be obtained as

$$\Phi_{ess1}(s) = \frac{s^3}{(s + \omega_c)(s^2 + k_p s + k_I)} \tag{33}$$

When the motor is accelerated or decelerated, the input is the acceleration equivalent to the angle. Assuming that a unit acceleration input is added to the above transfer function, according to the final value theorem, the steady-state error can be obtained as

$$\begin{aligned} e_{ss1} &= \lim_{s \to 0} s\Phi_{ss1}(s)\frac{1}{s^3} \\ &= \lim_{s \to 0} s \cdot \frac{s^3}{(s + \omega_c)(s^2 + k_p s + k_I)} \cdot \frac{1}{s^3} \\ &= 0 \end{aligned} \tag{34}$$

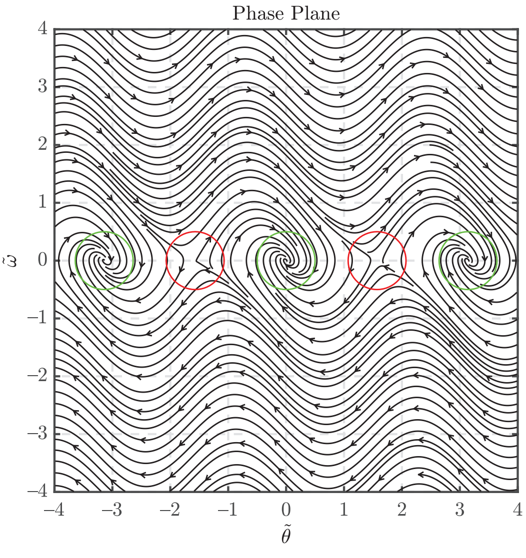


Figure 8. Improved PLL phase plane.

The improved PLL can reduce the rotor position error to 0 caused by the PLL when the motor is accelerating and decelerating. Therefore, the improved PLL can extract the position information of the rotor more accurately than the traditional PLL.

To sum up, there are two main improvements of the improved PLL:

(1) Improve the inability of the conventional PLL to reliably extract position information during motor reversal;

(2) During motor acceleration and deceleration, the conventional PLL's error problem has been resolved.

Therefore, the structure of the PMSM position sensorless control system based on SMO is shown in Figure 9.

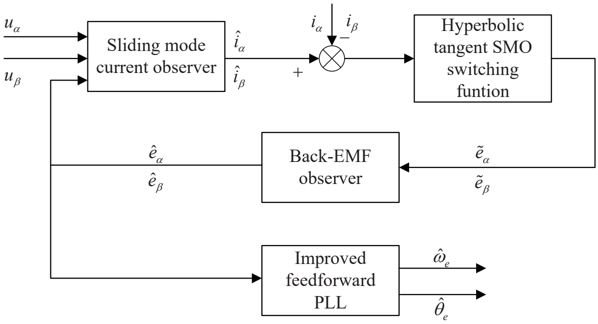


Figure 9. Structure of the PMSM position sensorless control system based on SMO.

5. Simulation Verification

To verify the feasibility of the proposed PMSM sensorless control algorithm based on hyperbolic tangent SMO and improved PLL, a simulation model is built in Matlab/Simulink according to Figure 9, and the simulation parameters are shown in Table 1.

Table 1. Main parameters of PMSM.

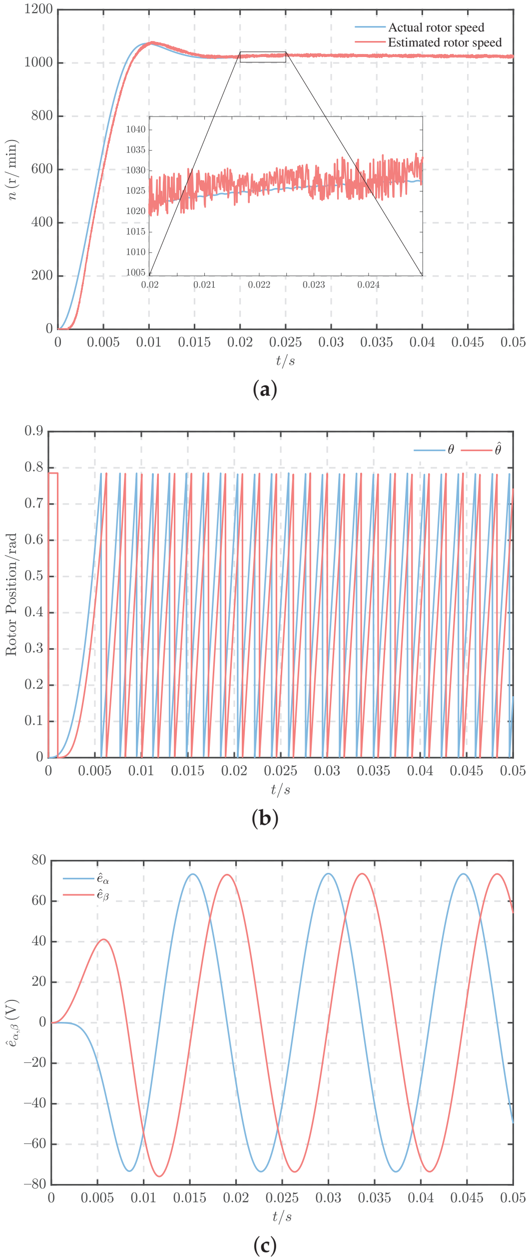
Parameter	Value
number of pole pairs p	4
stator resistance R_s	2.875 Ω
stator inductor L_s	8.5 mH
rotational inertia J	$0.001\ \text{kg} \cdot \text{m}^2$
permanent magnet flux ψ_f	0.175 Wb
DC voltage U_{dc}	311 V

5.1. Steady-State Performance

This experiment focuses on examining the steady-state performance differences between conventional SMO combined with conventional PLL and improved SMO combined with feedforward PLL.

Set the parameters of the improved SMO to $h = 0.01$, $\lambda = 100$, $\mu = 300$, $m = 100$. The parameter of traditional SMO is set to $\lambda = 1000$, $\omega_c = 2000$. The parameters of both feedforward PLL and conventional PLL are set to $k_p = 100$, k_I = 10,000.

When the motor is unloaded and its speed is set to 1000 r/min, the steady-state performance of the motor under each algorithm is compared. From Figures 10a and 11a, it can be observed that both the conventional algorithm and the improved algorithm can keep the motor running steadily, with the conventional algorithm having a speed fluctuation range of ±5 r/min and the improved algorithm having a speed fluctuation range of ±0.1 r/min. Therefore, the improved SMO may effectively mitigate the speed chattering issue. The peak rotor position estimation error is 0.255 rad and the steady-state error is 0.212 rad for the conventional algorithm, whereas the peak rotor position estimation error is 0.0043 rad and the steady-state error is 0.0042 rad for the improved algorithm (Figures 10b, 11b and 12). Consequently, the improved feedforward PLL provides superior position tracking performance.

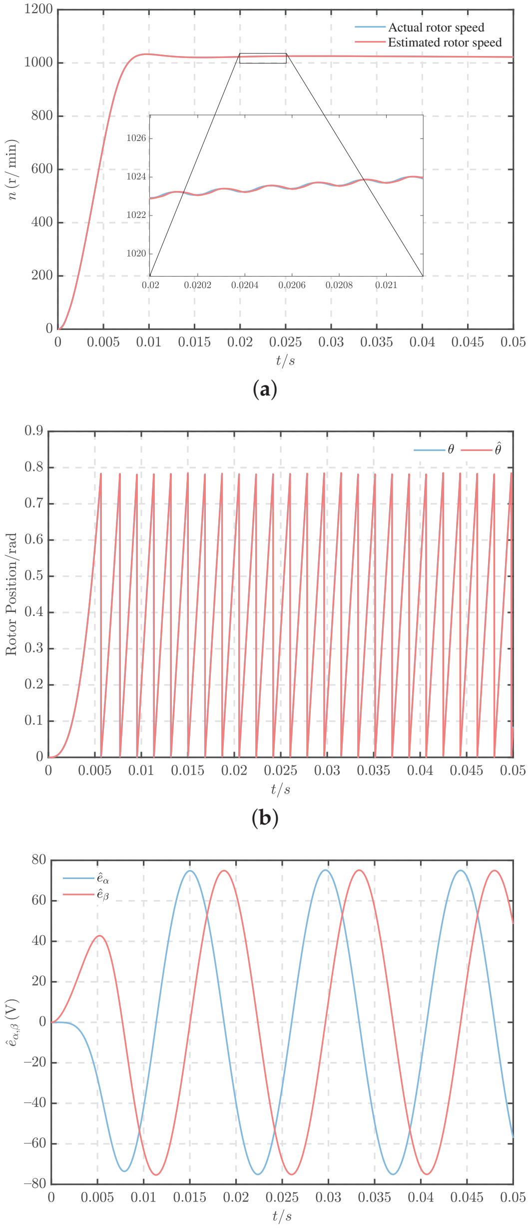


(**a**)

(**b**)

(**c**)

Figure 10. Responses of the simulation with conventional SMO combined with conventional PLL (steady-state). (**a**) Rotor speed. (**b**) Rotor position. (**c**) Estimation of the back-EMF.



(**a**)

(**b**)

(**c**)

Figure 11. Responses of the simulation with improved SMO combined with feedforward PLL (steady-state). (**a**) Rotor speed. (**b**) Rotor position. (**c**) Estimation of the back-EMF.

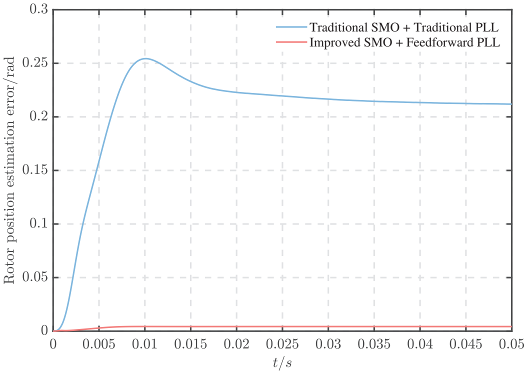


Figure 12. Rotor position observation errors (steady-state).

5.2. Dynamic Performance

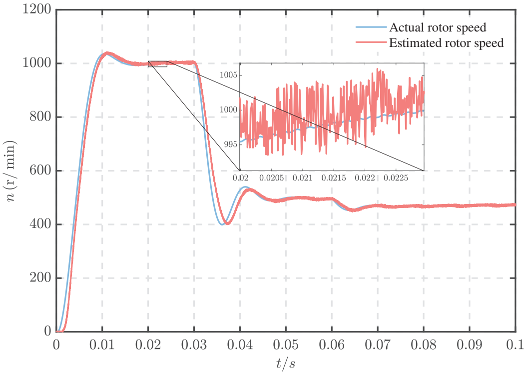
This experiment focuses on comparing the dynamic performance differences between the conventional SMO combined with the conventional PLL algorithm and the improved SMO combined with feedforward PLL when acceleration, deceleration, and sudden load addition are encountered.

First, set the initial load of the motor to 2 N·m and the initial given speed to 1000 r/min. When the motor runs to 0.03 s, set the given speed to 500 r/min to decelerate the motor. Add a sudden 4 N·m load to the motor while it has been running for 0.06 s. From Figures 13a and 14a, it can be seen that the conventional algorithm has a speed fluctuation range of ±5 r/min in the initial state and an overshoot of 4%. The improved algorithm has a speed fluctuation range of ±0.1 r/min and an overshoot of 0.3% in the initial state. In the case of motor deceleration, the adjustment time of the conventional algorithm is 15 ms, while the adjustment time of the improved algorithm is 6 ms. In the case of load surge, the speed drop of the conventional algorithm is 50 r/min and the adjustment time is 10 ms, while the speed drop of the improved algorithm is 40 r/min and the adjustment time is 6 ms. Figures 13b, 14b and 15 reveal that the traditional algorithm rotor position estimation error peaks at 0.25 rad, while the improved algorithm rotor position estimation error peaks at 0.0043 rad. After motor deceleration and sudden load application, the conventional algorithm rotor position estimation error is 0.1 rad, but the improved algorithm rotor position estimation error can quickly converge to 0.002 rad. Therefore, the improved SMO can make the motor maintain good stability and low chattering during dynamic operation. The improved PLL can keep the rotor position estimation error low during acceleration and deceleration, so that the rotor position information can be extracted more accurately.

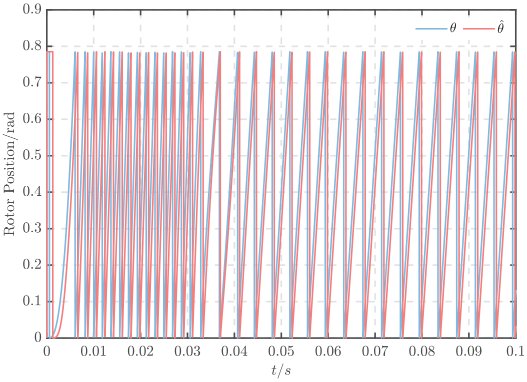
5.3. Forward and Reverse

The main purpose of this experiment is to verify that the improved feedforward PLL can accurately extract rotor position information in both forward and reverse motor rotation.

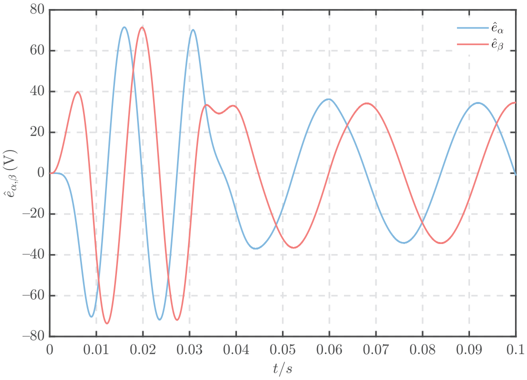
The motor load is initially set to 2 N·m and the initial given speed is set to 1000 r/min. To simulate the motor reverse, the given speed is adjusted to −500 r/min for 0.05 s. From Figures 16–18, it is evident that the improved PLL can accurately identify the rotor position information when the motor is reversed and maintain stable motor operation, whereas the conventional PLL cannot extract the rotor position information correctly when the motor is reversed, resulting in unstable motor operation. Consequently, the improved PLL can better maintain the stability of the ship motor control system in the face of the frequent forward and reverse rotation requirements of ship motors.



(**a**)

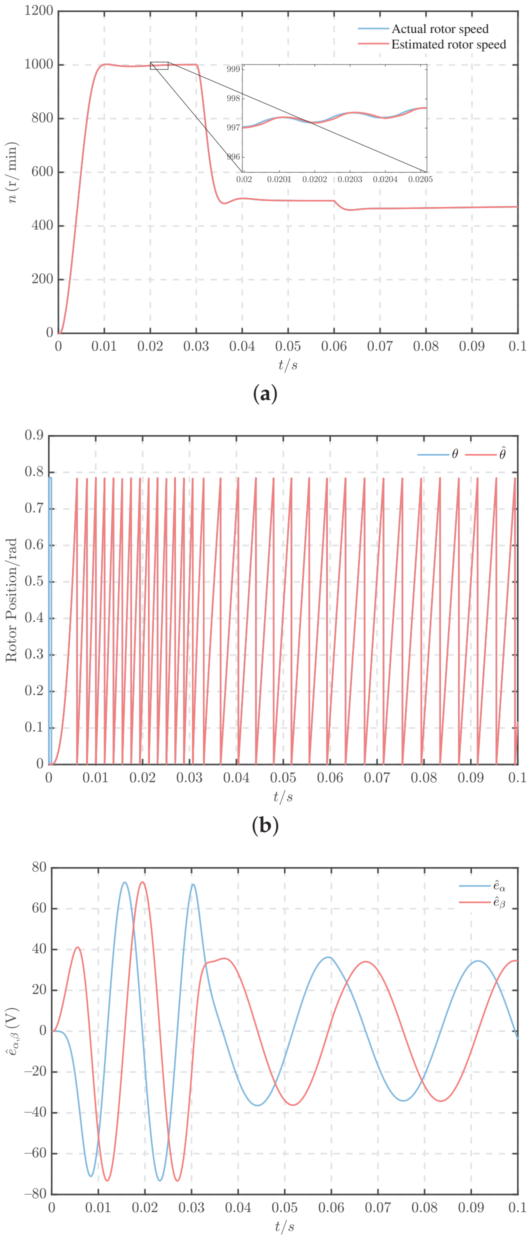


(**b**)



(**c**)

Figure 13. Responses of simulation with conventional SMO combined with conventional PLL (dynamic). (**a**) Rotor speed. (**b**) Rotor position. (**c**) Estimation of the back-EMF.



(**a**)

(**b**)

(**c**)

Figure 14. Responses of simulation with improved SMO combined with feedforward PLL (dynamic). (**a**) Rotor speed. (**b**) Rotor position. (**c**) Estimation of the back-EMF.

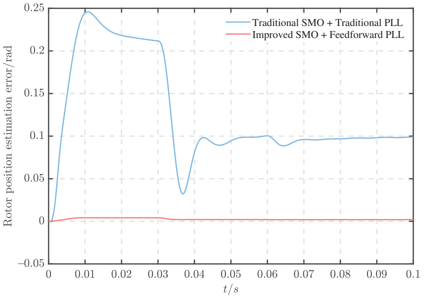
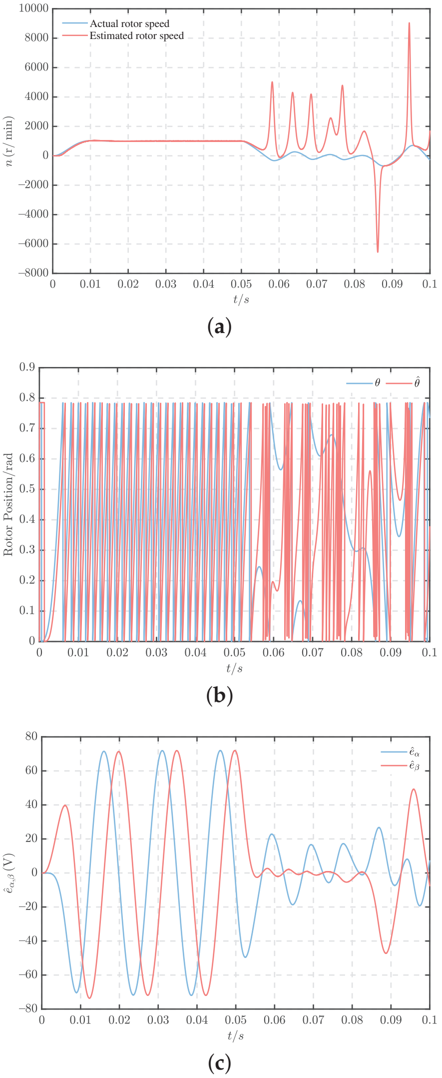


Figure 15. Rotor position observation errors (dynamic).

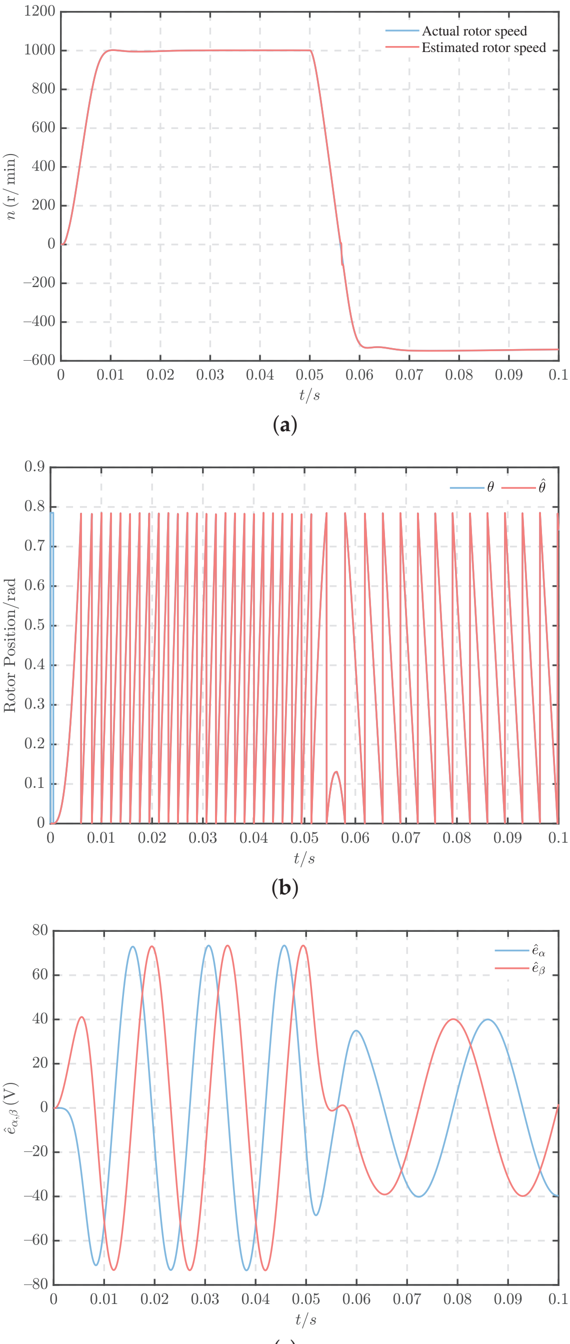


(**a**)

(**b**)

(**c**)

Figure 16. Responses of the simulation with conventional SMO combined with conventional PLL (forward and reverse). (**a**) Rotor speed. (**b**) Rotor position. (**c**) Estimation of the back-EMF.



(**a**)

(**b**)

(**c**)

Figure 17. Responses of simulation with the improved SMO combined with feedforward PLL (forward and reverse). (**a**) Rotor speed. (**b**) Rotor position. (**c**) Estimation of the back-EMF.

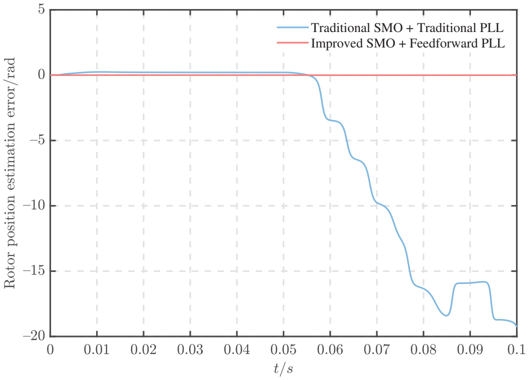


Figure 18. Rotor position observation errors (forward and reverse).

5.4. Performance under Ship Propeller Load

The motor position sensorless control algorithm proposed in this paper is mainly applied to electric propulsion ships, so it is necessary to verify the effectiveness of the algorithm under the ship propeller load. In addition, to verify the robustness of the algorithm in this paper, a random noise is applied to the load torque to simulate the disturbance of wind and waves in the ocean.

According to Ref. [30], the dynamic load model of the ship propeller is shown in Figure 19. Based on the propeller load model and the model parameter values shown in Table 2, the simulation model is established. The relationship between the motor speed and its load when the ship is sailing can be obtained. The effectiveness of the improved SMO and improved feedforward PLL is proposed in this paper for the dynamic environment of ship navigation. In Figure 19, n is the propeller speed of the motor, unit is rad/s, T_L is the propeller torque, unit is N·m, J is the advance ratio of the propeller, D_p is the diameter of the propeller, unit is m, P is the thrust generated by the rotation of the propeller, unit is N, M_s is the mass of the ship, unit is kg, R is the resistance of the ship, unit is N, V_s is the speed of the ship, unit is m/s, V_p is the propeller advance speed, unit is m/s, ω is the wake coefficient, t is the thrust derating coefficient, k is the water attachment coefficient, K_p is the dimensionless coefficient of propeller thrust, and K_m is the dimensionless coefficient of drag torque.

$$\begin{cases} K_p = 0.38955 - 0.27115J - 0.10256J^2 \\ K_m = 0.049543 - 0.021832J - 0.02079J^2 \end{cases} \tag{35}$$

The PMSM load perturbation was set to propeller load while introducing random perturbations. The initial speed was set to 1000 r/min, and after the motor ran for 0.05 s, the speed was set to −500 r/min to simulate motor reversal. As shown in Figure 20a, in the presence of random noise, the conventional algorithm ensures system stability in forward rotation, with a speed fluctuation range of ±50 r/min. However, in reverse rotation, the conventional algorithm causes the system to become unstable. From Figure 21a, it can be observed that the improved algorithm ensures system stability in both forward and reverse rotation, with a speed fluctuation range of ±20 r/min. The improved algorithm effectively reduces speed fluctuation under disturbance while ensuring stable forward and reverse rotation. From Figure 22, the maximum rotor position estimation error of the conventional algorithm is 0.231 rad, whereas the maximum rotor position estimation error of the improved algorithm is only 0.008 rad. Thus, the improved algorithm can significantly reduce the rotor estimation error and improve the position estimation accuracy. In conclusion, the algorithm exhibits excellent anti-interference performance and can adapt to complex marine environments.

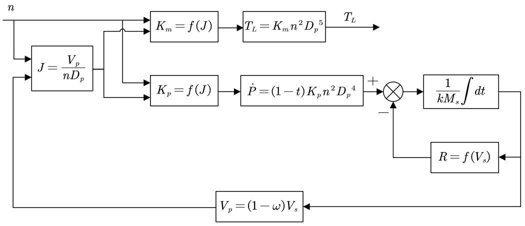


Figure 19. Ship propeller load model.

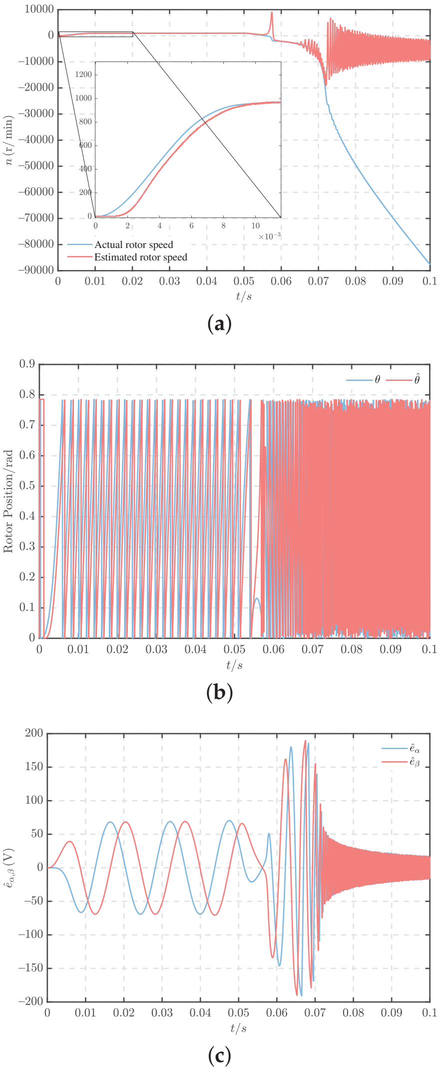
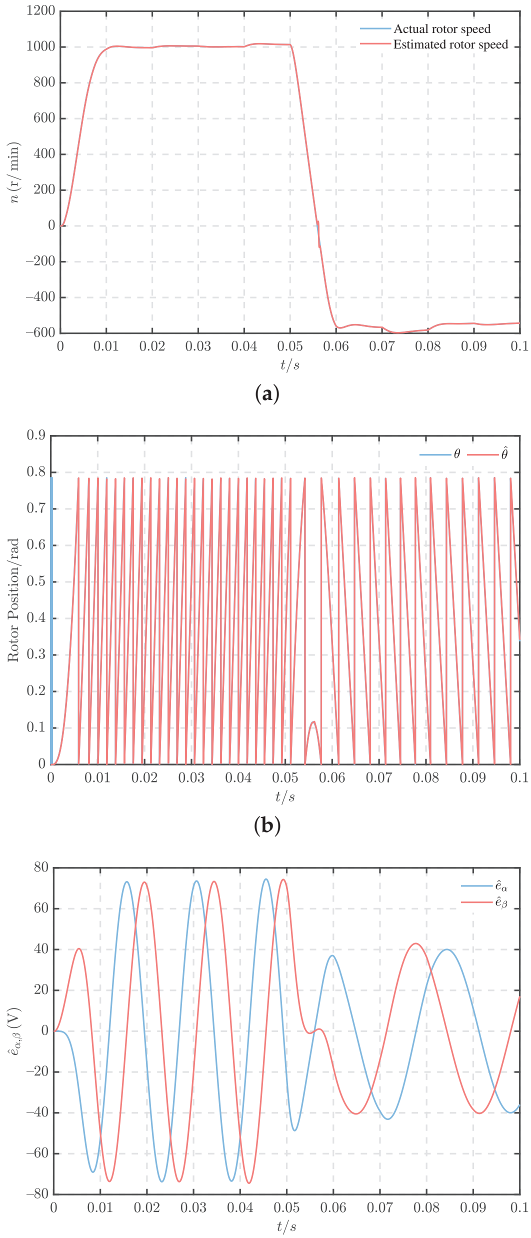


Figure 20. Responses of the simulation with conventional SMO combined with conventional PLL (with propeller load). (**a**) Rotor speed. (**b**) Rotor position. (**c**) Estimation of the back-EMF.



(**a**)

(**b**)

(**c**)

Figure 21. Responses of the simulation with improved SMO combined with feedforward PLL (with propeller load). (**a**) Rotor speed. (**b**) Rotor position. (**c**) Estimation of the back-EMF.

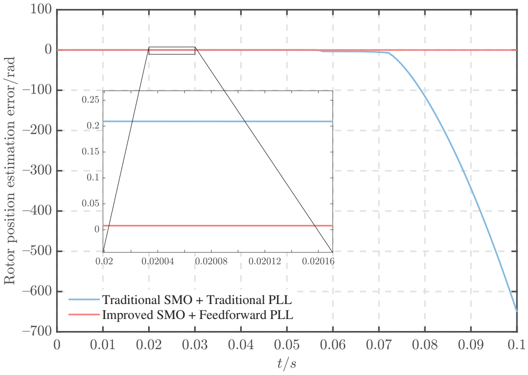


Figure 22. Rotor position observation errors (with propeller load).

Table 2. Main parameters of PMSM.

Parameter	Value
Propeller diameter D_p/m	0.1
Hull mass M_s/kg	100
Water attachment coefficient k	1.1
Wake coefficient ω	0.12285
Thrust derating coefficient t	0.146

6. Conclusions and Outlook

This paper presents an improved algorithm for position sensorless control of PMSM, which is specifically designed for electric propulsion ships equipped with shaftless RDTs. The primary goal of this algorithm is to enhance the reliability of the motor control system. The algorithm utilizes continuous hyperbolic tangent functions instead of discontinuous symbolic functions to address the chattering problem of SMO. An observer based on the back-EMF model is constructed to extract the back-EMF accurately and minimize speed fluctuations, which effectively reduces the sliding mode gain of the SMO and eliminates the phase delay caused by LPF. Furthermore, the improved feed-forward PLL is introduced to minimize rotor position estimation errors during acceleration and deceleration, as well as to achieve accurate extraction of rotor position information during motor reversal, outperforming the conventional PLL.

To verify the reliability of the algorithm in a complex marine environment, a ship propeller load model and random load are introduced. The simulation results demonstrate the superiority of the improved algorithm over the conventional algorithm consisting of a conventional SMO and a conventional PLL. The improved algorithm achieves smaller speed fluctuations in the face of disturbances, which is beneficial for the stable operation of the ship. It also shows good performance in bi-directional rotation of the motor and can stably track the rotor position, which is important for the frequent forward and reverse rotation of the ship's RDT motor.

Our future work will validate the algorithm in a real motor and apply it to a real ship, which will greatly contribute to the realization of green and sustainable ship technology.

Author Contributions: Conceptualization, H.B. and B.Y.; methodology, H.B. and B.Y.; software, H.B. and B.Y.; validation, H.B. and B.Y.; formal analysis, H.B. and B.Y.; investigation, B.Y.; resources, B.Y.; data curation, B.Y.; writing—original draft preparation, H.B., B.Y. and W.G.; writing—review and editing, H.B. and B.Y.; visualization, H.B. and W.G.; supervision, H.B. and W.G.; project administration, H.B. and W.G.; funding acquisition, H.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partly funded by supported by China Postdoctoral Science Foundation grant number 2020M672431, partly funded by project from Key Laboratory of Marine Power Engineering and Technology grant number KLMPET2020-05).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bai, H.; Yu, B. Position estimation of fault-tolerant permanent magnet motor in electric power propulsion ship system. *IEEJ Trans. Electr. Electron. Eng.* **2022**, *17*, 890–898. [CrossRef]
2. Du, J.; Li, J.; Lewis, F.L. Distributed 3D Time-Varying Formation Control of Underactuated AUVs with Communication Delays Based on Data-Driven State Predictor. *IEEE Trans. Ind. Inform.* **2022**, 1–9, *early access*. [CrossRef]
3. Qin, J.; Du, J. Robust adaptive asymptotic trajectory tracking control for underactuated surface vessels subject to unknown dynamics and input saturation. *J. Mar. Sci. Technol.* **2022**, *27*, 307–319. [CrossRef]
4. Qin, J.; Du, J.; Li, J. Adaptive finite-time trajectory tracking event-triggered control scheme for underactuated surface vessels subject to input saturation. *IEEE Trans. Intell. Transp. Syst.* **2023**, 1–11, *early access*. [CrossRef]
5. Chen, X.; Wang, Z.; Hua, Q.; Shang, W.; Luo, Q.; Yu, K. AI-Empowered Speed Extraction via Port-like Videos for Vehicular Trajectory Analysis. *IEEE Trans. Intell. Transp. Syst.* **2022**, 1–12, *early access*.
6. Qin, J.; Du, J. Minimum-learning-parameter-based adaptive finite-time trajectory tracking event-triggered control for underactuated surface vessels with parametric uncertainties. *Ocean Eng.* **2023**, *271*, 113634. [CrossRef]
7. Chen, X.; Liu, S.; Liu, W.; Wu, H.; Han, B.; Zhao, J. Quantifying Arctic oil spilling event risk by integrating analytic network process and fuzzy comprehensive evaluation model. *Ocean Coast. Manag.* **2022**, *228*, 106326. [CrossRef]
8. Bai, H.; Yu, B.; Ouyang, W.; Yan, X.; Zhu, J. HF-based sensorless control of a FTPMM in ship shaftless rim-driven thruster system. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 16867–16877. [CrossRef]
9. Yan, X.; Liang, X.; Ouyang, W; Liu, Z.; Liu, B.; Lan, J. A review of progress and applications of ship shaft-less rim-driven thrusters. *Ocean Eng.* **2017**, *144*, 142–156. [CrossRef]
10. Sulligoi, G.; Vicenzutti, A.; Menis, R. All-electric ship design: From electrical propulsion to integrated electrical and electronic power systems. *IEEE Trans. Transp. Electrif.* **2016**, *2*, 507–521. [CrossRef]
11. Lin, S.; Zhang, W. An adaptive sliding-mode observer with a tangent function-based PLL structure for position sensorless PMSM drives. *Int. J. Electr. Power Energy Syst.* **2017**, *88*, 63–74. [CrossRef]
12. Bai, H.; Zhu, J.; Qin, J.; Sun, J. Fault-tolerant control for a dual-winding fault-tolerant permanent magnet motor drive based on SVPWM. *IET Power Electron.* **2017**, *10*, 509–516. [CrossRef]
13. Qiao, Z.W.; Shi, T.N.; Wang, Y.D.; Yan, Y.; Xia, C.L.; He, X.N. New sliding-mode observer for position sensorless control of permanent-magnet synchronous motor. *IEEE Trans. Ind. Electron.* **2013**, *60*, 710–719. [CrossRef]
14. Yang, Z.; Yan, X.; Ouyang, W.; Bai, H. Sensor lessPMSM Control Algorithm for Rim-Driven Thruster Based on Improved PSO. In Proceedings of the 2021 6th International Conference on Transportation Information and Safety, Wuhan, China, 22–24 October 2021; pp. 285–292.
15. Seo, D.; Bak, Y.; Lee, K. An Improved Rotating Restart Method for a Sensorless Permanent Magnet Synchronous Motor Drive System Using Repetitive Zero Voltage Vectors. *IEEE Trans. Ind. Electron.* **2020**, *67*, 3496–3504. [CrossRef]
16. Wang, G.L.; Valla, M.; Solsona, J. Position Sensorless Permanent Magnet Synchronous Machine Drives-A Review. *IEEE Trans. Ind. Electron.* **2020**, *67*, 5830–5842. [CrossRef]
17. Zhang, X.G.; Li, Z.X. Sliding-Mode Observer-Based Mechanical Parameter Estimation for Permanent Magnet Synchronous Motor. *IEEE Trans. Power Electron.* **2016**, *31*, 5732–5745. [CrossRef]
18. Yang, H.; Yang, R.; Hu, W.; Huang, Z.M. FPGA-Based Sensorless Speed Control of PMSM Using Enhanced Performance Controller Based on the Reduced-Order EKF. *IEEE J. Emerg. Sel. Top. Power Electron.* **2021**, *9*, 289–301. [CrossRef]
19. Abo-Khalil, A.G.; Eltamaly, A.M.; Alsaud, M.S.; Sayed, K.; Alghamdi, A.S. Sensorless control for PMSM using model reference adaptive system. *Int. Trans. Electr. Energy Syst.* **2021**, *31*, e12733. [CrossRef]
20. Bernard, P.; Praly, L. Estimation of Position and Resistance of a Sensorless PMSM: A Nonlinear Luenberger Approach for a Nonobservable System. *IEEE Trans. Autom. Control.* **2021**, *66*, 481–496. [CrossRef]
21. Wang, G.L.; Hao, X.F.; Zhao, N.N.; Zhang, G.Q.; Xu, D.G. Current Sensor Fault-Tolerant Control Strategy for Encoderless PMSM Drives Based on Single Sliding Mode Observer. *IEEE Trans. Transp. Electrif.* **2020**, *6*, 679–689. [CrossRef]
22. Wu, L.G.; Liu, J.X.; Vazquez, S.; Mazumder, S.K. Sliding Mode Control in Power Converters and Drives: A Review. *IEEE-CAA J. Autom. Sin.* **2022**, *9*, 392–406. [CrossRef]
23. Kang, W.X.; Li, H. Improved sliding mode observer based sensorless control for PMSM. *IEICE Electron. Express* **2017**, *14*, 20170934. [CrossRef]

24. Wibowo, W.K.; Jeong, S.K. Improved estimation of rotor position for sensorless control of a PMSM based on a sliding mode observer. *J. Cent. South Univ.* **2016**, *23*, 1643–1656. [CrossRef]
25. Ren, N.N.; Fan, L.; Zhang, Z. Sensorless PMSM Control with Sliding Mode Observer Based on Sigmoid Function. *J. Electr. Eng. Technol.* **2021**, *16*, 933–939. [CrossRef]
26. Ye, S.C.; Yao, X.X. An Enhanced SMO-Based Permanent-Magnet Synchronous Machine Sensorless Drive Scheme with Current Measurement Error Compensation. *IEEE J. Emerg. Sel. Top. Power Electron.* **2021**, *9*, 4407–4419. [CrossRef]
27. Yang, Z.B.; Ding, Q.F.; Sun, X.D.; Lu, C.L.; Zhu, H.M. Speed sensorless control of a bearingless induction motor based on sliding mode observer and phase-locked loop. *ISA Trans.* **2022**, *123*, 346–356. [CrossRef] [PubMed]
28. Gong, C.; Hu, Y.H.; Gao, J.Q.; Wang, Y.G.; Yan, L.M. An Improved Delay-Suppressed Sliding-Mode Observer for Sensorless Vector-Controlled PMSM. *IEEE Trans. Ind. Electron.* **2020**, *67*, 5913–5923. [CrossRef]
29. Song, X.D.; Fang, J.C.; Han, B.C.; Zheng, S.Q. Adaptive Compensation Method for High-Speed Surface PMSM Sensorless Drives of EMF-Based Position Estimation Error. *IEEE Trans. Power Electron.* **2016**, *31*, 1438–1449. [CrossRef]
30. Peng, H.; Zhu, X.; Yang, L.; Zhang, G. Robust controller design for marine electric propulsion system over controller area network. *Control Eng. Pract.* **2020**, *101*, 104512. [CrossRef]

Journal of
Marine Science
and Engineering

Article

A Study on the Correlation between Ship Movement Characteristics and Ice Conditions in Polar Waters

Liang Chen [1], Changhai Huang [1,*] and Yanhao Wang [2]

[1] Merchant Marine College, Shanghai Maritime University, Shanghai 201306, China; chenliang@shmtu.edu.cn
[2] China Construction Harbour and Channel Engineering Bureau Group Co., Ltd., Shanghai 200433, China
* Correspondence: chhuang@shmtu.edu.cn

Abstract: The opening of arctic routes provides a new option for international navigation ships. The correlation between ship movement characteristics and ice conditions should be known, which will help ships adapt to the polar waters. Based on the voyage data and sea ice manual observation data of the 'XUE LONG' ship's six voyages in polar waters, a correlation analysis model of ice conditions and ship movement characteristics was established in this work. First, the ship movement characteristics in polar waters were analyzed, such as the distribution characteristics of ship speeds, courses, and variation characteristics by using the descriptive statistical analysis method and data visualization analysis method. Then, by using multivariate correlation analysis and univariate controlled correlation analysis methods, the correlation between movement characteristics and ice conditions, such as ice concentration and thickness, and the correlation between different ice conditions themselves, were quantitatively analyzed. The result shows that the correlation analysis model of ice conditions and ship movement characteristics is reliable and effective and can obtain quantitative correlation analysis results. On the one hand, sea ice thickness has almost no significant correlation with ship movement characteristics, excluding the influence of sea ice concentration. On the other hand, excluding the influence of sea ice thickness, sea ice concentration is still significantly correlated with the absolute value of speed, speed variation, and course variation. The conclusions of this work have important reference significance for polar scientific investigations, commercial ships' voyages in icy waters, and ships' designs for icy waters.

Keywords: ship movement characteristics; polar waters; correlation analysis; polar navigation

Citation: Chen, L.; Huang, C.; Wang, Y. A Study on the Correlation between Ship Movement Characteristics and Ice Conditions in Polar Waters. *J. Mar. Sci. Eng.* **2023**, *11*, 729. https://doi.org/10.3390/jmse11040729

Academic Editor: Sasan Tavakoli

Received: 3 March 2023
Revised: 22 March 2023
Accepted: 26 March 2023
Published: 27 March 2023

1. Introduction

As the global climate warms, the melting of the Arctic sea ice has accelerated, which makes the Arctic waterways gradually navigable. The opening of the Arctic route will provide Asia with two more convenient routes to Europe and North America, reducing the voyage by 10–13 days [1] and saving many ship fuel costs, canal costs, security costs, personnel costs, and many more. The advantages of the Arctic route will inevitably attract a large number of merchant ships to choose it [2]. However, the current navigation environment of the Arctic waterway is still dominated by icy waters. Merchant ships choosing Arctic routes need to improve and optimize ship design, route planning, and ship maneuvering [3].

The ship movement characteristics in icy waters can provide reference information for ship design, route planning, and ship maneuvering [4]. Different sea ice conditions have different effects on ship maneuverability, ship speed, and other characteristics [5]. Due to the lack of experience in the Arctic route, there are few studies on the ship movement characteristics in icy water, and there is a lack of quantitative methods to clarify the influence of sea ice conditions on the ship movement characteristics under multiple observed variables. Therefore, a correlation analysis model of ice conditions and ship movement characteristics is established in this work based on the voyage data and sea ice manual observation data of the 'XUE LONG' ship's six voyages in polar waters. The ship movement characteristics in polar waters are analyzed, such as the distribution characteristics of ship speeds, courses, and variation characteristics using the method of descriptive statistical analysis and data visualization analysis. The correlation of ship movement characteristics and ice conditions, such as ice concentration and thickness, and the correlation between different ice conditions themselves, are quantitatively analyzed by using multivariate correlation analysis and univariate controlled correlation analysis methods.

The remainder of this study is organized as follows. In Section 2, a literature review is presented. Section 3 presents the materials and methods. The discussion is provided in Section 4. Section 5 states the conclusion.

2. Literature Review

Sea ice is an important factor that affects ship movement characteristics in polar waters. The Northern Sea Route (NSR) has a large difference in ice conditions in summer and winter [6]. To ensure navigational safety, there is widespread consensus to reduce the speed of ships navigating icy waters. With the continuous improvement of the feasibility of the NSR, how to improve the efficiency and safety of ship navigation in the sea ice area has become a focus of scholars. Firstly, the relationship between ship speed and ice thickness is studied by numerical and simulation methods [7,8]. The second approach is to study the navigation performance of ships in polar open waters. Chen, C. et al., analyzed the effects of ship type, ship speed, and wave steepness on added resistance of the polar research vessel based on the three-dimensional full nonlinear time domain potential flow theory [9]. In the process of ship operation in the polar sea, some scholars have established the model of key parameters in the process of ship–ice interaction and applied it to the structural design loads [10,11]. The methods of numerical simulation and theoretical analysis provide a theoretical basis for the analysis of the ship movement characteristics in the polar sea, but it still needs to be combined with the actual observation data of the navigation environment and state for further research [12].

With the development of information technology, it has become easier to obtain Automatic Identification System (AIS) data and navigation environment data. The research results of ship movement characteristics based on a statistical analysis of data are developing continuously, but there is still a lot of potential value for massive spatiotemporal data that has not been fully utilized. The correlation between massive spatiotemporal data and ship movement characteristics needs to be deeply analyzed [13]. In terms of the characteristic state of ship movement, Zheng J. et al., studied the nonlinear characteristics of ship movement using control theory [14]. Montewka J. et al., established a hybrid model of ship performance in ice-covered waters, which can be used in ice region navigation planning [15]. With the AIS and navigation environment data, the ship movement characteristics can be further explored. Goerlandt F. et al., showed summary statistics of speed in convoy and escort operations in icy waters under different sea ice thickness situations primarily based on AIS data [16]. Zhang C. et al., pointed out that while the ice becomes denser in real life Arctic navigation, ship speed will gradually decrease [17]. The analysis of navigation environment data can also be extended to the study of traffic state characteristics [18–20].

At present, some conclusions about ship movement characteristics have been obtained from ship navigation status data and navigation environment data, but there are few studies on the quantitative correlation between sea ice density and ship movement characteristics in polar waters. Correlation research is an effective method to reveal the correlation between external variables and the target itself [21]. Zhao P. et al., proposed an analytical framework for exploring the relationship between intra-urban logistics and urban transport planning by integrating spatial analysis, network analysis, and spatial interaction analysis [22]. Gagic R. et al., determined the correlation between cruise ship activities in ports with an ambient concentration of pollutants [23]. The correlation analysis can also be extended to reveal the relationships among water traffic factors such as environmental conditions, traffic characteristics, ship collision frequency, ship defects, and so on [24,25]. In terms of polar ship navigation, Zambon A. et al., identified the correlation between ice-induced propeller loads and sea ice conditions by experimental measurements and numerical analysis [26]. Yuen P.C. et al., analyzed the correlation of cargo damage risks for the planning of marine container transportation voyages [27]. The premise of these studies is to accurately obtain the analytical data of each target. For ships in the polar sea, how to effectively obtain the environmental state data and ship movement state data is also the focus of the research [28–31].

There are some results on the relationship between ship performance and ice conditions. In the study by Montewka J. et al., the correlation coefficient between ship speed and sea ice concentration is given; there are no detailed studies on the correlation coefficient between ship speed and different ice concentrations [32]. In fact, the sea ice concentration and sea ice thickness are also related; they are not independent of each other [33]. Kimmritz M. et al., analyzed the relationship between sea ice concentration and thickness in the sea ice data assimilation method [34]. It is assumed that sea ice concentration and sea ice thickness are both related to ship movement characteristics. In that case, it is necessary to clarify which factor is more decisive: the characteristics of the ship course or ship course variation. The relationship between the ship course and sea ice conditions also needs to be further studied. These studies can clarify whether the ship uses ice breaking or bypasses the ice when sailing in different ice regions and chooses waters with thin ice layers or waters without ice. At the same time, the ship movement characteristics of ships are not only associated with sea ice concentration but are additionally related to sea ice thickness. Then, the relationship between sea ice concentration and sea ice thickness also needs to be explored. Moreover, while sea ice concentration and sea ice thickness are each associated with ship movement characteristics, it is essential to make clear which element is more decisive.

3. Materials and Methods

3.1. Correlation Analysis Framework

The correlation analysis between ship movement characteristics and ice conditions should meet three requirements: (1) analyze the basic information of ship movement characteristics, (2) explore the interaction between ice conditions variables, (3) obtain the interaction between ship movement characteristics and ice conditions, and design a comprehensive correlation analysis framework on this basis. The framework is mainly composed of three parts: data source, core analysis method, and analysis result output, as shown in Figure 1.

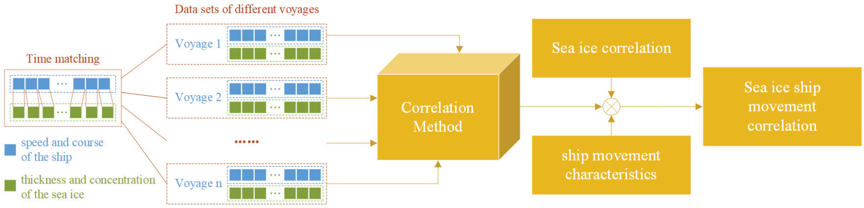


Figure 1. Framework for analysis of the correlation between ice conditions and ship movement characteristics.

In terms of data sources, the voyage data of the scientific research ship 'XUE LONG' in the Arctic and Antarctic, as well as the ice data collected during the voyage, will be used. Because the ice data collected by the ship during its voyage contained the actual information, they have great value for analyzing the navigation characteristics of ships in polar waters. In terms of core analysis methods, this paper studies a correlation analysis method between ice conditions and ship movement characteristics, which can output the correlation of the navigation environment, as well as the correlation between navigation environment and ship movement characteristics. In the aspect of result output, the ship navigation process is taken as the time series, the data types of the analysis results are combined, and a two-dimensional cross relationship of the correlation of data points is formed to form a graph, and the mutual relationship between the ice conditions and the ship movement characteristics is visualized.

3.2. Data Collection

This study's original data came from the 5–9th Arctic voyage and the 34th Antarctic voyage of the icebreaker 'XUE LONG,' including 69,098 voyage data points from GPS equipment and 4085 manual observation data points on sea ice. The icebreaker 'XUE LONG' is China's largest polar science research ship. It is 167 m long and 22.6 m wide, with a full draft of 9 m and a gross tonnage of 15,352 tons. It has a maximum speed of 17.9 knots, an endurance of 20,000 nautical miles, and is capable of continuously breaking through 1.2 m of ice (including 0.2 m of snow) at a speed of 1.5 knots.

3.3. Data Preprocessing

The voyage data and the manual observation data on sea ice were matched in time. Among these two types of data, there are 836 records with the same recording time, 2435 records with a time difference of less than 1 min, and 3394 records with a time difference of less than 10 min. Considering the ship's speed and sea ice variations, 3394 records with a time difference within 10 min between the two types of data were analyzed.

(1) Sea ice concentration and its classification and measurement

Sea ice concentration represents the sea ice coverage area ratio to the total area of the sea area. It is often expressed in percentages (tenths).

In this study, the sea ice concentration data are manual observation data, and the observation interval is about 30 min.

Among them, C_t represents the total concentration of sea ice in the view field with a diameter of 5 km (in tenths). C_a represents the concentration of sea ice with the largest concentration in the view field with a diameter of 5 km (Type A). C_b represents the concentration of sea ice with the second largest concentration in the view field with a diameter of 5 km (Type B). C_c represents sea ice concentration with the third largest concentration in the view field with a diameter of 5 km (Type C).

(2) Sea ice thickness and its classification and measurement

Sea ice thickness is defined as the vertical distance between the sea ice surface and the ice bottom, in cm.

In this study, the sea ice thickness data are also manual observation data, and the observation interval is about 30 min. In the actual sea ice observation data, the interval is usually used to express sea ice thickness, such as 50–70 cm. In this study, the interval data are processed, and their values are the average values of the upper and lower boundaries of the interval.

Among them, Th_a represents the thickness of the Type A sea ice. Th_b represents the thickness of the Type B sea ice. Th_c represents the thickness of the Type C sea ice.

Th_{AVG} represents the total average thickness of sea ice in the view field with a diameter of 5 km. Th_{AVG} is not directly derived from observational data and is calculated according to Formula (1) in this study.

$$Th_{AVG} = (C_a \cdot Th_a + C_b \cdot Th_b + C_c \cdot Th_c)/(C_a + C_b + C_c) \tag{1}$$

3.4. Data General Characteristics and Visualization

This study performed descriptive statistical analysis on the preprocessed data to obtain ship movement characteristic information, such as ship speed and course, in the research waters. To convey and communicate information clearly and effectively with the aid of graphical means, this study conducted a data visualization analysis on the collected data to visually display the data distribution characteristics of ship movement characteristics. Histograms, rose graphs, box graphs, and so on were adopted for data visualization analysis of ship movement characteristics and ice conditions.

3.5. Modeling

In order to explore the quantitative correlation between sea ice conditions and ship movement characteristics, this study established a correlation analysis model of ice conditions and ship movement characteristics. The model took ice condition and ship movement state variables as inputs and included two parts: multivariate correlation analysis and univariate controlled correlation analysis, which is shown in Figure 2. In multivariate correlation analysis, the correlation between ship movement characteristics and sea ice thickness and sea ice concentration, and other state variables is mainly studied. Since the relationship between multivariate variables is very complex, ship movement characteristics may be affected by multiple variables; therefore, univariate controlled correlation analysis is established on the basis of multivariate correlation analysis, which is partial correlation analysis. By eliminating the influence of the third variable, the correlation between other variables is analyzed to obtain a more accurate relationship between ice conditions and ship movement characteristics.

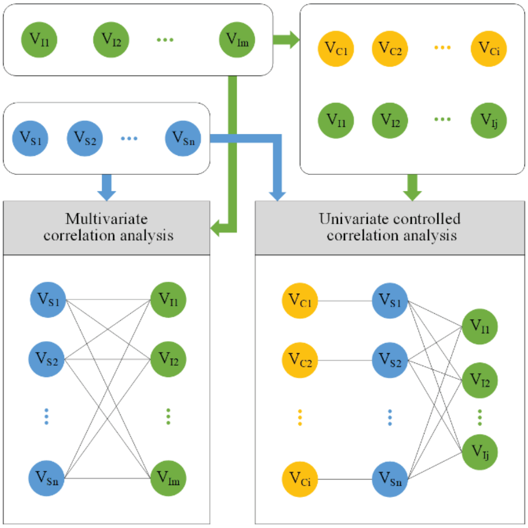


Figure 2. Model of correlation between ice conditions and ship movement characteristics.

As shown in Figure 2, the mathematical relationship of correlation analysis between ice conditions and ship movement characteristics is expressed as follows:

$$C_{Mi,j} = cov\left(V_{Si}, V_{Ij}\right), i \in \{1,\cdots,N\}, j \in \{1,\cdots,M\} \tag{2}$$

$$C_{UMi,j} = cov\left(V_{Si}, V_{Ij}\right)|V_{Ck}, i \in \{1,\cdots,N\}, j \in \{1,\cdots,M-1\}, k \in \{1,\cdots,M\} \tag{3}$$

where $C\_M_{i,j}$ is multivariate correlation analysis, $C\_UM_{i,j}$ is the univariate controlled correlation analysis, $cov()$ is the correlation analysis function, V_{Si} is the ship movement state variable, N is the total number of ship movement state variables, V_{Ij} is the ice condition state variable, M is the total number of ice condition state variables, and V_{Ck} is the selected control variable in ice condition state variables.

3.5.1. Multivariate Correlation Analysis

There are three types of data analysis methods in multivariate correlation analyses: Pearson, Spearman, and Kendall [35,36]. For quantitative variables with a normal distribution, the Pearson correlation coefficient can be used. If the data do not follow a normal distribution or have a sorted category, the Spearman coefficient or Kendall's tau-b coefficient can be adopted [37]. The latter two measure the correlation between ranks.

Spearman's correlation is also known as rank correlation, which is used when one or both variables are ranks or ordinal scales. It is applicable to determine the degree of correlation between two variables in the case of ordinal data.

The Spearman rank correlation coefficient is defined by:

$$r_s = 1 - \frac{6\sum d_i^2}{n(n^2-1)} \tag{4}$$

where d_i is the difference among ranks of ith pair of the two variables and n is the number of pairs of observations.

3.5.2. Univariate Controlled Correlation Analysis

The univariate controlled correlation analysis calculates the partial correlation coefficient, which describes the linear relationship between two variables while controlling the effect of one or more additional variables. It can be used to determine if the relationship between two variables is direct, spurious, or intervening, controlling each of these variables' correlation with a third related variable.

The formula for the partial correlation coefficient for X and Y, controlling for Z, is as follows:

$$r_{yx.z} = \frac{r_{yx} - r_{yz}r_{xz}}{\sqrt{\left(1-r_{yz}^2\right)\left(1-r_{xz}^2\right)}} \tag{5}$$

$r_{yx.z}$ is the partial correlation coefficient between variable X and variable Y, controlling for Z.

Before solving the above formula, the zero-order coefficients between all possible pairs of variables (Y and X, Y and Z, X, and Z) must be calculated first. The formula for the zero-order coefficients is as follows:

$$r_{xy} = \frac{\sum_{i=1}^{n}\left[(x_i-\overline{x})(y_i-\overline{y})\right]}{\sqrt{\sum_{i=1}^{n}(x_i-\overline{x})^2 \cdot \sum_{i=1}^{n}(y_i-\overline{y})^2}} \tag{6}$$

r_{xy} is the zero-order coefficient between variable X and variable Y.

The following ranges are used for the interpretation of the strength of the correlation. Complete correlation: correlation with $|r| = 1$; high correlation or strong correlation: correlation with $0.7 \leq |r| < 1$; moderate correlation: correlation with $0.4 \leq |r| < 0.7$; low correlation or weak correlation: $|r| < 0.4$ correlation; and zero correlation: $r = 0$.

4. Results

4.1. Analysis of Ship Movement Characteristics in Polar Waters

4.1.1. Characteristics of Ship Speed in Polar Waters

The speed histogram of the "XUE LONG" ship (at intervals of 1 knot) is shown in Figure 3, and the speed statistics table is shown in Table 1.

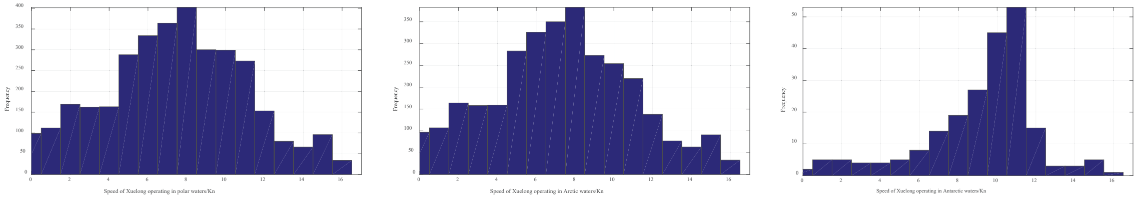


Figure 3. The speed distribution of the ship movement in polar waters.

Table 1. Statistics of ship speeds in polar waters.

	Minimum	**Maximum**	**Mean**	**S.D.**	**Median**	**Mode**	**Count**
Antarctic waters	0.20	15.60	9.20	2.98	9.90	10.90	218
Arctic waters	0.00	16.00	7.38	3.65	7.40	8.30	3176
Polar waters	0.00	16.00	7.50	3.63	7.60	8.30	3394

From the data in the statistical table, the average ship speed in the Antarctic waters is nearly 2 knots higher than that in the Arctic waters, and ship speed variation in the Arctic waters is larger than that in the Antarctic waters.

4.1.2. Characteristics of Variation of Ship Speed in Polar Waters

In this study, the speed variation refers to the difference between the speed of the current observation time and the speed of the last observation time, and its value is a vector. The box diagram of the speed variation of the "XUE LONG" ship is shown in Figure 4, and the statistics table of speed variation is shown in Table 2. It can be seen from the box chart that the quartiles of speed variation for different waters are relatively close.

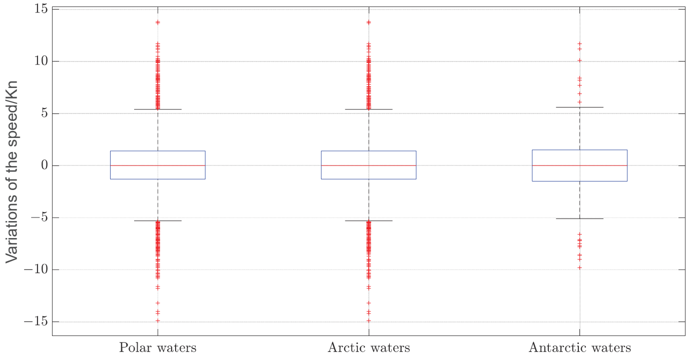


Figure 4. Box plot of ship speed variation in polar waters.

Table 2. Statistics of ship speed variation in polar waters.

	Minimum	**Maximum**	**Mean**	**S.D.**	**Median**	**Mode**	**Count**
Antarctic waters	−9.80	11.70	−0.02	3.21	0.00	−0.20	218
Arctic waters	−14.90	13.80	0.00	2.91	0.00	0.00	3176
Polar waters	−14.90	13.80	0.00	2.93	0.00	0.00	3394

From the statistical table, the mean of the speed variation in Antarctic waters and Arctic waters basically tends to be 0; the mode of speed variation in Arctic waters is 0, indicating that a certain speed is always maintained, and the mode of speed variation in Antarctic waters is −0.2 knots, which may be due to the frequency of observation during deceleration. The S.D. of speed variation in Antarctic waters is greater than that in Arctic waters, indicating a higher dispersion of speed variation.

4.1.3. Characteristics of Ship Course in Polar Waters

The rose chart of the course of the "XUE LONG" ship (with a 5° interval) is shown in Figure 5. It can be seen intuitively in the figure that, regardless of the Antarctic routes or the Arctic routes, the "XUE LONG" ship's course distribution is relatively scattered. Although there are several main lobes, the directions of the main lobes are still scattered.

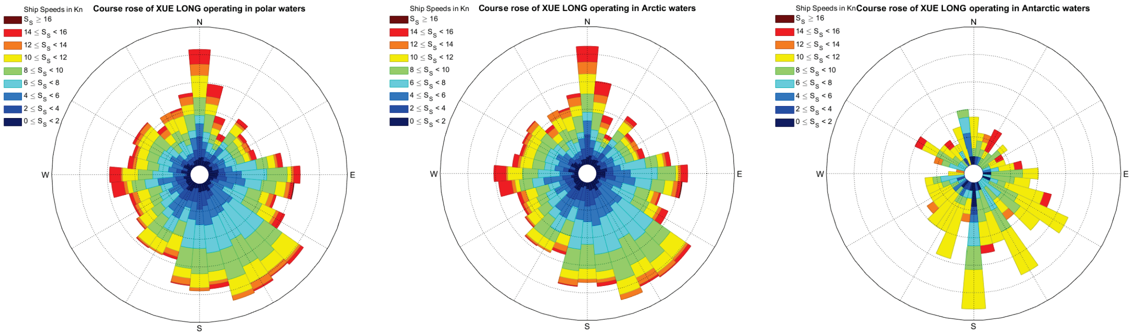


Figure 5. Rose chart of the ship's course distribution in polar waters.

4.1.4. Characteristics of the Ship's Course Variations in Polar Waters

In this study, the course variation refers to the difference between the course at the current observation time and the course at the last observation time, and its value is a vector. The box chart of the course variation of the "XUE LONG" ship is shown in Figure 6, and the statistical table of the course variation is shown in Table 3. In the box chart, it can be seen that the median of course variations in the Arctic waters is basically the same, but the interquartile range of the course variation in the Antarctic waters is larger, indicating that the course variation in the Antarctic waters is more scattered.

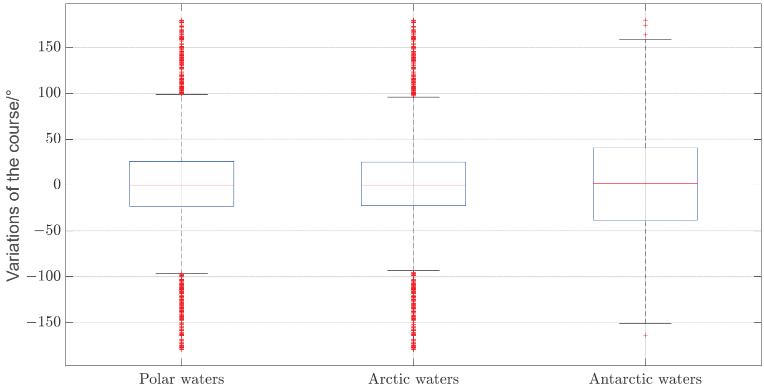


Figure 6. Box plot of the ship's course variations in polar waters.

Table 3. Statistics of the ship's course variations in polar waters.

	Minimum	Maximum	Mean	S.D.	Median	Mode	Count
Antarctic waters	−163.60	179.70	2.95	68.52	1.95	−20.00	218
Arctic waters	−179.30	179.80	−0.13	54.62	0.00	−3.20	3176
Polar waters	−179.30	179.80	0.07	55.61	0.00	−3.20	3394

From the statistical table, the mean of the ship's course variation in Antarctic waters deviates more from 0° than in Arctic waters, the median and mode also deviate more from 0°, and the standard deviation is larger than in Arctic waters. The ship's course variation is more scattered, and the variation is greater.

4.2. Correlation Analysis between Sea Ice Conditions and Ship Movement Characteristics

4.2.1. Correlation Analysis between Sea Ice Concentration and Ship Movement Characteristics

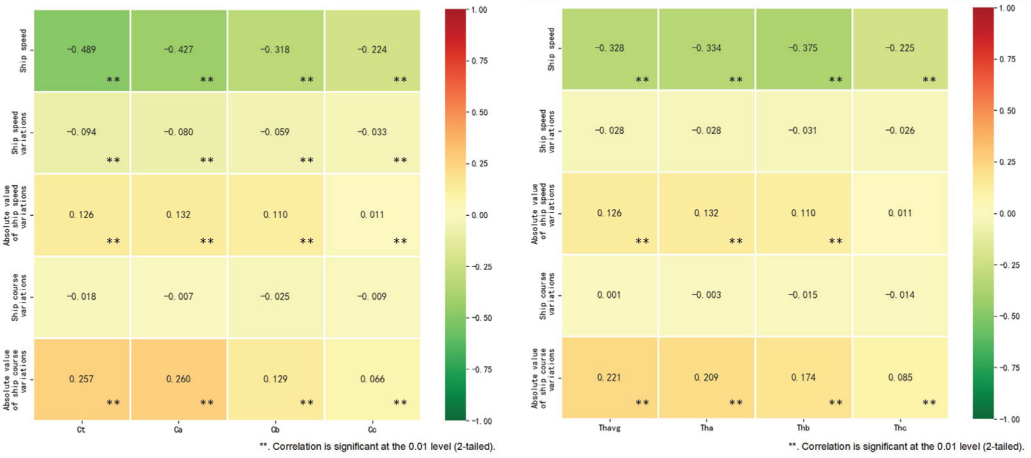
A bivariate correlation analysis was carried out on various types of sea ice concentrations and ship movement parameters. After testing, each variable's data does not conform to the normal distribution, so Pearson's correlation analysis cannot be used. Moreover, because the navigation parameters are not ordered variables, it is also inappropriate to use Kendall's tau-b correlation coefficient. Therefore, this study used the Spearman correlation analysis method to calculate the Spearman correlation coefficient and conduct a significance test.

The results of the correlation analysis between sea ice concentration and ship movement parameters are shown in Table 4 and Figure 7a.

Table 4. The correlation coefficient between sea ice concentration and ship movement characteristics.

Spearman's Correlation Coefficient		C_t	C_a	C_b	C_c
Ship speed	Correlation Coefficient	−0.489 **	−0.427 **	−0.318 **	−0.224 **
	Sig. (2-tailed)	0	0	0	0
	N	3394	3394	3394	3394
Ship speed variation	Correlation Coefficient	−0.094 **	−0.080 **	−0.059 **	−0.033
	Sig. (2-tailed)	0	0	0	0
	N	3394	3394	3394	3394
The absolute value of ship speed variation	Correlation Coefficient	0.126 **	0.132 **	0.110 **	0.011
	Sig. (2-tailed)	0	0	0	0
	N	3394	3394	3394	3394
Ship course variations	Correlation Coefficient	−0.018	−0.007	−0.025	−0.009
	Sig. (2-tailed)	0.291	0.671	0.151	0.618
	N	3394	3394	3394	3394
The absolute value of ship course variations	Correlation Coefficient	0.257 **	0.260 **	0.129 **	0.066 **
	Sig. (2-tailed)	0	0	0	0
	N	3394	3394	3394	3394

**. Correlation is significant at the 0.01 level (2-tailed).



(**a**) (**b**)

Figure 7. The correlation coefficient. (**a**) The correlation coefficient between sea ice concentration and ship movement characteristics; (**b**) the correlation coefficient between sea ice thickness and ship speed.

It can be seen in Table 4 and Figure 7a that total sea ice concentration, Type A sea ice concentration, and ship speed have a significant and moderately negative correlation. Type B sea ice concentration, Type C sea ice concentration, and ship speed have a significant and low negative correlation. Total sea ice concentration, Type A sea ice concentration, and Type B sea ice concentration are all significantly negatively correlated with ship speed variation. Total sea ice concentration, Type A sea ice concentration, and Type B sea ice concentration are all also significantly positively correlated with the absolute value of ship speed variation. The absolute value of the correlation coefficient with the latter is greater than with the former, indicating that the correlation degree between total sea ice concentration, Type A sea ice concentration, Type B sea ice concentration, and the absolute value of the speed variation is greater than the correlation with the speed variation. There is no significant correlation between total sea ice concentration and ship course variation. Still, all sea ice concentrations have a significant weak positive correlation with the absolute value of ship course variation.

4.2.2. Correlation Analysis between Sea Ice Thickness and Ship Movement Characteristics

The correlation analysis results between sea ice thickness and ship movement characteristics are shown in Table 5 and Figure 7b.

Table 5. The correlation coefficient between sea ice thickness and ship speed.

Spearman's Correlation Coefficient		Th_{AVG}	Th_a	Th_b	Th_c
Ship speed	Correlation Coefficient	−0.328 **	−0.334 **	−0.375 **	−0.225 **
	Sig. (2-tailed)	0	0	0	0
	N	3394	3394	3394	3394
Ship speed variation	Correlation Coefficient	−0.028	−0.028	−0.031	−0.026
	Sig. (2-tailed)	0.104	0.102	0.074	0.131
	N	3394	3394	3394	3394
The absolute value of ship speed variation	Correlation Coefficient	0.126 **	0.132 **	0.110 **	0.011
	Sig. (2-tailed)	0	0	0	0.516
	N	3394	3394	3394	3394
Ship course variations	Correlation Coefficient	0.001	−0.003	−0.015	−0.014
	Sig. (2-tailed)	0.976	0.858	0.391	0.407
	N	3394	3394	3394	3394
The absolute value of ship course variations	Correlation Coefficient	0.221 **	0.209 **	0.174 **	0.085 **
	Sig. (2-tailed)	0	0	0	0
	N	3394	3394	3394	3394

**. Correlation is significant at the 0.01 level (2-tailed).

It can be seen in Table 5 and Figure 7b that there is no significant correlation between the thickness of sea ice and ship speed variation. However, the average thickness of sea ice, the thickness of Type A sea ice, the thickness of Type B sea ice, and the absolute value of ship speed variation show a significant and weak positive correlation. There is no significant correlation between sea ice thickness and ship course variation. Still, each type of sea ice thickness has a significant and weak positive correlation with ship course variations' absolute value.

4.2.3. Correlation Analysis between Sea Ice Concentration and Sea Ice Thickness

The relationship between sea ice conditions' characterizing factors was analyzed to explore the relationship between sea ice conditions and ship movement characteristics. The bivariate correlation analysis of various types of sea ice concentration and sea ice thickness was carried out, and the analysis results are shown in Table 6 and Figure 8.

Table 6. The correlation coefficient between sea ice concentration and sea ice thickness.

Spearman's Correlation Coefficient		Th_{AVG}	Th_a	Th_b	Th_c
C_t	Correlation Coefficient	0.597 **	0.602 **	0.609 **	0.356 **
	Sig. (2-tailed)	0	0	0	0
	N	3394	3394	3394	3394
C_a	Correlation Coefficient	0.551 **	0.541 **	0.478 **	0.179 **
	Sig. (2-tailed)	0	0	0	0
	N	3394	3394	3394	3394
C_b	Correlation Coefficient	0.493 **	0.527 **	0.693 **	0.330 **
	Sig. (2-tailed)	0	0	0	0
	N	3394	3394	3394	3394
C_c	Correlation Coefficient	0.301 **	0.292 **	0.465 **	0.932 **
	Sig. (2-tailed)	0	0	0	0
	N	3394	3394	3394	3394

**. Correlation is significant at the 0.01 level (2-tailed).

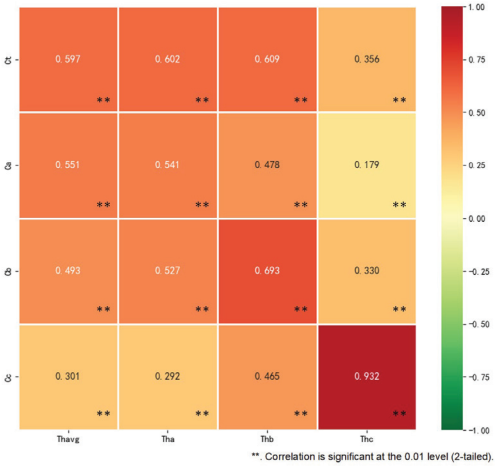


Figure 8. The correlation coefficient between sea ice concentration and sea ice thickness.

The total sea ice concentration is significantly positively correlated with the thickness of various sea ice types, especially with the average sea ice thickness, Type A sea ice thickness, and Type B sea ice thickness. Type A sea ice concentration has a significant and positive correlation with the thickness of various sea ice types. There is a significant and moderate positive correlation with the average sea ice thickness, Type A sea ice thickness, and Type B sea ice thickness.

Type B sea ice concentration is significantly positively correlated with the thickness of various sea ice types. The average sea ice thickness, Type A sea ice thickness, and Type B sea ice thickness are significantly positively correlated. Type C sea ice concentration has a significant and positive correlation with the thickness of various sea ice types. It has a strong positive correlation with the thickness of Type C sea ice and a significant and moderate positive correlation with Type B sea ice thickness.

4.2.4. Partial Correlation Analysis between Sea Ice Conditions and Ship Movement Characteristics

In view of the fact that there is a positive correlation between sea ice concentration and sea ice thickness, which characterize sea ice conditions, a partial correlation analysis method was used to analyze the relationship between sea ice conditions and ship movement characteristics. The partial correlation analysis result is shown in Tables 7 and 8 and Figure 9a,b.

Table 7. Partial correlation analysis between sea ice conditions and ship movement characteristics with the control variable of ice thickness.

Control Variable	Spearman's Correlation Coefficient		C_t	C_a	C_b	C_c
Th_{AVG}	Ship speed	correlation coefficient	−0.393 **	−0.333 **	−0.190 **	−0.141 **
		Sig. (2-tailed)	0	0	0	0
		N	3391	3391	3391	3391
Th_a	Ship speed	correlation coefficient	−0.395 **	−0.335 **	−0.181 **	−0.148 **
		Sig. (2-tailed)	0	0	0	0
		N	3391	3391	3391	3391
Th_b	Ship speed variation	correlation coefficient	−0.076 **	−0.062 **	−0.035 *	−0.02
		Sig. (2-tailed)	0	0	0.04	0.254
		N	3391	3391	3391	3391
Th_c	Ship speed variation	correlation coefficient	−0.074 **	−0.060 **	−0.033	−0.019
		Sig. (2-tailed)	0	0	0.051	0.263
		N	3391	3391	3391	3391
Th_{AVG}	The absolute value of ship speed variation	correlation coefficient	−0.006	0.013	−0.014	−0.032
		Sig. (2-tailed)	0.74	0.449	0.411	0.063
		N	3391	3391	3391	3391
Th_a	The absolute value of ship speed variation	correlation coefficient	−0.006	0.013	−0.016	−0.031
		Sig. (2-tailed)	0.742	0.436	0.346	0.073
		N	3391	3391	3391	3391
Th_{AVG}	Ship course variation	correlation coefficient	−0.031	−0.017	−0.027	−0.023
		Sig. (2-tailed)	0.074	0.312	0.114	0.176
		N	3391	3391	3391	3391
Th_a	Ship course variation	correlation coefficient	−0.027	−0.014	−0.025	−0.022
		Sig. (2-tailed)	0.118	0.409	0.146	0.204
		N	3391	3391	3391	3391
Th_{AVG}	The absolute value of ship speed variation	correlation coefficient	−0.006	0.013	−0.014	−0.032
		Sig. (2-tailed)	0.74	0.449	0.411	0.063
		N	3391	3391	3391	3391
Th_a	The absolute value of ship course variation	correlation coefficient	0.170 **	0.206 **	0.006	−0.015
		Sig. (2-tailed)	0	0	0.737	0.38
		N	3391	3391	3391	3391

**. Correlation is significant at the 0.01 level (2-tailed). *. Correlation is significant at the 0.05 level (2-tailed).

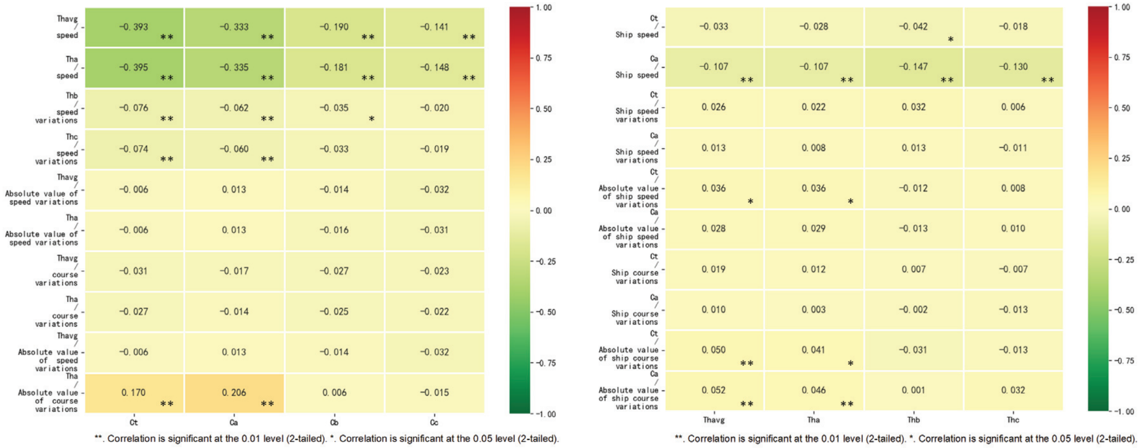
Table 8. Partial correlation analysis between sea ice conditions and ship movement characteristics with the control variable of ice concentration.

Control Variable	Spearman's Correlation Coefficient		Th_{AVG}	Th_a	Th_b	Th_c
C_t	Ship speed	correlation coefficient	−0.033	−0.028	−0.042 *	−0.018
		Sig. (2-tailed)	0.057	0.103	0.015	0.298
		N	3391	3391	3391	3391
C_a	Ship speed	correlation coefficient	−0.107 **	−0.107 **	−0.147 **	−0.130 **
		Sig. (2-tailed)	0	0	0	0
		N	3391	3391	3391	3391
C_t	Ship speed variation	correlation coefficient	0.026	0.022	0.032	0.006
		Sig. (2-tailed)	0.131	0.2	0.064	0.732
		N	3391	3391	3391	3391

Table 8. *Cont.*

Control Variable	Spearman's Correlation Coefficient		Th_{AVG}	Th_a	Th_b	Th_c
C_a	Ship speed variation	correlation coefficient Sig. (2-tailed) N	0.013 0.461 3391	0.008 0.634 3391	0.013 0.448 3391	−0.011 0.528 3391
C_t	The absolute value of ship speed variation	correlation coefficient Sig. (2-tailed) N	0.036 * 0.034 3391	0.036 * 0.035 3391	−0.012 0.49 3391	0.008 0.625 3391
C_a	The absolute value of ship speed variation	correlation coefficient Sig. (2-tailed) N	0.028 0.097 3391	0.029 0.097 3391	−0.013 0.454 3391	0.01 0.559 3391
C_t	Ship course variation	correlation coefficient Sig. (2-tailed) N	0.019 0.277 3391	0.012 0.485 3391	0.007 0.701 3391	−0.007 0.672 3391
C_a	Ship course variation	correlation coefficient Sig. (2-tailed) N	0.01 0.563 3391	0.003 0.841 3391	−0.002 0.918 3391	−0.013 0.445 3391
C_t	The absolute value of ship course variation	correlation coefficient Sig. (2-tailed) N	0.050 ** 0.003 3391	0.041 * 0.016 3391	−0.031 0.073 3391	−0.013 0.443 3391
C_a	The absolute value of ship course variation	correlation coefficient Sig. (2-tailed) N	0.052 ** 0.002 3391	0.046 ** 0.007 3391	0.001 0.962 3391	0.032 0.059 3391

**. Correlation is significant at the 0.01 level (2-tailed). *. Correlation is significant at the 0.05 level (2-tailed).



	Ct	Ca	Cb	Cc
Thavg / speed	−0. 393 **	−0. 333 **	−0. 190 **	−0. 141 **
Tha / speed	−0. 395 **	−0. 335 **	−0. 181 **	−0. 148 **
Thb / speed variations	−0. 076 **	−0. 062 **	−0. 035 *	−0. 020
Thc / speed variations	−0. 074 **	−0. 060 **	−0. 033	−0. 019
Thavg / Absolute value of speed variations	−0. 006	0. 013	−0. 014	−0. 032
Tha / Absolute value of speed variations	−0. 006	0. 013	−0. 016	−0. 031
Thavg / course variations	−0. 031	−0. 017	−0. 027	−0. 023
Tha / course variations	−0. 027	−0. 014	−0. 025	−0. 022
Thavg / Absolute value of speed variations	−0. 006	0. 013	−0. 014	−0. 032
Tha / Absolute value of course variations	0. 170 **	0. 206 **	0. 006	−0. 015

**. Correlation is significant at the 0.01 level (2-tailed). *. Correlation is significant at the 0.05 level (2-tailed).

	Thavg	Tha	Thb	Thc
Ct / Ship speed	−0. 033	−0. 028	−0. 042 *	−0. 018
Ca / Ship speed	−0. 107 **	−0. 107 **	−0. 147 **	−0. 130 **
Ct / Ship speed variations	0. 026	0. 022	0. 032	0. 006
Ca / Ship speed variations	0. 013	0. 008	0. 013	−0. 011
Ct / Absolute value of ship speed variations	0. 036 *	0. 036 *	−0. 012	0. 008
Ca / Absolute value of ship speed variations	0. 028	0. 029	−0. 013	0. 010
Ct / Ship course variations	0. 019	0. 012	0. 007	−0. 007
Ca / Ship course variations	0. 010	0. 003	−0. 002	−0. 013
Ct / Absolute value of ship course variations	0. 050 **	0. 041 *	−0. 031	−0. 013
Ca / Absolute value of ship course variations	0. 052 **	0. 046 **	0. 001	0. 032

**. Correlation is significant at the 0.01 level (2-tailed). *. Correlation is significant at the 0.05 level (2-tailed).

(**a**) (**b**)

Figure 9. Partial correlation analysis. (**a**) Partial correlation analysis between sea ice conditions and ship movement characteristics with the control variable of ice thickness; (**b**) partial correlation analysis between sea ice conditions and ship movement characteristics with the control variable of ice concentration.

(1) Partial correlation analysis between sea ice conditions and ship speed

While controlling for C_t, there is almost no significant correlation between ship speed and sea ice thickness. While controlling for C_a, the speed and the thickness of various sea ice types show a significant and weak negative correlation. While controlling for Th_{AVG}, the speed and various types of sea ice concentration all show significant and weak negative correlations. While controlling for Th_a, the speed and various types of sea ice concentration all show significant and weak negative correlations.

(2) Partial correlation analysis between sea ice conditions and ship speed variations

While controlling for C_t, there is no significant correlation between the speed variations and the sea ice thickness. While controlling for C_a, there is no significant correlation between the speed variations and the sea ice thickness. While controlling for Th_{AVG}, there are significant and weak correlations between ship speed variations and total sea ice concentration, Type A sea ice concentration, and Type B sea ice concentration, but there is no significant correlation with Type C sea ice concentration. While controlling for Th_a, the speed variations have significant and weak correlations with total sea ice concentration and Type A sea ice concentration but do not significantly correlate with Type B and Type C sea ice concentration.

(3) Partial correlation analysis between sea ice conditions and the absolute value of speed variations

While controlling for C_t, ship speed variations' absolute value has a significant and weak correlation with sea ice's average thickness and Type A sea ice thickness. Still, it does not significantly correlate with the thickness of Type B and Type C sea ice. While controlling for C_a, the absolute value of the speed variations does not significantly correlate with the thickness of various sea ice types. While controlling for Th_{AVG}, the absolute value of the speed variations does not significantly correlate with the concentration of various sea ice types. While controlling for Th_a, the absolute value of the speed variations does not significantly correlate with the concentration of various sea ice types.

(4) Partial correlation analysis between sea ice conditions and ship course variations

While controlling for C_t, there is no significant correlation between ship course variations and the thickness of various sea ice types. While controlling for C_a, ship course variations have no significant correlation with the thickness of various sea ice types. While controlling for Th_{AVG}, there is no significant correlation between ship course variations and the concentration of various sea ice types. While controlling for Th_a, there is no significant correlation between ship course variations and the concentration of various sea ice types.

(5) Partial correlation analysis between sea ice condition and the absolute value of ship course variations

While controlling for C_t, ship course variations' absolute value has a significant and weak positive correlation with sea ice's average thickness and Type A sea ice thickness. Still, it does not significantly correlate with the thickness of Type B and Type C sea ice. While controlling for C_a, the absolute value of ship course variations is significantly and positively correlated with the average sea ice thickness and the thickness of Type A sea ice, but not significantly correlated with the thickness of the Type B and Type C sea ice. While controlling for Th_{AVG}, the absolute value of ship course variations does not significantly correlate with the concentration of various sea ice types. While controlling for Th_a, the absolute value of ship course variations is significantly and positively correlated with sea ice concentration and Type A sea ice concentration, but not significantly correlated with Type B and Type C sea ice concentration.

5. Discussion

5.1. Ship Movement Characteristics in Polar Waters

The ship movement characteristics in icy waters are a reference value for ship design and management. This study extracts the ship movement characteristics in the Arctic and Antarctic icy waters based on statistical data.

(1) Frequent changes in speed

The speed distribution chart in Figure 3 and the speed variation box chart in Figure 4 show that the speed distribution of the "XUE LONG" ship is relatively scattered, and the speeds vary more frequently. This may be due to the narrow navigable waters between the ice, the ship's change of heading being hindered, or the ship needing to break the ice, which usually requires the ship to keep changing speed.

(2) Frequent changes in the course

From the ship's course rose chart in Figure 5 and the course variation box chart in Figure 6, it can be seen that the course distribution of the "XUE LONG" ship is relatively scattered, and the course varies more frequently. A popular explanation is that to avoid icy conditions, ships often have to change their course.

(3) Navigating at a low speed

The "XUE LONG" has a maximum speed of 17.9 knots. The average speed is lower than 10 knots in actual navigation, and the speed is relatively low. Once the ship enters an icy area, it will usually be prepared to slow down before the situation is determined.

(4) The difference exists between the Antarctic and Arctic routes

The Antarctic routes' speeds are more concentrated, most of which are distributed in the 8–12 knot range; the kurtosis of the speed distribution of the Arctic route is smaller than that of the Antarctic route, and the speeds appear more frequently in the range of 2–12 knots. The Antarctic route's average speed is nearly 2 knots higher than that of the Arctic route, and the standard deviation of the speeds of the Arctic route is large, and the speeds vary greatly. The distribution patterns of speed variations of different routes are relatively similar, but the Antarctic route's dispersion degree of speed variations is higher. Whether in the Antarctic routes or the Arctic routes, the distribution of the courses of the "XUE LONG" ship is relatively scattered, but the distribution of the course variation is more concentrated and the left and right forms of the distribution curve are closer; the course varies in the Antarctic routes because they are more scattered than the Arctic routes, and the degree of change in course variation is greater.

5.2. The Influence of Sea Ice Conditions on Ship Movement Characteristics

This study shows that, in general, both sea ice concentration and sea ice thickness will affect the speed and course variations of ships. Correlation analysis was carried out on different sea ice conditions and the "XUE LONG" ship's ship movement characteristics. The main conclusions are as follows.

5.2.1. The Relationship between Sea Ice Concentration and Ship Movement Characteristics

(1) There is a significant negative correlation between sea ice concentration and ship speed. This shows that the speed decreases with the increase in sea ice concentration and vice versa. In the study presented by Montewka [16], the correlation coefficient between speed and level ice concentration is -0.610, and the correlation coefficient between speed and ridged ice concentration is -0.31; the correlation coefficient between speed and total ice concentration calculated in this study is -0.489, which is between the above two values. The negative correlation between sea ice concentration and ship speed can also be clearly seen in the study's time series of analyzed parameters.

(2) The sea ice concentration has a significant negative correlation with speed variation. In contrast, it has a significant and weak positive correlation with the absolute value of speed variation, and the correlation degree is greater than the correlation with the speed variation. With the increase in sea ice concentration, the magnitude of speed variation increases, and the direction of speed variation can be decelerated or accelerated.

(3) There is no significant correlation between sea ice concentration and course variation. Still, the concentrations of all sea ice types have a significant and weak positive correlation with the absolute value of course variation. As the concentration of sea ice increases, the course variation's magnitude increases, but the course variation's direction is uncertain.

5.2.2. The Relationship between Sea Ice Thickness and Ship Movement Characteristics

(1) The thickness of all sea ice types has a significant and weak negative correlation with ship speed. This shows that the ship's speed decreases with the increase in sea ice thickness and vice versa. In the study by Montewka [16], the correlation coefficient between speed and level ice thickness is -0.120, and the correlation coefficient between speed and

rafted ice thickness is −0.500. The correlation coefficient between speed and average ice thickness calculated in this study is −0.328, which is between the above two values. The negative correlation between sea ice thickness and ship speed can also be found in the h–V curve, which describes the ship's maximum speed in level ice in the study by Izumiyama [38].

(2) There is no significant correlation between sea ice's average thickness and speed variation. However, the average thickness, Type A sea ice thickness, and Type B sea ice thickness are all positively correlated with the speed variation's absolute value; as the thickness of sea ice increases, the magnitude of speed variation increases.

(3) There is no significant correlation between sea ice thickness and course variation. Still, all sea ice thickness types have a significant and weak positive correlation with the absolute value of course variation. As the thickness of sea ice increases, the magnitude of course variation increases.

5.2.3. The Correlation between Sea Ice Concentration and Sea Ice Thickness

The overall sea ice concentration has a significant positive correlation with the thickness of various sea ice types. The average sea ice thickness, Type A sea ice thickness, and Type B sea ice thickness have a significant and moderate positive correlation, which can be explained by reduced horizontal melting for thicker ice.

5.2.4. Partial Correlation Analysis between Sea Ice Conditions and Ship Movement Characteristics

Since sea ice concentration and sea ice thickness have a significant positive correlation, a partial correlation analysis of sea ice conditions and ship movement characteristics was performed. Excluding sea ice concentration, sea ice thickness has almost no significant correlation with ship movement characteristics; excluding the influence of sea ice thickness, sea ice concentration is still significantly correlated with the absolute value of speed, speed variation, and course variation. That is to say, when the sea ice concentration is high, the speed of the ship will be reduced, and the magnitude of the course will be increased. Overall this can be explained by the fact that the ship tends to navigate in more open waters with the same ice thickness.

5.3. Shortcomings and Prospects

The data sampling in this study is not equal in the Antarctic and Arctic routes. The data collection and analysis need to be improved to further analyze the difference in ship movement characteristics in the Antarctic and Arctic routes. There may be errors in the observation data itself, and the data should be verified and preprocessed to improve accuracy. In addition to the sea ice concentration and sea ice thickness discussed in this study, the effects of sea ice types, etc., need to be further studied.

6. Conclusions

In this paper, we introduced a correlation analysis model for ice conditions and ship motion characteristics. The original work on correlation analysis was first developed since the complexity of Arctic navigation and the lack of sufficient navigational experience may make it difficult to discern the relationship between all the factors that influence the ship movement characteristics. The whole correlation analysis model consists of two sub-processes: one is multivariate correlation analysis and the other is univariate controlled correlation analysis. With regard to the former, the Spearman correlation method was chosen because it is more suitable for determining the degree of correlation between two variables in the case of ordinal data, and is applicable to the actual voyage and ice data used in this paper. According to the analysis result, the correlation coefficient between speed and total ice concentration calculated in this paper is −0.489. The sea ice concentration has a significant negative correlation with the speed variation and the concentrations of all sea ice types have a significant and weak positive correlation with the absolute value of course variation.

Since the relationships between multivariate variables is very complex, the ship movement characteristics may be affected by more than one variable. In order to further explore the relationship between ice conditions and ship movement characteristics, a univariate controlled correlation analysis was developed based on the multivariate correlation analysis. The results show that excluding sea ice concentration, sea ice thickness has almost no significant correlation with ship movement characteristics. Excluding the influence of sea ice thickness, sea ice concentration is still significantly correlated with the absolute value of speed, speed variation, and course variation. The conclusions of this work have important reference significance for polar scientific investigations, commercial ships' voyages in icy waters, and ships' designs for icy waters. Notwithstanding this, there are certain limitations that require further effort in future research. First of all, the input parameters of the current correlation analysis model would benefit from including a greater number of related components, such as flow velocity, flow direction, wind conditions, etc., which would facilitate a more comprehensive analysis of the ship movement characteristics in polar waters. Second, since the current study is based on the "XUE LONG" ship and the data are not extensive enough, the next step is to try to obtain more data from different types of ships. Finally, the relationship between ship motion characteristics and icy conditions in polar waters is not only a static correlation analysis problem but also feedback on ship maneuvers. This means that the next research effort will combine the captain's maneuvering and decision-making processes in polar waters to provide a practical reference for more ships attempting to navigate Arctic routes.

Author Contributions: Conceptualization, L.C., C.H. and Y.W.; methodology, L.C. and C.H.; validation, C.H.; investigation, C.H. and Y.W.; resources, L.C. and C.H.; data curation, C.H.; writing—original draft preparation, C.H. and Y.W.; writing—review and editing, L.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (Grant No. 51909156, 51709167) and the China Postdoctoral Science Foundation (Grant No. 2016M591651).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: 3rd Party Data. Restrictions apply to the availability of these data. Data was obtained from the "XUE LONG" ship and are available from the authors with the permission of the "XUE LONG" ship.

Acknowledgments: The authors acknowledge the National Arctic and Antarctic Data Center, Polar Research Institute of China for supporting the ship voyage data and manual sea ice observation data for the present work. The authors also pay high tribute to the Chinese Polar Research Team.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Melia, N.; Haines, K.; Hawkins, E. Sea ice decline and 21st century trans-Arctic shipping routes. *Geophys. Res. Lett.* **2016**, *43*, 9720–9728. [CrossRef]
2. Aksenov, Y.; Popova, E.E.; Yool, A.; Nurser, A.J.G.; Williams, T.D.; Bertino, L.; Bergh, J. On the future navigability of Arctic sea routes: High-resolution projections of the Arctic Ocean and sea ice. *Mar. Policy* **2017**, *75*, 300–317. [CrossRef]
3. Fu, S.; Goerlandt, F.; Xi, Y. Arctic shipping risk management: A bibliometric analysis and a systematic review of risk influencing factors of navigational accidents. *Saf. Sci.* **2021**, *139*, 105254. [CrossRef]
4. Grochowalski, S.; Hermanski, G. Ship resistance and propulsion in ice-covered waters: An experimental study. *Trans. Soc. Nav. Archit. Mar. Eng.* **2011**, *119*, 67–92.
5. Mueller, F.; Schoyen, H. Polar research and supply vessel capabilities—An exploratory study. *Ocean Eng.* **2021**, *224*, 108671. [CrossRef]
6. Liu, J.; Yao, R.; Yu, L.; Gan, X.; Wang, X. Line design and optimization for polar expedition cruise ships with transoceanic voyage characteristics. *J. Mar. Sci. Technol.* **2023**, *28*, 270–287. [CrossRef]

7. Ni, B.; Chen, Z.; Zhong, K.; Li, X.; Xue, Y. Numerical Simulation of a Polar Ship Moving in Level Ice Based on a One-Way Coupling Method. *J. Mar. Sci. Eng.* **2020**, *8*, 692. [CrossRef]
8. Guo, C.; Zhang, C.; Feng, F.; Wang, C.; Wang, C. Predicting ship ramming performance in thick level ice via experiments. *Ships Offshore Struct.* **2022**, *17*, 2141–2149. [CrossRef]
9. Chen, C.; Liu, Y.; He, Y.; Chen, Z.; Zheng, G. Numerical analysis of added resistance in head waves on a Polar Research Vessel and conventional ships. *Ocean Eng.* **2021**, *233*, 108888. [CrossRef]
10. Leira, B.J.; Chai, W.; Radhakrishnan, G. On Characteristics of Ice Ridges and Icebergs for Design of Ship Hulls in Polar Regions Based on Environmental Design Contours. *Appl. Sci.* **2021**, *11*, 5749. [CrossRef]
11. Xie, C.; Zhou, L.; Ding, S.; Liu, R.; Zheng, S. Experimental and numerical investigation on self-propulsion performance of polar merchant ship in brash ice channel. *Ocean Eng.* **2023**, *269*, 113424. [CrossRef]
12. Tavakoli, Y.; Pena-Castillo, L.; Soares, A. A Study on the Geometric and Kinematic Descriptors of Trajectories in the Classification of Ship Types. *Sensors* **2022**, *22*, 5588. [CrossRef] [PubMed]
13. Jiang, B.; Zhou, W.; Guan, J.; Jin, J. A method for identifying marine targets based on mining of multi-characteristic movement patterns. *Comput. Electr. Eng.* **2021**, *95*, 107434. [CrossRef]
14. Zheng, J.; Yan, M.; Li, Y.; Huang, C.; Ma, Y.; Meng, F. An online identification approach for a nonlinear ship motion model based on a receding horizon. *Trans. Inst. Meas. Control* **2021**, *43*, 3000–3012. [CrossRef]
15. Montewka, J.; Goerlandt, F.; Lensu, M.; Kuuliala, L.; Guinness, R. Toward a hybrid model of ship performance in ice suitable for route planning purpose. *Proc. Inst. Mech. Eng. Part O J. Risk Reliab.* **2019**, *233*, 18–34. [CrossRef]
16. Goerlandt, F.; Montewka, J.; Zhang, W.B.; Kujala, P. An analysis of ship escort and convoy operations in ice conditions. *Saf. Sci.* **2017**, *95*, 198–209. [CrossRef]
17. Zhang, C.; Zhang, D.; Zhang, M.; Mao, W. Data-driven ship energy efficiency analysis and optimization model for route planning in ice-covered Arctic waters. *Ocean Eng.* **2019**, *186*, 106071. [CrossRef]
18. Chen, X.; Xu, X.; Yang, Y.; Huang, Y.; Chen, J.; Yan, Y. Visual ship tracking via a hybrid kernelized correlation filter and anomaly cleansing framework. *Appl. Ocean Res.* **2021**, *106*, 102455. [CrossRef]
19. Chen, X.; Ling, J.; Wang, S.; Yang, Y.; Luo, L.; Yan, Y. Ship detection from coastal surveillance videos via an ensemble Canny-Gaussian-morphology framework. *J. Navig.* **2021**, *74*, 1252–1266. [CrossRef]
20. Liu, H.; Xu, X.; Chen, X.; Li, C.; Wang, M. Real-Time Ship Tracking under Challenges of Scale Variation and Different Visibility Weather Conditions. *J. Mar. Sci. Eng.* **2022**, *10*, 444. [CrossRef]
21. Zhu, Y.; Zhou, S.; Feng, Y.; Hu, Z.; Yuan, L. Influences of solar energy on the energy efficiency design index for new; building ships. *Int. J. Hydrogen Energy* **2017**, *42*, 19389–19394. [CrossRef]
22. Zhao, P.; Liu, X.; Shi, W.; Jia, T.; Li, W.; Chen, M. An empirical study on the intra-urban goods movement patterns using logistics big data. *Int. J. Geogr. Inf. Sci.* **2020**, *34*, 1089–1116. [CrossRef]
23. Gagic, R.; Skuric, M.; Djukanovic, G.; Nikolic, D. Establishing Correlation between Cruise Ship Activities and Ambient PM Concentrations in the Kotor Bay Area Using a Low-Cost Sensor Network. *Atmosphere* **2022**, *13*, 1819. [CrossRef]
24. Weng, J.; Liao, S.; Wu, B.; Yang, D. Exploring effects of ship traffic characteristics and environmental conditions on ship collision frequency. *Marit. Policy Manag.* **2020**, *47*, 523–543. [CrossRef]
25. Fu, J.; Chen, X.; Wu, S.; Shi, C.; Wu, H.; Zhao, J.; Xiong, P. Mining ship deficiency correlations from historical port state control (PSC) inspection data. *PLoS ONE* **2020**, *15*, e0229211. [CrossRef] [PubMed]
26. Zambon, A.; Moro, L.; Kennedy, A.; Oldford, D. Torsional vibrations of Polar-Class shaftlines: Correlating ice-propeller interaction torque to sea ice thickness. *Ocean. Eng.* **2023**, *267*, 113250. [CrossRef]
27. Yuen, P.C.; Sasa, K.; Kawahara, H.; Chen, C. Statistical estimation of container condensation in marine transportation between Far East Asia and Europe. *J. Navig.* **2022**, *75*, 176–199. [CrossRef]
28. Guo, W.; Zhao, Q.; Tian, Y.; Zhang, W. The research on floe ice force acting on the ”Xue Long” icebreaker based on synthetic ice test and virtual mass numerical method. *J. Hydrodyn.* **2021**, *33*, 271–281. [CrossRef]
29. Mikulic, A.; Katalinic, M.; Corak, M.; Parunov, J. The effect of spatial correlation of sea states on extreme wave loads of ships. *Ships Offshore Struct.* **2021**, *16*, 22–32. [CrossRef]
30. Pennino, S.; Angrisano, A.; Della Corte, V.; Ferraioli, G.; Gaglione, S.; Innac, A.; Martellato, E.; Palumbo, P.; Piscopo, V.; Rotundi, A.; et al. Sea State Monitoring by Ship Motion Measurements Onboard a Research Ship in the Antarctic Waters. *J. Mar. Sci. Eng.* **2021**, *9*, 64. [CrossRef]
31. Zhang, Y.; Meng, W.; Xie, X.; Yang, S.; Hang, S.; Han, Y.; Hong, Z. Antarctic sea ice detection using a shipborne GPS reflectometry setup. *Meas. Control.* **2021**, *54*, 618–626. [CrossRef]
32. Montewka, J.; Sinclair, H.; Kujala, P.; Haapala, J.; Lensu, M. Modelling ship performance in ice using Bayesian Networks. In Proceedings of the 22nd International Conference on Port and Ocean Engineering under Arctic Conditions, POAC 2013, Espoo, Finland, 9–13 June 2013.
33. Mishra, P.; Alok, S.; Rajak, D.R.; Beg, J.; Bahuguna, I.M.; Talati, I. Investigating optimum ship route in the Antarctic in presence of sea ice and wind resistances—A case study between Bharati and Maitri. *Polar Sci.* **2021**, *30*, 100696. [CrossRef]
34. Kimmritz, M.; Counillon, F.; Bitz, C.M.; Massonnet, F.; Bethke, I.; Gao, Y. Optimising assimilation of sea ice concentration in an Earth system; model with a multicategory sea ice model. *Tellus Ser. A Dyn. Meteorol. Oceanogr.* **2018**, *70*, 1–23. [CrossRef]

35. Rong, H.; Teixeira, A.P.; Soares, C.G. Spatial correlation analysis of near ship collision hotspots with local maritime traffic characteristics. *Reliab. Eng. Syst. Saf.* **2021**, *209*, 107463. [CrossRef]
36. Tong, L.; Zhang, C.; Peng, Z.; Wang, L. Spatial-Temporal Distribution Characteristics and Correlation Analysis of Air Pollutants from Ships in Inland Ports. *Sustainability* **2022**, *14*, 14214. [CrossRef]
37. Schober, P.; Boer, C.; Schwarte, L.A. Correlation Coefficients: Appropriate Use and Interpretation. *Anesth. Analg.* **2018**, *126*, 1763–1768. [CrossRef]
38. Izumiyama, K. *Ship Performance in Ice*; John Wiley & Sons, Ltd.: Hoboken, NJ, USA, 2017.

Journal of
Marine Science and Engineering

Article

An Improved NSGA-II Based on Multi-Task Optimization for Multi-UAV Maritime Search and Rescue under Severe Weather

Yue Ma [1], Bo Li [2], Wentao Huang [3] and Qinqin Fan [1,*]

[1] Logistics Research Center, Shanghai Maritime University, Shanghai 201306, China
[2] Institute of Logistics Science and Engineering, Shanghai Maritime University, Shanghai 201306, China
[3] Key Laboratory of Control of Power Transmission and Conversion, Shanghai Jiao Tong University, Shanghai 200240, China
* Correspondence: forever123fan@163.com; Tel.: +86-135-2446-0557

Abstract: The international trade heavily relies on maritime transportation. Due to the vastness of the ocean, once an accident happens, fast maritime search and rescue (MSR) is a must, as it is of life-and-death matter. Using unmanned air vehicles (UAVs) is an effective approach to completing complex MSR tasks, especially when the environment is dangerous and changeable. However, how to effectively plan paths for multi-UAVs under severe weather, e.g., to rescue the most urgent targets in the shortest time, is a challenging task. In this study, an improved NSGA-II based on multi-task optimization (INSGA-II-MTO) is proposed to plan paths for multi-UAVs in the MSR tasks. In the INSGA-II-MTO, a novel population initialization method is proposed to improve the diversity of an initial population. Further, two tasks are introduced during the execution of the search algorithm. Namely, one assistant task, which solves a simplified MSR problem through multi-task optimization, is implemented to provide necessary evolutional knowledge to a main task that solves an original MSR problem. The performance of the proposed INSGA-II-MTO is compared with other competitors in three MSR scenarios. Experimental results indicate that the proposed algorithm performs best among the compared ones. It is observed that the INSGA-II-MTO can find a set of shorter total paths and handle the most urgent task in the shortest possible time. Therefore, the proposed method is an effective and promising approach to solving multi-UAVs MSR problems to reduce human causalities and property losses.

Keywords: maritime search and rescue; path planning; unmanned air vehicle; multi-objective optimization; non-dominated sorting genetic algorithm-II; multi-task optimization

Citation: Ma, Y.; Li, B.; Huang, W.; Fan, Q. An Improved NSGA-II Based on Multi-Task Optimization for Multi-UAV Maritime Search and Rescue under Severe Weather. *J. Mar. Sci. Eng.* **2023**, *11*, 781. https://doi.org/10.3390/jmse11040781

Academic Editor: Mihalis Golias

Received: 7 March 2023
Revised: 26 March 2023
Accepted: 1 April 2023
Published: 4 April 2023

1. Introduction

With the rapid development of the global economy and trade, the throughput and scale of cargo transportation between countries have sharply increased in recent years. Due to the low freight, large transport capacity, and strong adaptability, maritime transportation is playing an increasingly important role in cargo transportation [1,2]. However, because of the rough and unpredictable marine environment due to the ever-change weather, maritime accidents occur frequently [3], which has caused many human causalities [4], huge property losses [5], and terrible damages to the marine ecological environment [6].

To carry out the MSR tasks in complex marine environments, assigning manual driving ships is a traditional way, which has many limitations, such as high cost, low efficiency, and poor applicability. In comparison, using UAVs is a much more promising and effective method to solve MSR problems due to high mobility, wide view field, and no risk of injury or death [7,8]. More importantly, UAVs can adapt to various challenging marine environments. Until now, a large number of related studies have been proposed. For example, Raap et al. [9] proposed a novel model to achieve search-trajectory planning for a single dynamic target. Kilic and Mostarda [10] proposed a new framework for path planning by optimizing the charging station grid to use a single UAV to reach multiple static ships. To further make the planned path more suitable and realistic for MSR tasks, a weight related to the path based on the time of accident is introduced into the Ant Colony Optimization (ACO) [11]. It is worth noting that the best path obtained by the improved ACO may not be the shortest. However, a single UAV can only complete a limited number of tasks, and its efficiency is often low. Therefore, some research has focused on implementing a group of UAVs. For example, Yang et al. [12] proposed a cognitive mobile computing network composed of UAVs and unmanned surface vehicles (USVs) for collaborative MSR tasks. An iteration of the Markov decision process (MDP) is used in sub-areas to find the paths with the highest reward for multiple pieces of equipment. The experimental results show that this method can successfully avoid obstacles and find an optimal path. Subsequently, some studies pointed out that there should be priorities according to urgency level or high-value tasks among different MSR tasks [13,14]. Furthermore, Huang et al. [15] proposed a multi-objective maritime patrolling problem, in which the vessels that need to be visited are regarded as circular areas. Then, a novel method inspired by the immune-endocrine short feedback system is proposed to solve the problem. Compared with other algorithms, the proposed algorithm showed good global and local search abilities for a given instance.

Although the existing studies can effectively solve MSR problems, they considered a relatively simple marine environment model with relatively few MSR targets. Generally, the solution complexity of multi-objective MSR problems increases exponentially with the increased number of tasks. Additionally, the previous study [16] pointed out that severe weather is a key factor causing marine accidents. Therefore, the present study aims to propose an advanced algorithm to solve complex multi-objective MSR problems under severe weather conditions.

To implement the above objective, the path planning problems of multi-objective MSR under severe weather are proposed in the present study. Moreover, an improved NSGA-II based on multi-task optimization (INSGA-II-MTO) is proposed. In the INSGA-II-MTO, a novel population initialization method is proposed to improve population diversity. Additionally, a multi-task optimization method, which can share knowledge among different tasks, is incorporated into the improved NSGA-II. Specifically, the original MSR problem is considered the main task, and a simplified MSR problem is used as the assistant task. The performance of the proposed INSGA-II-MTO is compared with the other two competitors in three test scenarios. The simulation results demonstrate that the INSGA-II-MTO is superior to the compared algorithms in three MSR scenarios. Therefore, the proposed algorithm is an effective and competitive approach to solving complex multi-objective MSR problems under severe weather.

The main contributions of this study are as follows: (1) the complex multi-objective MSR problems under severe weather are proposed. Different from previous studies, the number of tasks is more; (2) an improved NSGA-II based on multi-task optimization (INSGA-II-MTO) is proposed, where the multi-task optimization method is used to share knowledge among different tasks to speed up the convergence of the algorithm.

The rest of this paper is organized as follows. Section 2 introduces some preliminary knowledge of multi-objective optimization problems. Section 3 reviews the related literature on MSR. The problem definition and mathematical model of the multi-objective MSR are given in Section 4. Section 5 describes the proposed INSGA-II-MTO. Section 6 demonstrates the effectiveness of this proposed algorithm in solving multi-objective MSR through comparison with other algorithms on three testing examples. Conclusions are shown in Section 7.

2. Preliminary Knowledge

2.1. Multi-Objective Optimization Problem

A multi-objective optimization problem (MOP) involves multiple conflicting objectives that need to be optimized concurrently. Moreover, these objectives cannot be achieved optimally at the same time [17].

Without loss of generality, the minimization MOP can be mathematically described as:

$$\min \boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_M(\boldsymbol{x}))^T \tag{1}$$

where $\boldsymbol{x} = (x_1, x_2, \ldots, x_D) \in \Omega$ is a D-dimensional decision variable; $\Omega \subset \mathbb{R}^D$ is the decision space; $\boldsymbol{f}(\boldsymbol{x})$ is an objective vector with M objective functions; $f(\boldsymbol{x}) \subset \mathbb{R}^M$ is the objective vector.

Some basic concepts of MOP are described as follows [18–20]:

Definition 1. *Dominant relationship. Suppose two variables* $\boldsymbol{u} = (u_1, u_2, \ldots, u_D)$, $\boldsymbol{v} = (v_1, v_2, \ldots, v_D)$, $\boldsymbol{u}$ *is called to dominate* $\boldsymbol{v}$ *(denoted as* $\boldsymbol{u} \succ \boldsymbol{v}$*) if and only if* $\boldsymbol{u}$ *is no more than* $\boldsymbol{v}$*, that is:*

$$\boldsymbol{u} \succ \boldsymbol{v} \Leftrightarrow \forall d \in \boldsymbol{\Phi}, \boldsymbol{u}_k \leq \boldsymbol{v}_k \wedge \exists b \in \boldsymbol{\Phi}, \boldsymbol{u}_h < \boldsymbol{v}_h \tag{2}$$

where $\boldsymbol{\Phi} = (1, 2, \ldots, D)$

Definition 2. *Pareto optimal solution set. Assuming a solution* $\boldsymbol{x}^* \in \Omega$, $\boldsymbol{x}^*$ *is called to be a Pareto optimal solution if and only if there are not any* $\boldsymbol{x}$ *satisfy* $\boldsymbol{f}(\boldsymbol{x}) \succ \boldsymbol{f}(\boldsymbol{x}^*)$*. All of Pareto optimal solutions compose a Pareto optimal set (PS), marked as* $\boldsymbol{X}^*$.

Definition 3. *Pareto front. The objective vector corresponding to the PS in the object space is called the Pareto frontier (PF), represented as* $\mathrm{PF} = \{\boldsymbol{f}(\boldsymbol{x}^*) \mid \boldsymbol{x}^* \in \boldsymbol{X}^*\}$.

To evaluate the performance of multi-objective evolutionary algorithms (MOEAs), various performance indicators have been proposed [21,22]. Two commonly used performance metrics are applied in the current study, i.e., the Hypervolume (HV) and the Non-dominance ratio (NR). The HV proposed by Zitzler and Thiele [23] is a comprehensive performance metric, which can evaluate the convergence and diversity of the PF approximation at the same time. A larger HV value means that the algorithm performance is better. Moreover, the NR proposed by Goh and Tan [24] is used to evaluate which MOEA can find more Pareto solutions. A larger NR means that the algorithm performance is better.

2.2. Non-Dominated Sorting Genetic Algorithm-II

The Non-dominated Sorting Genetic Algorithm (NSGA-II) [25] is one of the most popular MOEAs, and has been successfully used to solve MOPs in various fields [26,27]. The main steps of the NSGA-II are as follows [28]:

First, generate the initial population P randomly and calculate the fitness value of each individual. Then, the offspring population OP is generated by the binary tournament selection [25], crossover and mutation strategies. For the integer-coded NSGA-II, the two-point crossover and multi-point mutation operator [29] are commonly adopted to generate offspring individuals. For the real-coded NSGA-II, the Simulated Binary Crossover (SBX) and polynomial mutation (PM) [30] operator are commonly used to produce offspring individuals. Note that the SBX is an operator simulating single-point binary crossover. Assuming that $x^1 = (x_1^1, x_2^1, \ldots, x_D^1)$ and $\mathrm{x}^2 = (x_1^2, x_2^2, \ldots, x_D^2)$ are two parent individuals, two offspring individuals $c^1 = (c_1^1, c_2^1, \ldots, c_D^1)$ and $c^2 = (c_1^2, c_2^2, \ldots, c_D^2)$ can be obtained by the SBX:

$$\begin{gathered} \begin{cases} c_k^1 = 0.5 \times \left[(1+\beta) \cdot x_k^1 + (1-\beta) \cdot x_k^2\right] \\ c_k^2 = 0.5 \times \left[(1-\beta) \cdot x_k^1 + (1+\beta) \cdot x_k^2\right] \end{cases} \\ \beta = \begin{cases} (2 \times rand)^{\frac{1}{1+\eta_c}}, rand \leq 0.5 \\ \left(\frac{1}{2(1-rand)}\right)^{\frac{1}{1+\eta_c}}, otherwise \end{cases} \end{gathered} \tag{3}$$

where *rand* is a random number between 0 and 1; k is an integer between 1 and D; η_c is a customization parameter.

The mathematical formula of the PM is as follows:

$$\begin{array}{l} x_k^1 = x_k^1 + \Delta t \\ \Delta t = \begin{cases} (2 \times rand)^{\frac{1}{1+\eta u}} - 1, rand < 0.5 \\ 1 - [2 \times (1 - rand)]^{\frac{1}{1+\eta u}}, otherwise \end{cases} \end{array} \tag{4}$$

where η_u is a customization parameter.

Next, the parent and offspring populations are merged to form P_{all}, which is sized $2NP$. Subsequently, the non-dominated sorting and crowding distance method [25] are used to select NP individuals from P_{all} to obtain a new parent population. Repeat these steps until the termination conditions are met. The pseudocode of NSGA-II is given in Algorithm 1.

Algorithm 1: NSGA-II

```
Input: Population size: $NP$; maximum generation: $G$
1: Generate and evaluate the initial population $P$
2: Set $G = 1$
3: while termination criterion not satisfied do
4:     $OP \leftarrow$ Generate offspring by crossover and mutation strategies
5:     $P_{all} \leftarrow OP \cup P$
6:     $P \leftarrow$ Select $NP$ individuals from $P_{all}$ based on the non-dominated sorting and crowding distance
7:     $G = G + 1$
8: end while
Output: The PS and PF
```

3. Literature Review

To reduce the damages caused by maritime accidents and complete the MSR tasks, a large number of methods have been proposed in previous studies. As MSR stations are often set up on coastal lands and far from the accident scenes, determining their locations is an optimization problem. Namely, selecting appropriate MSR stations can improve the MSR capability and reduce potential losses. Zhou et al. [31] proposed a new framework to plan the locations of the MSR stations. First, the response time of candidate islands, which are determined according to potential demands, can be calculated. Then, the best islands are selected as the MSR stations by solving the maximal covering location problem (MCLP). Experimental results indicate that this method can improve the primary coverage and reduce rescue time. Peng et al. [32] proposed a hybrid algorithm, in which the tabu search algorithm is incorporated into the ACO to solve a bi-level programming model of the location-routing problem. The proposed hybrid algorithm performs better than the ACO.

Apart from the location optimization of MSR stations, a number of methods have been proposed to support the planning of search areas. For example, Otote et al. [33] proposed a decision-making model to implement an MSR plan based on the support of optimal search theory. Specifically, they introduced the concept of density ratio and a random detection function to improve the accuracy of probability of containment (POC) and probability of detection (POD), respectively. Experiments confirm that the values of POC and POD are obviously improved and can support MSR tasks. Based on the above studies, Xiong et al. [34] proposed a time domain-based iterative planning (TIP) method to plan the search areas. In the TIP, the probability map is updated based on the mean drift direction in each iteration. Subsequently, an iterative search method is used to determine the optimal search areas based on the grid with the highest POC in the map. It was concluded that the probability of success (POS) of the search areas obtained by the TIP is higher than other methods. However, the TIP may easily fall into local optima, especially when the potential search area is large. To alleviate this issue, a method based on the minimum bounding rectangle and K-means clustering (MBRK) is proposed [35]. In the MBRK, the K-means clustering method is used to divide the potential locations of survivors into multiple regions. Moreover, the minimum bounding rectangle is used to generate probability maps for each region. Finally, the optimal search areas can be further optimized by adding or subtracting multiple cells. Simulation experiments confirm that MBRK can further improve the POS values of the search areas.

Apart from the above studies in the MSR, task allocation and path planning are also two key tasks. For task allocation, Ai et al. [36] proposed a regional task allocation algorithm to quickly respond to maritime accidents. The proposed method considers both the temporal and spatial characteristics of task assignment. Based on experimental tests, the proposed algorithm can avoid repeated searches and improve MSR efficiency. For path planning, Cho et al. [37] proposed a two-stage coverage path planning (CPP) method for multiple UAVs to minimize the completion time. In the first stage, a grid-based decomposition method is used to decompose the search area into squares. In the second stage, a three-phases randomized search heuristic (RSH) algorithm is proposed to find optimal paths in large-scale instances. Extensive numerical experiments show that the RSH has a faster convergence speed than other methods. To make the moving direction more diverse, Cho et al. [38] proposed a mixed-integer linear programming (MILP) model based on a hexagonal grid-based decomposition method. According to the results, the proposed model can obtain high-quality paths. However, Ho et al. [39] established new MSR route models, which consider path length and people information in distress simultaneously. Moreover, the grey relational analysis is used to quantify the people's information and then convert them into distance weight, which is combined with the distance matrix of the Floyd–Warshall algorithm. The experiment shows that the model can be applied to the actual decision-making situation to provide reasonable schemes. In addition, it should be noted that task allocation and path planning are coupled in most cases. Therefore, Yan et al. [40] proposed an improved particle swarm optimization combined with a genetic algorithm (GA-PSO) to solve task allocation and path planning problems. Namely, the partial matching crossover and second transposition mutation are introduced to improve the performance. The simulation results show that this algorithm can improve the efficiency of task allocation and path planning when compared with competitors. Except for the application scenarios of MSR, some researchers have proposed search and rescue strategies for other scenarios, which can be used as references for MSR. Yazdani et al. [41] proposed using public transportation systems to improve evacuation capability in extreme weather disasters. Gharib et al. [42] developed a comprehensive model to plan the delivery of construction materials to post disaster reconstruction projects.

4. Multi-Objective Maritime Search and Rescue Problem under Severe Weather

Maritime accidents can cause huge damage and often occur under severe weather, such as storms. Therefore, how to carry out MSR tasks efficiently in a complex environment is important. However, the cost and time consumption of a task are usually conflicting goals in a rescue operation. Additionally, people or vessels in a high-risk area need to be visited as soon as possible. Therefore, a multi-objective MSR problem under severe weather is considered in the current study.

4.1. Description of Multi-Objective MSR Problem under Severe Weather

In the current study, a scenario of one MSR station equipped with a group of UAVs is used. If the MSR station receives signals for help from a vessel in danger, it will dispatch UAVs to provide various necessary assistance as well as to collect real-time information about the troubled vessel. This can help decision-makers develop the next rescue plans to reduce damages and save time. After completing the MSR tasks, the UAVs return to the MSR station. It should be noted that the UAV does not have to reach the exact position of the vessel. However, the position where the UAV actually arrives must be within the contact range of the vessel. Assuming that this position between the UAV and the vessel is called a node, and then a task is considered successfully completed when the UAV reaches a node. Additionally, a storm is considered in the current study. Therefore, the UAV should prioritize the vessels which are in the high-risk area affected by the storm. An example of the problem is illustrated in Figure 1. As shown in Figure 1, two UAVs are dispatched to accomplish the MSR tasks. One will visit two vessels, and another will visit three vessels. Moreover, the red circle represents the high-risk area affected by the storm.

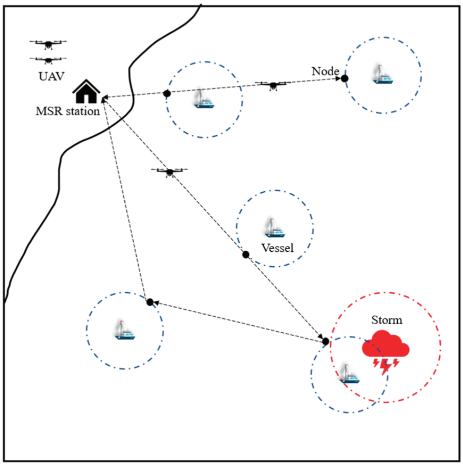


Figure 1. An example of multi-objective MSR problem under severe weather.

4.2. Mathematical Model of Multi-Objective MSR Problem under Severe Weather

Supposing that the position of an MSR station P is (x_P, y_P) and the number of UAVs is n. Moreover, there are q vessels in distress. The coordinate and contact range of the i-th vessel is represented as (x_i, y_i) and r_i, respectively. The center position of the storm S is (x_s, y_s) and its influencing radius is r_S. The j-th UAV is assigned to visit a set of vessels. Moreover, the nodes corresponding to these vessels are denoted as $T(j)$, in which N_i is denoted as the corresponding node of the i-th vessel. Therefore, all the targets visited by the j-th UAV is $W(j) = \{P, T(j), P\}$.

Furthermore, E_i is used to determine whether the i-th vessel is within the influencing scope of the storm. It is calculated by the following formula:

$$\begin{aligned} E_i &= \begin{cases} 1, d_{iS} < r_S \\ 0, oherwise \end{cases}, \\ d_{iS} &= \sqrt{(x_i - x_S)^2 + (y_i - y_S)^2} \end{aligned} \tag{5}$$

where d_{iS} denotes the Euclidean distance between the i-th vessel and the center position of the storm. If $E_i = 1$, the i-th vessel is within the influencing scope of the storm and should be visited as soon as possible. Otherwise, the i-th vessel is relatively safe.

If the j-th UAV is assigned to visit the vessels which should be visited as soon as possible, $\gamma_{j\max}$ is represented as the order number of the last urgent vessel in the visiting sequence. Therefore, the MSR station and the nodes of vessels whose order numbers are before $\gamma_{j\max}$ in j-th path compose a set $G(j)$. Moreover, $\delta_{uv} \in \{0, 1\}$ is a binary variable, which denotes the movement between the node u and v. If $\delta_{uv} = 1$, it means that UAV moves from node u to v. If $\delta_{uv} = 0$, it means that there is no path between the two nodes. Additionally, the d_{uv} stands for the Euclidean distance between the two nodes.

Based on the research of Huang et al. [15], the objectives of the model of multi-objective MSR problem under severe weather can be defined as follows:

$$\begin{aligned} \min F &= (f_1, f_2, f_3) \\ f_1 &= \sum_{j=1}^{n} d_{uv}\delta_{uv}, u, v \in W(j) \\ f_2 &= \max_{1 \le j \le n} d_{uv}\delta_{uv}, u, v \in W(j) \\ f_3 &= \max_{1 \le j \le n} d_{uv}\delta_{uv}, u, v \in G(j) \end{aligned} \tag{6}$$

The f_1 is to minimize the total path length of multi-UAVs. The shorter the total path length of UAVs, the less energy they consume. The f_2 aims at minimizing the total task completion time, which is equivalent to minimizing the longest single path length. The f_3 aims at minimizing the completion time of urgent tasks. That is, UAVs are required to visit the vessels within the affected area of the storm as early as possible. Concretely, it is represented to minimize the path length from the MSR station to the vessels which need priority.

5. The Proposed Method

Solving multi-objective MSR problems is a difficult task, thus it is important to develop advanced MOEAs. It is worth mentioning that multi-task optimization is considered an effective method to solve various complex optimization problems [43,44]. To effectively solve multi-objective MSR problems, an improved NSGA-II based on multi-task optimization (INSGA-II-MTO) is proposed in this study.

5.1. Encoding and Decoding Method

In this subsection, the encoding and decoding methods of the main and assistant tasks are introduced. Moreover, a boundary-based encoding approach [45] is adopted since it can reduce the size of the search space. The polar coordinate is used to represent the node position.

Individual encoding of the main task: The individuals in the main task are all encoded by real numbers, where each individual is 3 × q. Moreover, the individual is divided into three segments in the current study. The first segment (i.e., Chromosome Segment I) represents the task allocation of UAVs. It consists of q integer numbers within the range of [1, n]. The second segment (Chromosome Segment II) contains a series of vessel numbers, whose order will be mapped to the visiting sequence in the UAV paths. It also consists of q integer numbers within the range of [1, q]. To ensure that the UAVs traverse all vessels without repetition and omission, all genes in Chromosome Segment II are different from each other. The locations of nodes are indicated in the third segment (Chromosome Segment III), in which genes are represented as angles between the vessels and nodes. The Chromosome Segment III is composed of q floating-point numbers within the range of [0, 360]. An example of a main task containing 5 vessels and 3 UAVs is shown in Figure 2.

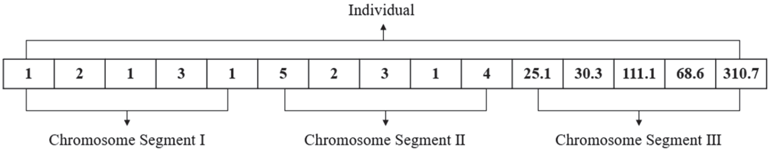


Figure 2. An example for encoding method of main task.

Individual encoding of the assistant task: Unlike the encoding of the main task, a new encoding for the simplified MSR problem (i.e., the target is regarded as an ideal point) is used. Therefore, the length of each individual in the assistant task is 2 × q. All the individuals are encoded by integer numbers. Moreover, the individual is divided into two segments. The first segment (i.e., Chromosome Segment I) represents the task allocation of

UAVs, and the second segment (i.e., Chromosome Segment II) denotes the visiting sequence of each UAV. An example of an assistant task containing 5 vessels and 3 UAVs is shown in Figure 3.

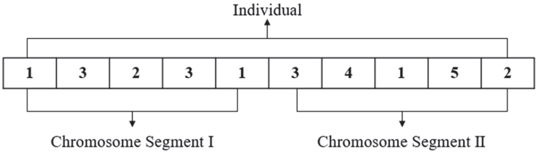


Figure 3. An example of encoding method of assistant task.

Since the encoding method of the assistant task is the same as that of part of the main task, the decoding methods of three chromosome segments in the main task are given as follows:

Chromosome Segment I: Each gene represents the number of a UAV, i.e., which UAV will visit the matching vessel. The same genes mean that these vessels are assigned to the same UAV.

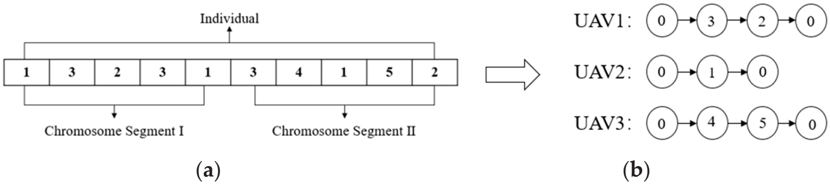
Chromosome Segment II: Each gene represents the number of a vessel. For a given UAV, its visiting sequence is determined by the corresponding order in the Chromosome Segment II.

Chromosome Segment III: Each node location is computed by the following formulas:

$$\begin{aligned} x_{N_i} &= x_i + r_i \cos\theta_i \\ y_{N_i} &= y_i + r_i \sin\theta_i \end{aligned} \tag{7}$$

where θ_i is represented as the angle between the i-th vessel and its corresponding node; x_{N_i} and y_{N_i} are the abscissa and ordinate of the node, respectively.

To further illustrate the decoding method of Chromosome Segments I and II, an example is given in Figure 4. It can be seen from Figure 4a that, for the Chromosome Segment I, UAV "2" visits one vessel, and UAVs "1" and "3" visit two vessels, respectively. It should be noted that the MSR station does not necessarily dispatch all UAVs for MSR tasks. For the Chromosome Segment II, the visiting sequence of UAV "2" is "0→1→0"; the visiting sequences of the UAVs "1" and "3" are "0→3→2→0" and "0→4→5→0", respectively. All paths are shown in Figure 4b, where "0" is denoted as an MSR station.



(**a**) (**b**)

Figure 4. The decoding method. (**a**) An individual for decoding. (**b**) The corresponding decoded paths.

5.2. Population Initialization Method

A good initial population can accelerate the convergence of the algorithm and improve the quality of final solutions. Therefore, how to generate a high-quality initial population is vital. Although a randomly generated population can enhance the exploration ability of algorithms in the early stage, it has some limitations in solving MSR problems. Specifically, the probability of generating Chromosome Segment I with identical genes via random initializing is extremely small. Namely, it is hard to generate individuals that dispatch one UAV to accomplish the MSR tasks. In addition, this probability will be further reduced with the expansion of search space.

To alleviate the above problem, a two-stage population initialization method is proposed, which is described as follows:

Step 1: The total number of UAVs and population size is set to n and NP, respectively. Generally, NP is much greater than n. The number of individuals of each particular type is defined as $Num = [NP/n]$. Then Num individuals that dispatch one UAV and Num individuals that dispatch two UAVs are generated, respectively.

Step 2: For the remaining individuals in the initial population, they are generated randomly.

Figure 5 shows an example of the proposed initialization method containing 5 vessels and 3 UAVs. It can be observed from Figure 5 that the first Num individuals are generated to dispatch one UAV to complete the MSR tasks. The next Num individuals are generated to dispatch two UAVs. Moreover, there is a wide variety of individuals among the rest of the population.

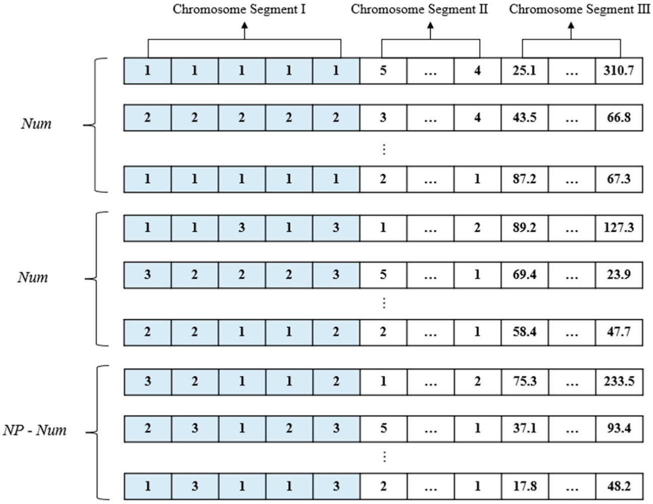
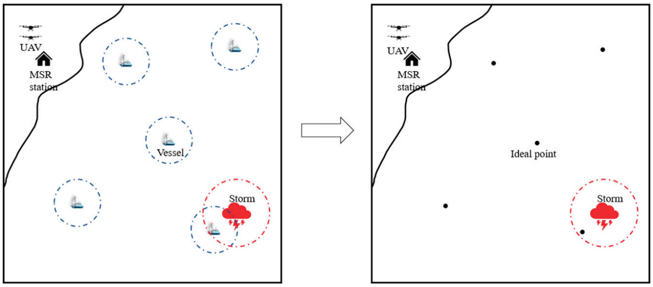


Figure 5. The proposed population initialization method.

5.3. *Knowledge Transfer in the Multi-Task Optimization*

The multi-task optimization is an effective method to solve complex problems, thus it is incorporated into the proposed algorithm to improve the performance in solving multi-objective MSR problems. A main task and an assistant task are used in the INSGA-II-MTO. The schematic diagrams of the original and simplified MSR tasks are shown in Figure 6. It can be seen from Figure 6a that the main task focuses on solving the original MSR problems. Figure 6b shows that the visiting target can be considered as an ideal point by ignoring the contact range, thus the assistant task can focus on solving the simplified MSR problem.



(**a**) (**b**)

Figure 6. The scenes of the original and simplified MSR tasks. (**a**) The scene of the original MSR task. (**b**) The scene of the simplified MSR task.

According to Section 5.1, the encoding methods of these two tasks are not identical. Therefore, the knowledge between the main task and the assistant task cannot be transferred directly. To illustrate the transfer process, an example is given in Figure 7.

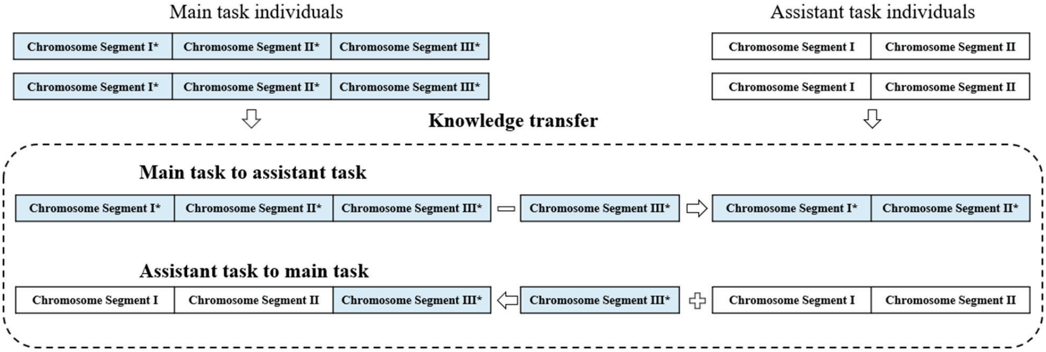


Figure 7. The transfer process of multi-task optimization. The symbol "*" denotes the chromosome segment comes from a main task individual.

Knowledge transfer from main task to assistant task: As mentioned in Section 5.1, the individual in the main task has three parts (i.e., Chromosome Segments I, II, and III), while there are two components of the individual in the assistant task. Therefore, if the knowledge of the main task is transferred to the assistant task, then the third part (i.e., Chromosome Segment III) should be deleted. It can be observed from Figure 7 that, to achieve knowledge transfer from the main task to the assistant task, the Chromosome Segment III of the individual in the main task is deleted to obtain a transferred individual, which can help the assistant task.

Knowledge transfer from assistant task to main task: The first two parts of individuals between two tasks are versatile. However, individuals in the assistant task do not contain the third part. Therefore, if the knowledge of the assistant task is transferred to the main task, the third part of the individual in the main task can be added to the transferred individual. It can be observed from Figure 7 that a Chromosome Segment III is randomly selected as a reference from the PS of the main task, which can provide high-quality node information. Then, combine this Chromosome Segment III with the assistant task individual to form a transferred individual.

5.4. The Overall Process of the INSGA-II-MTO

In the INSGA-II-MTO, the main task and the assistant task are incorporated to improve the NSGA-II. It optimizes the main task and the assistant task independently and simultaneously, and the two tasks are expected to assist each other during the search process through a knowledge transfer process in multi-task optimization. Its pseudocode is described in Algorithm 2.

In line 1, generate the initial populations P_1 and P_2 with NP individuals according to the proposed initialization method. In line 2, all individuals in P_1 are evaluated via the original multi-objective MSR problem; in line 3, all individuals in P_2 are evaluated via the simplified multi-objective MSR problem.

Next, the main loop begins. In lines 5–6, select $NP/2$ individuals from P_1 and P_2 via the binary tournament method, which are denoted as MP_1 and MP_2. Then, the SBX and PM are adopted to generate an offspring population OP_1 sized $NP/2$ in line 7; the two-point crossover and multi-point mutation operator are adopted to generate an offspring population OP_2 sized $NP/2$ in line 8. After generating two offspring populations, in lines 9–10, OP_1 and OP_2 are evaluated via the original multi-objective and simplified multi-objective MSR problem, respectively. Subsequently, knowledge sharing between two tasks is achieved by transferring individuals. In line 11, generate $OP_{1\text{new}}$ as a transferred population by deleting the Chromosome Segment III in OP_1; in line 12, generate $OP_{2\text{new}}$ as another transferred population by randomly adding the Chromosome Segment III to the individual in OP_2. In lines 13–14, P_1, OP_1 and $OP_{2\text{new}}$ are combined as $P_{1\text{all}}$; P_2, OP_2 and $OP_{1\text{new}}$ are combined as $P_{2\text{all}}$. Next, in lines 15–16, NP individuals are selected from $P_{1\text{all}}$ and $P_{2\text{all}}$ for the next iteration according to the non-dominated sorting and crowding distance, respectively. Finally, the PS and PF of the main task is output when the termination condition is satisfied.

Algorithm 2: INSGA-II-MTO

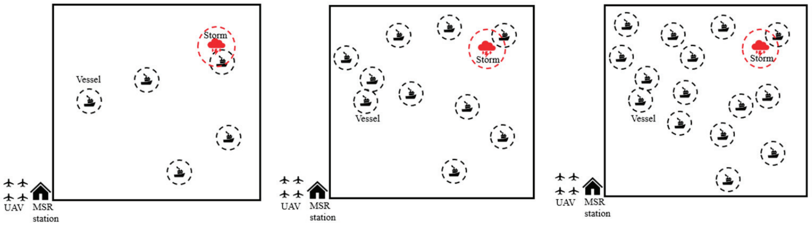
```
Input: Population size: $NP$; maximum generation: $G_{\max}$; population of the main task: $P_1$;
population of the assistant task: $P_2$
1: Initialize $P_1$ and $P_2$ of size $NP$ via the proposed initialization method in Section 5.2
2: Evaluate $P_1$ on the original multi-objective MSR
3: Evaluate $P_2$ on the simplified multi-objective MSR
4: while termination criterion not satisfied do
5:     $MP_1 \leftarrow$ Select $NP/2$ individuals from $P_1$ using binary tournament selection method
6:     $MP_2 \leftarrow$ Select $NP/2$ individuals from $P_2$ using binary tournament selection method
7:     $OP_1 \leftarrow$ Generate $NP/2$ offspring by $MP_1$ according to SBX and PM
8:     $OP_2 \leftarrow$ Generate $NP/2$ offspring by $MP_2$ according to two-point crossover and multi-point
mutation operator
9:     Evaluate $OP_1$ on the original multi-objective MSR problem
10:    Evaluate $OP_2$ on the simplified multi-objective MSR problem
11:    $OP_{1\text{new}} \leftarrow OP_1$ delete the Chromosome Segment III to generate a transferred population
according to;
12:    $OP_{2\text{new}} \leftarrow OP_2$ randomly add the Chromosome Segment III to generate a transferred
population according to Section 5.3;
13:    $P_{1\text{all}} \leftarrow P_1 \cup OP_1 \cup OP_{2\text{new}}$;
14:    $P_{2\text{all}} \leftarrow P_2 \cup OP_2 \cup OP_{1\text{new}}$;
15:    $P_1 \leftarrow$ Select $NP$ individuals from $P_{1\text{all}}$ based on the non-dominated sorting and crowding
distance
16:    $P_2 \leftarrow$ Select $NP$ individuals from $P_{2\text{all}}$ based on the non-dominated sorting and crowding
distance
17: end while
Output: The PS and PF of the main task
```

6. Experimental Results and Analysis

In this section, the performance of the INSGA-II-MTO is verified via comparing with the other excellent multi-objective optimization algorithms in three scenarios. All of the following experiments are tested via MATLAB R2021.

Three task scenarios are used and set to be 1000 × 1000 km, which are shown in Figure 8. Although the three scenarios are the same, the number of vessels is different. It can be seen from Figure 8a that 5 vessels should be visited. Figure 8b,c show that UAVs should visit 10 and 15 vessels. Clearly, the solution difficulty of the three scenarios is different. Moreover, the number of UAVs is set to be 4 in the MSR station in all scenarios. In addition, the radius of the affected range of the storm and the contact range of each vessel are set to 120 km and 60 km, respectively.

To ensure the fairness of comparisons, for all compared algorithms, the total number of fitness evaluations in each scenario is the same, i.e., 20,000 is set in scenario 1 (see Figure 8a), 40,000 is set in scenario 2, and 60,000 is set in scenario 3. Each comparison algorithm runs 20 times independently in each scenario. Moreover, the performance metrics HV and NR are selected in the following experiments. The same reference point is set for three algorithms in each MSR scenario. Additionally, parameter settings of all compared algorithms are shown in Table 1.



(**a**) (**b**) (**c**)

Figure 8. Three MSR task scenarios. (**a**) Scenario 1. (**b**) Scenario 2. (**c**) Scenario 3.

Table 1. Parameter settings of three compared algorithms.

Name	Value	Algorithms
pc_1	0.9	INSGA-II-MTO, NSGA-II, NSGA-II-GLS
pm_1	0.6	INSGA-II-MTO, NSGA-II, NSGA-II-GLS
pc_1	0.8	INSGA-II-MTO
pm_1	1	INSGA-II-MTO
NP	100	INSGA-II-MTO, NSGA-II, NSGA-II-GLS

6.1. Comparison Results with Other Algorithms

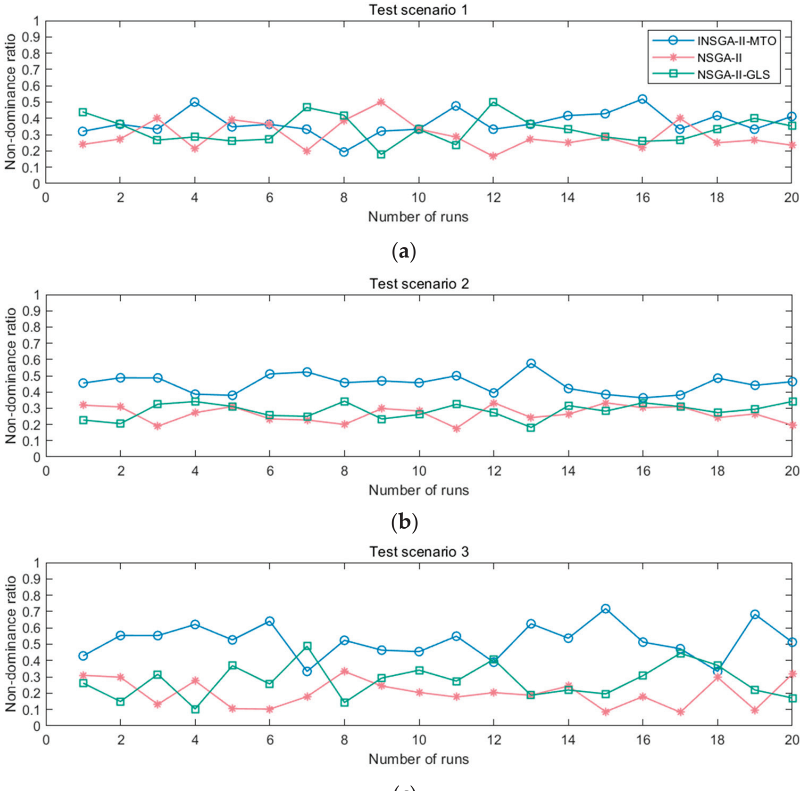
To verify the performance of the proposed algorithm, it is compared with the NSGA-II [25] and the NSGA-II-GLS [46] in three scenarios. The NSGA-II is a classical algorithm for solving MOPs, and the NSGA-II-GLS is an improved version of the NSGA-II.

The average and standard values of HV of the three algorithms are provided in Table 2. The best results are marked in bold, and Wilcoxon's rank sum test is employed to analyze the results from the statistical perspective. Specifically, the symbols "+", "−", and "=" represent that the compared algorithms are better than, worse than, and similar to the INSGA-II-MTO, respectively. From the results shown in Table 2, it can be observed that the performance of the proposed algorithm is superior to that of the two compared algorithms. Due to the relative simplicity of scenario 1, HV obtained by the three algorithms are similar. However, Table 2 shows that the INSGA-II-MTO significantly outperforms the NSGA-II and the NSGA-II-GLS in scenarios 2 and 3 in terms of HV. The main reason may be that the initialization method can improve the diversity of the solutions and the performance of an algorithm. In addition, the knowledge transfer in multi-task optimization can enhance the search capability and save computational resources.

Table 2. HV results of all comparison algorithms in three scenarios.

	Test Scenario 1	Test Scenario 2	Test Scenario 3
INSGA-II-MTO	0.1929 (2.46×10^{-4})	0.2846 (1.1×10^{-2})	0.2567 (1.2×10^{-2})
NSGA-II	0.1923 (5.8×10^{-4}) +	0.2099 (2.5×10^{-2}) +	0.1571 (4.1×10^{-2}) +
NSGA-II-GLS	0.1924 (5.98×10^{-4}) +	0.2211 (3.6×10^{-2}) +	0.1774 (3.9×10^{-2}) +

Besides the above comparisons, the other performance indicator NR is employed to assess the performance of three algorithms. Specifically, NR is used to explore the dominant relationship between the optimal solutions obtained by three algorithms. The NR values in three scenarios are plotted in Figure 9. From Figure 9a, the NR values obtained by the three algorithms are similar, while Figure 9b,c show that the NR values of the INSGA-II-MTO are larger than that of the other two algorithms during twenty experiments. This means that the INSGA-II-MTO can find more high-quality solutions which can dominate the most of non-dominated solutions obtained by the other two algorithms. Same as the above experiment, this is mainly because using the initialization method and multi-task optimization method is effective. It is worth noting that this advantage will be more evident in relatively complex scenarios.



(**a**)

(**b**)

(**c**)

Figure 9. The NR values of three algorithms. (**a**) The NR values obtained by three algorithms in scenario 1. (**b**) The NR values obtained by three algorithms in scenario 2. (**c**) The NR values obtained by three algorithms in scenario 3.

Based on the above analyses, it can be concluded that the INSGA-II-MTO is an effective and efficient approach to solving complex multi-objective MSR problems.

6.2. Experimental Analysis

As mentioned above, the population initialization method and the multi-task optimization method are used to improve the performance of the proposed algorithm. In this subsection, their effectiveness is verified by the following experiments.

(1) The effectiveness of the population initialization method

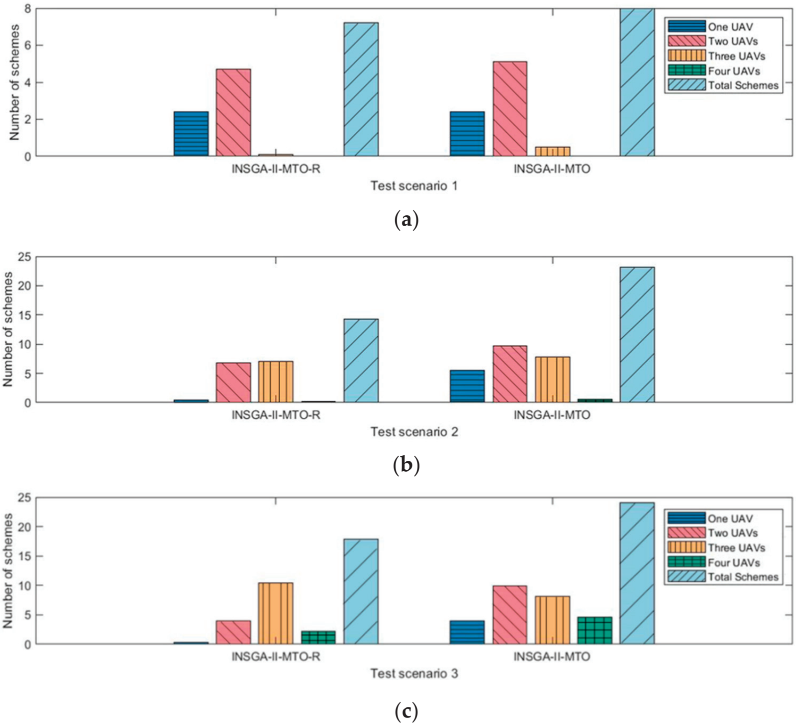
To illustrate the effectiveness of the initialization method, the INSGA-II-MTO and INSGA-II-MTO without the proposed initialization method (denoted as INSGA-II-MTO-R) are applied to solve three multi-objective MSR problems.

The average and standard values of HV achieved by the two algorithms are provided in Table 3. As shown in Table 3, the HV of the INSGA-II-MTO is better than that of the INSGA-II-MTO-R in all three scenarios. Therefore, the population initialization method can help the proposed algorithm improve the search performance and provide a high-quality initial population.

Table 3. HV results of the INSGA-II-MTO and INSGA-II-MTO-R.

	Test Scenario 1	Test Scenario 2	Test Scenario 3
INSGA-II-MTO	0.1929 (2.46×10^{-4})	0.2846 (1.1×10^{-2})	0.2567 (1.2×10^{-2})
INSGA-II-MTO-R	0.1926 (3.7×10^{-4}) +	0.2465 (3.0×10^{-2}) +	0.1908 (3.1×10^{-2}) +

Furthermore, the average number of various schemes obtained by two algorithms in twenty experiments is shown in Figure 10. “One UAV” means that the station dispatches one UAV to complete MSR tasks; “Two UAVs” represents two UAVs dispatched to complete tasks. “Three UAVs” and “Four UAVs” refer to schemes that dispatch three UAVs and four UAVs, respectively. The average number of schemes that can be obtained in experiments by each algorithm is denoted as “Total Schemes”. From Figure 10a, except for the “Four UAVs”, the INSGA-II-MTO and INSGA-II-MTO-R can find three other schemes. It can be seen from Figure 10b,c, the number of “One UAVs” and “Four UAVs” obtained by the INSGA-II-MTO is larger than the INSGA-II-MTO-R. Moreover, the results show that the total number of schemes obtained by the INSGA-II-MTO is always more than the INSGA-II-MTO-R in all three scenarios. Therefore, the proposed population initialization method can assist the proposed algorithm in finding more different schemes.



(**a**)

(**b**)

(**c**)

Figure 10. The number of various schemes obtained in experiments. (**a**) The number of schemes obtained by two algorithms in scenario 1. (**b**) The number of schemes obtained by two algorithms in scenario 2. (**c**) The number of schemes obtained by two algorithms in scenario 3.

Based on the above analyses, it can be inferred that the proposed population initialization method is effective and can help the proposed algorithm find more diverse and high-quality schemes.

(2) The effectiveness of the multi-task optimization

To verify the effectiveness of the multi-task optimization, the INSGA-II-MTO without assistant task (called INSGA-II-MTO-I) and the INSGA-II-MTO are used to solve three multi-objective MSR problems.

Table 4 lists the HV values of two algorithms in three scenarios. It can be seen from Table 4 that the performance of the INSGA-II-MTO-I and the INSGA-II-MTO is similar in scenario 1. The main reason may be that scenario 1 is relatively simple, and the multi-task cannot play an important role. However, Table 4 shows that the INSGA-II-MTO outperforms the INSGA-II-MTO-I in the remaining two scenarios. Compared with the first scenario, these two cases are more complex. Therefore, it can be concluded that multi-task optimization can help the proposed algorithm improve its performance in solving complex scenarios. This may mainly be because the computational resources of the INSGA-II-MTO-I are evenly distributed in a huge search space, which will lead to a large amount of wasted computational resources. However, the INSGA-II-MTO can allocate more resources in a specific search space due to the existence of an assistant task, which can contribute greatly to the performance.

Table 4. HV results of the INSGA-II-MTO and INSGA-II-MTO-I.

	Test Scenario 1	Test Scenario 2	Test Scenario 3
INSGA-II-MTO	0.1929 (2.46×10^{-4})	0.2846 (1.1×10^{-2})	0.2567 (1.2×10^{-2})
INSGA-II-MTO-I	0.1926 (2.8×10^{-4}) +	0.2593 (1.2×10^{-2}) +	0.2272 (1.8×10^{-2}) +

To further test the effectiveness of the multi-task optimization method, the proposed algorithm is used to solve the MSR task in scenario 2. Moreover, the successfully transferred individuals in each generation between two tasks are illustrated in Figure 11. "Individuals from assistant task" represents the number of individuals successfully transferred from the assistant task to the main task. "Individuals from main task" means the number of individuals successfully transferred from the main task to the assistant task. It can be observed from Figure 11 that the knowledge can be effectively transferred between the two tasks. This means that the multi-task optimization method can greatly improve search efficiency. Figure 11 also shows that the assistant task can provide a lot of useful knowledge to the main task in the early and middle stages. Especially, the number of individuals from the assistant task is extraordinary more than 4/5 at the first generation. As the assistant task may contain more useful knowledge when compared with the main task in the beginning, this result is as expected. In addition, the problem that the genetic algorithm is easy to fall into the local optimization can be alleviated by the multi-task optimization in the subsequent generations.

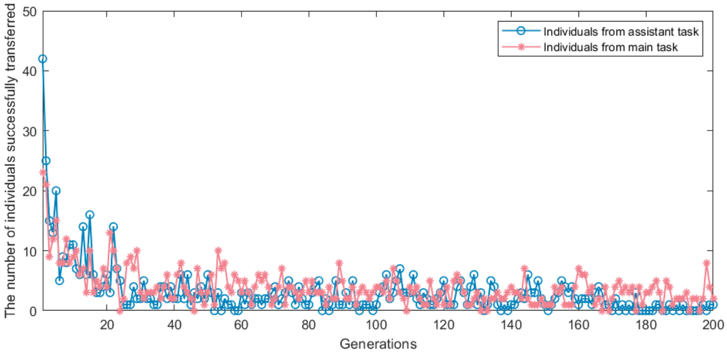
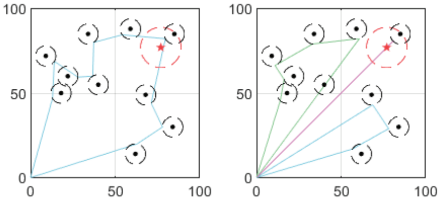


Figure 11. The number of individuals successfully transferred of INSGA-II-MTO.

Based on the above experimental analyses, for complex multi-objective MSR problems, multi-task optimization is an effective approach for improving the performance of the proposed algorithm.

6.3. Diversity of the Solutions

To demonstrate the performance of the INSGA-II-MTO, scenario 2 is used in this experiment. Moreover, two typical solutions are illustrated in Figure 12 and their objective values are shown in Table 5.



(**a**) (**b**)

Figure 12. Two typical solutions in scenario 2. (**a**) One UAV is used to complete the MSR task. (**b**) Three UAVs are dispatched to complete the MSR task.

Table 5. The typical objective values.

	Objective 1	**Objective 2**	**Objective 3**
(a)	297.876	297.876	138.7735
(b)	636.1917	228.4096	114.2048

From Table 5 and Figure 12, we can see that the total path length in the first row of Table 5 is the shortest. However, the total completion time of all tasks and urgent tasks is the longest. Moreover, Figure 12a shows that the MSR station dispatched one UAV to complete the tasks. Clearly, using one UAV can save cost, but it will reduce the efficiency of MSR. It is suitable for situations where decision-makers pursue low cost. It can be observed from Figure 12b that the MSR station dispatches three UAVs to visit vessels. Moreover, Table 5 indicates that the total path length of the second scheme is the longest, while the completion time is the shortest and the vessels in the risk region can be visited in the shortest possible time. This scheme pays more attention to the urgency of the tasks. Therefore, the MSR cost and the MSR efficiency are conflicting. Decision-makers can flexibly select different schemes to adapt to different MSR environments based on the obtained solution set.

7. Conclusions

Severe weather is the main factor leading to marine accidents, thus solving MSR problems under unpredictable weather conditions is an urgent problem. Using UAVs is an effective method to complete MSR tasks facing a harsh environment. In the present study, an improved NSGA-II based on multi-task optimization (INSGA-II-MTO) is proposed to solve the multi-objective MSR problem under severe weather. A main task aims to solve the original MSR problem, and an assistant task aims to solve the simplified MSR problem. Moreover, the knowledge can be transferred between the two tasks. Additionally, a novel population initialization method is proposed to improve population diversity. The experimental results show that the INSGA-II-MTO obtained maximum values in terms of HV and NR in three testing cases, that is, the proposed algorithm outperforms the other two competitors. In addition, the effectiveness of the multi-task optimization and the population initialization method is verified. Namely, the INSGA-II-MTO can find shorter total paths and handle the most urgent task in the shortest possible time. Therefore, the proposed algorithm is a competitive method to solve complex MSR problems under severe weather for providing decision-makers with effective MSR schemes when a marine accident occurs.

For future work, the energy consumption constraints of UAVs and the time window of tasks will be considered in the MSR tasks. Further, UAV-Ship collaborative systems can also be studied for the MSR tasks. Moreover, it is promising to extend this work to other application scenarios such as data collection.

Author Contributions: Conceptualization, Y.M. and Q.F.; methodology, Y.M. and Q.F.; writing—original draft preparation, Y.M. and Q.F.; writing—review and editing, Y.M., Q.F., B.L. and W.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the Shanghai Pujiang Program (No. 22PJD030), the National Nature Science Foundation of China (Nos. 61603244, 71904116), and the National Natural Science Foundation of China-Shandong joint fund (U2006228).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ma, W.; Ma, D.; Ma, Y.; Zhang, J.; Wang, D. Green maritime: A routing and speed multi-objective optimization strategy. *J. Clean. Prod.* **2021**, *305*, 127179. [CrossRef]
2. Chen, X.; Wei, C.; Zhou, G.; Wu, H.; Wang, Z.; Biancardo, S.A. Automatic Identification System (AIS) Data Supported Ship Trajectory Prediction and Analysis via a Deep Learning Model. *J. Mar. Sci. Eng.* **2022**, *10*, 1314. [CrossRef]
3. Ai, B.; Jia, M.; Xu, H.; Xu, J.; Wen, Z.; Li, B.; Zhang, D. Coverage path planning for maritime search and rescue using reinforcement learning. *Ocean. Eng.* **2021**, *241*, 110098. [CrossRef]
4. Ng, C.T.; Cheng, T.C.E.; Levner, E.; Kriheli, B. Optimal bi-criterion planning of rescue and evacuation operations for marine accidents using an iterative scheduling algorithm. *Ann. Oper. Res.* **2020**, *296*, 407–420. [CrossRef]
5. Cai, L.; Wu, Y.; Zhu, S.; Tan, Z.; Yi, W. Bi-level programming enabled design of an intelligent maritime search and rescue system. *Adv. Eng. Inform.* **2020**, *46*, 101194. [CrossRef]
6. Chen, X.; Liu, S.; Liu, R.W.; Wu, H.; Han, B.; Zhao, J. Quantifying Arctic oil spilling event risk by integrating an analytic network process and a fuzzy comprehensive evaluation model. *Ocean. Coast. Manag.* **2022**, *228*, 106326. [CrossRef]
7. Zhang, H.; Dou, L.; Xin, B.; Chen, J.; Gan, M.; Ding, Y. Data Collection Task Planning of a Fixed-Wing Unmanned Aerial Vehicle in Forest Fire Monitoring. *IEEE Access* **2021**, *9*, 109847–109864. [CrossRef]
8. Liu, H.; Ge, J.; Wang, Y.; Li, J.; Ding, K.; Zhang, Z.; Guo, Z.; Li, W.; Lan, J. Multi-UAV Optimal Mission Assignment and Path Planning for Disaster Rescue Using Adaptive Genetic Algorithm and Improved Artificial Bee Colony Method. *Actuators* **2021**, *11*, 4. [CrossRef]
9. Raap, M.; Zsifkovits, M.; Pickl, S. Trajectory optimization under kinematical constraints for moving target search. *Comput. Oper. Res.* **2017**, *88*, 324–331. [CrossRef]
10. Kilic, K.I.; Mostarda, L. Heuristic Drone Pathfinding Over Optimized Charging Station Grid. *IEEE Access* **2021**, *9*, 164070–164089. [CrossRef]
11. Zhang, H.; Sun, J.; Yang, B.; Shi, Y.; Li, Z. Optimal search and rescue route design using an improved ant colony optimization. *Inf. Technol. Control* **2020**, *49*, 438–447. [CrossRef]
12. Yang, T.; Jiang, Z.; Sun, R.; Cheng, N.; Feng, H. Maritime Search and Rescue Based on Group Mobile Computing for Unmanned Aerial Vehicles and Unmanned Surface Vehicles. *IEEE Trans. Ind. Inform.* **2020**, *16*, 7700–7708. [CrossRef]
13. Jin, X.; Zhang, G.; Yang, M.; Peng, X.; Shi, B. Cooperative multi-task traversing with complex marine environment for multiple unmanned surface vehicles inspired by membrane computing. *Ocean. Eng.* **2022**, *266*, 112586. [CrossRef]
14. Jin, X.; Er, M.J. Cooperative path planning with priority target assignment and collision avoidance guidance for rescue unmanned surface vehicles in a complex ocean environment. *Adv. Eng. Inform.* **2022**, *52*, 101517. [CrossRef]
15. Huang, L.; Zhou, M.; Hao, K. Non-Dominated Immune-Endocrine Short Feedback Algorithm for Multi-Robot Maritime Patrolling. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 362–373. [CrossRef]
16. Huang, J.; Han, N.; Jiang, B.Q.; Cao, Y.; Han, X.T. Water search and rescue (SAR) for ship accidents in China: Analysis of 12 years' data. *Ann. Transl. Med.* **2022**, *10*, 1207. [CrossRef]
17. Konak, A.; Coit, D.W.; Smith, A.E. Multi-objective optimization using genetic algorithms: A tutorial. *Reliab. Eng. Syst. Saf.* **2006**, *91*, 992–1007. [CrossRef]
18. Fan, Q.; Yan, X. Solving Multimodal Multiobjective Problems through Zoning Search. *IEEE Trans. Syst. Man Cybern. Syst.* **2021**, *51*, 4836–4847. [CrossRef]
19. Fan, Q.; Ersoy, O.K. Zoning Search with Adaptive Resource Allocating Method for Balanced and Imbalanced Multimodal Multi-Objective Optimization. *IEEE/CAA J. Autom. Sin.* **2021**, *8*, 1163–1176. [CrossRef]

20. Ji, H.; Chen, S.; Fan, Q. Zoning Search and Transfer Learning-based Multimodal Multi-objective Evolutionary Algorithm. In Proceedings of the 2022 IEEE Congress on Evolutionary Computation (CEC), Padua, Italy, 18–23 July 2022; pp. 1–8.
21. Tian, Y.; Cheng, R.; Zhang, X.; Li, M.; Jin, Y. Diversity Assessment of Multi-Objective Evolutionary Algorithms: Performance Metric and Benchmark Problems [Research Frontier]. *IEEE Comput. Intell. Mag.* **2019**, *14*, 61–74. [CrossRef]
22. He, Z.; Yen, G.G. Visualization and Performance Metric in Many-Objective Optimization. *IEEE Trans. Evol. Comput.* **2016**, *20*, 386–402. [CrossRef]
23. Zitzler, E.; Thiele, L. Multiobjective evolutionary algorithms a comparative case study and the strength Pareto approach. *IEEE Trans. Evol. Comput.* **1999**, *3*, 257–271. [CrossRef]
24. Goh, C.K.; Tan, K.C. A Competitive-Cooperative Coevolutionary Paradigm for Dynamic Multiobjective Optimization. *IEEE Trans. Evol. Comput.* **2009**, *13*, 103–127. [CrossRef]
25. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [CrossRef]
26. Song, M.; Chen, D. An improved knowledge-informed NSGA-II for multi-objective land allocation (MOLA). *Geo-Spat. Inf. Sci.* **2018**, *21*, 273–287. [CrossRef]
27. Liu, X.; Sun, J.; Zheng, L.; Wang, S.; Liu, Y.; Wei, T. Parallelization and Optimization of NSGA-II on Sunway TaihuLight System. *IEEE Trans. Parallel Distrib. Syst.* **2021**, *32*, 975–987. [CrossRef]
28. Huang, Y.; Fei, M. Motion Planning of Robot Manipulator Based on Improved NSGA-II. *Int. J. Control. Autom. Syst.* **2018**, *16*, 1878–1886. [CrossRef]
29. Wang, Z.; Liu, L.; Long, T.; Wen, Y. Multi-UAV reconnaissance task allocation for heterogeneous targets using an opposition-based genetic algorithm with double-chromosome encoding. *Chin. J. Aeronaut.* **2018**, *31*, 339–350. [CrossRef]
30. Deb, K.; Agrawal, R.B. Simulated Binary Crossover for Continuous Search Space. *Complex Syst.* **1995**, *9*, 115–148.
31. Zhou, X.; Cheng, L.; Zhang, F.; Yan, Z.; Ruan, X.; Min, K.; Li, M. Integrating Island Spatial Information and Integer Optimization for Locating Maritime Search and Rescue Bases: A Case Study in the South China Sea. *ISPRS Int. J. Geo-Inf.* **2019**, *8*, 88. [CrossRef]
32. Peng, Z.; Wang, C.; Xu, W.; Zhang, J. Research on Location-Routing Problem of Maritime Emergency Materials Distribution Based on Bi-Level Programming. *Mathematics* **2022**, *10*, 1243. [CrossRef]
33. Agbissoh Otote, D.; Li, B.; Ai, B.; Gao, S.; Xu, J.; Chen, X.; Lv, G. A Decision-Making Algorithm for Maritime Search and Rescue Plan. *Sustainability* **2019**, *11*, 2084. [CrossRef]
34. Xiong, P.; Liu, H.; Tian, Y.; Chen, Z. A Time Domain-Based Iterative Method for Helicopter Maritime Search Area Planning and Construction of the Simulation Environment. *IEEE Access* **2020**, *8*, 191460–191471. [CrossRef]
35. Xiong, P.; Liu, H.; Tian, Y.; Chen, Z.; Wang, B.; Yang, H. Helicopter maritime search area planning based on a minimum bounding rectangle and K-means clustering. *Chin. J. Aeronaut.* **2021**, *34*, 554–562. [CrossRef]
36. Ai, B.; Li, B.; Gao, S.; Xu, J.; Shang, H. An Intelligent Decision Algorithm for the Generation of Maritime Search and Rescue Emergency Response Plans. *IEEE Access* **2019**, *7*, 155835–155850. [CrossRef]
37. Cho, S.W.; Park, H.J.; Lee, H.; Shim, D.H.; Kim, S.Y. Coverage path planning for multiple unmanned aerial vehicles in maritime search and rescue operations. *Comput. Ind. Eng.* **2021**, *161*, 107612. [CrossRef]
38. Cho, S.W.; Park, J.H.; Park, H.J.; Kim, S. Multi-UAV Coverage Path Planning Based on Hexagonal Grid Decomposition in Maritime Search and Rescue. *Mathematics* **2021**, *10*, 83. [CrossRef]
39. Ho, W.C.; Shen, J.H.; Liu, C.P.; Chen, Y.W. Research on Optimal Model of Maritime Search and Rescue Route for Rescue of Multiple Distress Targets. *J. Mar. Sci. Eng.* **2022**, *10*, 460. [CrossRef]
40. Yan, M.; Yuan, H.; Xu, J.; Yu, Y.; Jin, L. Task allocation and route planning of multiple UAVs in a marine environment based on an improved particle swarm optimization algorithm. *EURASIP J. Adv. Signal Process.* **2021**, *2021*, 1–23. [CrossRef]
41. Yazdani, M.; Mojtahedi, M.; Loosemore, M. Enhancing evacuation response to extreme weather disasters using public transportation systems: A novel simheuristic approach. *J. Comput. Des. Eng.* **2020**, *7*, 195–210. [CrossRef]
42. Gharib, Z.; Yazdani, M.; Bozorgi Amiri, A.; Tavakkoli Moghaddam, R.; Taghipourian, M.J. Developing an integrated model for planning the delivery of construction materials to post-disaster reconstruction projects. *J. Comput. Des. Eng.* **2022**, *9*, 1135–1156. [CrossRef]
43. Gupta, A.; Ong, Y.S.; Feng, L. Multifactorial Evolution: Toward Evolutionary Multitasking. *IEEE Trans. Evol. Comput.* **2016**, *20*, 343–357. [CrossRef]
44. Feng, L.; Zhou, L.; Zhong, J.; Gupta, A.; Ong, Y.S.; Tan, K.C.; Qin, A.K. Evolutionary Multitasking via Explicit Autoencoding. *IEEE Trans. Cybern.* **2019**, *49*, 3457–3470. [CrossRef]
45. Yang, Z.; Xiao, M.; Ge, Y.; Feng, D.; Zhang, L.; Song, H.; Tang, X. A double-loop hybrid algorithm for the traveling salesman problem with arbitrary neighbourhoods. *Eur. J. Oper. Res.* **2018**, *265*, 65–80. [CrossRef]
46. Zhang, Z.; Lu, B. Improving NSGA-II by a Local Search Strategy with Gaussian Mutation. In Proceedings of the 2021 40th Chinese Control Conference (CCC), Shanghai, China, 26–28 July 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1628–1633. [CrossRef]

Journal of
Marine Science and Engineering

Article

Modelling, Linearity Analysis and Optimization of an Inductive Angular Displacement Sensor Based on Magnetic Focusing in Ships

Zhipeng Li, Bonan Wang *, Xianbin Wang, Chao Zhang and Xu Meng

School of Traffic and Transportation, Northeast Forestry University, Harbin 150040, China;
lizp-nefu@nefu.edu.cn (Z.L.); xbwang10@l63.com (X.W.); 15663712907@163.com (C.Z.);
mengxu2016@nefu.edu.cn (X.M.)
* Correspondence: wangbonan126@126.com

Abstract: A sensor for measuring the crankshaft angle of the main engine in ships is designed. Compared with the existing crankshaft angle encoder, this design's advantage is that there is no need to add a gear system at the free end of the crankshaft, reducing machining complexity. The purpose of providing high angle resolution over a wide speed range is achieved. Inductive angular displacement sensors (IADSs) require an eddy current magnetic field as a medium to generate the induced voltage. The induced voltage also requires a complex linearization calculation to obtain a linear relationship between angle and voltage. Therefore, a model of the inductive angular displacement sensor based on magnetic focusing (IADSMF) is proposed. Magnetic focusing is introduced into the IADS to replace the eddy current magnetic field with a focusing magnetic field. The main disadvantage of traditional IADSs, which is that they cannot reduce the eddy current magnetic field, is mitigated. An approximate square–shaped focusing magnetic field (12.4 × 12.4 mm^2) is formed using the magnetic field constraint of the magnetic conductor. When the receiving coil undergoes a position change relative to the square–shaped focusing magnetic field, the voltage generated via the receiving coil is measured using the electromagnetic induction principle to achieve angular displacement measurement. A mathematical model of the IADSMF is derived. Induced voltages at different frequencies and rotational speeds are simulated and analyzed via MATLAB. The results show that frequency is the main factor affecting the induced voltage amplitude. The sensitivity of the IADSMF is 0.2023 mV/°. The resolution and measurement of the IADSMF range from 0.06° and 0–360°. Compared with a conventional planar coil–based IADS, the eddy current loss is reduced from 2.1304 to 0.3625 W. Direct linearization of the angular displacement with the induced voltage is achieved through designing a square–shaped focusing field and receiving coil. After optimizing the sensor structure with the optimization algorithm, the linearity error is 0.6012%. Finally, this sensor provides a theoretical basis and research ideas for IADS development in ships and navigation.

Keywords: crankshaft angle of marine main engines; angular displacement sensor; magnetic focusing; induced voltage analysis; linearity error optimization; eddy current loss

Citation: Li, Z.; Wang, B.; Wang, X.; Zhang, C.; Meng, X. Modelling, Linearity Analysis and Optimization of an Inductive Angular Displacement Sensor Based on Magnetic Focusing in Ships. *J. Mar. Sci. Eng.* **2023**, *11*, 1028. https://doi.org/10.3390/jmse11051028

Academic Editor: Rafael Morales

Received: 25 April 2023
Revised: 8 May 2023
Accepted: 10 May 2023
Published: 11 May 2023

1. Introduction

With the rapid development of sensor technology, angular displacement sensors have become an important research direction. The detection of angular displacement is related to the performance of ship systems and offshore platforms, as expressed in terms of energy consumption, service life and safety [1–4]. The detection of angular displacement plays an important role in modern ships, navigation and ocean transportation [5,6].

Angular displacement measurement is widely used in the shipbuilding industry [7,8] to provide an estimation of the ship test [9], ship turning performance [10], underwater transmission [11], localization and detection of targets in underwater [12], mooring lines [13] and a berthing assistant system [14].

Currently, most of the propulsion systems used in ships are driven using diesel engines. Most of the diesel engines are manufactured under license from MAN Diesel and Wärtsilä Corporation, which are the world's largest marine diesel engine manufacturers. Crankshaft angle sensor is an important part of the main engine. It provides crankshaft position information to the electronic control unit (ECU) for the purpose of starting the main engine and controlling the injection timing, injection volume, injection pressure and opening and closing of the exhaust valve.

The gear at the free end of the main engine of the ship has hundreds of teeth, and the crankshaft angle encoder is above the gear [15]. The encoder gives a rectangular voltage signal due to the change in gear air gap. This tool records the number of high and low electrical levels, and the final measurement of the angular position is achieved. The system has several disadvantages: (1) the crankshaft has to be machined with teeth; and (2) if a higher resolution is needed, the gear should be reprocessed.

Angle, torque and power are important control parameters in ship propulsion systems. They determine whether the ship can operate normally, as well as its fuel consumption and safety. It is necessary to measure and monitor these factors. The torque can be calculated using the torsion angle on the rotating shaft, and the power is calculated using the torque through the formula [16,17]. Thus, the premise of monitoring them is to measure the angle.

The common ways to measure the torque on the main engine are as follows.

(1) Torque is measured through measuring the surface deformation of the rotating shaft during engine operation. The use of strain gauges as sensors is one of the most common [15]. Four half-bridge strain gauges are affixed directly to the rotating shaft. The measurement principle for torque is based on the relationship between shear stress and normal stress for pure torsion. The torque is then calculated using Equation (1):

$$M = \frac{\varepsilon}{\pi \cdot G r^3}, \tag{1}$$

where M—torque, ε—strain, G—shear modulus and r—shaft radius.

(2) This method is based on an optical sensor and, through two teeth fixed on the shaft, the distance is "1" teethed rings, achieving torque measurement. This measurement method requires only one optical sensor [18]. The relationship between teethed rings is proportional to the torsion angle of the shaft. When no torque is applied to the shaft, the torsion angle $\varphi = 0$. At this time, the pulse period T_1 and T_2 of the two tooth rings corresponding to the optical sensor are equal. When there is a torque applied to the shaft, $\varphi \neq 0$ and $T_1 \neq T_2$. The torque is then calculated using Equation (2):

$$M = \frac{k_T \cdot (T_1 - T_2)}{n}, \tag{2}$$

where k_T—coefficient depends on construction of shaft and teethed wheels, n—revolution and T_1, T_2—time of pulses received from optical sensor.

There are several disadvantages of using this method (1): the measurement method of strain gauge is contact measurement, and contact friction due to long-term use will cause the measurement results to deteriorate; (2): a geared system is required, and the number of teeth of the gearing after processing is fixed. Thus, machining becomes more difficult as the resolution of the measurement angle increases.

Therefore, this paper proposes an inductive angular displacement sensor based on magnetic focusing (IADSMF) to replace the crankshaft angle encoder. The aim of this study is to provide high angular resolution over a wide speed range without using machining tooth rings. Furthermore, through adding an IADSMF to the other end of the shaft, the torsion angle φ and torque of the shaft system can be measured [17]. Not only does the sensor have the function of measuring angles, but it can also be extended to measure torque. Ref. [15] mentioned that the cost of measuring shaft torque is five times higher than measuring angular displacement in a ship's main engine system. As a result, versatility is addressed while measurement costs are reduced.

1.1. Related Work

1.1.1. Angular Displacement Sensor

According to how angular displacement measurement is implemented, angular displacement sensors can be divided into contact and non–contact sensors. Traditional contact sensors are mainly potentiometric and resistance strain sensors [19–21]. The main disadvantages are contact friction, large differences in measurement results due to the attachment process, limited resolution, and strain gauge creep after long-term use. These problems can lead to increased wear, affecting measurement accuracy and service life.

Non–contact sensors mainly include optical sensors, capacitive sensors [22,23], magneto–resistive sensors [24–26] and inductive sensors [27,28]. Optical sensors are categorized as photoelectric encoders [29–31] or grating sensors [32–34]. Their advantages are high measurement resolution and accuracy. Their disadvantages [35] are structural complexity and expense. They also have special requirements in specific environments. Their disadvantages limit their application in measurement. Capacitive sensors [22] have the advantages of high sensitivity, such as a simple structure and small non–linear error, but the disadvantages are evident. For example, capacitive sensors are prone to parasitic capacitance, leading to low measurement accuracy and poor load capacity that is susceptible to external interference, affecting sensor stability. Magneto–resistive sensors [24,25] are sensitive to changes in orientation over an angular measurement range of 0–180°. Their measurement results are more accurate than those of potentiometric sensors, and they are widely used in automotive pedal detection. However, the anisotropy of their magneto–resistive shapes can cause harmonic distortion, which affects measurement.

An inductive angular displacement sensor (IADS) is an electromagnetic sensor [27,36–38] that typically uses a printed circuit board (PCB) to support a stator and rotor made from a metal sheet. The stator contains an excitation coil and m receiving coils. The physical space angle difference of each receiving coil is 1/m of the angle corresponding to a measurement period, where the m value is generally 3. The excitation coil generates a magnetic field, and rotation of the rotor causes electromagnetic induction between the rotor and receiving coil. The change in position causes the receiving coil to produce an induced voltage. The magnitude of the induced voltage depends on the rotor position, allowing measurement of the angular displacement. An IADS has the following advantages; (1) it has a simple structure and no friction loss, no parasitic capacitance, a moderate price, and no special requirements for operating environments; (2) the sensor is composed of PCBs and metal sheets without additional materials; (3) the measuring range of angular displacement can reach 360°, which is larger than that of a magneto–resistive sensor.

Table 1 shows the advantages and disadvantages of non–contact angular displacement sensors. In summary, the IADS is an ideal sensor for engineering applications. IADS is especially suitable for the harsh environment of navigation ships, which are characterized by high temperatures, humid air, poor ventilation and high salinity gas erosion.

Table 1. Advantages and disadvantages of non–contact angular displacement sensors.

	Optical	**Magneto–Resistive**	**Capacitive**	**Inductive**
Measurement range	0–360°	0–180°	0–360°	0–360°
Output is linear or not	No	No	Yes	Yes
Measurement resolution	0.35°	0.08°	0.1°	0.15°
Manufacturing difficulty	Harder	Moderate	Moderate	Simple
External environmental impact	Large	Small	Moderate	Small

1.1.2. Magnetic Focusing and Application

Magnetic focusing in magnetism is the spiral line motion of charged particles in a magnetic field. It is implemented when a charged particle beam enters uniform magnetic field B with an initial velocity v. Moreover, the angle between v and B is small, with $v_{//} = v\cos\theta \approx v$, $v_{\perp} = v\sin\theta \approx v\theta$. Thus, each particle has a spiral motion. Since the $v_{//}$ of each particle is almost the same, the pitch is approximately equal. Therefore, particle beams converge at the same point after completing the spiral motion.

With the development of technology, material processing techniques using focus–adjustable laser beams and electron beams as high–energy density heat sources emerged in the field of magnetics. Among them, high–energy electron beam processing technology has the advantages of a vacuum environment and high energy utilization rate, and was applied to metal additive manufacturing technology [39]. In recent years, the use of the above–mentioned particle beam motion drove technological developments in the microscopic field, such as the processing of microscopic materials [40,41]. Moreover, the technology's application for electron microscopes and scanning electron microscopes are growing [42,43].

Magnetic focusing is generally achieved using the magnetic field excited in a current-carrying solenoid. In most practical applications, the non–uniform magnetic field excited via the coil is commonly used to achieve magnetic focusing. Magnetic focusing has long been used in physics and medicine. For example, Siegbahn et al. [44] studied the focusing of two–dimensional electrons in an inhomogeneous magnetic field. Walton et al. [45] used a uniform magnetic field with a straight–line boundary for high–order focusing, enabling a mass spectrograph to better separate and detect charged ions. Damadian et al. [46] visualized tumours in living animals through field-focusing nuclear magnetic resonance (FONAR). In recent years, magnetic focusing was widely used in transcranial magnetic stimulation [47] (TMS), magnetic resonance imaging [48], magnetic drug targeting [49–51] (MDT), underground pipeline localization [52], metal soldering [53], metal surface detection [54] and quantum science [55].

Traditionally, the TMS magnetic focusing process used a figure–eight coil [56], double–butterfly coil [57] or other structures [58]. Following years of development, TMS can now be used to activate neurons [59], thus playing a vital role in the treatment of neurological disorders and rehabilitation. Philip et al. [60] reported on the use of TMS to stimulate tissues, such as the amygdala, deep in the brain. This study involved the treatment of advanced brain stimulation. TMS based on magnetic focusing has even made positive contributions in stuttering improvement [61] and detoxification [62]. The localization and image acquisition of cerebral haemorrhages in patients was achieved via portable magnetic resonance imaging [48]. While simplifying the detection process, TMS reduces the potential risk of patients exposed to a high–intensity magnetic field environment. MDT [50] is a process that uses the magnetic force of an external focusing magnetic field to remotely deliver a drug containing magnetic particles to a lesion site. Non–invasive or minimally invasive drug therapy was achieved through MDT, avoiding the potential risks associated with surgery.

Using the magnetic focusing method (MFM) [63], a relatively high magnetic field can be formed in a tiny area on an outer metal surface using the needle tip of a magnetic conductor (MC). When the MC scans a metal surface with defects, the stable magnetic field is disturbed and changed. The induced signal in the outer coil of the MC is made to change, and metal surface defect detection is realized. The MFM, thus, became a mainstream nondestructive testing method. Chen et al. [55] simplified quantum processing devices using ultra–high enhanced field–matter interaction generated via magnetic focusing. This study promoted the application of magnetic focusing in integrated quantum information processing and high–sensitivity quantum sensing.

In summary, magnetic focusing is widely used in medicine, equipment testing, materials science and quantum science. To the best of our knowledge, it is less commonly used in measurement applications, especially in angular displacement sensors with ships.

1.2. Contribution

The main contributions of this paper are as follows.

(1) An IADS based on the MFM is proposed. A breakthrough in the measurement method of the IADS is realized. Compared with the existing crankshaft angle sensors in the ship, the angle resolution is improved. The angular resolution is improved from the existing 0.35 or 0.5 degrees to 0.06 degrees when the speed is 100 r/min.

(2) A mathematical model of the proposed sensor is derived, and the accuracy and feasibility of the sensor are verified theoretically via simulations.

(3) Direct linearization of the induced voltage and angular displacement is realized using this sensor. Compared with the traditional IADS case, there is no need for linearization design and calculation. The corresponding linearization error is only 0.6239%.

(4) The sensor has a lower eddy current loss than the traditional IADS. The eddy current loss is reduced from 2.103 W to 0.3625 W.

2. Methods and Models

2.1. Excitation Methods and Sensor Modelling

The common excitation methods for magnetic focusing and IADS are shown in Table 2. As shown in Table 2, most of the traditional focusing methods are two–dimensional planes or combinations of coil arrays. They only complete the coil to focus the magnetic field at a point below it, achieving an increase in the magnetic field at only one point. The magnetic field and its magnetic flux leakage in a certain area of the plane is ignored. Based on this defect and combined with the excitation method of IADS, this paper proposed the combination of Archimedes coil, hollow–core MC and solid–core MC. Due to the emergence of hollow– and solid–core MC, the focusing effect was increased and the surrounding magnetic flux leakage was reduced. Focusing was achieved in a certain area below the MC. The magnetic flux leakage outside the area was reduced to a negligible level, and this method was used to meet the measurement requirements of IADSMF.

Table 2. Magnetic focusing and IADS excitation method.

	Magnetic Focusing	**IADS**	**IADSMF**
Excitation method	Figure–eight coil [56]	Circular coil [27,64]	Archimedes coil, hollow–core MC and solid–core MC
	Double–butterfly coil [57]	Semicircular arc coil [65]	
	Hemispherical solenoid array [66]	Four square Archimedes coils [67]	
	C–type core coil [68]	Planar spiral coil [69]	
	Coil arrays in hemispherical, plane and torus shapes [70]	-	
	Biconical stimulation coil system [71]	-	

The sensor was modeled using the modelling function that came with COMSOL software. The sensor model is shown in Figure 1. The modelling parameters are shown in Table 3. The sensor stator was composed of two Archimedes coils, a solid–core MC and two hollow–core MCs. The Archimedes coil was arranged on the PCB. Two PCBs arranged with Archimedes coils were taped to each end of a solid–core MC (see Figure 1 for a schematic of the sensor structure). The Archimedes coil was used as an excitation source for generating a magnetic field. The functions of the solid–core MC were to conduct magnetism and constrain the Archimedes coil. The PCB with Archimedes coil was adhered to solid–core MC surface to prevent detachment. Constraining the magnetic field and reducing magnetic flux leakage were the roles of the hollow–core MC as a magnetic conduction device and shielding layer. The cross–section of the hollow–core MC was square–shaped; thus, the Archimedes coil on the inner wall was located 1 mm below the bottom, generating a square–shaped magnetic field (12.4 × 12.4 mm^2).

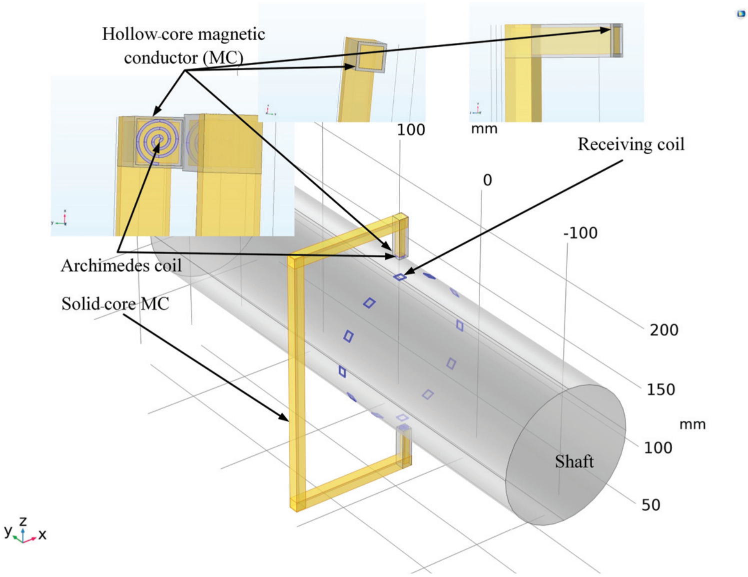


Figure 1. Sensor structure diagram.

Table 3. The modelling parameters.

Parameter Name	Value (mm)	Material
Radial side length of solid–core MC	8	Iron
Radial thickness of hollow–core MC	1	Iron
Maximum radius of Archimedes coil	4	Copper
Receiving coil side length	12.4	Copper
Radius of rotating shaft	47.4	Iron

A flexible printed circuit (FPC) had characteristics of arbitrary bending, and FPC technology was used to manufacture the receiving coil. An FPC was wound on the surface of the shaft to realize the requirement of the receiving coil being wound around the shaft. Through wrapping the FPC around the shaft, the need for the rotor to wrap around the shaft circumference was satisfied. The receiving coil in the FPC was square–shaped, and the position of one coil between two adjacent receiving coils was empty. The size of the receiving coil was the same as that of the square–shaped magnetic field. The rotor consists of 12 receiving coils with a difference of 15° between adjacent receiving coils. These dimensions represented the overall structural design of the sensor detailed in this subsection. Through focusing the magnetic field, the direct linearization of the induced voltage and angular displacement and sensor loss reduction can be realized via this design.

2.2. Receiving Coil Structure Arrangement

The planar structure of the FPC receiving coils was designed as shown in Figure 2a. The FPC had 4 layers. Odd receiving coils were arranged in the first layer, and even receiving coils were arranged in the second layer (Figure 2b). Wires connected to odd coils were arranged in the third layer, and wires connected to even coils were arranged in the fourth layer.

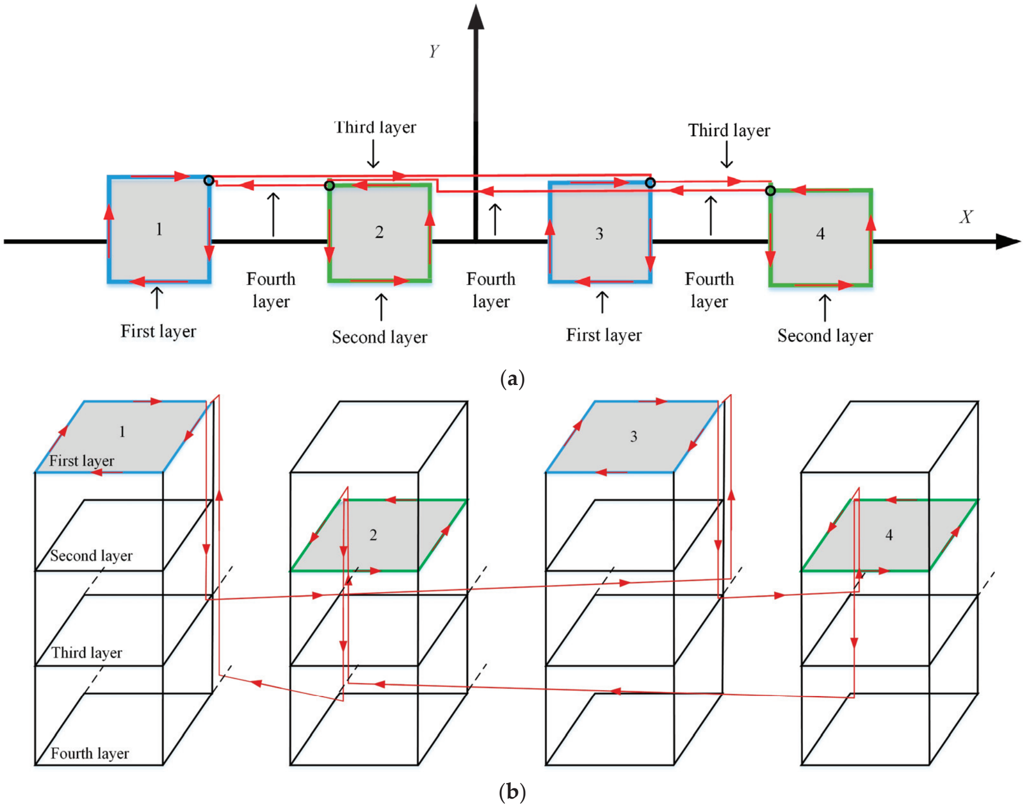


Figure 2. FPC receiving coil plane connection diagram: (**a**) two-dimensional diagram; (**b**) three-dimensional diagram.

The induced currents in odd coils, such as receiving coils 1 and 3, moved in the same direction, while the induced currents in even coils, such as receiving coils 2 and 4, also moved in the same direction. The induced current directions of the odd and even receiving coils were opposite. The odd coil and even coils were connected to form a complete set of receiving coils. Black circles indicate the position of the over-holes between the different layers (Figure 2a). Every over–hole position had a small offset from the receiving coil of this layer, which ensured that the two sides of the same receiving coil did not overlap.

The red arrow represents the current direction. The blue line represents the winding method of the odd receiving coils. The green line represents the winding method of the even receiving coils. The red line represents the winding method of connecting the receiving coil wires (Figure 2b). In Figure 2b, the small offset of the over-hole is enlarged for clarity.

The spatial arrangement was such that the center points of all receiving coils were located on the same axis (*X*–axis), allowing the receiving coils on the shaft surface to be arranged stably for one week. We noted that to show the staggered relationship between the receiving coil and the wire, the center point of the receiving coil (shown in Figure 2a) was not on the same axis (*X*–axis). However, the center points of the actual FPC receiving coils were on the same axis.

2.3. Measurement Principle and Mathematical Modelling

2.3.1. Measurement Principle of Angular Displacement

According to the principle of electromagnetic induction, the induced voltage u [72] can be expressed as in (3):

$$u = -N\frac{d\psi}{dt} = -N\frac{d}{dt}\boldsymbol{B}_{\mathrm{Z}} \cdot S, \tag{3}$$

where ψ and N represent the magnetic flux and the turn number, respectively; $\boldsymbol{B}_{\mathrm{Z}}$ is the magnetic flux density in the Z–axis direction; and S is the receiving coil area.

A stable focusing magnetic field (square) was formed using the MFM, and the receiving coil shape was the same as it (Figure 3). When the focusing magnetic field did not coincide with the receiving coil (position 1), there was no magnetic field in the area enclosed using the receiving coil; thus, the magnetic flux was zero. When the shaft rotated, the area where the receiving coil coincided with the focusing magnetic field increased, and the magnetic flux began to increase (the receiving coil was between positions 1 and 2). When the focusing magnetic field reached position 2, the receiving coil completely coincided with the focusing magnetic field, and the negative magnetic flux was maximum. The receiving coil continued to move. When the focusing magnetic field was between positions 2 and 3, the overlap area between the receiving coil and the focusing magnetic field decreased. When the focusing magnetic field reached position 3, the magnetic flux was 0. When the focusing magnetic field reached between 3 and 4, the positive magnetic flux began to increase. When the focusing magnetic field reached position 4, the receiving coil coincided completely with the focusing magnetic field for the second time, and the positive magnetic was maximum (the magnetic flux at position 2 was opposite to that at position 4, which was achieved through winding the receiving coil, as shown in Figure 2). When the focusing magnetic field was located between positions 4 and 5, the overlap area between the receiving coil and the focusing magnetic field decreased, and the forward magnetic flux decreased. When the focusing magnetic field reached position 5, the receiving coil did not coincide with the focusing magnetic field, and the magnetic flux was 0.

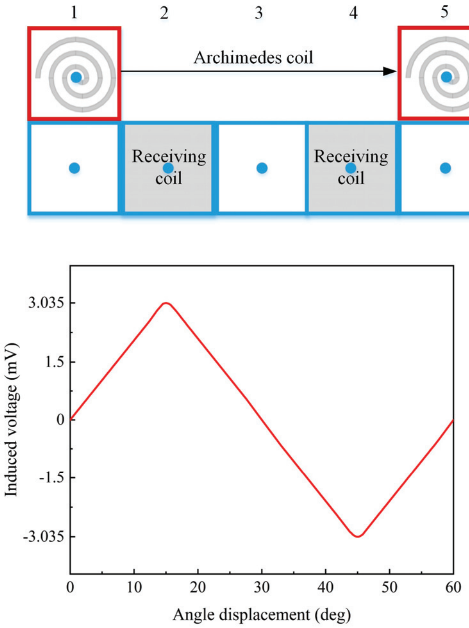


Figure 3. Induced voltage of receiving coils over one cycle.

The process of measuring angular displacement was the process of changing receiving coils on the rotor with respect to the Archimedes coil via a corresponding angle. The rotor rotated one measurement period (60°), and the corresponding change in the magnetic flux of receiving coils for a period was $0 \to -\psi \to 0 \to \psi \to 0$. According to the principle of electromagnetic induction, the receiving coil generated the induced voltage for one cycle as follows: $0 \to u \to 0 \to -u \to 0$. The relationship between the induced voltage of the receiving coils and the shaft angular displacement was approximately a sine wave (Figure 3).

A linear relationship between the induced voltage and angular displacement was induced using the FPC receiving coils, enabling the measurement of angular displacement (Figure 3). The difficulty of sensor design and linearization calculation was reduced through the linear relationship between angular displacement and induced voltage. The sensor had the advantages of being easy to calibrate, avoiding non–linear compensation and being easy to analyze.

For the induced voltage analysis, mathematical modelling of $\boldsymbol{B}_z$ and S was performed first, and the simulation analysis was carried out.

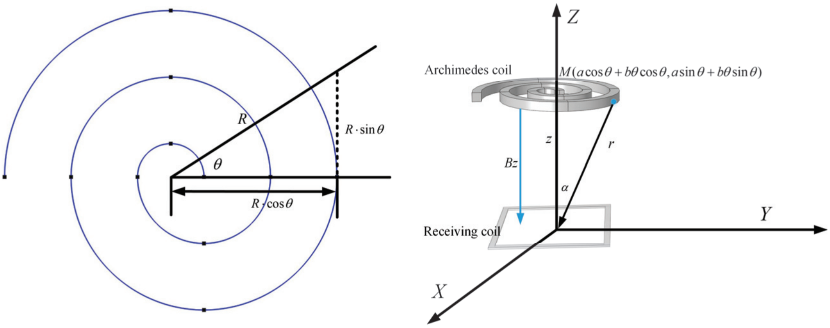
2.3.2. Archimedes Coil Magnetic Field Model

The Archimedes coil polar coordinate [73,74] is shown in Equation (4):

$$R = a + b\theta, \tag{4}$$

where a is the initial radius, b is the coil pitch and θ is the rotation angle. Equation (4) is expanded into parametric Equation (5) (Figure 4a):

$$\begin{array}{l} x = R \cdot \cos\theta \\ y = R \cdot \sin\theta \end{array}. \tag{5}$$



(**a**) (**b**)

Figure 4. Related schematic diagram of Archimedes coil: (**a**) parameter equation diagram; (**b**) generated magnetic field diagram.

The polar coordinate equation is then transformed into the parameter equation in the Cartesian coordinate system. Equation (5) can be described via (6):

$$\begin{array}{l} x = (a + b\theta) \cdot \cos\theta \\ y = (a + b\theta) \cdot \sin\theta \end{array}. \tag{6}$$

The lower magnetic flux leakage was neglected in the theoretical derivation of this paper to reduce the difficulty with theoretical derivation. According to Biot-Savart's law [75], the Archimedes coil with the MC in the Z magnetic flux density generated in the direction was $\boldsymbol{B}_z$ (Figure 4b). $\boldsymbol{B}_z$ can be obtained as follows (7):

$$\begin{aligned} \boldsymbol{B}_z &= \oint_{l_0} d\boldsymbol{B} \cdot \cos\alpha = \int \boldsymbol{B} \cdot \cos\alpha\, dl_0 = \int \frac{\mu_0 u_r}{4\pi r^2} \cdot \frac{u_1 \cdot \sin(2\pi f_1 t)}{R_1} dl_0 \\ &= \frac{\mu_0 u_r I_1 \cdot \sin(2\pi f_1 t)}{4\pi} \cdot \int_0^{k\cdot 2\pi} \frac{a+b\theta}{(a^2+b^2\theta^2+2ab\theta+z^2)^{3/2}} d\theta \end{aligned}, \tag{7}$$

where l_0 and α represent the circumference of the Archimedes coil and the angle between $\boldsymbol{B}$ and the Z–axis, respectively; u_0 and u_r represent the permeability of vacuum and the MC relative permeability, respectively; r is the distance from point M on the Archimedes coil to the center of the receiving coil; k and u_1 represent the number of coil loops and the excitation voltage, respectively; R_1 and I_1 represent the Archimedes coil resistance and the excitation current with frequency f_1, respectively; and z is the vertical distance from the Archimedes coil to the receiving plane.

To reduce the difficulty of deriving the equations, the values of $\boldsymbol{B}_z$ at the center point and in the region of the receiving coil are considered to be the same.

2.3.3. Receiving Coil Area Change Model

The three–dimensional space rotation of the Archimedes coil and the receiving coil was simplified into a two–dimensional–plane relative motion (Figure 5). The mathematical model of receiving coil area change was implemented, and the curvature of the receiving coil was neglected both here and subsequently in this study. Here, only the trend of the magnetic flux change from zero to the maximum value is shown, and the trend of the other phases was the same but in different directions.

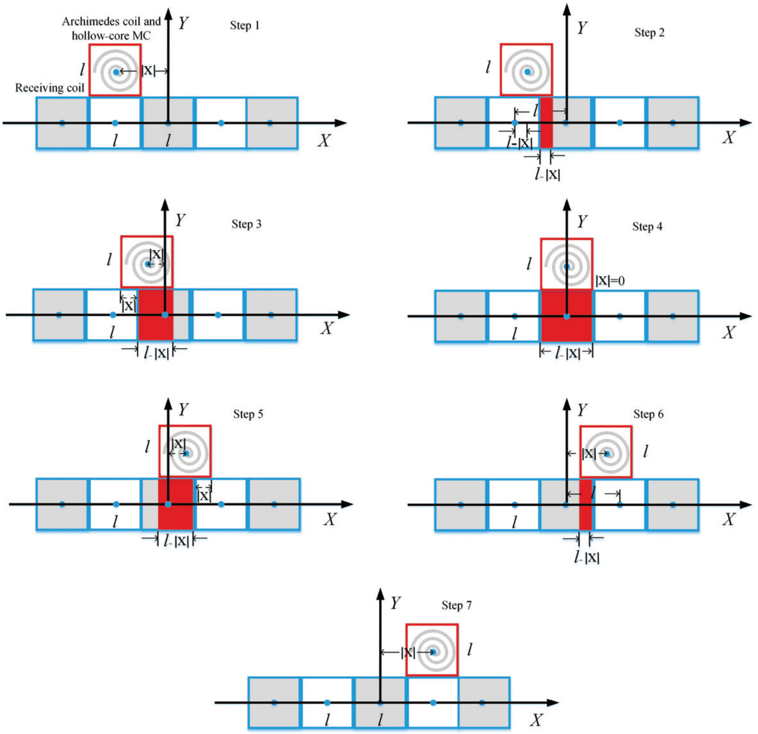


Figure 5. Relative motion diagram of coils.

The red–bordered square shape represents the focusing magnetic field generated using the Archimedes coil (side length is l) in Figure 5. The grey square–shaped coil is the receiving coil with side l. The white square–shaped coil is vacant. The absolute value of x represents the distance between the Archimedes coil and the Y–axis. The length of the Archimedes coil coinciding with the receiving coil is equal to $l - \mid x \mid$. The overlap area of the Archimedes coil and the receiving coil can be expressed as $S(x) = l \cdot (l - \mid x \mid)$, $x \in (-l, +l)$.

The Fourier expansion of $S(x)$ [76,77] and $S(x)$ is an even function. $S(x)$ can be described via (8)–(10):

$$S(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left(a_\mathrm{n} \cos \frac{n\pi x}{l} \right), \tag{8}$$

$$a_0 = \frac{2}{l} \int_0^l l \cdot (l - x) dx = \left. \frac{l \cdot x(2l - x)}{l} \right|_0^l = l^2, \tag{9}$$

$$\begin{aligned} a_n &= \tfrac{2}{l} \int_0^l l \cdot (l - x) \cdot \cos \tfrac{n\pi x}{l} dx = \tfrac{2l \cdot (l-x)}{n\pi} \sin \tfrac{n\pi x}{l} - \tfrac{2l^2}{n^2\pi^2} \cos \tfrac{n\pi x}{l} \Big|_0^l, \\ &= -\tfrac{2l^2}{n^2\pi^2} (\cos n\pi - 1) = \tfrac{2l^2}{n^2\pi^2} (1 - \cos n\pi) = \tfrac{4l^2}{n^2\pi^2} \cdot (\sin \tfrac{n\pi}{2}) 2 \end{aligned} \tag{10}$$

where the fundamental wave period is $l = \frac{2\pi R_\mathrm{a}}{N_1}$.

The angular displacement y can be obtained via (11):

$$y = vt = \omega R_\mathrm{a} t = \frac{2\pi n_\mathrm{a} R_\mathrm{a} t}{60}, \tag{11}$$

where v and ω represent the linear velocity of the receiving coil and the angular velocity of the receiving coil, respectively; R_a and n_a represent the radius of the shaft and the rotational speed of the receiving coil, respectively; and N_1 is twice the number of receiving coils.

$$S(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left(a_\mathrm{n} \cos \frac{n\pi x}{l} \right) = \frac{l^2}{2} + \sum_{n=1}^{\infty} \frac{4l^2}{n^2\pi^2} \cdot \left(\sin \frac{n\pi}{2} \right)^2 \cdot \cos \frac{nN_1 \pi n_\mathrm{a} t}{60}, \tag{12}$$

where n is the number of Fourier expansion terms.

2.3.4. Receiving Coil Induced Voltage Model

The induced voltage model was derived in this subsubsection. To avoid confusion, the excitation voltage generated using the Archimedes coil is set to u_1, and the induced voltage of the receiving coil is set to u_2. According to Lenz's law [72], u_2 can be obtained via (13):

$$u_2 = -n_2 \frac{d\psi}{dt} = -n_2 \frac{d}{dt} \boldsymbol{B}_z \cdot S. \tag{13}$$

The frequency–independent terms are set as coefficients (K_1, K_2 and K_3) in the expression to highlight the relationship between frequency and induced voltage in the equation. K_1, K_2 and K_3 can be obtained via (14)–(16):

$$K_1 = \frac{\boldsymbol{B}_\mathrm{z}}{\sin(2\pi f_1 t)} = \frac{\mu_0 u_\mathrm{r} I_1}{4\pi} \cdot \int_0^{k \cdot 2\pi} \frac{a + b\theta}{(a^2 + b^2\theta^2 + 2ab\theta + z^2)^{3/2}} d\theta, \tag{14}$$

$$K_2 = \frac{a_0}{2} = \frac{l^2}{2}, \tag{15}$$

$$K_3 = a_n = \frac{4l^2}{n^2\pi^2} \cdot (\sin \frac{n\pi}{2})^2. \tag{16}$$

Here, u_2 is split into three expressions for the excitation frequency f_1, and u_2 is shown in (17). Thus, u_{21}, u_{22} and u_{23} can be obtained via (18), (19) and (20):

$$u_2 = u_{21} + u_{22} + u_{23}, \tag{17}$$

$$u_{21} = -n_2 K_1 K_2 \cdot \cos(2\pi f_1 t) \cdot 2\pi f_1, \tag{18}$$

$$u_{22} = -n_2 \sum_{n=1}^{\infty} K_1 K_3 \cdot \cos \frac{nN_1 \pi n_{\mathrm{a}} t}{60} \cdot \cos(2\pi f_1 t) \cdot 2\pi f_1, \tag{19}$$

$$u_{23} = -n_2 \sum_{n=1}^{\infty} K_1 K_3 \cdot \sin(2\pi f_1 t) \cdot \left(-\frac{nN_1 \pi n_{\mathrm{a}}}{60} \cdot \sin \frac{nN_1 \pi n_{\mathrm{a}} t}{60} \right). \tag{20}$$

The final expression for u_2 can be described via (21):

$$\begin{aligned} u_2 &= -n_2 K_1 K_2 \cdot \cos(2\pi f_1 t) \cdot 2\pi f_1 - n_2 \sum_{n=1}^{\infty} K_1 K_3 \cdot \cos \tfrac{nN_1 \pi n_{\mathrm{a}} t}{60} \cdot \cos(2\pi f_1 t) \cdot 2\pi f_1 - n_2 \sum_{n=1}^{\infty} K_1 K_3 \cdot \sin(2\pi f_1 t) \cdot \left(-\tfrac{nN_1 \pi n_{\mathrm{a}}}{60} \cdot \sin \tfrac{nN_1 \pi n_{\mathrm{a}} t}{60} \right) \\ &= -n_2 \left[K_1 K_2 \cdot 2\pi f_1 + \sum_{n=1}^{\infty} K_1 K_3 \cdot \cos \tfrac{nN_1 \pi n_{\mathrm{a}} t}{60} \cdot (2\pi f_1 t - \tfrac{nN_1 \pi n_{\mathrm{a}}}{60}) \right] \cdot \cos(2\pi f_1 t) - n_2 \sum_{n=1}^{\infty} K_1 K_3 \cdot \tfrac{nN_1 \pi n_{\mathrm{a}}}{60} \cdot \cos(2\pi f_1 + \tfrac{nN_1 \pi n_{\mathrm{a}}}{60}) t \end{aligned} . \tag{21}$$

For a concise and clear representation of the effect of excitation frequency and rotational speed on the induced voltage, the initial Equation (3) for the induced voltage is expressed in (24), where the excitation current *i*(*t*) is described via (22):

$$i(t) = I \cdot \sin(\omega t). \tag{22}$$

The magnetic flux of the receiving coil can be obtained as follows (23):

$$\psi = k_4 y \sin\,(\omega t), \tag{23}$$

where k_4 is the proportionality factor and y is the angular displacement.

Equations (22) and (23) are substituted into Equation (3) to obtain (24):

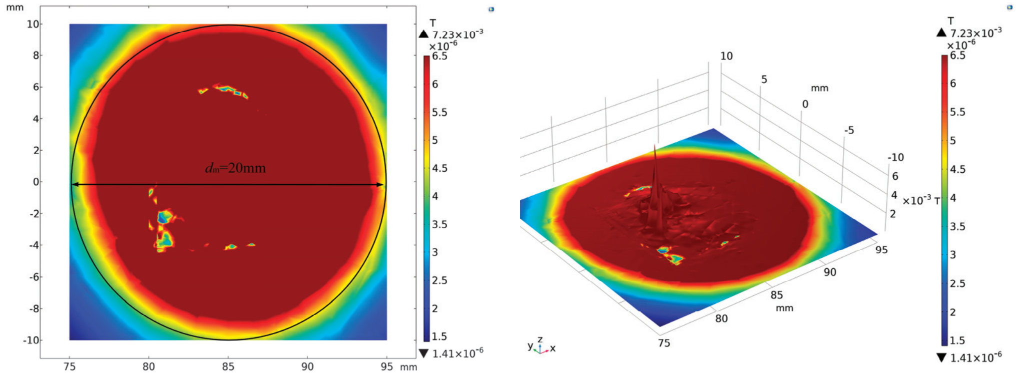
$$\begin{aligned} u &= -N \tfrac{d\psi}{dt} \quad = -N \tfrac{d[k_4 y \sin(\omega t)]}{dt} = -N k_4 \tfrac{dy}{dt} \cdot \sin(\omega t) - N k_4 y \omega \cdot \cos(\omega t) \\ &= -N k_4 v \cdot \sin(\omega t) - N k_4 y \omega \cdot \cos(\omega t) \; = -N k_4 \sqrt{v^2 + (y \cdot 2\pi f_1)^2} \cdot \sin(2\pi f_1 t + \varphi) \end{aligned} . \tag{24}$$

At f_1 = 10 kHz, the linear velocities of 5.6 m/s and 5.6 × 10^{-3} m/s, corresponding to 1000 r/min and 1 r/min, respectively, are substituted into Equation (24), and the ratio is then calculated. The induced voltage amplitude is increased by a factor of merely 1.000023 when the rotational speed is increased from 1 r/min to 1000 r/min. Therefore, when the rotational speed change value is less than 1000 r/min, the induced voltage change is very small and can be directly ignored. From Equation (24), it can be seen that the magnitude of the excitation frequency is the main factor affecting the induced voltage amplitude when the excitation frequency is in the kHz range. Only when the rotational speed increases to 10^4 r/min does the rotational speed affect the induced voltage amplitude, with the amplitude increasing by a factor of 1.002; however, the impact is still small.

3. Results

3.1. Verification and Analysis of the Focusing Magnetic Field

COMSOL software is used to simulate the Archimedes coils with and without an MC. The two sets of Archimedes coils are separately fed with an excitation voltage of 10 mV to verify whether the above non–MC and MC setups can generate a square–shaped focusing magnetic field. This set–up provides a basis for the next step to measure the dynamic induced voltage. A circular magnetic field with a diameter of 20 mm is formed in the *X-Y* plane at a distance of 1 mm below it (Figure 6). The magnetic field is in a divergent state without the MC. The ratio of the magnetic flux density of the red circular magnetic field to that of the surrounding blue area is 4.61. This ratio is defined as the magnetic field focusing intensity λ. In engineering, if the value of the same parameter differs by more than one order of magnitude, the lower value can be ignored. The λ_1 value formed without an MC is only 4.61, which cannot meet the measurement requirement of ignoring the surrounding magnetic flux leakage (Figure 6 and Table 4).



(**a**) (**b**)

Figure 6. Magnetic field 1 mm the Archimedes coil without an MC: (**a**) two–dimensional diagram; (**b**) three–dimensional contour diagram.

Table 4. Parameters obtained via two simulations.

Parameter Name	Magnetic Field without MC	Magnetic Field with MC
Magnetic field diameter, d_m (mm) Magnetic field side length, l (mm)	20	12.4
Magnetic field area, $S_m = \pi(d_m/2)^2$; $S_m = l^2$ (mm^2)	314	153.76
Maximum magnetic flux density, $\boldsymbol{B}_{max}$ (T)	7.23×10^{-3}	1×10^{-2}
Focusing magnetic flux density $\boldsymbol{B}_F$ (T)	6.5×10^{-6}	1.4×10^{-4}
Flux leakage, $\boldsymbol{B}_L$ (T)	1.41×10^{-6}	1.22×10^{-5}
Magnetic field focusing intensity, $\lambda = \boldsymbol{B}_F/\boldsymbol{B}_L$	4.61	11.48

The magnetic field is approximately square–shaped (12.4 × 12.4 mm^2), reducing the area by 160.24 mm^2 (Figure 7 and Table 4). The λ_2 value formed by the MC is 11.48, which is greater than an order of magnitude; thus, the magnetic field in the blue region can be ignored (Figure 7).

We note that the Archimedes coil without an MC is 1 mm above the rotation axis of the ferromagnetic material. Therefore, it can be assumed that the Archimedes coil has an MC, but its relative permeability is not as large. For this reason, the maximum magnetic flux density differs by only one order of magnitude, as shown in Figures 6 and 7. As shown in Figures 6 and 7, only the magnetic field value of the left minimal region is relatively large (1×10^{-2} T), while the other areas have essentially the same value and can be approximately considered to correspond to a uniform magnetic field.

The previous study was focused on a minuscule area, i.e., a point [63]. At this point, there was very little magnetic field that could be emitted independently. Therefore, the relative permeability u_r was changed from 300 to 60,000, and the increase in multiple K after magnetic focusing was not large, being only 1.08 times. Compared with the minuscule area of the previous study, the magnetic focusing area of this paper is large (12.4 × 12.4 mm^2). The original divergent magnetic field in this region is focused; thus, K corresponding to $\boldsymbol{B}_{max}$ and $\boldsymbol{B}_F$ is larger than the multiples of the previous study. However, the increasing trend of K is the same as the previous trend. The corresponding K increases by a factor of 1.38 at the position of $\boldsymbol{B}_{max}$ in this region and by a factor of 21.54 over the entire focusing magnetic field region.

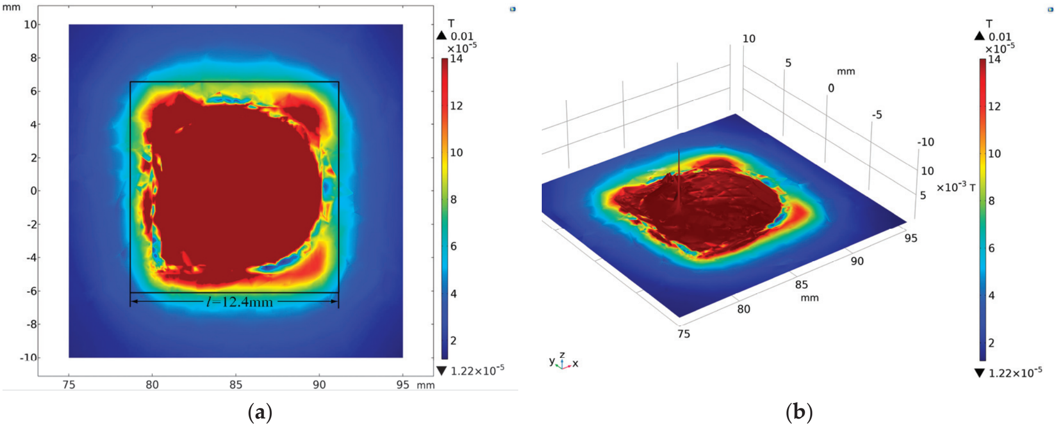


Figure 7. Magnetic field 1 mm below Archimedes coil with an MC: (**a**) two–dimensional diagram; (**b**) three–dimensional contour diagram.

Here, the increased K value is not equal to the relative permeability u_r of the MC for the following reasons: (1) the hollow–core MC can only constrain the magnetic field and cannot completely concentrate all the magnetic flux leakage in the square–shaped area; (2) the lower Archimedes coil is a rotating shaft, which is equivalent to having an MC with a small relative permeability. Thus, K does not simply represent an increase in the u_r factor of the MC. In summary, the focusing intensities simulated are basically consistent with previous research results.

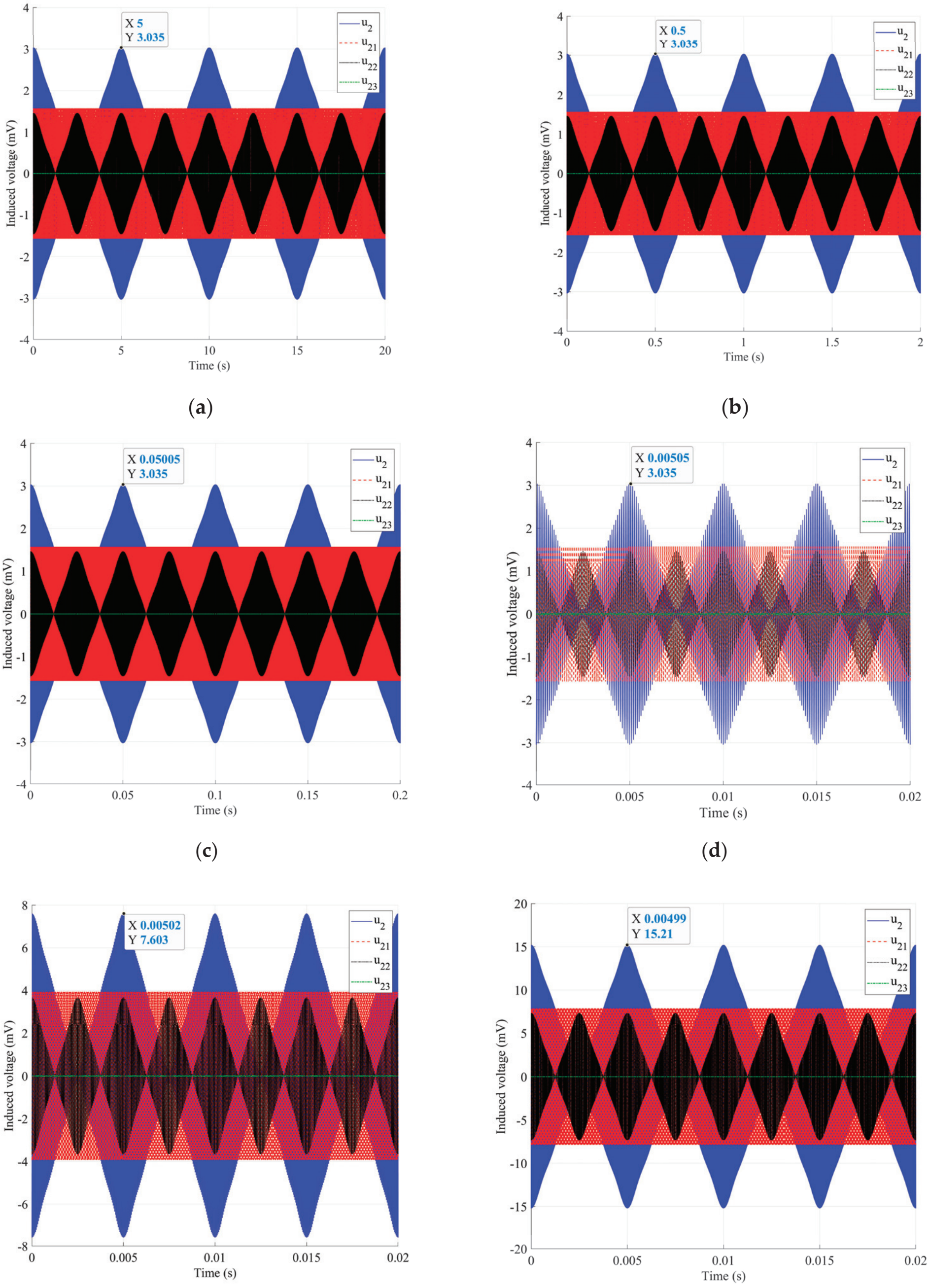
Due to the hollow–core MC constraint, the magnetic flux leakage is concentrated in the square–shaped area (Figure 7). The magnetic field is reduced from a circle (d_m = 20 mm) in Figure 6 to a square (l = 12.4 mm) in Figure 7. The focusing area of the magnetic field is reduced by 160.24 mm^2. With the reduction in the focusing area, the resolution of the sensor measurement can be improved; λ is increased from 4.61 times in Figure 6 to 11.48 times in Figure 7, which realizes the focusing of magnetic flux density.

Since the magnetic field is generated via Archimedes coil excitation, no matter how the shielding is added, it does not make the total amount of magnetic field intensity increase or decrease. The original magnetic flux leakage, which is dispersed to other locations in space, is constrained along the Z–axis through the outer wall of the hollow–core MC. This function reduces the magnetic flux leakage to a negligible level, which, in turn, meets the IADSMF measurement requirement.

3.2. Induced Voltage Simulation and Linearization Analysis

Since the equation for the induced voltage is mathematically derived from this paper, simulation is used to verify the derivation. COMSOL software cannot be edited to include equations; thus, MATLAB is used for simulations to verify the equations (see Figure 8 for the simulation results). It is verified that the IADS based on magnetic focusing can realize the measurement of angular displacement through the principle of electromagnetic induction. This finding is consistent with the measurement principle described in Section 2.3.

As can be seen in Figure 8d–f, the resolution of the IADSMF is 0.6 degrees for f_1 = 10 kHz, 0.24 degrees for f_1 = 25 kHz and 0.12 degrees for f_1 = 50 kHz. In summary, the increased resolution of the angle measurement can be achieved through simply increasing the excitation frequency while the speed is constant. The purpose of providing high angular resolution over a wide speed range is achieved.



(**a**) (**b**)

(**c**) (**d**)

(**e**) (**f**)

Figure 8. Simulation curves of induced voltage corresponding to different speeds and frequencies: (**a**) n_a = 1 r/min, f_1 = 10 kHz; (**b**) n_a = 10 r/min, f_1 = 10 kHz; (**c**) n_a = 100 r/min, f_1 = 10 kHz; (**d**) n_a = 1000 r/min, f_1 = 10 kHz; (**e**) n_a = 1000 r/min, f_1 = 25 kHz; and (**f**) n_a = 1000 r/min, f_1 = 50 kHz.

The reason for the approximately linear relationship between time and induced voltage u_2 is that when the rotational speed n_a and excitation frequency f_1 are constant, u_2 is related only to time in Equation (21), and u_2 is a sine-like function of time. Since the amplitude of the sine function in Equation (21) is large and the independent variable time is small, u_2 is approximately linear with time (Figure 8).

The reason for the linear relationship between the angular displacement y and time is that the receiving coil and excitation magnetic field are both square–shaped. The amount of change over time is the area $S(x)$, where the receiving coil coincides with the excitation magnetic field. Since the width l in $S(x)$ (Figure 7) is constant, the overlap area is related only to the overlap length $l - |x|$ in the X–axis direction (angular displacement $y = l - |x|$). Therefore, the angular displacement is linear with time.

Since the length of time in a cycle corresponds to the spatial angular displacement, u_2 and y of the shaft assume an approximate linear relationship (Figure 5). The advantage of the proposed square–shaped receiving coil, compared with the three sets of diamond-shaped coils previously proposed [27], is that the induced voltage obtained is directly linearized, eliminating the need for an algorithm to obtain the linear relationship between the induced voltage and angle and simplifying the post–processing work. The measuring requirements can be met using a set of receiving coils designed for this purpose; thus, the receiving coil structure is simplified, and the number of wires is reduced. This outcome is the original intention of designing a square–shaped receiving coil.

When the rotational speed is constant, the induced voltage amplitude increases with increasing excitation frequency f_1 (Figure 8d–f). Here, f_1 is increased to improve the resolution and achieve more accurate and smaller angle measurements. However, if f_1 is increased excessively, electric energy will be wasted.

Advantages of IADSMF:

(1) For post–processing

There are several reasons for having different post–processing formulas (Table 5). Firstly, the sensors have different metal rotor shapes and numbers of blades. Secondly, the angle of the space occupied by its corresponding one blade is different; thus the post–processing equation is different. Moreover, each researcher has a different understanding of the sensor that they designed; thus, the defined post–processing is also different, i.e., there is no standard answer, but only the correct answer. This makes post–processing poorly versatile. Inevitably, it has caused some difficulties for researchers in understanding in the field of just–contact sensors. Since IADSMF has no metal rotor blades, there is no post–processing formula. At the same time, the focusing magnetic field is consistent with the shape of the receiving coil; thus, the direct linearization is realized.

Table 5. Comparison of sensors.

Parameter	[27]	[64]	[69]	IADSMF
f_1	4 MHz	2 MHz	40 kHz	10 kHz
u_2 (mV)	3.25	21	1.5×10^{-6}	3.035
Post–processing formula	$U_0 = U_A$ $U_0 = U_{\overline{c}} + (U_{MAX} - U_{MIN})$ $U_0 = U_B + 3U_{MAX} - U_{MIN}$ $U_0 = U_{\overline{A}} + 5U_{MAX} - U_{MIN}$ $U_0 = U_C + 7U_{MAX} - U_{MIN}$ $U_0 = U_{\overline{B}} + 9U_{MAX} - U_{MIN}$	$\theta = \arctan\left(\frac{U_{rx1}(\theta)}{U_{rx2}(\theta)}\right)$	$\theta_p = \begin{cases} \theta_1 - \theta_2 & \text{if } (\theta_1 - \theta_2) \geq 0 \\ \theta_1 - \theta_2 + 2\pi & \text{if } (\theta_1 - \theta_2) < 0 \end{cases} \quad \theta_1, \theta_2, \theta_p \in [0, 2\pi)$	None

(2) Input and output aspects

As can be seen from Table 5, in conventional IADS, the excitation frequency is generally at the MHz level in order to obtain the induced voltage at the mV level. The excitation frequency cannot be too small as, as shown in [69], f_1 = 40 kHz; if the excitation frequency is too small, the voltage cannot reach mV level. As it removes the intermediate link of the metal rotor, the advantages of IADSMF are reflected. Instead of using the eddy current magnetic field on the metal rotor to induce the output voltage, the excitation field induces the voltage directly in the receiving coils. Compared with traditional IADSs, it can reduce the sensor components and input demand of the excitation end. Its eddy current loss will also be reduced, realizing the energy saving of the sensor. The specific loss is shown in Table 6 of Section 3.3.

Table 6. Eddy current loss comparison.

Parameter	[38]	IADSMF
f_1 (kHz)	10	10
$V = a^3$ (m^3)	9.3696×10^{-10}	1.2301×10^{-8}
$\boldsymbol{B}_\mathrm{F}$ (T)	1.2297×10^{-3}	1.4×10^{-4}
u_2 (mV)	2.934	3.035
P (W)	2.1304	0.3625

In the next subsection, the eddy current loss is calculated to quantify the analysis.

3.3. Eddy Current Loss Calculation

Since the shaft is a ferromagnetic material, there is eddy current loss; thus, it is necessary to study how to reduce the eddy current loss. While meeting measurement needs, the excitation frequency should be minimized; the lower the frequency is, the smaller the eddy current loss. Excessive eddy current also causes the shaft to heat up, and a quantity of heat is transferred to the FPC on the shaft surface. Excessive temperatures affect the stability of the FPC, resulting in unnecessary errors.

The eddy current loss [78] can be expressed via (25):

$$P = \int_v \frac{\left|\boldsymbol{J}_y\right|^2}{\gamma} \mathrm{d}V = \frac{1}{24} k_{yoke} \sigma \omega^2 \boldsymbol{B}^2 a^3 \quad = \frac{1}{6} k_{yoke} \sigma \pi^2 f^2 \boldsymbol{B}^2 a^3, \tag{25}$$

where $\boldsymbol{B}$ and $\boldsymbol{\omega}$ represent the magnetic flux density and the angular frequency, respectively; f and a^3 represent the excitation frequency and the volume, respectively; and k_{yoke} = 0.915 and σ represent the correction factors for eddy current loss and the material conductivity, respectively.

The penetration depth δ of the eddy current [79] can be expressed via (26):

$$\delta = \frac{1}{\sqrt{\pi f \mu_0 \mu_\mathrm{r} \sigma}}, \tag{26}$$

where u_0 and u_r represent the permeability of vacuum and the relative permeability, respectively; and σ is the material conductivity.

Substituting $u_0 = 4\pi \times 10^{-7}$, $u_\mathrm{r} = 400$, $\sigma = 1 \times 10^7$ S/m, π = 3.14 and f = 10 kHz, we obtain δ = 0.08 mm, i.e., a_3 = 0.08 mm. Therefore, $V_\mathrm{IADSMF} = a_1 \times a_2 \times a_3 = 12.4 \times 12.4 \times 0.08 \times 10^{-9} = 1.2301 \times 10^{-8}$ m^3, and $V_\mathrm{IADS} = a_1 \times a_2 \times a_3 = 58.56 \times 0.2 \times 0.08 \times 10^{-9} = 9.3696 \times 10^{-10}$ m^3.

From the conclusion of Section 3.1, it can be seen that the excitation frequency is reduced from 1 MHz to 10 kHz when the induced voltage is reduced by approximately 100 times. Thus, the induced voltage of [38] is reduced to 2.934 mV. At this point, the output voltage amplitudes of the IADS and IADSMF are almost equal, i.e., at the same level of 3 mV. We calculate the eddy current loss P (Table 6). P_IADSMF = 0.3625 W is only 17.02% of the IADS (2.1304 W) loss. The purpose of reducing eddy current loss is achieved through replacing the eddy current field with a focusing magnetic field through the MFM.

3.4. Sensor Optimization Parameter Selection

After obtaining the linear relationship between the angular displacement and induced voltage of the sensor, the measurement error needs to be analyzed. The linearity error of the induced voltage is used to reflect the measurement error of the sensor. This error is derived from the measured induced voltage curve through fitting the difference between the measured voltage and ideal voltage at the corresponding angle. The difference is divided by the full–scale voltage to express the linearity error. L [38] can be expressed via (27):

$$L = \frac{|(u_2 - u_i)_{\max}|}{u_{\mathrm{FS}}} \times 100\%, \tag{27}$$

where L represents the linearity error; u_2 and u_i represent the simulation voltage and the ideal voltage, respectively; and u_{FS} is the full–scale output voltage.

The main design parameters of the sensor are listed. The design parameters include the number of turns of the Archimedes coil, the width of the Archimedes coil, the air gap between the Archimedes coil and the receiving coil, and the side length of the receiving coil. The initial design parameters and ranges are shown in Table 7. The parameter that has the greatest influence on the linearity error is selected through changing the single variable. This parameter is optimized to reduce the sensor linearity error.

Table 7. Sensor model parameters.

Parameter	Initial Value	Setting Range
N_1	3	3–11
r_1 (mm)	0.2	0.2–0.4
d_g (mm)	1	0.6–1.4
l_r (mm)	12.4	12.4–12.8

N_1—Archimedes coil turn number; r_1—Archimedes coil width; d_g—air gap between the Archimedes coil and receiving coil; l_r—receiving coil side length.

The initial parameters of the sensor model are used to perform the simulation. The induced voltage curve is obtained, and the linearity error is calculated (Table 8). As seen from Table 8, a corresponding simulated voltage has the maximum error (0.038 mV) with the theoretical voltage at 10°. Therefore, the maximum linearity error of the sensor in a measurement period (0–60°) is 0.6239%.

Table 8. Sensor model simulation results.

θ_r (°)	u_2 (mV)	u_i (mV)	u_2–u_i (mV)	L (%)
0	3.035	3.045	−0.01	0.1642
10	1.049	1.011	0.038	0.6239
20	−1.059	−1.023	−0.036	0.5911
30	−3.035	−3.045	0.01	0.1642
40	−1.023	−0.999	−0.024	0.3941
50	1.071	1.0353	0.0357	0.5862
60	3.035	3.045	−0.01	0.1642

θ_r—rotation angle.

3.4.1. Archimedes Coil Turn Number

Only the number of turns of the Archimedes coil is changed, and the initial values of the other design parameters remain unchanged (Table 7). The number of turns is increased from 3 to 11. The variation in the maximum and minimum linearity error is 0.0318%. The number of turns has a small effect on the linearity error of the sensor (Table 9).

Table 9. Effect of number of turns on linearity error.

N_1	r_1 (mm)	d_g (mm)	l_r (mm)	L (%)	E_v (%)
3	0.2	1	12.4	0.6239	0.0318
5	0.2	1	12.4	0.6443	
7	0.2	1	12.4	0.6521	
9	0.2	1	12.4	0.6475	
11	0.2	1	12.4	0.6557	

N_1—Archimedes coil turn number; r_1—Archimedes coil width; d_g—air gap between the Archimedes coil and receiving coil; l_r—receiving coil side length; E_v—error variation.

3.4.2. Archimedes Coil Width

Only the Archimedes coil width is changed, and the initial values of the other design parameters remain unchanged (Table 7). The width is increased from 0.2 to 0.4 mm. The variation in the maximum and minimum linearity error is 0.0269% (Table 10). The Archimedes coil width has a small effect on the linearity error of the sensor.

Table 10. Effect of Archimedes coil width on linearity error.

N_1	r_1 (mm)	d_g (mm)	l_r (mm)	L (%)	E_v (%)
3	0.2	1	12.4	0.6239	0.0269
3	0.25	1	12.4	0.6488	
3	0.3	1	12.4	0.6479	
3	0.35	1	12.4	0.6498	
3	0.4	1	12.4	0.6508	

3.4.3. Air Gap between the Archimedes Coil and Receiving Coil

Only the air gap between the Archimedes coil and receiving coil is changed, and the initial values of the other design parameters remain unchanged (Table 7). The air gap is increased from 0.6 to 1.4 mm. The variation in the maximum and minimum linearity error is 0.6219% (Table 11). The air gap has a significant effect on the linearity error of the sensor.

Table 11. Effect of air gap on linearity error.

N_1	r_1 (mm)	d_g (mm)	l_r (mm)	L (%)	E_v (%)
3	0.2	0.6	12.4	0.7389	0.6219
3	0.2	0.8	12.4	0.6663	
3	0.2	1	12.4	0.6239	
3	0.2	1.2	12.4	0.8043	
3	0.2	1.4	12.4	1.2458	

3.4.4. Receiving Coil Side Length

Only the receiving coil side length is changed, and the initial values of the other design parameters remain unchanged (Table 7). The side length is increased from 12.4 to 12.8 mm. The variation in the maximum and minimum linearity error is 0.0305% (Table 12). The side length has a small effect on the linearity error of the sensor.

Table 12. Effect of side length on linearity error.

N_1	r_1 (mm)	d_g (mm)	l_r (mm)	L (%)	E_v (%)
3	0.2	1	12.4	0.6239	0.0305
5	0.2	1	12.5	0.6533	
7	0.2	1	12.6	0.6399	
9	0.2	1	12.7	0.6485	
11	0.2	1	12.8	0.6544	

3.4.5. Optimization Parameter Selection

The magnitude of the error variation indicates the magnitude of the voltage fluctuation. We calculate the magnitude of error variation in the Archimedes coil turn number, Archimedes coil width, air gap between the Archimedes coil and the receiving coil, and receiving coil side length. The maximum error variation for each parameter is shown in Table 13. From these, the parameter corresponding to the maximum error variation is selected and optimized.

Table 13. Effect of the main design parameters on E_v.

Parameter	**E_v (%)**
N_1	0.0318
r_1 (mm)	0.0269
d_g (mm)	0.6219
l_r (mm)	0.0305

When the distance of the air gap is changed from 0.6 to 1.4 mm, the magnitude of the error variation is 0.6219% (Table 13). E_v of the air gap is an order of magnitude larger than that of the other parameters. The air gap is a main parameter affecting the linearity error of the sensor. Therefore, the air gap is selected as an optimized parameter for the sensor.

3.5. Optimization Algorithm

Nowadays, the algorithm is being developed very rapidly, and new algorithms are everywhere. For example, the IbI Logics Algorithm (ILA) is based on Intelligible-in-time (IbI) Logics [80]. Its advantage over existing algorithms is that the optimization can be divided into three phases, which can be optimized in stages. If there is a problem in the second stage, the third stage will not be executed. Correct the problems in the second phase before implementing the next phase. It is suitable for scenarios where the entire optimization process needs to be monitored and the optimization parameters need to be continuously improved (e.g., changing the number of iterations). There is also an optimization algorithm applied to exoplanet exploration–Transit Search (TS) [81]. For the optimization approach used through TS, different optimization strategies are used at different stages of optimization. It optimizes 73 constraints, and the total average error of TS is the lowest compared with the existing algorithms. Therefore, it is suitable for outer space exoplanet exploration. The algorithm is suitable for multi–constrained problems, and its effect of equilibrium is good.

In recent years, researchers optimized sensor structures or related parameters through various optimization algorithms. The algorithms used are surface integral method [82], particle swarm optimization (PSO) [83] and response surface method [38]. PSO is suitable for solving global optimization problems. Compared with traditional algorithms, PSO has fast solving speed and guarantees global search ability [84]. However, PSO has the problem of premature convergence and may converge at the local optimum solution. Considering the above shortcomings, Jing et al. improved the PSO via adding inertia weights to the PSO. The inertia weights can decrease non–linearly with the number of iterations increasing. Thus, it ensures global search capability while avoiding trapping searches in local optimal solutions [85].

Linearly decreasing inertia weight particle swarm optimization (LIWPSO) and the finite element method (FEM) are combined to optimize the sensor parameters. The sensor parameter design problem is transformed into a particle swarm search for the best position. LIWPSO has two key factors for the position search: (1) particle velocity ($v_{i,j}$) and (2) particle position ($x_{i,j}$) [27]. The particle velocity represents the magnitude of the ability to find a solution. In the n–dimensional search space, each position of a particle represents a solution to the fitness function, i.e., a position corresponds to a set of sensor structure parameters. LIWPSO finds the optimal structure variable through solving for the optimal value of the fitness function in the search space. Using the above variable, the corresponding parameters are changed, and the sensor is simulated via the FEM.

Linearity error is calculated through comparing the simulated voltage profile with the ideal voltage profile, i.e., LIWPSO finds a set of structural parameters that minimize the sensor linearity error. The updated formulas [27] of the LIWPSO particle velocity and position can be obtained via (28) and (29), respectively:

$$v_{i,j}(t+1) = \omega v_{i,j}(t) + c_1 r_1(t)(p_{i,j}(t) - x_{i,j}(t)) + c_2 r_2(t)(g_{i,j}(t) - x_{i,j}(t)), \tag{28}$$

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t), \tag{29}$$

where $v_{i,j}$ represents the particle velocity; ω represents the inertia weight; $x_{i,j}$ represents the particle position; c_1 and c_2 represent the acceleration coefficients of individual particles and the particle swarm, respectively; and r_1 and r_2 represent random numbers uniformly distributed on the interval [0, 1].

The inertia weight formula [27] can be obtained via (30):

$$\omega = \omega_{\max} + \frac{t \cdot (\omega_{\min} - \omega_{\max})}{t_{\max}}, \tag{30}$$

where $\omega_{\max}$ represents the maximum inertia weight, $\omega_{\min}$ is the minimum inertia weight and $t_{\max}$ is the maximum number of iterative steps.

In LIWPSO, ω should be maintained at a large value at the beginning to ensure that particles jump out of the local optimal solution. Additionally, ω should be maintained at a small value when the number of iterations is large to ensure that this parameter tends to the global optimal solution and facilitate algorithm convergence. Generally, $\omega_{\min}$ = 0.4, and $\omega_{\max}$ = 0.9. Figure 9 shows the LINWPSO optimization process.

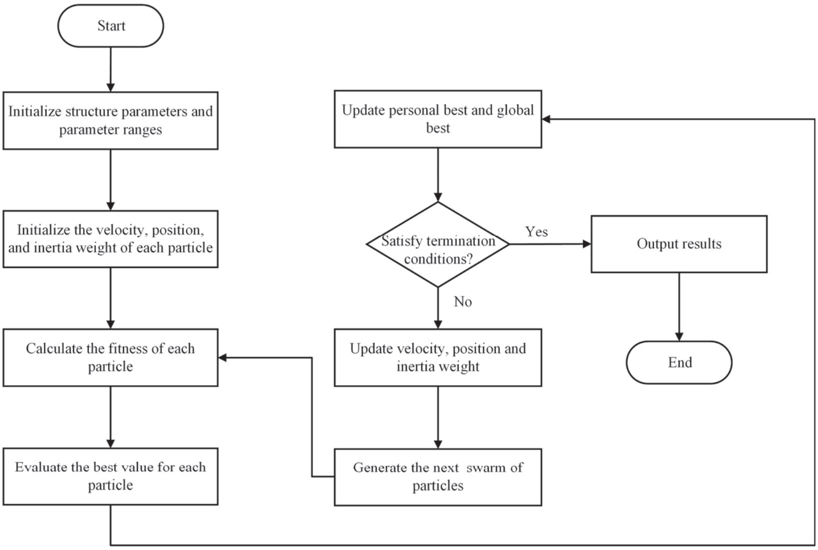


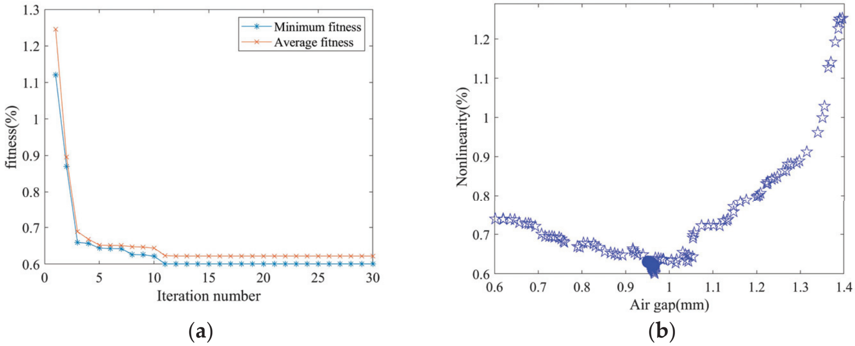
Figure 9. Flow chart of LINWPSO optimization process.

In LIWPSO–FEM, the fitness function is used to represent the sensor linearity error. The air gap between the Archimedes coil and receiving coil is used as the optimized parameter. The optimization design of the sensor is transformed into the problem of determining the values of the sensor variables through searching for the minimum fitness function via LIWPSO–FEM [27]. The fitness function [38] can be expressed via (31):

$$fitness = L = \frac{|(u_2 - u_i)_{\max}|}{u_{\mathrm{FS}}} \times 100\%, \tag{31}$$

where L is the sensor linearity error; u_2 and u_i represent the simulated voltage and the ideal voltage, respectively; and u_{FS} is the full–scale output voltage.

In the optimization process of LIWPSO–FEM, the particle swarm size is 120, the acceleration coefficients $c_1 = c_2 = 1.495$ and $-0.05 \leq v_{i,j} \leq 0.05$. The number of iterations is 30. The minimum fitness, i.e., the linearity error value, is obtained through 30 iterations. The linearity error is stable at 0.6012% when the number of iterations exceeds 11 (Figure 10a). The variation range of the air gap is 0.6–1.4 mm, and the air gap distance corresponding to the minimum linearity error is 0.96 mm (Figure 10b).



(**a**) (**b**)

Figure 10. Sensor optimization simulations: (**a**) fitness value with iteration number; (**b**) linearity error vs. the air gap obtained in optimization design.

The optimized linearity error, as shown in Table 14.

Table 14. Linearity error corresponding to optimized sensor parameter.

Parameter	*L* (%)
$d_g = 0.96$ (mm)	0.6012

In Table 15, a comparative study is given for the IADSMF and some inductive displacement sensors. After optimization, the proposed IADSMF has a lower linearity error. The advantage of direct linearization is, thus, demonstrated.

Table 15. Linearity comparison of IADSMF and other displacement sensors.

Sensor	*L* (%)
[27]	0.778
[86]	0.8
[87]	1.25
IADSMF	0.6012

4. Discussion

The achievements and shortcomings are as follows.

The main engines of large ships usually operate at low–to–medium speeds. Therefore, the main engine speed is usually in the range of 90–120 r/min [88]. As shown in Figure 8c, $n_a = 100$ r/min, and the angular resolution is 0.06 degrees. The resolution of IADSMF is higher than shown in [15,89] (0.5 and 0.35 degrees). Higher or lower angular resolution can be achieved through changing f_1.

In contrast to crankshaft angle measurement using an optical angle encoder set, the IADSMF does not require a separate electronic module for converting the optical signal into standard TTL signal [89]. IADSMF reduces the complexity of sensors.

Magnetic focusing is used as the excitation method to reduce one level of transmission compared with a traditional IADS. A traditional IADS [27,36–38] is composed of three parts—excitation, rotor and receiving coils—requiring two-level transmission to complete a measurement. The working mode occurs as follows: (1) the excitation coil produces a magnetic field due to rotor rotation, and an eddy current is created in the rotor. The rotor undergoes a position change relative to the receiving coil; (2) the receiving coil then induces a voltage through the eddy current magnetic field generated using the rotor to achieve the position measurement. An IADSMF forms a stator through means of the MC and the excitation coil. The generated focusing magnetic field directly acts on the FPC receiving coil on the rotor, and the FPC receiving coil directly induces a voltage.

The focusing magnetic field from the excitation end replaces the eddy current magnetic field of the IADS. The eddy current magnetic field inside the metal rotor is eliminated, again avoiding inducing a voltage at the receiving coil end. Thus, the eddy current field can be reduced, and the purpose of reducing eddy current loss can be achieved (Table 6).

The excitation frequency is 10 kHz for IADSMF and 4 MHz for [27], and the induced voltage amplitude of the two sensors are 3 mV. Therefore, when this sensor outputs the same magnitude of induced voltage as the previously studied sensor [27], there is no need for a MHz–level excitation frequency. The reason for this finding is that the introduction of magnetic focusing increases the magnetic induction intensity acting directly on the receiving end.

In terms of linearization, the focusing magnetic field and the receiving coil of the IADSMF are square–shaped to achieve direct linearization. The linearity error of the IADSMF before optimization is 0.6239% (0.6012% after optimization), which is less than the linearity error of a traditional IADS, which is optimized to 0.778% [27].

The sensitivity of the IADSMF is 0.2023 mV/°, while the sensor sensitivity shown in [27] is 0.4847 mV/°; the IADSMF value is lower. The definition of sensitivity shows that the higher the sensitivity, the narrower the measurement range of angular displacement, and the lower the sensor’s stability. Although the IADSMF sensitivity is low, its stability is better than that of [27].

In terms of the angle measurement range, the IADSMF can achieve 0–360° measurement without the constraint associated with the measuring angle range. However, in [37], it does not calculate the angle corresponding to the induced voltage of 0 mV, and there is a defect in the range of angle measurement. The advantage of direct linearization is further demonstrated.

In terms of the optimization algorithm, LINWPSO has a strong ability to jump out of local optimal solutions. LINWPSO has a faster calculation speed and shorter calculation time than the traditional particle swarm optimization method used in [37].

The MFM is used to change the traditional excitation method, and a breakthrough in the excitation method of an IADS is realized. The above breakthrough and improvements enabled the direct linearization of angular displacement and induced voltage, thus reducing post–processing design and calculation.

The following issues should be the focus of future research.

The focus of future research is a comprehensive study of signal transmission at the sensor output. The induced voltage would be sent to the host computer through a wireless transmission technology such as Bluetooth. Among them, electromagnetic interference and miniaturization of transmission devices are the focus of research.

(1) Electromagnetic interference

Through debugging different excitation and transmission frequencies, we would discover a suitable electromagnetic compatibility scheme. The effect of different directions for the transmitted signal on the excitation would be sought in space. We would aim to find the best decoupling scheme for the transmission signal and the excitation signal or the scheme with the least influence [90].

(2) Miniaturization aspect

We would use manufacturing technologies, such as inkjet printing or silicon micromachining technologies, to manufacture circuit boards. We would use these technologies to take advantage of their smaller size and ability to arrange multi–layer coils on a very small area, as well as their amplifiers, low-pass filters, analog–to–digital converters, Bluetooth transceivers, and other components to create transmission devices [91–93].

5. Conclusions

Based on the advantages of IADS, a magnetic focusing type sensor suitable for using in the environment of ships is designed. The sensor has good versatility and scalability. An IADSMF can realize the measurement of the crankshaft angle position. Two IADSMFs are combined to enable the measurement of torque on ships' rotating shafts. Compared to the optical method of torque measurement, the IADSMF does not require an additional geared system for torque measurement. Therefore, the machining of the shaft is simplified. Compared to traditional IADS, IADSMF does not require a metal rotor in either a planar or vertical structure. This fact results in one less component being in the overall sensor system. The purpose of creating a simpler system and higher reliability in the engineering application is realized. In addition, IADSMF is a non–contact sensor; thus, there is no need to worry about contact friction and service life.

Compared to the resolution of existing crankshaft angle encoders (0.35 and 0.5 degrees), the IADSMF increases the measurement resolution (0.06 degrees) by a factor of 5.8 to 8.3. If necessary, the resolution improvement could be achieved through increasing the existing excitation frequency (10 kHz). The increase in excitation frequency would be proportional to the increase in resolution (Figure 8d–f).

Through combining an Archimedes coil with a hollow–core MC, the magnetic field generated using the excitation coil is changed from a circular (d_m = 20 mm) to an approximately square–shaped magnetic field (12.4×12.4 mm^2). This method reduces the magnetic field's area by 51.03%. It improves the center–focusing degree and sensor resolution and realizes magnetic field focusing. The stator is composed of metal MCs and Archimedes coils. The rotor is composed of a set of 12 square–shaped receiving coils connected at the beginning and end, with the adjacent receiving coils separated by 15°. A measurement cycle is 60°, and the angular displacement measurement of rotating in a circle can be completed in six measurement cycles.

Mathematical models of the excitation coil magnetic field, the receiving coil area change and the induced voltage are derived. Subsequent simulations are performed for rotational speeds from 1 to 1000 r/min. The simulation results show that the relationship between the shaft angular displacement information and induced voltage of the receiving coil is approximately sinusoidal. The reason for the linearization of angular displacement with respect to the induced voltage is analyzed, being achieved using the square–shaped magnetic field and receiving coil design. Theoretically, the feasibility of the measurement principle of this sensor is demonstrated, and the direct linearization of the angular displacement and induced voltage is achieved. The IADSMF corresponds to an initial linearity error of 0.6239%. As can be seen from the data in Table 15, this error is also smaller than that of the other IADSs. The above finding shows that it is beneficial to the proposed direct linearization, while its effect is good. Simplifying the post–processing design reduces the difficulty of sensor development and avoids the calculation error that IADS may cause due to subsequent complex linearization calculation. According to equations and simulations, it is concluded that excitation frequency f_1 is the main factor affecting the induced voltage amplitude. The amplitude of u_2 increases proportionally with increasing f_1. At f_1 = 10 kHz, the influence of a low rotational speed change on the amplitude of u_2 can be ignored. The eddy current loss of a traditional IADS is 2.1304 W, while that of the IADSMF is 0.3625 W for almost the same induced voltage amplitude, achieving a reduction in the eddy current loss. The above eddy current loss calculation shows that the measurement method adopted using IADSMF changes the original IADS, which requires a large eddy current magnetic field to implement the angle measurement. A shortcoming in the design of IADS is solved, i.e., the eddy current field of IADS cannot be too small, otherwise the induced voltage cannot reach the mV level to achieve measurement. Moreover, an excessive eddy current field corresponds to a large eddy current loss, which is not conducive to the energy saving of the sensor. Therefore, IADSMF provides a research idea for the development of IADS family. The parameter that has the greatest influence on the linearity error of the IADSMF—the air gap—is identified through simulations. The sensor structure is then optimized via LIWPSO to find the minimum linearity error of 0.6012%, corresponding to an air gap of 0.96 mm.

During optimization, the change in air gap affects the focusing effect of magnetic field, which, in turn, affects the linearity. Thus, it is found that the better the focusing effect of magnetic field, the less the magnetic flux leakage outside the receiving coil of IADSMF, and the less it affects the measurement results. Therefore, future research can improve the focusing effect. For example, we could find a material with greater relative magnetic permeability than hollow–core MC to further constrain magnetic flux leakage. Moreover, different materials of solid–core MC are selected and combined with hollow–core MC. We could also find the best combination of hollow– and solid–core MC to improve the focusing effect. The introduction of MFM into IADS further advanced application of MFM in the measurement field. In particular, one potential application is angle measurement in ships, which is based on the advantages of direct linearization of IADSMF, which, in turn, contributed to the advancement in reducing the design difficulty and understanding aspect of IADS. As shown in Table 5, there is no need for designers to convert angles or voltages. For researchers who are not experts in the field of sensors, it is not necessary to understand the design principles and formulas of each sensor post–processing. The linearity of the voltage variation with time (angular displacement) can be visualized. The versatility and readability of IADS in post–processing are improved.

Since the IADSMF does not have a metal rotor, we allow the IADS to bypass the eddy current magnetic field. The excitation magnetic field is realized via inducing a voltage directly into the receiving coil. The improvement of measurement method is realized. For the designer, it avoids a situation in which the sensor, knowing that the greater the eddy current, the greater the loss (Table 6), cannot reduce the eddy current significantly. Otherwise, the induced voltage amplitude is too small to be measured, making the IADS useless as a sensor.

In ships, the cost of developing a separate torque sensor is five times that of an angle sensor [15]. While using IADSMF to replace the angle encoder of the ship main engine, there is no need to separately develop a torque sensor applied to the ship. Instead, we can simply place another IADSMF on the other end of the shaft. Through measuring the torsion angle between two IADSMFs, the torque can be measured using the formula, improving versatility and reducing costs. A potential application area for the IADSMF is torque measurement on ships. This function promotes the measurement and application of magnetic focusing knowledge in the field of torque.

An inductive sensor and the MFM are combined to realize the breakthrough of the IADS measurement method in theory. This method provides a theoretical basis and new research directions to promote further IADS development in ships and navigation. We will experiment with the application of IADSMF in ships and navigation in our next steps.

Experimental arrangement:

(1) According to the data in Table 3, the solid– and hollow–core MCs are processed, and they are combined with the PCB containing the Archimedes coil.
(2) Using a Gauss meter, we measured the magnetic field formed using the MFM proposed in Section 3.1 to verify that the focus was successful.
(3) According to the actual focused square magnetic field, the FPC receiving coil with the same size was processed.

(4) We processed the shaft, purchase and manufacture equipment, such as the drive motor, frequency inverter, excitation power supply, couplings and transmission equipment.
(5) After the sensor assembly was completed, the sensor was tested and calibrated to determine the measurement angle, the linearity error and repeatability, etc.

After the above experiments were completed, IADSMF was installed on the free end of the crankshaft of the main engine for board testing.

6. Patents

Li, Z.; Wang, B.; Guo, Y.; Zhang, C.; Qiu, F. A measurement system based on magnetic focusing type to measure static torque and its measurement method. CN110987259B, 30 November 2021.

Li, Z.; Zhang, C.; Qiu, F.; Wang, B.; Meng, X. A dual magnetic focusing speed, torque and angle sensor. CN210486909U, 8 May 2020.

Author Contributions: Conceptualization, Z.L. and B.W.; methodology, Z.L. and B.W.; software, B.W. and X.W.; validation, Z.L., B.W. and X.W.; formal analysis, X.M. and C.Z.; investigation, B.W.; resources, and C.Z.; data curation, B.W.; writing—original draft preparation, B.W.; writing—review and editing, Z.L.; supervision, Z.L.; project administration, Z.L.; funding acquisition, Z.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, funding number 52175497, and the Key R&D Program of Heilongjiang Province in China, funding number JD22A014.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to express their sincere thanks to all of the members of the research team for their contributions to this research. The authors are also grateful in advance to the reviewers and editors for their comments and suggestions on how to improve the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chung, W.C.; Jin, C.; Kim, M. Dual-algorithm hybrid method for riser structural health monitoring using the fewest sensors. *J. Mar. Sci. Eng.* **2022**, *10*, 1994. [CrossRef]
2. Kim, H.; Jin, C.; Kim, M. Real-time estimation of riser's deformed shape using inclinometers and extended Kalman filter. *Mar. Struct.* **2021**, *77*, 102933. [CrossRef]
3. Suzuki, R.; Tsukada, Y.; Ueno, M. Estimation of full-scale ship manoeuvrability in adverse weather using free-running model test. *Ocean Eng.* **2020**, *213*, 107562. [CrossRef]
4. Zhang, X.X.; Shan, X.B.; Xie, T.; Miao, J.M.; Du, H.J.; Song, R.J. Harbor seal whisker inspired self-powered piezoelectric sensor for detecting the underwater flow angle of attack and velocity. *Measurement* **2021**, *172*, 108866. [CrossRef]
5. Zheng, J.; Chen, J.; Wu, X.; Liang, H.; Zheng, Z.; Zhu, C.; Liu, Y.; Sun, C.; Wang, C.; He, D. Analysis and compensation of installation perpendicularity error in unmanned surface vehicle electro-optical devices by using sea-sky line images. *J. Mar. Sci. Eng.* **2023**, *11*, 863. [CrossRef]
6. Hu, P.; Cui, Y.; Zhao, C.; Li, Y.; Li, B. Numerical investigation on the hydrodynamic response of pentamaran-resistance analysis of different outrigger inclination angles. *J. Mar. Sci. Eng.* **2023**, *11*, 186. [CrossRef]
7. Lee, D.; Ku, N.; Kim, T.W.; Kim, J.; Lee, K.Y.; Son, Y.S. Development and application of an intelligent welding robot system for shipbuilding. *Robot. Comput. Integr. Manuf.* **2011**, *27*, 377–388. [CrossRef]
8. Chen, S.F.; Liu, J.; Chen, B.; Suo, X.Y. Universal fillet weld joint recognition and positioning for robot welding using structured light. *Robot. Comput. Integr. Manuf.* **2022**, *74*, 102279. [CrossRef]
9. Wu, W.; Chen, S.; Qin, S. Online estimation of ship dynamic flexure model parameters for transfer alignment. *IEEE Trans. Control Syst. Technol.* **2013**, *21*, 1666–1678. [CrossRef]
10. Hasanvand, A.; Hajivand, A.; Ali, N.A. Investigating the effect of rudder profile on 6DOF ship turning performance. *Appl. Ocean Res.* **2019**, *92*, 101918. [CrossRef]
11. Zhou, R.; Chen, J.; Tan, W.; Cai, C. Sensor selection for optimal target localization with 3-D angle of arrival estimation in underwater wireless sensor networks. *J. Mar. Sci. Eng.* **2022**, *10*, 245. [CrossRef]

12. Ullah, I.; Chen, J.; Su, X.; Esposito, C.; Choi, C. Localization and detection of targets in underwater wireless sensor using distance and angle based algorithms. *IEEE Access* **2019**, *7*, 45693–45704. [CrossRef]
13. Wang, P.; Tian, X.L.; Peng, T.; Luo, Y. A review of the state-of-the-art developments in the field monitoring of offshore structures. *Ocean Eng.* **2018**, *147*, 148–164. [CrossRef]
14. Mentjes, J.; Wiards, H.; Feuerstack, S. Berthing assistant system using reference points. *J. Mar. Sci. Eng.* **2022**, *10*, 385. [CrossRef]
15. Jiménez Espadafor, F.J.; Becerra Villanueva, J.A.; Guerrero, D.P.; García, M.T.; Trujillo, E.C.; Vacas, F.F. Measurement and analysis of instantaneous torque and angular velocity variations of a low speed two stroke diesel engine. *Mech. Syst. Signal. Process.* **2014**, *49*, 135–153. [CrossRef]
16. Doan, M.N.; Kai, Y.; Obi, S. Twin marine hydrokinetic cross-flow turbines in counter rotating configurations: A laboratory-scaled apparatus for power measurement. *J. Mar. Sci. Eng.* **2020**, *8*, 918. [CrossRef]
17. Li, Z.P.; Wang, B.N.; Meng, X.; Zhang, C. Principle, study status and development trend of the electromagnetic torque sensor. *Chin. J. Sci. Instrum.* **2021**, *42*, 1–14. [CrossRef]
18. Dudojc, B.; Mindykowski, J. New approach to analysis of selected measurement and monitoring systems solutions in ship technology. *Sensors* **2019**, *19*, 1775. [CrossRef]
19. Mohankumar, P.; Ajayan, J.; Yasodharan, R.; Devendran, P.; Sambasivam, R. A review of micromachined sensors for automotive applications. *Measurement* **2019**, *140*, 305–322. [CrossRef]
20. Noguchi, A.; Yamawaki, K.; Yamamoto, T.; Toratani, T. Development of a steering angle and torque sensor of contact-type. *Furukawa Rev* **2004**, *25*, 36–41.
21. Lee, K.T.; Chee, P.S.; Lim, E.H.; Kam, Y.H. Data glove with integrated Polyethylene-Carbon Composite-Based strain sensor for virtual reality applications. *Chem. Eng. Technol.* **2023**, *46*, 1–8. [CrossRef]
22. Fu, Y.C.; Fan, W.; Jin, H.X.; Chen, Q. A new capacitance angle sensor of concentric ring multi-layer differential. *Measurement* **2020**, *158*, 107625. [CrossRef]
23. Wu, J.; Meng, Z.; Zhang, X.; Mi, W.; Yan, Y. Capacitive angle sensor research using COMSOL Multiphysics. *Appl. Sci.* **2023**, *13*, 2937. [CrossRef]
24. Sreevidya, P.V.; Borole, U.P.; Kadam, R.; Khan, J.; Barshilia, H.C.; Chowdhury, P. A novel AMR based angle sensor with reduced harmonic errors for automotive applications. *Sens. Actuators A Phys.* **2021**, *324*, 112573. [CrossRef]
25. Cubells-beltrán, M.D.; Reig, C.; Madrenas, J.; Marcellis, A.D.; Santos, J.; Cardoso, S.; Freitas, P.P. Integration of GMR sensors with different technologies. *Sensors* **2016**, *16*, 939. [CrossRef]
26. Wang, Y.Y.; Qin, Y.; Chen, X.H.; Peng, D.L.; Tang, Q.F.; Zhang, T.H. Angular displacement sensor for gear position detection based on the tunneling magnetoresistance effect. *Meas. Sci. Technol.* **2023**, *34*, 025112. [CrossRef]
27. Li, Z.P.; Zhang, C.; Shi, S.Z.; Meng, X.; Wang, B.N. Design and parameter optimization of contactless vertical inductive angle sensor. *Vacuum* **2019**, *169*, 108865. [CrossRef]
28. Li, Y.; Li, R.; Yang, J.; Xu, J.; Yu, X. Effect of excitation signal on double-coil inductive displacement transducer. *Sensors* **2023**, *23*, 3780. [CrossRef]
29. Hou, H.; Cao, G.H.; Ding, H.C.; Li, K. Research on particle swarm compensation method for subdivision error optimization of photoelectric encoder based on parallel iteration. *Sensors* **2022**, *22*, 4456. [CrossRef]
30. Wang, X.J. Errors and precision analysis of subdivision signals for photoelectric angle encoders. *Opt. Precis. Eng.* **2012**, *20*, 379–386. [CrossRef]
31. Tan, N.Y.J.; Liu, K.; Senthil Kumar, A. Multiple axis ultra-precision freeform deviation control using encoder data analysis. *J. Manuf. Process.* **2023**, *90*, 242–256. [CrossRef]
32. Das, S.; Chakraborty, B. Design and realization of an optical rotary sensor. *IEEE Sens. J* **2018**, *18*, 2675–2681. [CrossRef]
33. Agarwal, S.; Shakher, C. In-plane displacement measurement by using circular grating Talbot interferometer. *Opt. Lasers Eng.* **2015**, *75*, 63–71. [CrossRef]
34. Guerrero-Mendez, C.D.; Arco, L.D.; Pontes, M.J.; Segatto, M.E.V.; Cifuentes, C.A.; Bastos-Filho, T.F.; Diaz, C.A.R. Fatigue test on optical fiber angle sensors based on polymeric materials for flexion–extension applications. *Opt. Fiber Technol.* **2023**, *78*, 103334. [CrossRef]
35. Twu, R.C.; Lin, B.L. Development and investigation of birefringent Beta-Barium borate for optical angle sensor. *Opt. Laser Technol.* **2022**, *155*, 108369. [CrossRef]
36. Dauth, R.A.; Gerlach, G.; Fella, S. An effective method to model and simulate the behavior of inductive angle encoders. *Sensors* **2022**, *22*, 7804. [CrossRef] [PubMed]
37. Ye, L.; Yang, M.; Xu, L.; Zhuang, X.Q.; Dong, Z.P.; Li, S.Y. Nonlinearity analysis and parameters optimization for inductive angle sensor. *Sensors* **2014**, *14*, 4111–4125. [CrossRef]
38. Ye, L.; Yang, M.; Xu, L.; Guo, C.; Li, L.; Wang, D.Q. Optimization of inductive angle sensor using response surface methodology and finite element method. *Measurement* **2014**, *48*, 252–262. [CrossRef]
39. Jia, H.Z. Current status of electron beam selective melting additive manufacturing technology. *J. Phys. Conf. Ser.* **2021**, *2044*, 012126. [CrossRef]
40. Hartog, S.D.; Neukermans, S.; Samanipour, M.; Vincent Ching, H.Y.; Breugelmans, T.; Hubin, A.; Ustarroz, J. Electrocatalysis under a magnetic lens: A combined electrochemistry and electron paramagnetic resonance review. *Electrochim. Acta* **2022**, *407*, 139704. [CrossRef]

41. Zhang, L.P.; Chang, L.; Yuan, X.G.; Zhang, J.H.; Zhou, H.S.; Luo, G.N. A RF plasma source with focused magnetic field for material treatment. *Plasma Chem. Plasma Process.* **2023**, *43*, 329–345. [CrossRef]
42. Konečná, A.; Javier García de Abajo, F. Electron beam aberration correction using optical near fields. *Phys. Rev. Lett.* **2020**, *125*, 030801. [CrossRef] [PubMed]
43. Grillo, V.; Tavabi, A.H.; Yucelen, E.; Lu, P.H.; Venturi, F.; Larocque, H.; Jin, L.; Savenko, A.; Gazzadi, G.C.; Balboni, R.; et al. Towards a holographic approach to spherical aberration correction in scanning transmission electron microscopy. *Opt. Express* **2017**, *25*, 21851–21860. [CrossRef]
44. Siegbahn, K.; Svartholm, N. Focusing of electrons in two dimensions by an inhomogeneous magnetic field. *Nature* **1946**, *157*, 872–873. [CrossRef]
45. Walton, E.T.S. High-order focusing by a uniform magnetic field with straight-line boundaries. *Nature* **1954**, *173*, 1147–1148. [CrossRef]
46. Damadian, R.; Minkoff, L.; Goldsmith, M.; Stanford, M.; Koutcher, J. Field focusing nuclear magnetic resonance (FONAR): Visualization of a tumor in a live animal. *Science* **1976**, *194*, 1430–1432. [CrossRef]
47. Lefaucheur, J.P. Transcranial magnetic stimulation. *Handb. Clin. Neurol.* **2019**, *160*, 559–580. [CrossRef]
48. Mazurek, M.H.; Cahn, B.A.; Yuen, M.M.; Prabhat, A.M.; Chavva, I.R.; Shah, J.T.; Crawford, A.L.; Brian-welch, E.; Rothberg, J.; Sacolick, L.; et al. Portable, bedside, low-field magnetic resonance imaging for evaluation of intracerebral hemorrhage. *Nat. Commun.* **2021**, *12*, 5119. [CrossRef]
49. Kee, H.; Lee, H.; Park, S. Optimized Halbach array for focused magnetic drug targeting. *J. Magn. Magn. Mater.* **2020**, *514*, 167180. [CrossRef]
50. Liu, Y.L.; Chen, D.; Shang, P.; Yin, D.C. A review of magnet systems for targeted drug delivery. *J. Control. Release* **2019**, *302*, 90–104. [CrossRef] [PubMed]
51. Alexiou, C.; Diehl, D.; Henninger, P.; Iro, H.; Rockelein, R.; Schmidt, W.; Weber, H. A high field gradient magnet for magnetic drug targeting. *IEEE Trans. Appl. Supercond.* **2006**, *16*, 1527–1530. [CrossRef]
52. Zhao, Y.Z.; Wang, X.H.; Sun, T.; Chen, Y.C.; Yang, L.; Zhang, T.; Ju, H.Y. Non-contact harmonic magnetic field detection for parallel steel pipeline localization and defects recognition. *Measurement* **2021**, *180*, 109534. [CrossRef]
53. Cui, P.; Zhu, W.B.; Ji, H.J.; Chen, H.T.; Hang, C.J.; Li, M.Y. Analysis and optimization of induction heating processes by focusing the inner magnetism of the coil. *Appl. Energy* **2022**, *321*, 119316. [CrossRef]
54. Zhang, X.; Li, W.W.; Li, B.; Tu, J.; Liao, C.H.; Wu, Q.; Feng, S.; Song, X.C. A new design of the dual-mode and pure longitudinal EMAT by using a radial-flux-focusing magnet. *Sensors* **2022**, *22*, 1316. [CrossRef]
55. Chen, X.D.; Wang, E.H.; Shan, L.K.; Feng, C.; Zheng, Y.; Dong, Y.; Guo, G.C.; Sun, F.W. Focusing the electromagnetic field to $10^{-6}\lambda$ for ultra-high enhancement of field-matter interaction. *Nat. Commun.* **2021**, *12*, 6389. [CrossRef]
56. Daskalakis, Z.J.; Farzan, F.; Barr, M.S.; Maller, J.J.; Chen, R.; Fitzgerald, P.B. Long-interval cortical inhibition from the dorsolateral prefrontal cortex: A TMS-EEG study. *Neuropsychopharmacology* **2008**, *33*, 2860–2869. [CrossRef]
57. Li, J.T.; Liang, Z.; Ai, Q.Y.; Yan, X.H.; Tian, J. Double butterfly coil for transcranial magnetic stimulation aiming at improving focality. *IEEE Trans. Magn.* **2012**, *48*, 3509–3512. [CrossRef]
58. Rotundo, S.; Brizi, D.; Flori, A.; Giovannetti, G.; Menichetti, L.; Monorchio, A. Shaping and focusing magnetic field in the human body: State-of-the art and promising technologies. *Sensors* **2022**, *22*, 5132. [CrossRef]
59. Lee, S.W.; Fallegger, F.; Casse, B.D.F.; Fried, S.I. Implantable microcoils for intracortical magnetic stimulation. *Sci. Adv.* **2016**, *2*, e1600889. [CrossRef]
60. Philip, N.S.; LaBar, K.S. Mapping a pathway to improved neuropsychiatric treatments with precision transcranial magnetic stimulation. *Sci. Adv.* **2022**, *8*, q7254. [CrossRef]
61. Amigo, I. Stimulating the brain may help people who stutter. *Science* **2022**, *376*, 1365–1366. [CrossRef] [PubMed]
62. Wadman, M. Zapping cocaine addiction. *Science* **2017**, *357*, 960–963. [CrossRef]
63. Liu, S.W.; Sun, Y.H.; He, L.S.; Kang, Y.H. Magnetic focusing method and sensor in surface topography testing for ferromagnetic materials. *Sens. Actuators A Phys.* **2019**, *285*, 531–542. [CrossRef]
64. Passarotto, M.; Qama, G.; Specogna, R. A fast and efficient simulation method for inductive position sensors design. In Proceedings of the 2019 IEEE Sensors of the Conference, Montreal, QC, Canada, 27–30 October 2019. [CrossRef]
65. Anandan, N.; Varma Muppala, A.; George, B. A flexible, planar-coil-based sensor for through-shaft angle sensing. *IEEE Sens. J.* **2018**, *18*, 10217–10224. [CrossRef]
66. Yang, C.; Tian, Y.; Yu, J.; Jin, M.; Xie, N.; Deng, Y. A 3D printing method of customized magnetic focusing generator for magnetic field therapy. *Acad. J. Med. Health Sci.* **2022**, *3*, 44–53. [CrossRef]
67. Anil Kumar, A.S.; George, B. A noncontact angle sensor based on eddy current technique. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 1275–1283. [CrossRef]
68. Liu, C.; Ding, H.; Fang, X.; Wang, Z. Optimal design of transcranial magnetic stimulation thin core coil With trade-off between stimulation effect and heat energy. *IEEE Trans. Appl. Supercond.* **2020**, *30*, 1–6. [CrossRef]
69. Zhang, Z.; Ni, F.; Dong, Y.; Guo, C.; Jin, M.; Liu, H. A novel absolute magnetic rotary sensor. *IEEE Trans. Ind. Electron.* **2015**, *62*, 4408–4419. [CrossRef]

70. Liu, J.; Lu, J.; Liu, C.; Hu, Y. Coil arrays modeling and optimization for transcranial magnetic stimulation. In Proceedings of the 2009 2nd International Conference on Biomedical Engineering and Informatics, Tianjin, China, 17–19 October 2009; pp. 1–5. [CrossRef]
71. Wu, Y.X.; Yu, H.Y.; Liu, Z.W. Numerical investigation of the magnetic and electric field distributions produced by biconical transcranial magnetic stimulation coil for optimal design. *IEEE Trans. Magn.* **2018**, *54*, 1–5. [CrossRef]
72. Zhang, X.; Wang, Y. Calculation of transient magnetic field and induced voltage in photovoltaic bracket system during a lightning stroke. *Appl. Sci.* **2021**, *11*, 4567. [CrossRef]
73. Zhang, G.Z.; Chen, K.; Li, X.H.; Wang, K.; Fang, R.X.; Liu, J.B. Flexible built-in miniature Archimedes spiral antenna sensor for PD detection in GIS. *High Volt. Eng.* **2022**, *48*, 2244–2254.
74. Song, Z.H.; Zhang, J.; Liu, M.J. Hemispherical Archimedean spiral antenna for multi-mode satellite navigation signals receiving. *J. Astronaut.* **2010**, *31*, 391–396.
75. Aditya, K. Analytical design of Archimedean spiral coils used in inductive power transfer for electric vehicles application. *Electr. Eng.* **2018**, *100*, 1819–1826. [CrossRef]
76. Rodríguez-Maldonado, J.; Posadas-Castillo, C.; Zambrano-Serrano, E. Alternative method to estimate the Fourier expansions and its rate of change. *Mathematics* **2022**, *10*, 3832. [CrossRef]
77. Pachuau, J.L.; Kashyap, P.; Kumar, A.; Paul, R.; Id, P.; Chandrakiran, B.; Debnath, S.; Saha, A.K. Segmentation of composite signal into harmonic Fourier expansion using genetic algorithm. *Int. J. Inf. Technol.* **2022**, *14*, 3507–3515. [CrossRef]
78. Liang, Z.G.; Tang, R.Y. Analytical solution of eddy current loss in core of an electromagnetic valve. *Proc. CSEE* **2005**, *25*, 153–157. [CrossRef]
79. Jiao, S.; Liu, X.; Zeng, Z. Intensive study of skin effect in eddy current testing with pancake coil. *IEEE Trans. Magn.* **2017**, *53*, 1–8. [CrossRef]
80. Mirrashid, M.; Naderpour, H. Incomprehensible but Intelligible-in-time logics: Theory and optimization algorithm. *Knowl.-Based Syst.* **2023**, *264*, 110305. [CrossRef]
81. Mirrashid, M.; Naderpour, H. Transit search: An optimization algorithm based on exoplanet exploration. *Results Control Optim.* **2022**, *7*, 100127. [CrossRef]
82. Hoxha, A.; Passarotto, M.; Qama, G.; Specogna, R. Design optimization of PCB-Based rotary-inductive position sensors. *Sensors* **2022**, *22*, 4683. [CrossRef]
83. Sharmin, S.; Ahmedy, I.; Md Noor, R. An energy-efficient data aggregation clustering algorithm for wireless sensor Networks using hybrid PSO. *Energies* **2023**, *16*, 2487. [CrossRef]
84. Shami, T.M.; El-Saleh, A.A.; Alswaitti, M.; Al-Tashi, Q.; Summakieh, M.A.; Mirjalili, S. Particle swarm optimization: A comprehensive survey. *IEEE Access* **2022**, *10*, 10031–10061. [CrossRef]
85. Wang, J.; Wang, X.; Li, X.; Yi, J. A hybrid particle swarm optimization algorithm with dynamic adjustment of inertia weight based on a new feature selection method to optimize SVM parameters. *Entropy* **2023**, *25*, 531. [CrossRef] [PubMed]
86. Anandan, N.; George, B. Design and development of a planar linear variable differential transformer for displacement sensing. *IEEE Sens. J.* **2017**, *17*, 5298–53051. [CrossRef]
87. Wang, K.; Zhang, L.S.; Le, Y.; Zheng, S.Q.; Han, B.C.; Jiang, Y.X. Optimized differential self-inductance displacement sensor for magnetic bearings: Design, analysis and experiment. *IEEE Sens. J.* **2017**, *17*, 4378–4387. [CrossRef]
88. Fonte, M.; Duarte, P.; Anes, V.; Freitas, M.; Reis, L. On the assessment of fatigue life of marine diesel engine crankshafts. *Eng. Fail. Anal.* **2015**, *56*, 51–57. [CrossRef]
89. Li, R.; Wen, C.W.; Meng, X.H.; Xie, Y.B. Measurement of the friction force of sliding friction pairs in low-speed marine diesel engines and comparison with numerical simulation. *Appl. Ocean Res.* **2022**, *121*, 103089. [CrossRef]
90. Tawadros, P.; Awadallah, M.; Walker, P.; Zhang, N. Using a low-cost bluetooth torque sensor for vehicle jerk and transient torque measurement. *Proc. Inst. Mech. Eng. Part D* **2019**, *234*, 423–437. [CrossRef]
91. Gryz, K.; Karpowicz, J.; Zradziński, P. Complex electromagnetic issues associated with the use of electric vehicles in urban transportation. *Sensors* **2022**, *22*, 1719. [CrossRef]
92. Jerančе, N.; Vasiljević, D.; Samardžić, N.; Stojanović, G. A compact inductive position sensor made by inkjet printing technology on a flexible substrate. *Sensors* **2012**, *12*, 1288–1298. [CrossRef]
93. Losby, J.E.; Sauer, V.T.K.; Freeman, M.R. Recent advances in mechanical torque studies of small-scale magnetism. *J. Phys. D Appl. Phys.* **2018**, *51*, 483001. [CrossRef]

Journal of
*Marine Science
and Engineering*

Article

Improved UNet-Based Shoreline Detection Method in Real Time for Unmanned Surface Vehicle

Jiansen Zhao [1], Fengchuan Song [1,*], Guobao Gong [2] and Shengzheng Wang [1]

[1] Merchant Marine College, Shanghai Maritime University, Shanghai 201306, China; jszhao@shmtu.edu.cn (J.Z.)
[2] Navigation College, Dalian Maritime University, Dalian 116026, China
* Correspondence: 202130110062@stu.shmtu.edu.cn

Abstract: Accurate and real-time monitoring of the shoreline through cameras is an invaluable guarantee for the safety of near-shore navigation and berthing of unmanned surface vehicles; existing shoreline detection methods cannot meet both these requirements. Therefore, we propose an improved shoreline detection method to detect shorelines accurately and in real time. We define shoreline detection as the combination of water surface area segmentation and edge detection, the key to which is segmentation. To detect shorelines accurately and in real time, we propose an improved U-Net for water segmentation. This network is based on U-Net, using ResNet-34 as the backbone to enhance the feature extraction capability, with a concise decoder integrated attention mechanism to improve the processing speed while ensuring the accuracy of water surface segmentation. We also introduce transfer learning to improve training efficiency and solve the problem of insufficient data. When obtaining the segmentation result, the Laplace edge detection algorithm is applied to detect the shoreline. Experiments show that our network achieves 97.05% MIoU and 40 FPS with the fewest parameters, which is better than mainstream segmentation networks, and also demonstrate that our shoreline detection method can effectively detect shorelines in real time in various environments.

Keywords: water surface segmentation; attention mechanism; edge detection; shoreline detection

Citation: Zhao, J.; Song, F.; Gong, G.; Wang, S. Improved UNet-Based Shoreline Detection Method in Real Time for Unmanned Surface Vehicle. *J. Mar. Sci. Eng.* **2023**, *11*, 1049. https://doi.org/10.3390/jmse11051049

Academic Editor: Sergei Chernyi

Received: 24 March 2023
Revised: 22 April 2023
Accepted: 9 May 2023
Published: 15 May 2023

1. Introduction

Unmanned surface vehicles (USVs) have dramatically developed in recent years thanks to technical advancements. These intelligent devices can be navigated by manual or programmed control to accomplish a variety of tasks. Real-time and accurate monitoring of the shoreline is important when using these autonomous surface vehicles, both for the safety of berthing and near-shore navigation.

On large vessels, several types of sensors are installed to monitor the surrounding environment, such as cameras [1,2] and radar [3,4]. These devices can provide various forms of environmental information to the ship, but considering the limitations of USVs themselves in terms of carrying capacity and energy supply, they cannot equip huge or a large number of sensors. As a result, visual sensors such as cameras that are lighter and more energy-efficient while still offering extensive environmental information are better-suited for USVs. Based on this analysis, a visual-based shoreline detection method is crucial for USVs.

Depending on their technical means, existing visual-based shoreline detection methods can be classified into traditional image-based methods and deep-learning-based methods. Traditional methods include local binary patterns combined with the gray-level co-occurrence matrix method [5], column-by-column logistic regression combined with the polynomial spline modeling method [6], the calculation of vertical gradients in gray space combined with the random sample consensus (RANSAC) algorithm fitting method [7], and the calculation of morphological gradients on HSV color space combined with the watershed algorithm and edge detection method [8]. The abovementioned methods are subject to limitations in their use and are susceptible to water surface reflections, light changes, waves, and long processing times, making them unable to meet the need for accurate and real-time detection of shorelines. In recent years, artificial intelligence has been greatly developed and been widely used in the maritime fields, such as for ship detection [9,10], ship trajectory analysis [11] and prediction [12], marine accident risk quantification [13], and environmental perception [14]. As for shoreline detection, some researchers have attempted to introduce semantic segmentation techniques into this field [15] by first extracting the water surface area with a semantic segmentation network and using an edge detection algorithm on the obtained result to detect the shoreline. Based on this, some works have improved existing semantic segmentation models, making them more suitable for water surface area segmentation [16–18], and other studies have introduced pretrained methods [19] or the use of transfer learning [20] to solve the problem of an insufficient amount of data for training, as well as use self-supervised training approaches [21] to address the problem of insufficient labeled data. The trained models have strong robustness, which is good for solving the interference of environmental factors present in traditional image-based approaches but still cannot address the processing speed problem due to the use of a large network architecture, the large scale of the feature maps, etc.

To address the abovementioned problem, i.e., that existing visual-based shoreline detection methods cannot meet the requirement of shoreline detection both accurately and in real time, we constructed a better method to achieve real-time and accurate shoreline detection. We define shoreline detection as the combination of water surface segmentation and edge detection. According to our definition, the key to shoreline detection is water surface segmentation, which directly determines the accuracy and inference speed of our method. Edge detection has almost no impact on either of these factors. Therefore, to achieve the abovementioned target, we propose an improved U-Net network to perform water surface segmentation accurately and in real time. This network is based on U-Net combined with a residual network [22] and a squeeze-and-excitation (SE) attention module [23] to increase the segmentation accuracy and processing efficiency and named the Residual Squeeze-and-excitation U-Net (RS-UNet). According to experimental verification, the network proposed in this paper achieved a processing speed of 40 FPS and 97.05% MIoU, outperforming some mainstream methods of semantic segmentation, meeting the demand for real-time and accurate water surface area segmentation, and shoreline detection when combined with an edge detection algorithm. Other experiments also demonstrate the generalization capability of our method. Specifically, the contributions of this paper include:

- An encoder is built based on ResNet-34 to enhance the feature extraction capability of the network in complicated environments, with the introduction of transfer learning using pretrained ResNet-34 weights to improve the training efficiency and solve the problem of insufficient training data;
- To reduce the amount of computation, a lightweight decoder is built, and an attention mechanism is added to the decoder to force the network to pay more attention to the data in the critical part throughout the segmentation process, increasing the computational speed and maintaining the segmentation quality;
- Construction of a shoreline detection method based on the proposed RS-UNet, which can accurately detect the shoreline in real time and be applied in various environments.

The remainder of this paper is organized as follows. Section 2 presents a brief review of related works. Section 3 explains our proposed method. Section 4 shows the experimental results and analyses of these results. Section 5 provides a summary of our work and directions for future work. We also provide a list of abbreviations in Abbreviations for a better reading experience.

2. Related Work

2.1. Traditional Shoreline Detection Method

Traditional shoreline detection methods detect the shoreline through image processing; for example, Kristan et al. [24] proposed the imposition of weak structural constraints and a Markov random field to account for the semantic structure of the marine environment in real time. At present, the accuracy of this method is relatively low. Kröhnert [6] proposed the use of column-by-column logistic regression and polynomial spline modeling to detect shorelines, which has a good detection effect for nearly straight shorelines but a relatively poor effect for more curved shorelines. Wei and Zhang [5] used local binary patterns and a gray-level co-occurrence matrix to calculate river texture information and used structure detection to eliminate the effects of wind and light, which is also a suitable method for more flat shorelines. Zhan et al. [7] calculated the vertical gradient on a gray image with the background texture removed, obtained the water shoreline candidate points from the vertical gradient of each column, and fitted the shoreline from the candidate points using the RANSAC algorithm. The processing speed of this method is faster, but it is susceptible to the influence of water reflection or texture. Feng et al. [8] computed morphological gradients in HSV color space to highlight edges, then used the watershed algorithm to segment the image area, combined with the use of a filtering operator to detect a river shoreline, which achieved real-time shoreline detection but was still subject to the influence of ambient lighting to some extent. Peng et al. [25] analyzed the differences in the characteristics of images in HSV color space under different lighting conditions. Different regions in the image were segmented, and the shoreline was detected based on the differences in saturation and brightness between land and water areas. This method is less stable and easily affected by environmental changes and lighting variations.

2.2. Deep-Learning-Based Shoreline Detection Method

The key to the deep-learning-based method is the segmentation of the water surface area. After obtaining the segmented result, the corresponding shoreline can be obtained using the edge detection algorithm, so it is essentially a semantic segmentation problem. For example, Steccanella et al. [15] used a fully convolutional neural network to detect the water surface area and obtained a high segmentation accuracy rate. However, this method could not meet the requirement of real-time processing. Steccanella et al. [16] further improved this method and achieved 98.8% pixel segmentation accuracy and 10 FPS on a 160 × 160 image. To address the problem of an insufficient amount of data for water surface segmentation training, Adam et al. [19] demonstrated that the use of a pretrained backbone can significantly improve the network's ability to segment water surface regions, and Vandaele et al. [20] proposed the use of a transfer learning approach that completes pretraining on the COCO dataset and is then fine-tuned on the water surface segmentation dataset. Zhan et al. [21] combined a semantic segmentation network with conditional random fields (CRFs) and superpixel mapping to propose an adaptive water surface segmentation network that effectively solves the training problem on datasets with limited labeled data. Shen et al. [18] used improved DeepLab v3+ to acquire water surface area segmentation results combined with an edge detection algorithm to detect the shoreline, which effectively overcomes the interference of factors such as reflection, although the processing speed of this method is 8 FPS, which is far from meeting the demand for real-time detection. Yao et al. [26] proposed ShrelineNet to detect shorelines for USVs, which segments the entire image into sky, land, and water sections. Then, the shoreline is detected based on the water region. Yin et al. [17] applied the improved PSPNet to the water surface segmentation task and later used the Canny edge detection algorithm to detect the shoreline, obtaining a segmentation MIoU of 96.87% on the USVInland dataset [27].

3. Method

3.1. U-Net Network

The U-Net network [28], which is a fully convolutional network (FCN) [29], achieves outstanding segmentation performance on small sample datasets for the job of segmenting medical images. As shown in Figure 1, the distinctive characteristic of U-Net is fully symmetric encoder–decoder composition. The feature maps input at each stage of the encoder and decoder are subjected to two consecutive 3 × 3 convolution processes without padding, while each downsampling in the encoder corresponds to a 2 × 2 upsampling in the decoder. In addition, U-Net crops the feature maps of each layer of the encoder for the decoding process by Crop and Copy operation to supplement part of the information lost in the downsampling and upsampling process.

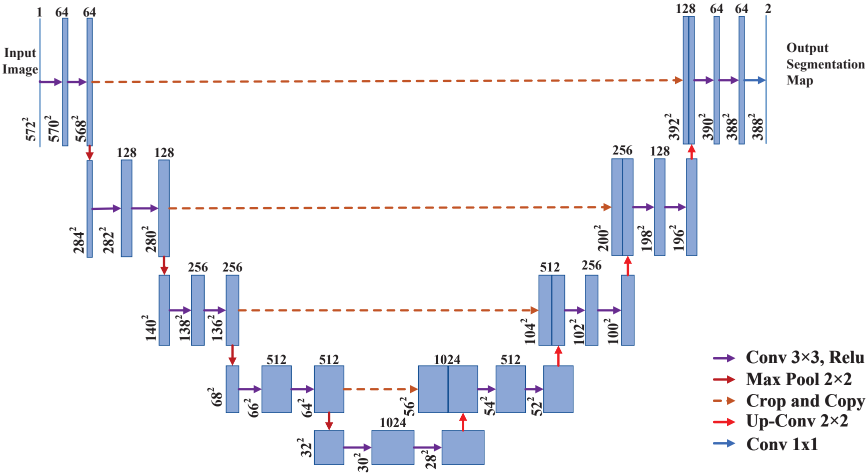


Figure 1. U-Net network architecture [28]. The input is a gray image, and the output is the probability that each pixel belongs to each category. The input image is downsampled four times to extract features of different levels and upsampled an equal number of times to recover the original resolution. Crop and Copy operation is used to first crop the feature map of different layers of the encoder, then copy them to the decoder for semantic segmentation.

3.2. RS-UNet Network

The RS-UNet network proposed in this paper is based on U-Net owing to the similarity in nature between water surface area segmentation and medical image segmentation. Medical image segmentation is essentially a binary segmentation task that segments lesion regions or other regions of interest in the input images. The water surface area segmentation problem that we wish to solve is also a binary segmentation task that involves segmenting the water surface area in the input image. Both tasks are simultaneously plagued by the problem of a limited quantity of training samples. For this reason, we think that in this study, we can take design inspirations from U-Net to build a water surface region segmentation network.

This network maintains the encoder–decoder architecture and employs equal amounts of downsampling and upsampling, as shown in Figure 2. To ensure that the final segmentation result is consistent with the resolution of the input image and that the feature maps of the corresponding stages of the encoder and decoder have the same spatial resolution, the convolution operation of each stage is padded according to the filter size of the convolution layer. This enables us to fully utilize the output of various stages of the encoder to make up for the information loss caused by sampling and to make comprehensive use of the contextual information at various scales to better complete the task of water surface area segmentation by using a skip connection instead of Crop and Copy opreation as in the original U-Net.

For the water surface area segmentation task, the effectiveness of the extracted features from the input image directly affects the segmentation results. Natural images used for this task are more complicated and contain more information than medical images, so the 10 layer convolutional network in the original U-Net encoder is unable to extract a sufficient amount of useful features from such complex scenes. To improve the feature extraction capability of the encoder while controlling the FLOPs of the network, we use

ResNet-34 [22] with the final average pooling layer and fully connected layer removed as the backbone to extract features. Because of the deeper network architecture and fewer FLOPs of the backbone, the encoder of RS-UNet can effectively extract more high-level and richer contextual features, which lays a foundation for the network to better complete the segmentation task with less inference time. In addition, the utilization of ResNet provides conditions for the introduction of the transfer learning approach, which solves the problem of insufficient training data and improves the training efficiency of the network.

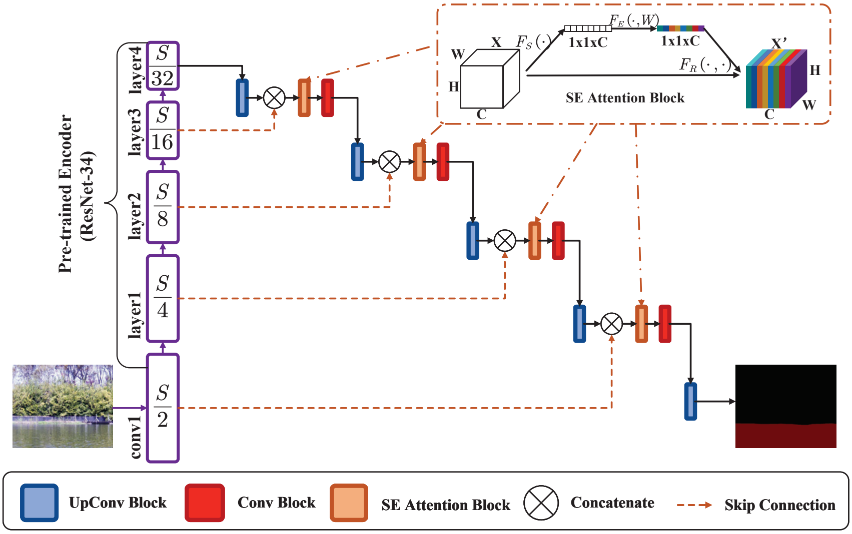


Figure 2. RS-UNet network architecture. S indicates the spatial resolution of the input image. The input is an RGB image, and the output is the segmentation result. The Encoder is the pretrained ResNet-34 and is fine-tuned during training. There are two input feature maps with different resolutions for each layer in the decoder. The lower feature is upsampled by the UpConv block and concatenated with the bigger larger feature through a skip connection. Then, interdependence between channels is modeled through the SE attention block and processed by the Conv block. At the end of the network, one UpConv block recovers the spatial resolution to the resolution of the input image and classifies each pixel into a category.

Theoretically, the real-time processing capability means that the network has low FLOPs. According to this theory, we built a very simple decoder for the network. The main computational body of each layer contains only one transposed convolution operation and one convolution operation; the former is responsible for recovering the spatial resolution, and the latter is responsible for processing the concatenated feature map. This architecture guarantees that the decoder contains low FLOPs, but it also leads to a loss of the computational capacity of the decoder. To overcome this weakness, the SE attention module [23] is introduced to each layer of the decoder. By modeling the interdependence between different channels of the concatenated feature map, this module can help the decoder focus its limited computational capacity on important features to improve performance. The additional computation required to introduce this kind of module is almost negligible. Table 1 shows the details of the network architecture.

Table 1. The architectural details of the network. Up: upsampling multiplier; $Channels$: the number of channels of each input and output module; In and Out: spatial resolutions of the feature maps; $Input$: input content of the module; $\otimes$: concatenation operation; S: spatial resolution of the original image; $F(B)$: the output corresponding to block B.

Encoder

Block	Filter Size	Stride	Channels	In	Out	Input
conv1	7 × 7	2	3/64	S	$S/2$	Input image
MaxPooling	2 × 2	2	64/64	$S/2$	$S/4$	$F(conv1)$
layer1	3 × 3	1	64/64	$S/4$	$S/4$	$F(MaxPooling)$
layer2	3 × 3	2	64/128	$S/4$	$S/8$	$F(layer1)$
layer3	3 × 3	2	128/256	$S/8$	$S/16$	$F(layer2)$
layer4	3 × 3	2	256/512	$S/16$	$S/32$	$F(layer3)$

Decoder

Block	Filter Size	Up	Channels	In	Out	Input
de4upconv	2 × 2	2	512/256	$S/32$	$S/16$	$F(layer4)$
de4se		1	512/512	$S/16$	$S/16$	$F(de4upconv \otimes layer3)$
de4conv	3 × 3	1	512/256	$S/16$	$S/16$	$F(de4se)$
de3upconv	2 × 2	2	256/128	$S/16$	$S/8$	$F(de4conv)$
de3se		1	256/256	$S/8$	$S/8$	$F(de3upconv \otimes layer2)$
de3conv	3 × 3	1	256/128	$S/8$	$S/8$	$F(de3se)$
de2upconv	2 × 2	2	128/64	$S/8$	$S/4$	$F(de3conv)$
de2se		1	128/128	$S/4$	$S/4$	$F(de2upconv \otimes layer1)$
de2conv	3 × 3	1	128/64	$S/4$	$S/4$	$F(de2se)$
de1upconv	2 × 2	2	64/64	$S/4$	$S/2$	$F(de2conv)$
de1se		1	128/128	$S/2$	$S/2$	$F(de1upconv \otimes conv1)$
de1conv	3 × 3	1	128/64	$S/2$	$S/2$	$F(de1se)$
upconv	3 × 3	2	64/2	$S/2$	S	$F(de1conv)$

The semantic segmentation task is essentially a pixel-level classification task, so the most commonly used loss function is the cross-entropy loss function. However, considering the potential positive and negative sample imbalance problem in the water segmentation task, a joint loss function (Equation (3)) based on Dice loss (Equation (1)) and focal loss (Equation (2)) is constructed in this paper to replace the cross-entropy loss function to supervise the training of the network.

$$L_{dice}(X,Y) = 1 - \sum_{k=0}^{K} \frac{2\omega_k \sum_{i=1}^{H} \sum_{j=1}^{W} p(X(i,j),k) g(Y(i,j),k)}{\sum_{i=1}^{H} \sum_{j=1}^{W} p(X(i,j),k) + \sum_{i=1}^{H} \sum_{j=1}^{W} g(Y(i,j),k)} \tag{1}$$

where H and W denote the height and width of the image, respectively; X and Y denote the predicted result of the network and the ground truth, respectively; K denotes the number of categories except the background; w_k represents the weight of each category; $p(X(i,j),k)$ denotes the probability of $X(i,j)$ being predicted as category k; and $g(Y(i,j),k)$ denotes the truth label of $Y(i,j)$ corresponding to category k.

$$L_{focal}(X,Y) = -\frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \sum_{k=1}^{K} (1 - p_t(X(i,j),k))^{\gamma} g(Y(i,j),k) \log\left(p_t(X(i,j),k)\right) \tag{2}$$

where $p_t(X(i,j),k)$ denotes the probability of $X(i,j)$ being predicted as category k, and γ is the focusing parameter.

$$L(X,Y) = \omega_{dice} L_{dice}(X,Y) + \omega_{focal} L_{focal}(X,Y) \tag{3}$$

where ω_{dice} and ω_{focal} denote the weight coefficients of Dice loss and focal loss in the loss function, respectively.

3.3. Shoreline Detection

The flow chart of our shoreline detection method is shown in Figure 3. We define shoreline detection as the combination of water surface segmentation and edge detection. The shoreline, which is the edge of the water surface area, can be obtained using the eight-neighborhood Laplace edge detection algorithm [30] based on the results of water surface segmentation. The extracted shoreline is then superimposed on the original picture for display. The outcome of shoreline detection is shown in Figure 4, demonstrating that a simple eight-neighborhood Laplace operator can effectively detect the complete shoreline based on the segmentation results, with only a small amount of additional computation generated.

(a) Input Image **(b)** Semantic Segmentation **(c)** Segmentation Result **(d)** Laplacian Filter **(e)** Detected Shoreline

Figure 3. Flow chart of shoreline detection; from left to right: (**a**) input image; (**b**) semantic segmentation; (**c**) obtained segmentation result; (**d**) process executed by the Laplacian edge detection algorithm; (**e**) display of the detected shoreline.

(**a**) Original image. (**b**) Detected shoreline.

Figure 4. Shoreline detection results. (**a**) The original image; (**b**) the detected shoreline (green lines).

4. Results and Discussion

4.1. Experimental Implementation

Dateset and Evaluation Metrics According to our definition of shoreline detection introduced in Section 3.3, the key to shoreline detection is the semantic segmentation of water surface areas, the accuracy of which can be equivalent to the accuracy of shoreline detection. To enhance the water segmentation capability of the network and its adaptability to different scenes, a new dataset was constructed for the water surface segmentation task. The dataset consists of 433 images of various scenes; the resolutions of these images range from 220 × 165 to 5792 × 4344. These images were collected through the Internet or photographed by ourselves. The dataset includes a wide range of shooting angles and lighting conditions to make the dataset more representative and more widely applicable.

As the accuracy of shoreline detection can be equivalent to the accuracy of water surface segmentation, we directly quantitatively evaluated the performance of our shoreline detection method using the metrics of semantic segmentation, that is, the Dice coefficient (Dice), mean intersection over union (MIoU), category mean pixel accuracy (MPA), and pixel accuracy (PA).

Training Setting For network training, the dataset is divided into a training set and a test set according in a 9-to-1 ratio. We set $\omega_k = \frac{1}{K+1}$ in Equation (1), $\gamma = 2$ in Equation (2), and $\omega_{dice} = \omega_{focal} = 1$ in Equation (3) according to the theory that Dice loss (Equation (1)) and focal loss (Equation (2)) constrain the network updating toward the same target from different perspectives and that their equal status benefits the capability of the network and saves on computation during training. We used adaptive moment estimation (Adam) [31] for optimization. Since pretraining weights were introduced, the combination of freeze training and unfreeze training was adopted. In a total of 150 epochs of the training process, we first went through 30 epochs of freeze training, in which the weights of the backbone were not updated, with an initial learning rate of 10^{-3} and a batch size of 4. The following 120 epochs trained the backbone, together with the rest of the network, with an initial learning rate of 10^{-4} and a batch size of 2. In addition, a learning rate decay coefficient of 0.96 was used throughout the whole training process.

For data augmentation, we employed some fundamental data augmentation techniques, such as random flipping, random scaling, and augmentation through HSV color space. Specifically, each input image was rescaled to between 25% and 200% of its original resolution, then horizontally flipped with a probability of 50%. Finally, its hue, saturation, and value were randomly adjusted to between 50% and 150% of the original value. All these processes were executed automatically and randomly.

Training was conducted on one NVIDIA GeForce RTX 3080 GPU.

4.2. Ablation Studies

We hypothesized that the potential imbalance of positive and negative samples in the segmentation task would affect the training effect of the network, so the joint loss function of Dice loss, which measures the similarity of segmentation results, and focal Loss, which boosts the weights of small samples, was employed in training to replace the cross-entropy loss function. In this section, experiments were conducted to compare the impact of different loss functions on the training of the network, which used only the cross-entropy loss function or the joint loss function. As shown in Table 2, the semantic segmentation performance of the network trained with the joint loss function is better. This indicates that considering the positive and negative sample imbalance problem is more beneficial to training semantic segmentation networks than simply measuring the pixel-level classification accuracy in the water surface area segmentation task.

Table 2. The impact of different loss functions on the training of the network. Best results are in bold.

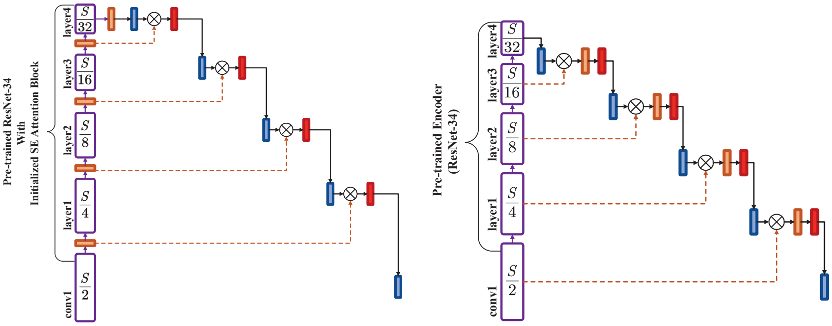
Loss Function	Dice	MIoU (%)	MPA (%)	PA (%)
Cross-entropy loss	0.9748	96.65	98.27	98.37
Joint loss	**0.9763**	**97.05**	**98.49**	**98.56**

The effect of introducing attention mechanisms at different nodes on the network's performance is shown in Table 3. The two networks, RS-UNet-1 and RS-UNet-2, are depicted in Figure 5 as Figure 5a,b, respectively. RS-UNet-1 adds the attention mechanism to the pretrained backbone used for feature extraction, while RS-UNet-2 adds the attention mechanism to the decoder (ours). For training, the same hyperparameters are employed. According to the experimental results, integrating the attention mechanism into the pretrained decoder is preferable to integrating it into the backbone in an interpolated manner.

Owing to this phenomenon, we found that because the backbone was pre-trained on ImageNet, while the added SE attention module was not pre-trained but only initialized, this kind of interleaved combination of pre-trained module and non-pre-trained module destroyed the consistency of weight in the encoder. When training the network, the same learning rate cannot have a uniform effect on both parts, making the network converge to a local optimum instead of a global optimum, which results in a suboptimal final output. During training, the convergence speed of RS-UNet-1 also lags behind that of RS-UNet-2, which is considered to have the same cause.

Table 3. Effect of adding the attention mechanism to different locations on network performance. The network architectures of RS-UNet-1 and RS-UNet-2 are shown in Figure 5. Best results are in bold.

Architecture	Dice	MIoU (%)	MPA (%)	PA (%)
RS-UNet-1	0.9735	96.66	98.26	98.37
RS-UNet-2	**0.9763**	**97.05**	**98.49**	**98.56**



(**a**) RS-UNet-1 (**b**) RS-UNet-2

Figure 5. RS-UNet-1 and RS-UNet-2 network architectures. RS-UNet-1 adds the attention mechanism in the encoder (integrated into the backbone), and RS-UNet-2 adds the attention mechanism in the decoder. All symbols are the same as in Figure 2.

4.3. Experimental Results and Analysis of Water Surface Segmentation

The comparison between our network and some mainstream convolution-based semantic segmentation networks for water surface segmentation performance and parameters is shown in Table 4. All the networks compared here were retrained on our dataset with its best hyperparameters. Note that all the networks were trained on one NVIDIA GeForce RTX 3080 GPU and tested on one NVIDIA GeForce RTX 3050Ti GPU; the spatial resolution of the images used in the test of processing speed was uniform, at 640 × 320. The improvement over [28,32,33] validates the effectiveness of introducing the attention mechanism. The improvement over [32–35] favorably validates the effectiveness of integrating contextual information at all scales using a skip connection. In summary, the combination of the attention mechanism and skip connection help the network successfully overcome the influence of inherent properties of the water surface, such as reflection and irregular boundary shapes. We can see that the processing speed of our network is the fastest due to the concise network architecture.

Table 4. Comparison of the water surface segmentation performance and parameters of different networks. We compare our network against some mainstream segmentation networks. All the numbers reported here are from our experiment. Best results are in bold.

Network	Params (M)	Dice Coefficient	MIoU (%)	MPA (%)	Pixel Accuracy (%)	FPS
U-Net [28]	31.0	0.8791	80.16	88.24	89.60	9.5
PSPNet [32]	65.6	0.9052	88.83	93.82	94.36	20
DeepLab v3+ [33]	59.3	0.9449	92.22	95.90	94.36	35
DANet [34]	66.6	0.9705	95.61	97.70	97.93	34
CondNet [35]	44.1	0.9664	95.36	97.67	97.80	37
RS-UNet(ours)	23.7	**0.9763**	**97.05**	**98.49**	**98.56**	**40**

In addition to the quantitative comparison, some qualitative results are also presented in Figure 6 to visually show the differences between these networks after being trained on the same dataset. The first column is the original input image, the associated ground truth is displayed in the second column, and the following columns are the segmentation results corresponding to different methods. The result of our network is the closest to the ground truth. Other networks have obvious incorrect segmentation problems due to the inherent properties of the water surface.

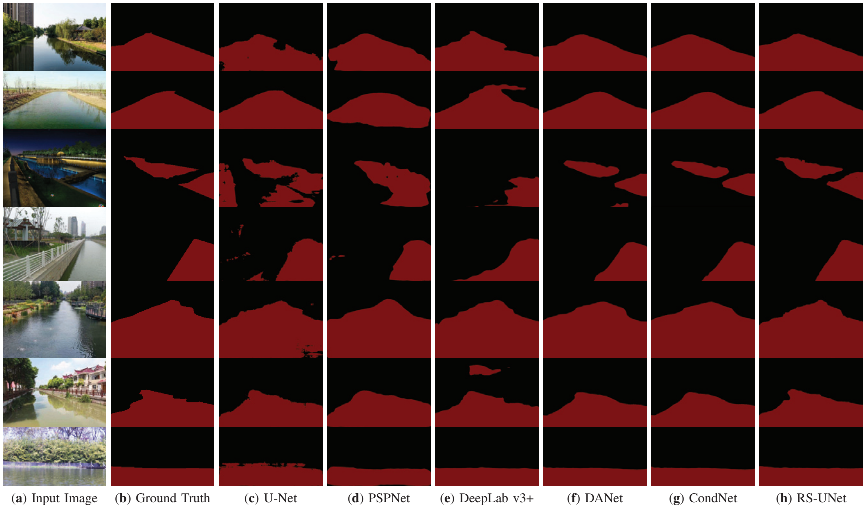


Figure 6. Some qualitative results of water surface segmentation of different networks. The first column is the input image, the second column is the corresponding ground truth, and the other columns are the segmentation results of the water surface area obtained by different methods. This figure visually shows the performance differences between different networks after training on the same dataset. Our results match the ground truth best.

4.4. Comparison with Other Shoreline Detection Methods

In addition to the comparison of water surface area segmentation results with those of mainstream segmentation methods, we also compared our shoreline method with other shoreline detection methods in the professional field. Due to the lack of open-source resources, here, our method was only compared with that of Yin et al. [17], one of the newest and best shoreline detection methods, on the USVInland dataset [27]. We trained our network on this dataset with the same hyperparameters as in Section 4.1 and followed their dataset division strategies. The results reported in [17] were directly used here. For fairness, this experiment was conducted on one RTX2080Ti. Because in these two methods, the segmentation performance is directly related to the shoreline detection performance, Table 5 only shows the comparison of these two methods in water surface segmentation.

Table 5. Comparison with other shoreline detection methods in segmentation performance and inferring speed. The data reported in the first row are from [17]. Best results are in bold.

Method	MIoU (%)	PA (%)	FPS
Yin et al. [17]	96.87	98.49	-
Ours	**97.21**	**98.60**	32

It can be seen that when compared with the other methods in the professional field, our method still shows a more favorable result. The inferring speed of our method is very attractive, which satisfies the need for real-time shoreline detection. Furthermore, comparison of the improvements in MIoU and PA also demonstrates the effectiveness of the use of joint loss during training.

4.5. Experiment on Generalization Capability

As discussed in Section 4.3, the inherent properties of the water surface, such as reflection, and irregular boundary shapes, have a significant impact on the detection result. However, these common features are easily influenced by the environment; for example, the landscape on the shore affects the reflection. This causes the same feature to behave differently in different environments, which challenges the generalization capability of the network. To verify the generalization ability of our method in different scenes, we tested it on the USVInland dataset [27] and the port scenes collected by our team without any fine tuning. The segmentation performance of RS-UNet on these two datasets is shown in Table 6. Obviously, our network generalizes well on the two datasets.

As for the shoreline detection performance, some qualitative results on the USVInland and port datasets are shown in Figures 7 and 8, respectively. The green line indicates the shoreline detected by our method, the red line is the artificially delineated reference shoreline, and the yellow line indicates the overlap of the two. It can be seen that, although there are reflection interference problems in the USVInland dataset [27] and the port scene is not included in the training data, the shoreline detected by our method matches the reference shoreline closely in both environments. This excellent result demonstrates that our shoreline detection method can generalize well in various environments.

Table 6. The performance of RS-UNet on other datasets. The network was not re-trained on these datasets, just use the weights trained in Section 4.3.

Dateset	Dice	MIoU (%)	MPA (%)	PA (%)
USVInland [27]	0.9763	95.60	97.73	97.75
Port	0.9830	97.26	98.68	98.76

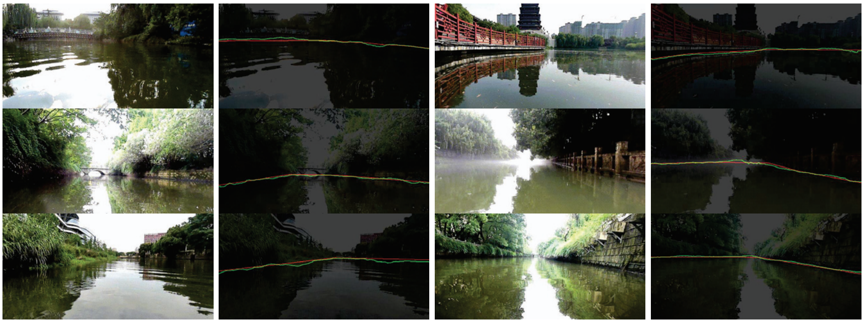

(**a**) Original Image (**b**) Detected Shoreline (**c**) Original Image (**d**) Detected Shoreline

Figure 7. The results of our method on the USVInland dataset for shoreline detection. The green line is the detected shoreline, the red line is the reference shoreline, and the yellow portions represent the overlap between the two.


(**a**) Original Image (**b**) Detected Shoreline (**c**) Original Image (**d**) Detected Shoreline

Figure 8. The results of our method on the port dataset for shoreline detection.The green line is the detected shoreline, the red line is the reference shoreline, and the yellow portion represents the overlap between the two.

5. Conclusions

In this paper, we discussed the current state of research in the field of shoreline detection and analyzed the reasons for the shortcomings of these methods in accurately detecting shorelines in real time. Accordingly, we constructed a more accurate and real-time shoreline detection method based on the proposed RS-UNet network. Our method defines shoreline detection as the combination of water surface area segmentation and edge detection. As the key of our method, we proposed RS-UNet as the water segmentation network, which can segment the water area accurately and in real time. Experiments show that our RS-UNet achieves a 97.05% MIoU and 40 FPS processing speed in the task of segmenting the water surface area, which is better than some existing mainstream semantic segmentation methods and deep-learning-based shoreline detection methods. We also demonstrated the generalization capability of our method through experiments. In summary, our shoreline detection method can accurately detect shorelines in real time and in various environments.

Although our method performs well in shoreline detection, it is still to some limitations. The main limitation is the insufficient amount of training data with annotations, which limits the generalization ability of our method. Our future work will be focused in two directions. The first direction is exploring training our network in an unsupervised manner. By employing an unsupervised training process, we can use a large number of images without annotations for training and significantly reduce the impact of a lack of training data. Another direction is to treat this work as a foundation, integrating the proposed method with other downstream tasks. such as obstacle detection on the water surface, automatic visual positioning, and the monitoring of distance between USVs and the shore, in order to provide guarantees for the safety of navigation of USVs.

Author Contributions: Conceptualization, J.Z. and F.S.; Data curation, F.S.; Funding acquisition, J.Z. and S.W.; Investigation, G.G. and F.S.; Methodology, J.Z. and F.S.; Project administration, J.Z. and S.W.; Resources, J.Z. and G.G.; Software, F.S.; Supervision, S.W.; Validation, F.S.; Visualization, F.S.; Writing—original draft, F.S.; Writing—review and editing, J.Z., S.W. and F.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Key Research and Development Program, China (Grant no. 2021YFC2801004) and the National Natural Science Foundation of China (Grant nos. 52071199 and 52102397).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Access to the data will be considered upon request by the authors.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

As there is a large number of acronyms and terms in our manuscript, we provide an abbreviations list here for a better reading experience.

Abbreviation	**Full Name**
MIoU	Mean intersection over union
FPS	Frames per second
USV	Unmanned surface vehicle
RANSAC	Random sample consensus
HSV	Hue saturation value
SE	Squeeze and excitation
RS-UNet	Residual squeeze-and-excitation U-Net
CRF	Conditional random fields
FCNs	Fully convolutional Networks

References

1. Yang, P.; Song, C.; Chen, L.; Cui, W. Image Based River Navigation System of Catamaran USV with Image Semantic Segmentation. In Proceedings of the 2022 WRC Symposium on Advanced Robotics and Automation (WRC SARA), Beijing, China, 20 August 2022; pp. 147–151.
2. Sinisterra, A.J.; Dhanak, M.R.; Von Ellenrieder, K. Stereovision-based target tracking system for USV operations. *Ocean Eng.* **2017**, *133*, 197–214. [CrossRef]
3. Ji, X.; Zhuang, J.Y.; Su, Y.M. Marine radar target detection for USV. In Proceedings of the Advanced Materials Research, Lille, France, 26–30 May 2014; Trans Tech Publications: Bach, Switzerland, 2014; Volume 1006, pp. 863–869.
4. Wang, Z.; Zhang, Y. Estimation of ship berthing parameters based on Multi-LiDAR and MMW radar data fusion. *Ocean Eng.* **2022**, *266*, 113155. [CrossRef]
5. Wei, Y.; Zhang, Y. Effective waterline detection of unmanned surface vehicles based on optical images. *Sensors* **2016**, *16*, 1590. [CrossRef] [PubMed]
6. Kröhnert, M. Automatic waterline extraction from smartphone images. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *41*, 857. [CrossRef]
7. Zhan, W.; Xiao, C.; Yuan, H.; Wen, Y. Effective waterline detection for unmanned surface vehicles in inland water. In Proceedings of the 2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA), Montreal, QC, Canada, 28 November–1 December 2017; pp. 1–6.
8. Feng, T.; Xiong, J.; Xiao, J.; Liu, J.; He, Y. Real-time riverbank line detection for USV system. In Proceedings of the 2019 IEEE International Conference on Mechatronics and Automation (ICMA), Tianjin, China, 4–7 August 2019; pp. 2546–2551.
9. Liu, R.W.; Yuan, W.; Chen, X.; Lu, Y. An enhanced CNN-enabled learning method for promoting ship detection in maritime surveillance system. *Ocean Eng.* **2021**, *235*, 109435. [CrossRef]
10. Chen, P.; Li, Y.; Zhou, H.; Liu, B.; Liu, P. Detection of Small Ship Objects Using Anchor Boxes Cluster and Feature Pyramid Network Model for SAR Imagery. *J. Mar. Sci. Eng.* **2020**, *8*, 112. [CrossRef]
11. Liang, M.; Liu, R.W.; Li, S.; Xiao, Z.; Liu, X.; Lu, F. An unsupervised learning method with convolutional auto-encoder for vessel trajectory similarity computation. *Ocean Eng.* **2021**, *225*, 108803. [CrossRef]
12. Feng, H.; Cao, G.; Xu, H.; Ge, S.S. IS-STGCNN: An Improved Social spatial-temporal graph convolutional neural network for ship trajectory prediction. *Ocean Eng.* **2022**, *266*, 112960. [CrossRef]
13. Chen, X.; Liu, S.; Liu, R.W.; Wu, H.; Han, B.; Zhao, J. Quantifying Arctic oil spilling event risk by integrating an analytic network process and a fuzzy comprehensive evaluation model. *Ocean Coast. Manag.* **2022**, *228*, 106326. [CrossRef]
14. Xue, H.; Chen, X.; Zhang, R.; Wu, P.; Li, X.; Liu, Y. Deep Learning-Based Maritime Environment Segmentation for Unmanned Surface Vehicles Using Superpixel Algorithms. *J. Mar. Sci. Eng.* **2021**, *9*, 1329. [CrossRef]
15. Steccanella, L.; Bloisi, D.; Blum, J.; Farinelli, A. Deep learning waterline detection for low-cost autonomous boats. In Proceedings of the International Conference on Intelligent Autonomous Systems, Singapore, 1–3 March 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 613–625.
16. Steccanella, L.; Bloisi, D.D.; Castellini, A.; Farinelli, A. Waterline and obstacle detection in images from low-cost autonomous boats for environmental monitoring. *Robot. Auton. Syst.* **2020**, *124*, 103346. [CrossRef]
17. Yin, Y.; Guo, Y.; Deng, L.; Chai, B. Improved PSPNet-based water shoreline detection in complex inland river scenarios. *Complex Intell. Syst.* **2022**, *9*, 233–245. [CrossRef]
18. Shen, J.; Tao, Q.; Xiao, Z. Shoreline detection algorithm based on the improved Deeplab v3+ network. *J. Image Graph.* **2019**, *23*, 2174–2182.
19. Adam, M.A.M.; Ibrahim, A.I.; Abidin, Z.Z.; Zaki, H.F.M. Deep Learning-Based Water Segmentation for Autonomous Surface Vessel. In Proceedings of the IOP Conference Series: Earth and Environmental Science, Kuala Lumpur, Malaysia, 20–21 October 2020; IOP Publishing: Bristol, UK, 2020; Volume 540, p. 012055.

20. Vandaele, R.; Dance, S.L.; Ojha, V. Automated water segmentation and river level detection on camera images using transfer learning. In Proceedings of the DAGM German Conference on Pattern Recognition, Tubingen, Germany, 28 September–1 October 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 232–245.
21. Zhan, W.; Xiao, C.; Wen, Y.; Zhou, C.; Yuan, H.; Xiu, S.; Zou, X.; Xie, C.; Li, Q. Adaptive semantic segmentation for unmanned surface vehicle navigation. *Electronics* **2020**, *9*, 213. [CrossRef]
22. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27 June–1 July 2016; pp. 770–778.
23. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141.
24. Kristan, M.; Sulić Kenk, V.; Kovačič, S.; Perš, J. Fast Image-Based Obstacle Detection From Unmanned Surface Vehicles. *IEEE Trans. Cybern.* **2016**, *46*, 641–654. [CrossRef] [PubMed]
25. Peng, M.; WANG, J.; WEN, X.; Cong, X. Shoreline detection method by combining HSV spatial water image feature. *J. Image Graph.* **2018**, *23*, 526–533.
26. Yao, L.; Kanoulas, D.; Ji, Z.; Liu, Y. ShorelineNet: An Efficient Deep Learning Approach for Shoreline Semantic Segmentation for Unmanned Surface Vehicles. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 5403–5409.
27. Cheng, Y.; Jiang, M.; Zhu, J.; Liu, Y. Are we ready for unmanned surface vehicles in inland waterways? The usvinland multisensor dataset and benchmark. *IEEE Robot. Autom. Lett.* **2021**, *6*, 3964–3970. [CrossRef]
28. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 234–241.
29. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
30. Wang, X. Laplacian operator-based edge detectors. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 886–890. [CrossRef] [PubMed]
31. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
32. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
33. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 801–818.
34. Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H. Dual attention network for scene segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 3146–3154.
35. Yu, C.; Shao, Y.; Gao, C.; Sang, N. CondNet: Conditional classifier for scene segmentation. *IEEE Signal Process. Lett.* **2021**, *28*, 758–762. [CrossRef]

Journal of
Marine Science and Engineering

MDPI

Article

Path Planning of an Unmanned Surface Vessel Based on the Improved A-Star and Dynamic Window Method

Shunan Hu [1], Shenpeng Tian [2], Jiansen Zhao [2,*] and Ruiqi Shen [2]

[1] School of Automotive Engineering, Changshu Institute of Technology, Suzhou 215506, China
[2] Merchant Marine College, Shanghai Maritime University, Shanghai 201306, China
* Correspondence: jszhao@shmtu.edu.cn

Abstract: In order to ensure the safe navigation of USVs (unmanned surface vessels) and real-time collision avoidance, this study conducts global and local path planning for USVs in a variable dynamic environment, while local path planning is proposed under the consideration of USV motion characteristics and COLREGs (International Convention on Regulations for Collision Avoidance at Sea) requirements. First, the basis of collision avoidance decisions based on the dynamic window method is introduced. Second, the knowledge of local collision avoidance theory is used to study the local path planning of USV, and finally, simulation experiments are carried out in different situations and environments containing unknown obstacles. The local path planning experiments with unknown obstacles can prove that the local path planning algorithm proposed in this study has good results and can ensure that the USV makes collision avoidance decisions based on COLREGs when it meets with a ship.

Keywords: local path planning; dynamic window method; USV; path planning

1. Introduction

In recent years, due to global warming, increasing environmental pollution, and the scarcity of land resources, countries around the world have paid more attention to the development of marine resources [1], which has led to the rapid development of intelligent maritime technologies such as unmanned surface vehicles (USVs) and underwater robots. As an intelligent robot, ASV can perform tasks independently without human intervention [2,3]. It is a surface operating platform with the advantages of a small size and stealthiness. Therefore, it is mainly used to perform hazardous missions [4]. USVs play an important role in civil and military fields such as environmental monitoring, hydrographic exploration, offshore patrol, maritime search and rescue, and long-range reconnaissance [5,6]. During the navigation of USVs, they are affected not only by the known static environment (shore base, shoals, islands, etc.) but also by unknown dynamic vessels and unknown static obstacles (pontoons, offshore operating platforms, etc.) [7]. In order to ensure safe navigation, USVs will first perform global path planning based on known static obstacles in the navigation environment during operation, then perform real-time detection of the dynamic ocean environment [8], and finally implement local dynamic path planning in combination with COLREGs. During the process of the dynamic path planning, when the vessels are in sight of one another, only by considering the rules of collision avoidance can the USV in sailing be guided accurately [9]. At present, there are also many studies focusing on ship collision avoidance algorithms; Zhang et al. [10] proposed a distributed intelligent anti-collision decision support formulation and designed a linear extension algorithm for both course alteration and speed reduction to keep clearance of all the target ships, which proves that anti-collision formulation can avoid collision when all ships have complied with COLREGs as well as when some of them do not take actions. Cai et al. [11] proposed a Collision Risk Index of Ferry (CRIF) based on the behavior of ferries

Citation: Hu, S.; Tian, S.; Zhao, J.; Shen, R. Path Planning of an Unmanned Surface Vessel Based on the Improved A-Star and Dynamic Window Method. *J. Mar. Sci. Eng.* **2023**, *11*, 1060. https://doi.org/10.3390/jmse11051060

Academic Editor: Rafael Morales

Received: 16 April 2023
Revised: 11 May 2023
Accepted: 12 May 2023
Published: 16 May 2023

and multiple target ships, which makes it possible to assess real-time collision risk during crossing and to integrate the collision risk of each voyage based on historical encounter scenarios. Xu et al. [12] proposed a modified path-following control system using the vector field method for USV. Unlike other methods, they considered the coupled path following and collision avoidance task as a whole. Finally, their proposed method allows the USV to follow a predetermined path while automatically avoiding obstacles. Zhang et al. [13] proposed a modified path planning method called RRTes, which simultaneously considers the ship draft and UKC to avoid grounding as well as ship maneuverability constraints and COLREGs to identify ship dynamic collision risk. Therefore, in addition to global path planning, USVs need to have the ability to plan paths in a dynamic environment. In the current research on USVs [14], autonomous collision avoidance technology is a key technology that has not been surpassed in the development of unmanned vessels and is a prerequisite for USVs to navigate safely in dynamic environments.

Based on COLREGs, Almoaili et al. [15] propose a near-optimal algorithm to find a feasible path for UGVs (Unmanned Ground Vehicle) in a static environment, and the performance of the proposed algorithm was compared with other well-established algorithms in the path planning literature such as A* using a simulator developed for this purpose. Wang et al. [16] designed three-dimensional path planning and security event-triggered cooperative path tracking for a multi-disc autonomous underwater glider (AUG) considering the presence of underwater obstacles and denial-of-service (DoS) attacks. In the study by Jabbarpour et al. [17], a new ant-based path planning approach that considers UGV energy consumption in the IPTS planning strategy is proposed. This method is called Green Ant (G-Ant) and integrates an ant-based algorithm with a power/energy consumption prediction model to reach its main goal, which is providing a collision-free shortest path with low power consumption. Han et al. [18] propose a path planning method based on the multi-strategy evolutionary learning artificial bee colony algorithm. Thoresen et al. [19] propose a new method of path planning for UGVs on terrain, and the proposed path planning method is based on the Hybrid A* algorithm and uses estimated terrain traversability to find the path that optimizes both traversability and distance for the UGV. Shin et al. [20] proposes a model predictive path planning algorithm by employing a passivity-based model predictive control (MPC) optimization setup. Chen et al. [21] introduce a new meta-heuristic path planning algorithm, the Cuckoo-Beetle Swarm Search (CBSS) algorithm, to solve the path planning problem for heterogeneous mobile robots. Through domestic and international research, it has been found that there are few studies in global path planning that can ensure path safety while also considering path economy. In local path planning, unknown static obstacles are rarely considered to be introduced into the environment. Therefore, the improved A * algorithm is used in global path planning to generate routes that are both safe and cost-effective. Unknown static obstacle evaluation factors are introduced into the evaluation function of the dynamic window method (DWA) in local path planning. This improves the sensitivity of the algorithm in local path planning. This article integrates global and local path planning, which can provide certain feasible suggestions for route planning before unmanned vessel navigation and collision avoidance during navigation.

2. Global Path Planning of USV Based on an Electronic Navigation Chart

In order to realize the autonomous navigation of the USV under mutual visibility conditions, the navigation environment is established based on a 5 m depth contour according to web charts. It is based on the surface data in the ECDIS data, and rendering is performed using tools such as Mapbox to achieve the display of basic surfaces such as land, shoals, and bathymetric surfaces. Java's SpringBoot framework is used as the main body of the program, the ORM (Object Relational Mapping) framework of the MySQL database is Mybatis, the Druid is used as the Connection pool of the MySQL database, and the JedisPool is used as the Connection pool of the Redis database. A front and rear nonseparation strategy is adopted. The security optimization objectives for global path planning are determined, and a global path planning model is established. The flowchart of our work is shown in Figure 1. This work includes global path planning and local path planning. The global path planning is the generation of a path based on known static obstacles, including chart processing, path planning, path smoothing, etc. Local path planning is the dynamic adjustment made by ships when navigating along the globally planned path. During navigation, there may be static and dynamic obstacles that are not displayed on the chart, and safe collision avoidance actions need to be taken to avoid collisions. After avoiding collisions, it is necessary to return to the global planning path.

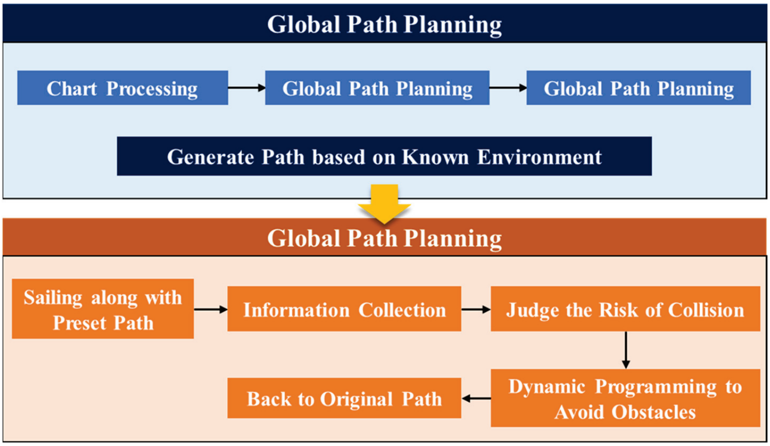


Figure 1. The flowchart of global and local path planning.

2.1. Chart Data Processing

First of all, the surface data rendered in different colors in the nautical chart are processed by pixel threshold segmentation. Among them, the processing that satisfies the water depth constraints is the navigable area, and the others are the obstacle area. Using electronic charts displayed on the web as the initial experimental environment, a global path planning experimental model is constructed for a portion of the chart areas with a latitude range of 30°9.627′ N–30°21.306′ N and a longitude range of 121°52.356′ E–122°8.543′ E. The values of different chart information (R, G, B) are shown in Table 1.

Table 1. Comparison table of rendering colors for different environmental information of the chart.

Chart Environmental Information	Color	Red	Green	Blue
Areas with a water depth of 30–50 m	🟩	216	244	225
Areas with a water depth of 10–20 m	🟦	185	225	222
Areas with a water depth of 5 m	🟦	152	211	237
Areas with a water depth of 1–2 m	🟦	123	193	241
shoal	🟩	115	185	142
land	🟨	203	199	131

Second, the point and line data that hinder navigation in the screening experiment area are drawn in the experiment environment model. Next, after constructing the nautical chart data into a rasterized environment model, the generated path is the grid position point in the Cartesian coordinate system. In order to be able to guide the navigation of the

USV in practice, it is necessary to convert the grid nodes and the geographic coordinate system (longitude, latitude). The coordinate conversion calculation principle is shown in Figure 2. Figure 2 shows a chart area where the maximum and minimum latitudes are $Lat_{\max}$ and $Lat_{\min}$ and the maximum and minimum longitudes are $Lon_{\max}$ and $Lon_{\min}$. The corresponding number of horizontal grids is m, and the number of vertical grids is n. The conversion formula between latitude and longitude and grid coordinates is as follows:

$$\begin{cases} Lat = \frac{y(Lat_{\max} - Lat_{\min})}{n} + Lat_{\min} \\ Lon = \frac{x(Lon_{\max} - Lon_{\min})}{m} + Lon_{\min} \end{cases} \tag{1}$$

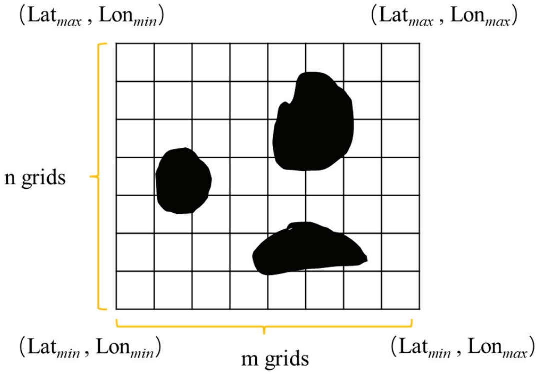


Figure 2. Coordinate transformation diagram.

Among them, Lat and Lon represent the position point coordinates of the unmanned ship during navigation, and x and y represent the horizontal and vertical coordinates in the grid map, respectively.

2.2. Improved A-Star Algorithm

As a commonly used heuristic path search algorithm, the A-star algorithm is widely used in the field of global path planning [22]. Its search methods include four-connected area search and eight-connected area search. Considering the movement characteristics of the USV in the ocean [23], this paper uses the search method of eight-connected areas to search navigable waters. The heuristic function of the improved A-star algorithm is defined as follows:

$$f(n) = g(n) + h(n) + D(n) \tag{2}$$

where $g(n)$ is the actual distance cost from the starting point to the current node, $h(n)$ is the estimated distance cost from the current node to the target node [24], and $D(n)$ is the hazard function that uses the Voronoi field to add hazards to the grid. The A-star algorithm only uses the path length as a heuristic function, and the planned path often approaches obstacles, which cannot effectively guide USV to navigate safely and smoothly. This article improves the evaluation function of the A-star algorithm by using the concept of establishing navigation boundaries based on the Voronoi field algorithm, the definition of which is as follows:

$$\rho_V(x,y) = \left(\frac{\alpha}{\alpha + d_o(x,y)}\right)\left(\frac{d_v(x,y)}{d_o(x,y) + d_v(x,y)}\right) * \left[\frac{(d_o - d_o^{\max})^2}{(d_o^{\max})^2}\right] \tag{3}$$

where $d_o(x,y)$ is the distance from the current node to the obstacle, $d_v(x,y)$ is the distance from the current node to the Voronoi boundary, $\alpha > 0$ is a constant that controls the decay rate of the potential field, and $d_o^{\max} > 0$ is a constant controlling the maximum operating range of the potential field.

In the process of unmanned ship navigation, not only should the constraint of water depth be considered, but a certain safe distance should also be maintained between the generated path and obstacles. The evaluation functions of each grid after adding hazard levels can be defined as:

$$f(n) = \begin{cases} g(n) + h(n), & \rho_v = 0 \text{ or } d_v(x,y) \geq D \\ \infty, & \rho_v > 0 \text{ or } d_v(x,y) < D \end{cases} \tag{4}$$

ρ_v is the Voronoi field value at the current node, and D is the safety constraint distance.

3. Local Path Planning of USV

3.1. Collision Avoidance Decision Based on the Dynamic Window Method

First, we establish the motion model of the USV (linear velocity, angular velocity, etc.). Second, a set of navigable speeds including linear and angular velocities $\{(v_1, w_1), \ldots, (v_i, w_i)\}$ is established under the constraints of the motion model and the distribution of obstacles in the motion environment [25]. Finally, the speed vectors under the navigable speed set are sampled, and the predicted trajectory of each speed vector (v, w) in the next time period is solved according to the sampled speed vectors [26]. Finally, the evaluation function is used to optimize the trajectory, and the speed combination corresponding to the optimal trajectory is used as the navigation speed in the next period of time [27].

The movement state of the USV during navigation is shown in Figure 3. Assume that the current given linear velocity and angular velocity vector is (v_0, w_0), the current two-dimensional position coordinate is (x_0, y_0), and the included angle with the abscissa is φ_0 [28]. The motion model of the USV is shown in Formula (5):

$$\begin{cases} x = x_0 + v_0 \Delta t \cos\varphi \\ y = y_0 + v_0 \Delta t \sin\varphi \\ \varphi = \varphi_0 + w_0 \Delta t \end{cases} \tag{5}$$

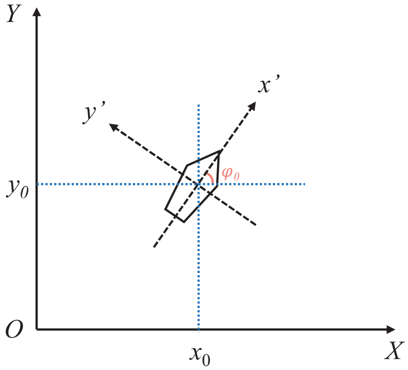


Figure 3. Kinematics Model of USV.

After the USV model is established, the trajectory of the USV in the next sampling period can be predicted according to the motion model. The speed space available for sampling in actual voyages is limited, which is determined by the following three aspects:

(1) Velocity constraints

Suppose the maximum linear velocity of the USV is $v_{\max}$, the minimum linear velocity is $v_{\min}$, the maximum angular velocity is $w_{\max}$, and the minimum angular velocity is $w_{\min}$ [29]. Then, the set of velocities V_m between the maximum value and the minimum value is:

$$V_m = \{(v, w) | v \in [v_{\min}, v_{\max}], w \in [w_{\min}, w_{\max}]\} \tag{6}$$

(2) Performance constraints of USV

Assuming that the linear velocity and angular velocity vector of a given USV at a certain moment are (v_0, w_0), and its maximum linear acceleration and maximum angular acceleration are α and w [30], then the speed set V_d constrained by the power device is:

$$V_d = \{(v, w) | v \in [v_0 - a\Delta t, v_0 + a\Delta t], w \in [w_0 - \dot{w}\Delta t, w_0 + \dot{w}\Delta t]\} \tag{7}$$

(3) Stopping distance constraint

Assuming that $dist(v, w)$ is the shortest distance between the end of the USV-predicted trajectory and the obstacle, v' and w' are the maximum stopping linear acceleration and maximum stopping angular acceleration, respectively [31]; then, the speed set V_d constrained by the stopping distance is:

$$V_a = \left\{(v, w) \middle| v \leq \sqrt{2 \cdot dist(v, w) \cdot v'}, w \leq \sqrt{2 \cdot dist(v, w) \cdot w'}\right\} \tag{8}$$

In summary, if the USV is to sail safely to the destination, the speed space needs to meet the above three constraints. Assuming that all navigable speed sets of USV in the space are V_r, then V_r is the intersection of the speed sets under the above three constraints.

3.2. Acquisition of Optimal Trajectory

The acquisition of the optimal trajectory is divided into two parts: trajectory preprocessing and trajectory evaluation, and the operation is as follows:

(1) Trajectory preprocessing

Dynamic ship risk detection is the process of screening and predicting trajectories, as well as the operation of determining the timing of collision avoidance. Its principle is shown in Figure 4. The distance between the sampling point of the predicted trajectory and the corresponding position point of the obstructed ship during the period is calculated, and the CPA is adopted. When the distance encountered recently is less than the set safety distance, it indicates a risk of collision. The length of the USV is set to 3 to 11 m, and the draft is less than 2 m. The size of the target ship is set to twice that of the USV, and the draft is 3 m.

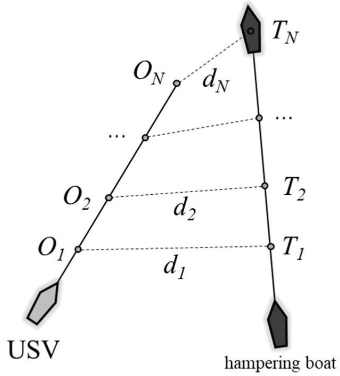


Figure 4. Dynamic ship risk detection.

When ships perform collision avoidance operations in head-on situations and crossing situations, according to the requirements of the COLREGs, the give-way ship should turn to starboard to avoid stand-on ships. Therefore, the range of angular velocity corresponding to the predicted trajectory will be limited to $\omega \in [\omega_{\min}, \omega_0]$. This operation needs to be maintained during the avoidance process until no obstructive ship is detected in the predicted trajectory; then, the process ends.

(2) Trajectory evaluation function

On the basis of the constraints of the COLREGs, this paper sets three aspects of speed function, heading function, and distance function to construct the trajectory evaluation function to evaluate the navigable predicted trajectory. The optimal trajectory evaluation function is shown in Formula (9):

$$G(v,\omega) = \alpha \cdot f_v(v,\omega) + \beta \cdot f_h(v,\omega) + \gamma \cdot f_d(v,\omega) + \lambda \cdot f_{du}(v,w) \tag{9}$$

where $G(v,\omega)$ is the trajectory evaluation function, and its maximum value is used to find the optimal solution of the predicted trajectory; the other functions are sub-functions of the predicted trajectory function, and α, β, γ, λ are the weight coefficients of each sub-function. The fastness function $f_v(v,\omega)$ is used to evaluate the linear velocity size of the current trajectory, the reachability function $f_h(v,\omega)$ is used to evaluate the deviation degree of the predicted trajectory end of the USV from the target point, and the safety functions of known and unknown obstacles $f_d(v,\omega)$ and $f_{du}(v,w)$ are used to evaluate the safety of predicted trajectories.

The trajectory evaluation function needs to be normalized before evaluation, as shown in Formula (10):

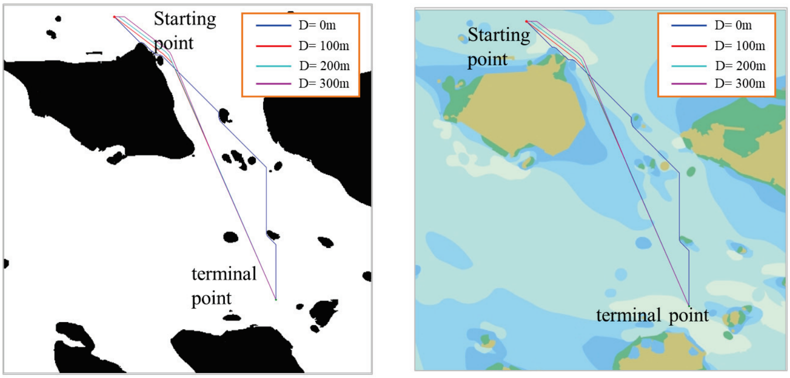
$$\begin{cases} \overline{f}_v(i) = \frac{f_v(i)}{\sum_{i=1}^{n} f_v(i)} \\ \overline{f}_h(i) = \frac{f_h(i)}{\sum_{i=1}^{n} f_h(i)} \\ \overline{f}_d(i) = \frac{f_d(i)}{\sum_{i=1}^{n} f_d(i)} \\ \overline{f}_d(i) = \frac{f_{du}(i)}{\sum_{i=1}^{n} f_{du}(i)} \end{cases} \tag{10}$$

4. Results and Discussion

4.1. Experimental Results of Global Path Planning

In this paper, the selected nautical chart area is rasterized and divided into 500 × 500 grids, and each grid represents 25 m. Aiming at the maneuverability of the USV, safety distance constraints of 100 m, 200 m, and 300 m are set. The results of global path planning based on different safety distance constraints and smooth optimization are shown in Figure 5, Figure 5a shows the display of the planned path in the grid environment, and Figure 5b shows the planning in the redrawn sea chart water depth environment display of the path.

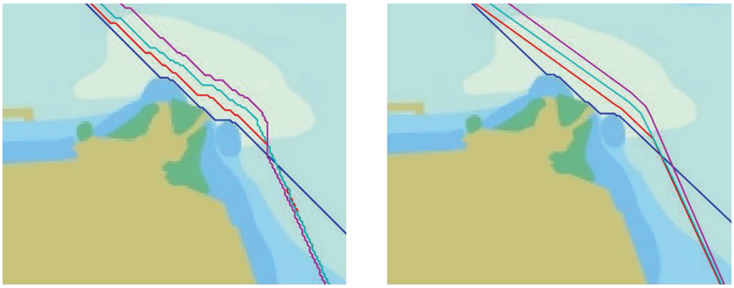
Figure 5 is a comparison of the experimental results of the traditional A-star algorithm and the improved A-star algorithm under different safety distance constraints. The solid blue line is the result of the global path planning of the A-star algorithm without adding a safety distance constraint. It can be seen that the planned path is close to the obstacle, which will undoubtedly increase the collision between the USV and the obstacle risk. Red, cyan, and magenta are the path planning results of maintaining 100 m, 200 m, and 300 m safety distance constraints, respectively. It can be seen that with the increase in distance constraints, the planned path can maintain distance constraints when approaching obstacles, and the path with a larger safety radius is farther away from obstacles. It can be seen from Figure 5b that the planned paths are all outside the water depth of 5 m, and the proposed method can guarantee the constraint of safe water depth.



（a） Path planning grid display （b） Path planning chart display

Figure 5. Global path planning results under different safety distance constraints.

Figure 6 shows the comparison of some planned paths before and after smooth optimization. Figure 6a shows the path before smooth optimization. The number of turns in the path is too many and not smooth, which does not meet the requirements of navigation practice and USV performance; Figure 6b shows the path after smooth optimization. The number of path turns is small, and the smoothness is high, which is of high guiding significance.



（a） path before smoothing （b） path after smoothing

Figure 6. Comparison of path smoothing effects under different safety distance constraints. The meaning of different colored lines is same as Figure 5.

Figure 7 shows the distance between the path node and the obstacle under three different safety distances. From the black horizontal dotted line set in the figure, it can be seen that the path node keeps the set safety distance from the obstacle, which can improve the safety factor of the USV during navigation.

Table 2 shows the comparison of path planning results under different safety distances, and it can be seen that the optimized path has good results in terms of length and number of nodes. In terms of path length, the smoothed path has a slight decrease. On the path nodes, there is a significant decrease in the number of smoothed path nodes. The experimental results demonstrate that the path smoothing algorithm proposed in this paper has good performance.

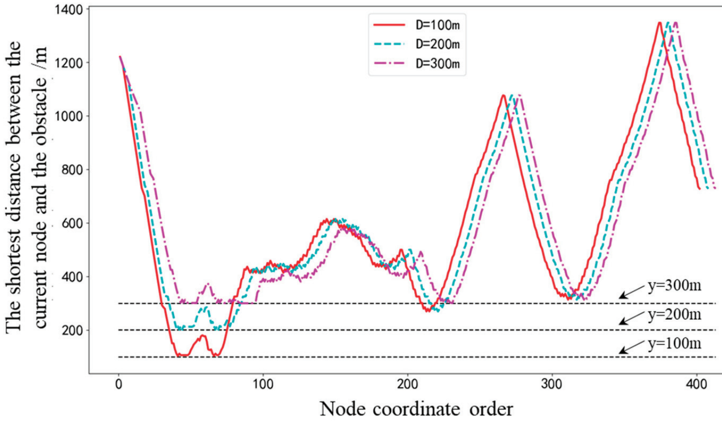


Figure 7. The distance between path nodes and obstacles under different safety distances.

Table 2. Comparison of path planning results under different safety distances.

Safe Distance/m	Original Path Length/km	Optimized Path Length/km	Number of Nodes	Number of Optimized Nodes
D = 0	12.08	-	398	-
D = 100	12.14	11.32	402	5
D = 200	12.23	11.38	409	6
D = 300	12.30	11.48	414	6

In summary, the global path planning method proposed in this article can achieve dual constraints of water depth and obstacle safety distance, which is in line with the actual requirements of USV navigation.

4.2. Experimental Simulation under Different Encounter Situations

In order to verify the effectiveness of the proposed algorithm in the collision avoidance process, the collision avoidance simulation experiment of USV is carried out in the range of 500 m × 500 m under the three scenarios of head-on, crossing, and overtaking between ships. The simulation parameter setting and algorithm setting of USV are shown in Tables 3 and 4.

Table 3. Simulation parameters of USV.

Parameter	Linear Velocity	Angular Velocity
minimum value	0 m/s	$-20^\circ/s$
maximum value	5 m/s	$20^\circ/s$
maximum acceleration	$2\ m/s^2$	$5^\circ/s^2$
sampling interval	0.2 m/s	$1^\circ/s$

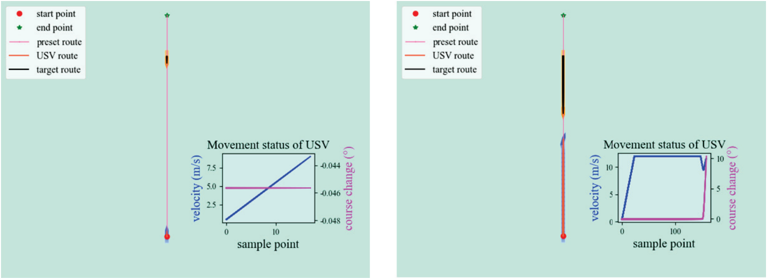
Table 4. Simulation parameters of USV.

α	β	γ	λ	Predicted Time/s	Safe Distance/m
0.02	0.4	0.2	0.2	5	50

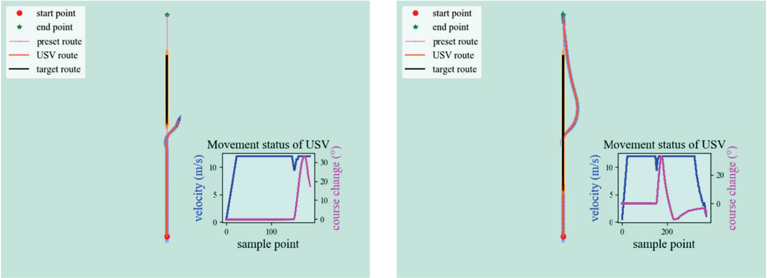
(1) Head-on situation

The collision avoidance simulation process in the head-on situation is shown in Figure 7, where the red circle is the starting point of the USV, the green star is the target point of the USV, the pink line is the preset route, the red line is the trajectory of the USV, and black is the trajectory of the target ship in the situation of encountering the USV.

Figure 8a shows the positional relationship between the USV and the target ship before sailing, which can be judged as a head-on situation according to the COLREGs. Figure 8b starts to make a decision for the USV to detect that there is a risk of collision with the target ship. Figure 8c shows that the USV follows the COLREGs and turns starboard to avoid the target ship. Figure 8d shows returning to the preset route after avoiding the target ship and reaching the target point safely.



(a) USV starts sailing (b) USV begins to avoid

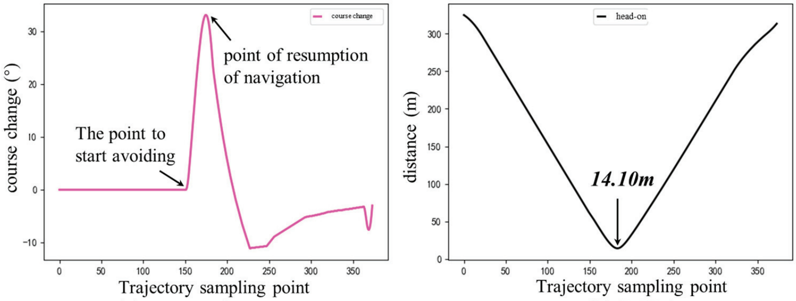


(c) USV successfully avoided (d) USV reaches target point

Figure 8. Simulation results of collision avoidance in a head-on situation.

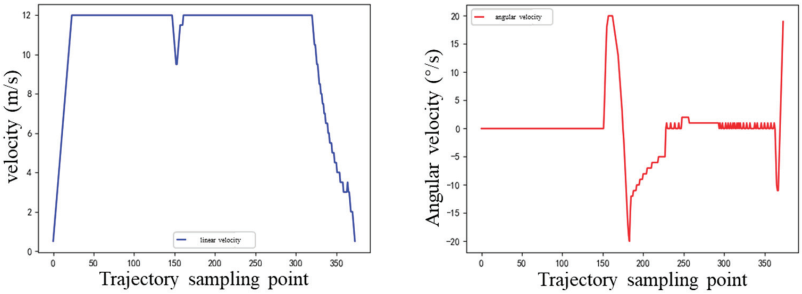
Figure 9 is the heading change curve and the DCPA diagram corresponding to the trajectory sampling points during the entire avoidance process in the USV head-on situation. It can be seen from Figure 9a that near the 150th sampling point, the USV begins to turn and avoid. At this time, the distance between the two ships corresponding to Figure 9b is 50 m, which is in line with the set safety distance. In the process of the encounter, the distance between the USV and the target ship decreases rapidly due to the relatively fast speed of the USV and the target ship. The distance decreases continuously throughout the process, and the nearest encounter distance is 14.10 m. As the process ends, the distance gradually increases.

Figure 10 shows the changing trend of linear velocity and angular velocity in the head-on situation. It can be seen from the linear velocity change diagram in Figure 10a that before the collision avoidance process, the USV first accelerates and then has a constant speed; during collision avoidance, deceleration measures are taken; after collision avoidance, the USV accelerates again until it approaches the target point and begins to decelerate. Figure 10b is the diagram of angular velocity change. Before the collision avoidance process, the USV angular velocity is 0, and the navigation is maintained at the predetermined course; during collision avoidance, the USV turns to starboard. From the analysis, it can be seen that the collision avoidance behavior in the encounter process conforms to the COLREGs and the common maneuvering behavior at sea.



(a) course change curve (b) DCPA

Figure 9. The heading change curve and the DCPA diagram in the head-on situation.



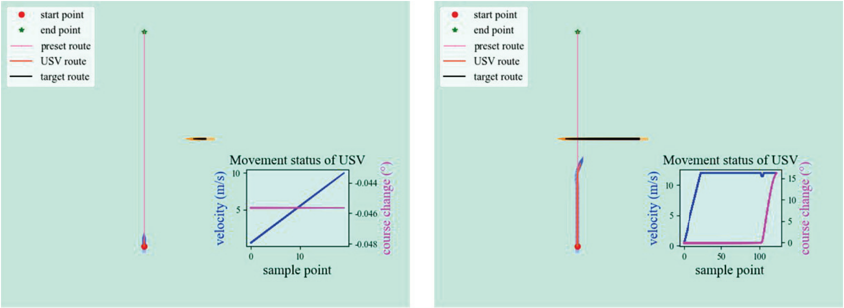
(a) The change trend of linear velocity in head-on situation (b) Variation trend of angular velocity in head-on situation

Figure 10. Trend diagram of speed change in a head-on situation.

(2) Crossing situation

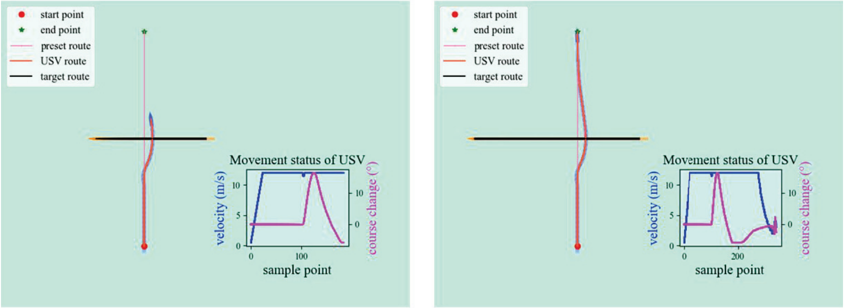
The simulation process of collision avoidance under the crossing situation is shown in Figure 11. Figure 11a shows the positional relationship between the USV and the target ship before sailing, which can be judged as a crossing situation by the COLREGs. Figure 11b starts to make a collision avoidance decision for the USV to detect that there is a risk of collision with the target ship. Figure 11c shows that the USV follows the constraints of the COLREGs and turns to the starboard to successfully avoid the target ship. Figure 11d shows that USV successfully returned to the preset route after safe avoidance and sailed safely to the destination.

Figure 12 shows the heading change diagram and the DCPA diagram corresponding to the trajectory sampling points during the entire avoidance process in the crossing situation. The first section of Figure 12a shows that the USV keeps a fixed heading when no target ship is detected; the second section is to start to turn starboard to avoid collision when the target ship is detected. The collision avoidance behavior complies with the requirements of the COLREGs. According to the COLREGs, the maximum collision avoidance range is set to 17° since the USV is the give-way ship in the crossing situation, and it should take substantial action to keep well clear and avoid crossing ahead the head-on ship under the allowed circumstances case. The range of collision avoidance ensures the safety of the collision avoidance process, which is consistent with the actual situation of navigation. The third section is the resumption process after collision until reaching the target point. Figure 12b shows the distance of the nearest encounter. It can be seen that the distance between the two ships decreases first and then increases. The distance of the nearest encounter is 40.32 m, which can safely avoid collision.



(a) USV starts sailing

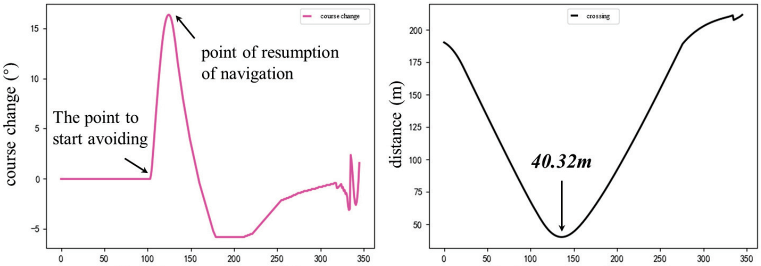
(b) USV begins to avoid



(c) USV successfully avoided

(d) USV reaches target point

Figure 11. Simulation results of collision avoidance in a crossing situation.

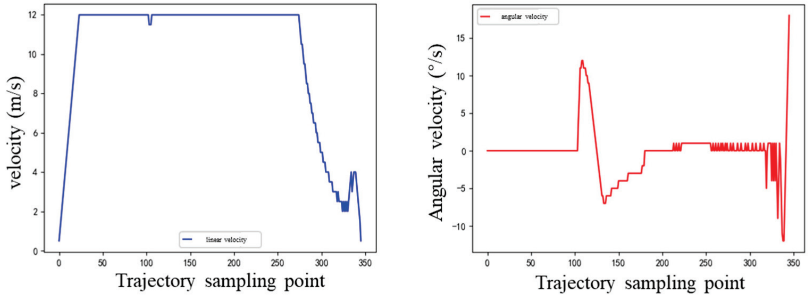


(a) course change curve

(b) DCPA

Figure 12. The course change curve and the DCPA diagram in the crossing situation.

Figure 13 shows the change trend of the line velocity and angular velocity under the crossing situation. Figure 13a shows the trend of linear speed change. During the whole process, speed up first, then sail at the maximum speed and uniform speed and decelerate near the target point. Figure 13b shows the change trend of angular velocity. It can be seen that the course is maintained before collision avoidance. During collision avoidance, turn to starboard to avoid the target ship. Due to the short process, the maximum angular velocity is about 8°/s. After collision avoidance, USV quickly resumes the course.



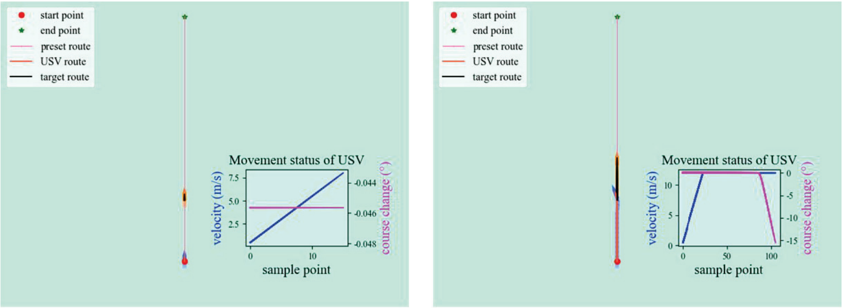
(a) Variation trend of linear velocity in crossing situation

(b) Variation trend of angular velocity in the crossing situation

Figure 13. Velocity variation trend diagram in a crossing situation.

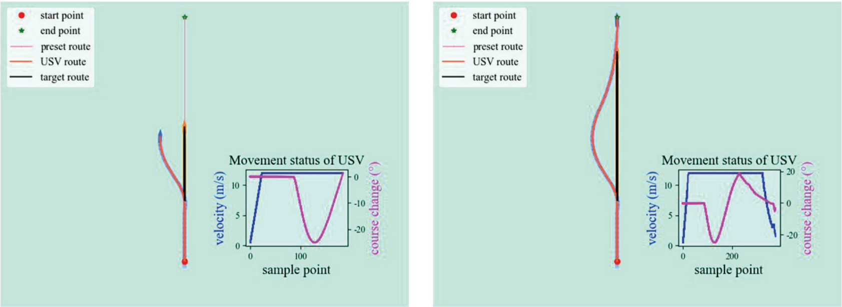
(3) Overtaking situation

The simulation process of collision avoidance in overtaking situations is shown in Figure 14. Figure 14a shows the overtaking situation. Figure 14b starts to make a decision for the USV to detect that there is a risk of collision with the target ship. Figure 14c shows that the USV follows the constraints of the COLREGs and turns to the port as much as possible to avoid the target ship. Figure 14d shows that the USV returns to the preset route after evading successfully and sails safely to the target point.



(a) USV starts sailing

(b) USV begins to avoid

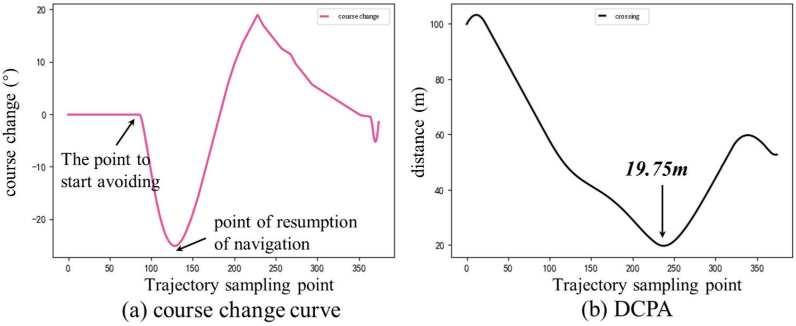


(c) USV successfully avoided

(d) USV reaches target point

Figure 14. Simulation results of collision avoidance in an overtaking situation.

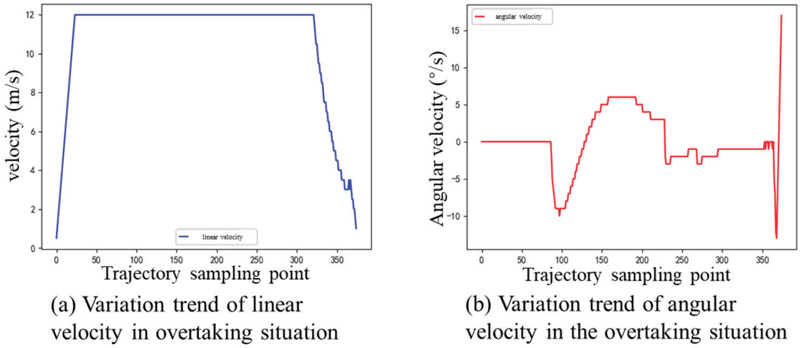
Figure 15 shows the course change curve and the DCPA diagram corresponding to the trajectory sampling point during the whole collision avoidance process under the overtaking situation. The first section of Figure 15a shows that the USV keeps a fixed heading when no target ship is detected. The second section is to start turning to the port to avoid collision when the target ship is detected. The collision avoidance behavior meets the requirements of the COLREGs, and the maximum collision avoidance range is about 27°, which is in line with the actual situation of navigation. The third section is the resumption process after collision until reaching the target point. In Figure 15b, when the encounter distance is less than 50 m, the collision distance will continue to decrease due to the close proximity of the two during the overtaking process, and the nearest encounter distance is 19.75 m.



(a) course change curve (b) DCPA

Figure 15. The course change curve and the DCPA diagram in the overtaking situation.

Figure 16 shows the variation trend of the linear velocity and angular velocity of the USV in the overtaking situation. Figure 16a shows the trend of linear speed change. During the whole process, the USV first sped up, then sailed at the maximum speed and uniform speed, and decelerated near the target point. Figure 16b shows the change trend of angular velocity and maintains the heading before overtaking. When overtaking, the USV has an obvious behavior turning port, the maximum turning rate reaches 10°/s, which is half of the maximum turning rate of the USV, and it moves towards the target point after overtaking.



(a) Variation trend of linear velocity in overtaking situation (b) Variation trend of angular velocity in the overtaking situation

Figure 16. Velocity variation trend diagram in the situation of overtaking.

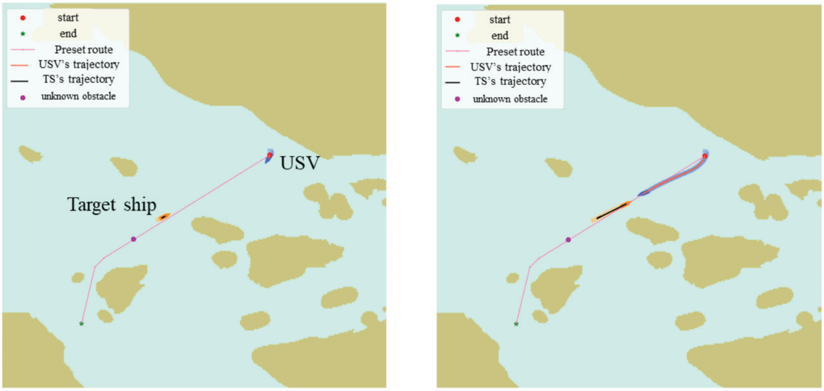
Through the analysis, it can be seen that the algorithm proposed in this paper can guide the USV to avoid the target ship safely. Since the algorithm in this paper considers the constraints of the COLREGs, it has certain practical guiding significance.

4.3. Experimental Simulation in a Local Environment

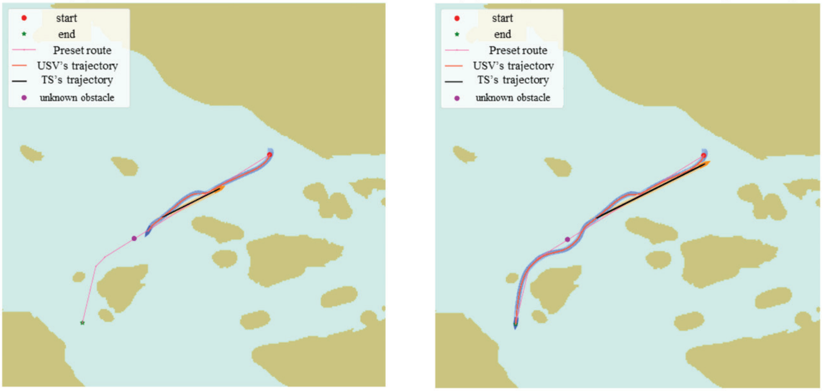
In order to verify the feasibility of the algorithm in the environment with unknown obstacles, local path planning experiments are carried out.

(1) Local path planning with dynamic obstructing ships and unknown static obstacles in the head-on situation.

Figure 17 shows the local path planning process in head-on situations and unknown static obstacles. Figure 17a shows that when the USV starts sailing, the pink line represents the preset route from the starting point to the target point to guide the navigation of the USV, and the purple dotted area represents the unknown static obstacles that are not considered in the global path planning. Figure 17a shows that when there are unknown static obstacles on the route, sailing along the preset route will collide with the obstacles. Therefore, the USV needs to avoid unknown obstacles during actual navigation. The algorithm proposed in this paper adds constraints on unknown obstacles to the trajectory evaluation function, which can ensure that the USV avoids unknown static obstacles in the navigation environment. Figure 17b shows that the USV detects the dynamic ship in the situation of collision and starts to avoid the ship to the starboard. Figure 17c shows the moment when the USV has completed the collision avoidance process and the unknown static obstacle is detected when it continues to sail along the route. Figure 17d shows the time at which the USV reaches the target point. It can be seen from the figure that the USV's navigation path can conduct collision avoidance operation between a single ship in a local environment and avoid unknown static obstacles.



(a) Before the USV sails (b) Begin to avoid the target ship

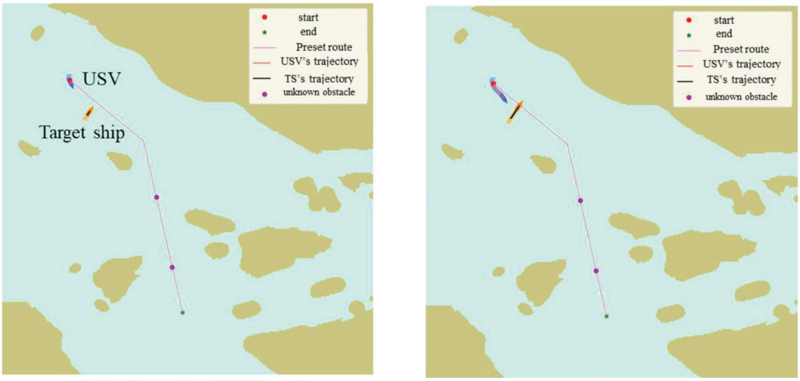


(c) Start avoiding unknown static obstacles (d) USV arrives at the terminal

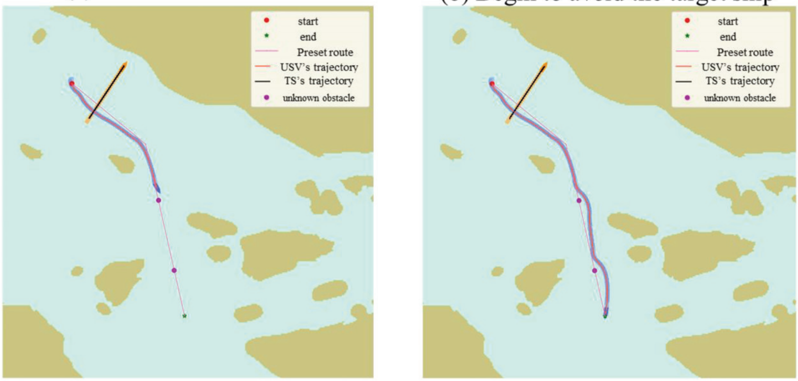
Figure 17. Partial path planning process in the head-on situation.

(2) Local path planning with dynamic obstructing ships and unknown static obstacles in crossing situations.

Figure 18 shows the local path planning process in crossing situations and unknown static obstacles. Figure 18a is when the USV is sailing, where the pink line represents the preset route that guides the unmanned ship from the starting point to the destination, and the two purple areas represent unknown static obstacles that are not considered in the global path planning. Figure 18b shows that the USV detects the dynamic ship in the crossing situation and starts to turn to the starboard to avoid collision. Figure 18c shows the moment when the USV completes the avoidance process in the crossing situation, detects an unknown static obstacle, and begins to avoid it. Figure 18d shows the time when the whole voyage is completed and the destination is reached. It can be seen from the figure that the navigation path of USV can avoid collision and unknown static obstacles under the crossing situation between a single ship in a local environment so as to ensure that the USV can sail smoothly to the destination.



(a) Before the USV sails (b) Begin to avoid the target ship

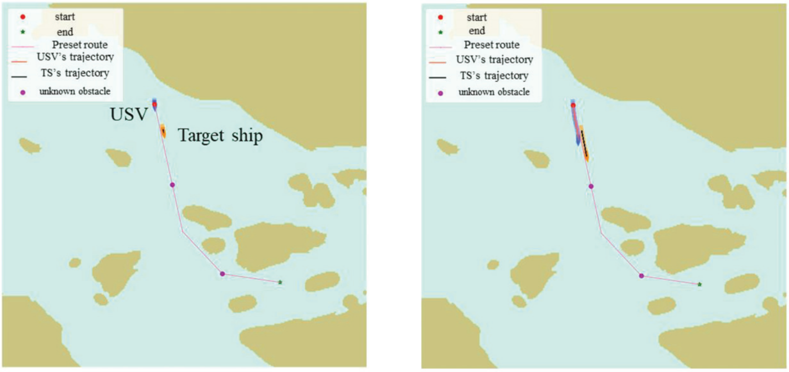


(c) Start avoiding unknown static obstacles (d) USV arrives at the terminal

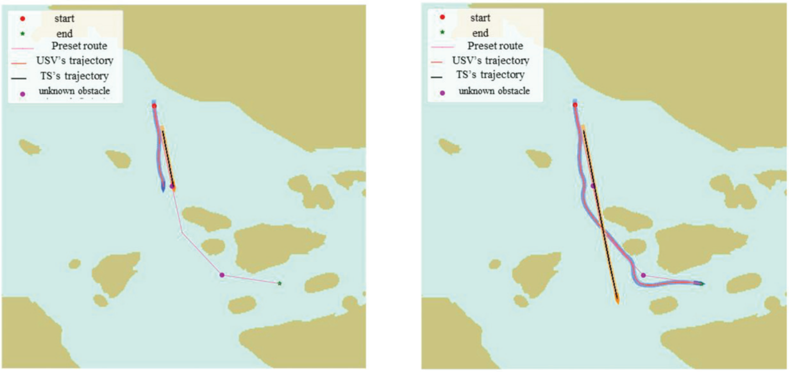
Figure 18. Local path planning process in the crossing situation.

(3) Local path planning in the overtaking situation with dynamic obstructing ships and unknown static obstacles.

Figure 19 shows the local path planning process with an overtaking situation and two unknown static obstacles. Figure 18a is the state when the USV is sailing. Figure 19b shows the dynamic ship when the USV detects the situation of overtaking and starts to turn to starboard to overtake the ship. Figure 19c shows that the USV has completed the avoidance process in the overtaking situation. When it continues to sail, it detects an unknown static obstacle and begins to avoid the unknown static obstacle. Figure 19d shows the moment when the entire voyage is completed and the destination is reached.



(a) Before the USV sails (b) Begin to avoid the target ship



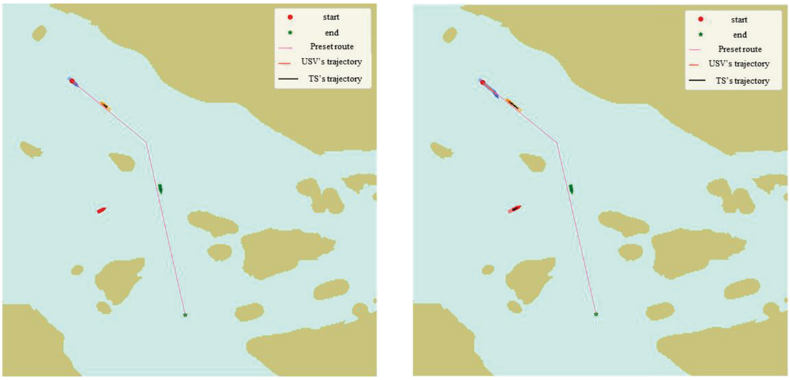
(c) Start avoiding unknown static obstacles (d) USV arrives at the terminal

Figure 19. Local path planning process in an overtaking situation.

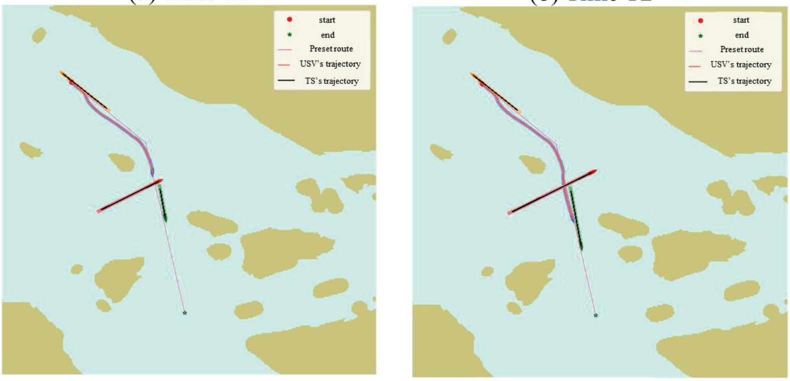
(4) Local path planning in three different situations containing dynamic obstructing ships and unknown static obstacles.

Figure 20 shows the path planning process of a multi-objective dynamic ship. At time T1, USV begins to sail. At time T2, USV starts sailing, forms a head-on situation with the target ship, and starts to turn to avoid. At time T3, the USV successfully evades and forms a crossing situation with the second target ship, and it begins to turn to the starboard to avoid. At time T4, USV has successfully evaded and returned to the preset route. Time T5 is when the USV finds the ship under the situation of overtaking and begins to implement the overtaking process. Time T6 is when the USV successfully reaches the target point.

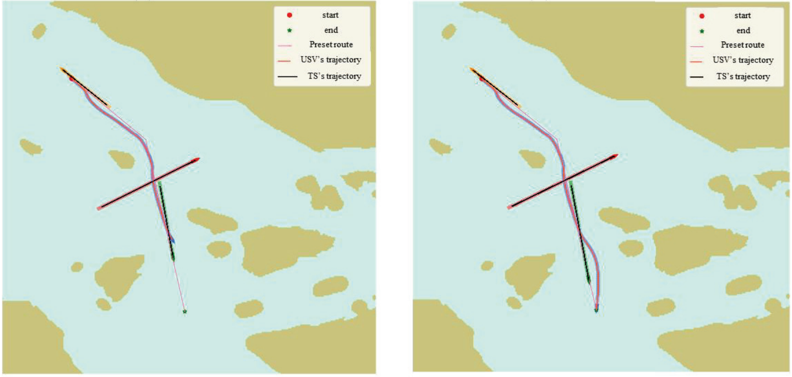
Overall, the innovation of this article is that the hazard function is introduced as the evaluation function in global path planning. In local path planning, the distance function of static unknown obstacles is introduced. The computational complexity is not high, the speed of generating routes is improved, and the sensitivity of path planning is improved when facing unknown static obstacles. In local path planning, this method is suitable for simple collision avoidance situations. In more complex scenarios, the accuracy of route generation is reduced. Next, we will focus on studying the path planning of ships in complex scenarios.



(a) Time T1 (b) Time T2



(c) Time T3 (d) Time T4



(e) Time T5 (f) Time T6

Figure 20. Path planning process of a multi-objective dynamic ship.

5. Conclusions

In the global path planning, the constraints of water depth and safety distance are mainly considered. Using the Voronoi Field algorithm to set the safety distance constraints, combined with the A* algorithm, this paper realizes the smoothing process under different safety distances. Different routes are generated by setting different safety distance constraints such as 100 m, 200 m, 300 m, etc. Finally, the data collected in path planning are visualized, and the rationality and feasibility of the algorithm are analyzed.

This paper describes the problems of USV navigation based on global path planning, that is, insufficient consideration of unknown obstacles in the environment (dynamic and static obstacles). On this basis, this paper proposes a collision avoidance decision of USV based on the dynamic window method, including the USV kinematics model, trajectory sampling considering the constraints of the COLREGs, and the optimal trajectory evaluation method. Finally, the effectiveness and rationality of the method are proved by the simulation results of collision avoidance in different situations of ships and local path planning with unknown obstacles.

Author Contributions: Conceptualization, S.H. and J.Z.; Data curation, S.H.; Funding acquisition, S.H. and J.Z.; Investigation, S.T. and R.S.; Methodology, J.Z. and S.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the National Key Research and Development Program, China (Grant no. 2021YFC2801003) and the National Natural Science Foundation of China (Grant no. 51709167).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Long, Y.; Liu, S.; Qiu, D.; Li, C.; Guo, X.; Shi, B.; AbouOmar, M.S. Local Path Planning with Multiple Constraints for USV Based on Improved Bacterial Foraging Optimization Algorithm. *J. Mar. Sci. Eng.* **2023**, *11*, 489. [CrossRef]
2. Liu, X.; Li, Y.; Zhang, J.; Zheng, J.; Yang, C. Self-Adaptive Dynamic Obstacle Avoidance and Path Planning for USV under Complex Maritime Environment. *IEEE Access* **2019**, *7*, 114945–114954. [CrossRef]
3. Liu, J.; Yan, X.; Liu, C.; Fan, A.; Ma, F. Developments and Applications of Green and Intelligent Inland Vessels in China. *J. Mar. Sci. Eng.* **2023**, *11*, 318. [CrossRef]
4. Song, R.; Liu, Y.; Bucknall, R. Smoothed A* Algorithm for Practical Unmanned Surface Vehicle Path Planning. *Appl. Ocean Res.* **2019**, *83*, 9–20. [CrossRef]
5. Han, X.; Zhang, X.; Zhang, H. Trajectory Planning of USV: On-Line Computation of the Double S Trajectory Based on Multi-Scale A* Algorithm with Reeds–Shepp Curves. *J. Mar. Sci. Eng.* **2023**, *11*, 153. [CrossRef]
6. Zhao, J.; Yan, Z.; Chen, X.; Han, B.; Wu, S.; Ke, R. k-GCN-LSTM: A k-hop Graph Convolutional Network and Long–Short-Term Memory for ship speed prediction. *Physica A* **2022**, *606*, 128107. [CrossRef]
7. Feng, Z.; Pan, Z.; Chen, W.; Liu, Y.; Leng, J. USV Application Scenario Expansion Based on Motion Control, Path Following and Velocity Planning. *Machines* **2022**, *10*, 310. [CrossRef]
8. Xu, P.F.; Ding, Y.X.; Luo, J.C. Complete Coverage Path Planning of an Unmanned Surface Vehicle Based on a Complete Coverage Neural Network Algorithm. *J. Mar. Sci. Eng.* **2021**, *9*, 1163. [CrossRef]
9. Ke, C.; Chen, H. Cooperative Path Planning for Air–Sea Heterogeneous Unmanned Vehicles Using Search-and-Tracking Mission. *Ocean Eng.* **2022**, *262*, 112020. [CrossRef]
10. Zhang, J.; Zhang, D.; Yan, X.; Haugen, S.; Soares, C.G. A distributed anti-collision decision support formulation in multi-ship encounter situations under COLREGs. *Ocean Eng.* **2015**, *105*, 336–348. [CrossRef]
11. Cai, M.; Zhang, J.; Zhang, D.; Yuan, X.; Soares, C.G. Collision risk analysis on ferry ships in Jiangsu Section of the Yangtze River based on AIS data. *J. Reliab. Eng. Syst. Saf.* **2021**, *215*, 107901. [CrossRef]
12. Xu, H.; Hinostroza, M.A.; Guedes Soares, C. Modified Vector Field Path-Following Control System for an underactuated Autonomous surface ship model in the presence of static obstacles. *J. Mar. Sci. Eng.* **2021**, *9*, 652. [CrossRef]
13. Zhang, J.; Zhang, H.; Liu, J.; Wu, D.; Soares, C.G. A Two-Stage Path Planning Algorithm Based on Rapid-Exploring Random Tree for Ships Navigating in Multi-Obstacle Water Areas Considering COLREGs. *J. Mar. Sci. Eng.* **2022**, *10*, 1441. [CrossRef]
14. Zhang, Y.; Shi, G.; Liu, J. Dynamic Energy-Efficient Path Planning of Unmanned Surface Vehicle under Time-Varying Current and Wind. *J. Mar. Sci. Eng.* **2022**, *10*, 759. [CrossRef]
15. Almoaili, E.; Kurdi, H. Path Planning Algorithm for Unmanned Ground Vehicles (UGVs) in Known Static Environments. *Procedia Comput. Sci.* **2020**, *177*, 57–63. [CrossRef]
16. Wang, H.; Lu, L.; Li, T.; Wang, A. Multi-AUG Three-Dimensional Path Planning and Secure Cooperative Path Following under DoS Attacks. *Ocean Eng.* **2023**, *274*, 113864. [CrossRef]
17. Jabbarpour, M.R.; Zarrabi, H.; Jung, J.J.; Kim, P. A Green Ant-Based Method for Path Planning of Unmanned Ground Vehicles. *IEEE Access* **2017**, *5*, 1820–1832. [CrossRef]

18. Han, Z.; Chen, M.; Shao, S.; Wu, Q. Improved Artificial Bee Colony Algorithm-Based Path Planning of Unmanned Autonomous Helicopter Using Multi-Strategy Evolutionary Learning. *Aerosp. Sci. Technol.* **2022**, *122*, 107374. [CrossRef]
19. Thoresen, M.; Nielsen, N.H.; Mathiassen, K.; Pettersen, K.Y. Path Planning for UGVs Based on Traversability Hybrid A*. *IEEE Robot. Autom. Lett.* **2021**, *6*, 1216–1223. [CrossRef]
20. Shin, J.; Kwak, D.; Kwak, K. Model Predictive Path Planning for an Autonomous Ground Vehicle in Rough Terrain. *Int. J. Control. Autom. Syst.* **2021**, *19*, 2224–2237. [CrossRef]
21. Chen, D.; Wang, Z.; Zhou, G.; Li, S. Path Planning and Energy Efficiency of Heterogeneous Mobile Robots Using Cuckoo–Beetle Swarm Search Algorithms with Applications in UGV Obstacle Avoidance. *Sustainability* **2022**, *14*, 15137. [CrossRef]
22. Liang, C.; Zhang, X.; Watanabe, Y.; Deng, Y. Autonomous Collision Avoidance of Unmanned Surface Vehicles Based on Improved A Star And Minimum Course Alteration Algorithms. *Appl. Ocean Res.* **2021**, *113*, 102755. [CrossRef]
23. Pasandi, L.; Hooshmand, M.; Rahbar, M. Modified A* Algorithm Integrated with Ant Colony Optimization for Multi-Objective Route-Finding; Case Study: Yazd. *Appl. Soft Comput.* **2021**, *113*, 107877. [CrossRef]
24. Li, J.; Zhang, W.; Hu, Y.; Fu, S.; Liao, C.; Yu, W. RJA-Star Algorithm for UAV Path Planning Based on Improved R5DOS Model. *Appl. Sci.* **2023**, *13*, 1105. [CrossRef]
25. Zou, A.; Wang, L.; Li, W.; Cai, J.; Wang, H.; Tan, T. Mobile Robot Path Planning Using Improved Mayfly Optimization Algorithm and Dynamic Window Approach. *J. Supercomput.* **2022**, *79*, 8340–8367. [CrossRef]
26. Wang, N.; Xu, H. Dynamics-Constrained Global-Local Hybrid Path Planning of an Autonomous Surface Vehicle. *IEEE Trans. Veh. Technol.* **2020**, *69*, 6928–6942. [CrossRef]
27. Han, S.; Wang, L.; Wang, Y.; He, H. A Dynamically Hybrid Path Planning for Unmanned Surface Vehicles Based on Non-Uniform Theta* and Improved Dynamic Windows Approach. *Ocean Eng.* **2022**, *257*, 111655. [CrossRef]
28. Ji, X.; Feng, S.; Han, Q.; Yin, H.; Yu, S. Improvement and Fusion of A* Algorithm and Dynamic Window Approach Considering Complex Environmental Information. *Arab. J. Sci. Eng.* **2021**, *46*, 7445–7459. [CrossRef]
29. Chi, W.; Ding, Z.; Wang, J.; Chen, G.; Sun, L. A Generalized Voronoi Diagram-Based Efficient Heuristic Path Planning Method for RRTs in Mobile Robots. *IEEE Trans. Ind. Electron.* **2022**, *69*, 4926–4937. [CrossRef]
30. Chi, W.; Wang, J.; Ding, Z.; Chen, G.; Sun, L. A Reusable Generalized Voronoi Diagram-Based Feature Tree for Fast Robot Motion Planning in Trapped Environments. *IEEE Sens. J.* **2022**, *22*, 17615–17624. [CrossRef]
31. Schoener, M.; Coyle, E.; Thompson, D. An Anytime Visibility–Voronoi Graph-Search Algorithm for Generating Robust and Feasible Unmanned Surface Vehicle Paths. *Auton. Robots* **2022**, *46*, 911–927. [CrossRef]

Journal of
Marine Science
and Engineering

Article

A System-Theory and Complex Network-Fused Approach to Analyze Vessel–Wind Turbine Allisions in Offshore Wind Farm Waters

Kai Yan [1,2,3,4], Yanhui Wang [1,2,*], Wenhao Wang [1,2], Chunfu Qiao [3,4], Bing Chen [3,4] and Limin Jia [1,2,4]

[1] State Key Laboratory of Advanced Rail Autonomous Operation, Beijing Jiaotong University, Beijing 100044, China
[2] School of Traffic and Transportation, Beijing Jiaotong University, Beijing 100044, China
[3] Transport Planning and Research Institute, Ministry of Transport, Beijing 100028, China
[4] Laboratory of Transport Safety and Emergency Technology, Beijing 100028, China
* Correspondence: wangyanhui@bjtu.edu.cn; Tel.: +86-18910186600

Abstract: Given the national goal of "emission peaking and carbon neutralization", China has become the largest country in the world for offshore wind farm construction. At the same time, navigational safety problems in offshore wind farm waters have become increasingly frequent. Owing to the complexity of offshore wind farm waters and the small number of accident data samples available for reference, the system theory method is more suitable for selection than the traditional method. Based on causal analysis based on system theory (CAST) and a complex network (CN), in this study, a qualitative and quantitative accident analysis model, CAST-CN, is constructed to analyze a complete case of vessel and wind turbine allision in offshore wind farm waters. The results show that, at the micro level, in addition to the master, crew, shipping company, and typhoon Hato, the maritime safety administration and the wind farm operation management department have a certain impact on the development of the accident discussed in this study. At the macro level, internal and external factors leading to the lack of system safety are identified, and measures and suggestions for system safety improvement are proposed based on analysis. This study can fill the research gap in the systematic analysis of traffic accidents in offshore wind farm waters and provide support for the safety assessment and decision-making of government management departments and research institutes.

Keywords: accident analysis; offshore wind farm; STAMP; CAST; complex network

Citation: Yan, K.; Wang, Y.; Wang, W.; Qiao, C.; Chen, B.; Jia, L. A System-Theory and Complex Network-Fused Approach to Analyze Vessel–Wind Turbine Allisions in Offshore Wind Farm Waters. *J. Mar. Sci. Eng.* **2023**, *11*, 1306. https://doi.org/10.3390/jmse11071306

Academic Editor: Mihalis Golias

Received: 8 June 2023
Revised: 20 June 2023
Accepted: 23 June 2023
Published: 27 June 2023

1. Introduction

1.1. Background

With the global vision of carbon neutrality, offshore wind power is going through a phase of rapid growth worldwide. China has always considered the development of the wind power industry as an important means to achieve "emission peaking and carbon neutralization". Supported by the subsidy policy, China has surpassed the UK to become the world's largest country in terms of total installed offshore wind power capacity at a stunning rate. As shown in Figure 1, approximately 21.1 GW of new offshore wind power capacity was added globally in 2021, and the cumulative installed capacity reached 57 GW [1]. Meanwhile, 16.9 GW of offshore wind power was added in China, and the cumulative installed capacity is 27.7 GW, accounting for 80.1% and 48.4% of the total, respectively. With advantages such as large wind farm areas, low visual impact, high wind speed, and low transmission cost, offshore wind power has great potential for development, and the scale of installed offshore wind power is expected to continue to exceed expectations [2]. Due to the high development trend of offshore wind power, the conflict of sea resources between the construction and operation of offshore wind farms (OWFs) and other sea-related activities has gradually increased, affecting the safety of ship navigation [3].

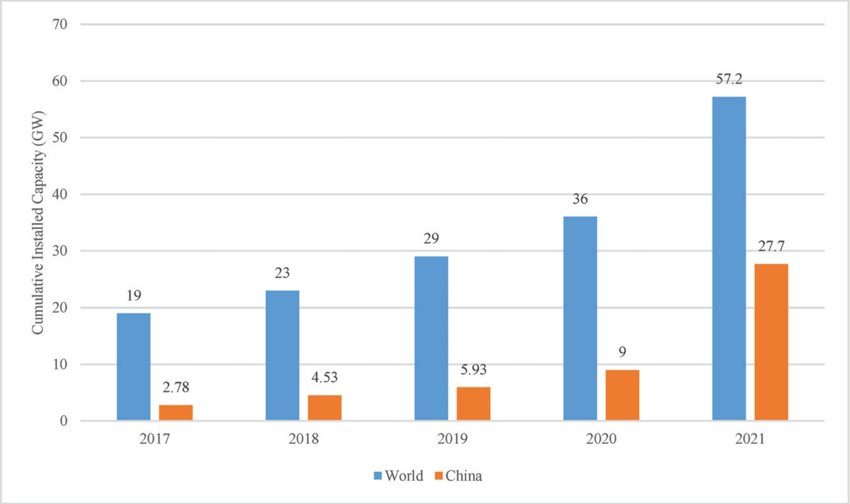


Figure 1. 2017–2021 Global and Chinese cumulative installed offshore wind power capacity.

Marine traffic is variously impacted by the construction of offshore wind farms. First, the presence of OWFs signifies that vessels must avoid more obstacles in the water and ensure no allisions between vessels and offshore wind facilities [4]. Second, OWFs may also limit the navigable space available to vessels, causing increased traffic density and an increased risk of collisions between vessels [5]. Additionally, the physical structure and electromagnetic characteristics of the wind turbine units of offshore wind turbines can interfere with—even block—shore-based perception and communication systems, such as the vessel traffic service system (VTS), the automatic identification system (AIS) and the very high frequency communication system (VHF), which may lead to the loss of the supervision and guidance for a vessel from the maritime safety administration (MSA), decreasing the navigational safety level [6,7], especially in extreme weather [3]. Moreover, due to the difficulty of the construction and operation, and maintenance of offshore wind farms, a large number of workers need to be transported by special engineering vessels (service vessels) to complete operations at sea, increasing safety risks.

Additionally, the combination of the above-mentioned influencing factors and strong wind energy significantly increases the risk of accidents in offshore wind farm waters, compared to other navigable waters [8]. The most common types of marine accidents in offshore wind farm waters are vessel–wind turbine allisions (VTAs), vessel–vessel collisions, and vessel sinking caused by wind disasters [9]. The analysis and prevention of VTAs are more complex and difficult than those of the latter two types of accidents, and this is because vessels have direct physical interactions with wind farm facilities. More companies and organizations are involved, especially during the construction phase of an offshore wind farm. Moreover, government safety supervision and legal policy formulation are categorized as two industry sectors. In China, for example, the safety management of vessels is handled by the China Maritime Safety Administration (China MSA) under the Ministry of Transport, while the site selection approval and safety supervision guidance of offshore wind farms is handled by the National Energy Administration under the Ministry of Natural Resources. Therefore, determining how to scientifically and effectively analyze and prevent VTA has become a focus of research on the navigational safety of offshore wind farms [10].

1.2. Related Works

The majority of accident analyses in offshore wind farm waters have been conducted based on the theory of accident causation, and the objective of these studies is to identify causal factors related to accidents, such as navigation environment, traffic flow conditions, machine failures, and human errors [3]. Scholars have analyzed various marine accidents arising from the construction and operation phases of offshore wind farms, as well as the corresponding risk influencing factors, to address the issue of navigational risk in offshore wind farm waters for vessels [11,12]. Commonly used methods include the Formal Safety Assessment (FSA) of the International Maritime Organization [13], fault tree [8,14,15], and Bayesian networks [16,17].

For example, Rawson and Brito used fault tree analysis to study the navigational risks associated with environmental changes caused by the construction of offshore wind farms, and their study showed that collisions were the most probable risks [18]. Dai established a system fault tree analysis of offshore wind farm operations, and the assessment showed that the key factors leading to accidents, such as offshore wind collapse, personal injury, ship collision, and damage to submarine cables, also included high winds, untimely maintenance, and collision avoidance failure [19]. Mehdi studied the dynamic risk assessment of vessels operating in the waters of offshore wind farms. The study showed that offshore wind farm facilities were detrimental to the navigational safety of passing vessels, the safe operation of wind farm support vessels, and emergency operations such as search and rescue (SAR), and that the risks of these operations resulted from the reduction of ocean space and the increase in traffic density [20]. To assess the overall navigational risk in offshore wind farm waters objectively and accurately, Mehdi selected several indicators from both natural conditions and the navigational environment, and constructed a model to assess the navigational risk of vessels in offshore wind farm waters, not considering factors such as personnel reliability, technical failures, and traffic management [21].

The traditional accident analysis methods mentioned above are widely employed in research in the field of ship navigation safety [22]. However, the issue of VTAs analysis in offshore wind farm waters requires simultaneous consideration of offshore wind farms, various types of vessels in the water, and other elements. This is a complex systemic problem, and the traditional risk assessment methods are unable to systematically analyze the connections between various elements [23]. Another key challenge in marine accident analysis in offshore wind farm waters concerns the scarcity of historical data on relevant accidents. Hence, qualitative methods such as expert judgment are more commonly used [24]. Qing Yu discussed the possibility of merchant vessel accidents due to offshore wind energy development off the Atlantic coast of the United States, enlisting the advice of nautical experts to assess the probability of allisions, collisions, or groundings of merchant vessels due to the presence of offshore wind farms [25,26]. In addition, owing to China's late start of offshore wind power, research literature and historical accident data are scarcer in China than in countries such as the UK [13,27]. Therefore, some machine learning methods that rely on large-scale data sets for Natural Language Processing(NLP) model training are difficult to apply to the analysis of VTA [28].

1.3. Objective and Outline

Against the background described above, the present study aims to construct a qualitative and quantitative accident analysis method based on system theory to analyze a complex VTA accident case [29]. The study comprehensively analyzes the development, cause, and impact degree of VTA accidents, and explores deeper influencing factors, thus providing new ideas for improving the intrinsic safety of ship navigation systems in offshore wind farm waters.

To this end, this paper introduces the causal analysis based on system theory (CAST) model, which is based on the Systems-Theoretic Accident Model and Process (STAMP), proposed by Professor Leveson of NASA Institute in 2004 [30], and creatively integrates complex network (CN) analysis methods [31]. A typical VTA accident occurring along the coast of China is selected as a case study [32]. Using the accident investigation report, the events chain is clarified, the accident causes are identified, the safety control structure model and complex network model are constructed, the importance of key nodes is evaluated, and the defects at the system level are analyzed from a macro perspective. Finally, suggestions for improvement are provided. In the second section, the framework of system analysis, methods used, and calculation indexes are introduced. The third section introduces the complete case analysis process. In the fourth section, the innovation, application significance, and limitations of research are discussed. Section 5 concludes the study and discusses future prospects.

2. Methodology

2.1. The Analysis Framework

The approach to analyzing VTA accidents in this study is based on the CAST model integrated complex network theory to achieve quantitative analysis results. As shown in Figure 2, the analysis framework is divided into three stages. Stage 1 is the initialization, and the accident narrative is completed by extracting valuable information from a VTA accident investigation report. Then the traffic system in offshore wind farm water (TSOWF) can be defined and the system hazard and the constraint can be identified as the fundamental step of the entire CAST analysis procedure. In Stage 2, microanalysis, a hierarchical safety control structure (HSCS) is designed first to depict and code both the system components and their relationships. The detailed analysis of all the system components is conducted based on the proximate events in the VTA accident and the coded HSCS. Then a V-T network/matrix is constructed and weighted according to the HSCS and the components analysis results. Finally, the network eigenvalues are computerized to attain the critical components in the TSOWF. In Stage 3, macroanalysis, the system deficiency in the VTA accident is identified and improvement recommendations are proposed.

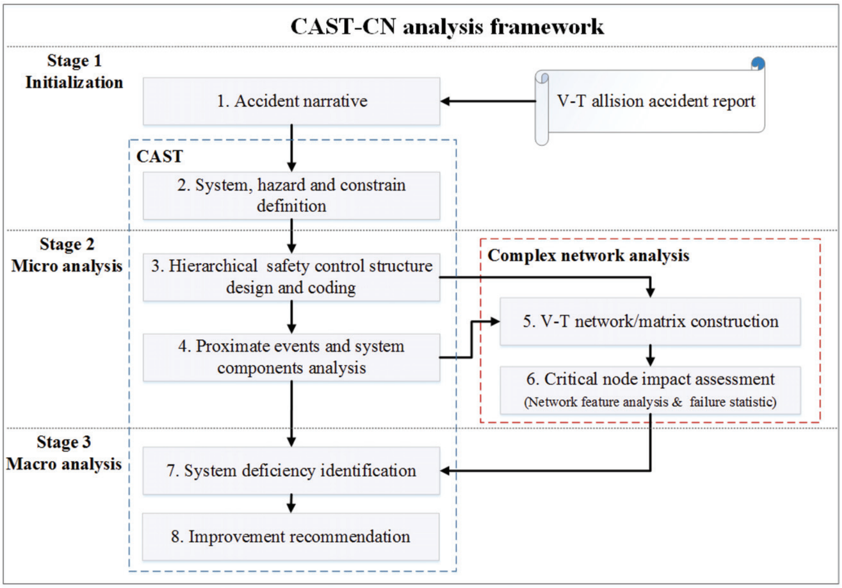


Figure 2. The analysis framework.

2.2. CAST Procedure

CAST is derived from STAMP, especially for accident qualitative analysis. Based on the idea underlying STAMP, CAST is created to fulfill the goals of analyzing all accident causes (optimized learning), reducing hindsight bias, systematically thinking about human behavior, providing blame-free explanations, and improving the safety control structure of the system [33]. The main analysis framework of CAST accords with STAMP, and comprises system and system hazard definitions, HSCS modeling, component analysis, control structure (system) flaw identification, and improvement suggestions. The difference lies in the fact that the system hazard identified by CAST is only related to the given accident scenario, and it is necessary to determine the proximate events leading to the loss [34]. Therefore, the procedure of CAST optimization and adjustment in this study is divided into five steps, as shown in Table 1.

Table 1. Experiment parameter setting.

Id	CAST Step	Details
1	Define system, hazard, constrain	(1) Define the system involved and the boundary of the analysis. (2) Describe the loss and identify the system's hazardous state (system hazard). (3) Identify the system safety requirements and constraints.
2	Design and code the hierarchical safety control structure (HSCS)	(1) Model the HSCS by learning from the existing system structure and the accident report. (2) Code the components (A–Z) and their relationships, including controls (C), feedback (F), communication (N), and physical impacts (I). For example, if component "A" controls component "B", there should be a relationship link coded "CAB".
3	Determine the proximate events and analyze components	(1) Find the proximate events in a timeline from the accident report. (2) Analyze the components by determining their responsibilities, safe and unsafe actions, contexts, and mental model flaws.
4	Identify the system's deficiencies	Identify flaws in the control structure as a whole (general systemic factors) that contributed to the loss.
5	Propose the improvement recommendation	Create recommendations for changes to the control structure to prevent a similar loss in the future.

Since all STAMP-derived models, including CAST, are only suitable for qualitative analysis [35,36], to improve the accuracy of the analysis and further weaken the subjective factors of manual qualitative analysis, this paper introduces important quantitative analysis indicators from complex network theory as a supplement to the CAST analysis process.

2.3. System Component Analysis Based on a Complex Network

Complex network theory can quantify and analyze complex systems well, providing a good method for identifying critical nodes in the system. A large number of documents have already applied complex network theory to model and analyze real systems. For example, Shaphari et al. analyzed the fragility of the Iranian power grid using weighted PageRank and identified critical fragile nodes [37]. Zhao et al. analyzed a weighted city infrastructure system network using biased PageRank, reflecting the importance of infrastructure in topology and functionality [38]. Kopsidas and Kepaptsoglou developed a public transportation network with subway stations as nodes and analyzed the importance of nodes using a combination of closeness centrality and betweenness centrality [31]. Tang et al. established a directed weighted network of unsafe behavior in building accidents and analyzed its characteristics using five network attributes: degree and degree distribution, node strength and node strength distribution, average path length and diameter, weighted clustering coefficient, and intermediary degree centrality [39].

In this study, the network formed by system components for a VTA accident scenario can be defined as a "V-T network". The nodes in the network represent the components in the system, while the edges represent their interactions based on the modeled HSCS. Thus, the V-T network is a directed network [37]. The V-T network comprising n nodes and m edges is converted into a directed and weighted graph $G = (V,\ E)$, and the node in the V-T network is represented as $v_n \in V$, while the link in the network is represented as $e_m \in E$.

The weight $w_{i,j}$ of the $e_{i,j}$ is determined by the number of failure controls/feedback $f_{i,j}$ from v_i to v_j according to the statistics from the accident report, as shown in the following formula. The smaller $w_{i,j}$ is, the more fragile the control/feedback relationship is and the closer the distance between v_i and v_j in the network is.

$$w_{i,j} = \frac{1}{1 + f_{i,j}} \tag{1}$$

Then a weighted adjacency matrix $M_{i,j}$ is constructed.

$$M_{i,j} = \begin{bmatrix} w_{1,1} & \cdots & w_{1,n} \\ \vdots & \ddots & \vdots \\ w_{n,1} & \cdots & w_{n,n} \end{bmatrix} \tag{2}$$

Based on this, the impact assessment process of system components is as follows:

Step 1. Analyze network structure features utilizing PageRank (PR). *PR* is a method used to calculate the number of important nodes connected by a node [40]. Typically, the value of PR must undergo multiple iterations before a stable outcome can be reached. This stable outcome serves as the ultimate basis for ranking [41,42]. The *PR* value of node v_i at iteration time t is

$$PR_t(v_i) = \frac{(1-s)}{n} + s \sum_{j=1}^{n} w_{ji} \frac{PR_{t-1}(v_j)}{k_{out}(v_j)} \tag{3}$$

where, $PR_t(v_i)$ is the *PR* of node v_i at time t, s is the random jumping probability, which is usually set around 0.85, w_{ji} is the weight of edge v_j, to v_i, $k_{out}(v_j)$ is the out-degree of node v_j, $PR_{t-1}(v_j)$ is the PageRank value of node v_j at time $t-1$, and n is the total number of nodes in the network.

When the difference between the PageRank values at time t and $t-1$ is less than a specific threshold (i.e., when Equation (4) is satisfied), the iteration is considered to be in a stable state, and the $PR_t(v_i)$ will be the final result.

$$\left| PR_t(v_i) - PR_{t-1}(v_j) \right| \leq \alpha \tag{4}$$

where α is a specific threshold, the value of α = 0.0001 is selected in this paper.

Step 2. Analyze Closeness Centrality (*CC*). In the V-T network, Closeness Centrality is defined as how close a node is to other nodes, usually expressed as the following formula [41].

$$CC(v_i) = \frac{n-1}{\sum_{j \neq i}^{n} g(v_i,\ v_j)} \tag{5}$$

where $CC(v_i)$ is the closeness centrality of the node vi, and $g(v_i,\ v_j)$ is the shortest-path distance between node v_i to v_j.

Step 3. Analyze Betweenness Centrality (*BC*). A higher *BC* means that the influence of a node on the entire network information flow is greater [43–45].

$$BC(v_i) = \sum_{v_s \neq v_i \neq v_t \in V,\ s<t} \frac{\sigma_{st}(v_i)}{\sigma_{st}} \tag{6}$$

where $BC(v_i)$ is the betweenness centrality of node v_i, σ_{st} is the number of the shortest path from the node v_s to v_t, and $\sigma_{st}(v_i)$ is the number of those paths that pass through v_i.

Step 4. Compute Network Importance (*NI*). *NI* is the comprehensive network feature value that combines the above three indicators, representing the influence of a node in the entire network structure.

$$NI(v_i) = \frac{S(v_i)}{\sum_{v_i \in\ V} S(v_i)} \tag{7}$$

$$S(v_i) = \frac{PR(v_i)}{\sum_{v_i \in\ V} PR(v_i)} + \frac{CC(v_i)}{\sum_{v_i \in\ V} CC(v_i)} + \frac{BC(v_i)}{\sum_{v_i \in\ V} BC(v_i)} \tag{8}$$

Step 5. Component Impact (*CI*) assessment. The *CI* represents the contribution of a component to a VTA accident, and its quantitative calculation integrates the *NI* of a component and the proportion of failure control/feedback actions performed by that component.

$$CI(v_i) = log_2\left((NI(v_i) + 1) \times \left(\frac{f_i}{\sum_{i=1}^{n} f_i} + 1\right)\right) \tag{9}$$

3. Case Study

3.1. Stage 1: Initialization

In Stage 1, the accident narrative is completed by extracting valuable information from a VTA accident investigation report [32]. After that, the TSOWF can be defined, and the system hazard and constraints can be identified as the fundamental step of the entire CAST analysis procedure.

3.1.1. Accident Narrative

At 21:00 on 22 August 2017, the vessel Rongxiang 66 of Bohai New Area Rongxiang Shipping Co., Ltd. of Cangzhou, China was carrying 5100 tons from Chi Bay, Shenzhen to the west side of Guishan Pilot Anchorage at the mouth of the Pearl River. At approximately 11:05 on 23 August, affected by super typhoon Hato, the vessel crashed into the base of the #02 wind turbine of the Guishan Offshore Wind Farm in Zhuhai, causing the cargo hold to sink into the water. Eleven people on board fell into the water, including five dead, three missing, and three rescued. The direct economic loss of the accident was approximately CNY 12,460,000. According to the accident investigation report, the causes of this VTA accident included the impact of severe weather and sea conditions caused by super typhoon Hato, the insufficient anchoring position of the vessel to stabilize against the strong typhoon, the insufficient guidance of the vessel safety management system document on typhoon prevention, the insufficient deployment of the master's typhoon prevention work, and the failure to actively enact typhoon prevention measures as early as possible. According to the accident investigation report, Rongxiang 66 had to be held responsible for the accident, and the master was the person responsible for the accident. It can be seen from Figure 3 that the vessel's anchoring position was at the periphery of the anchorage—only 0.8 nm away from the No. 2 pile foundation [32].

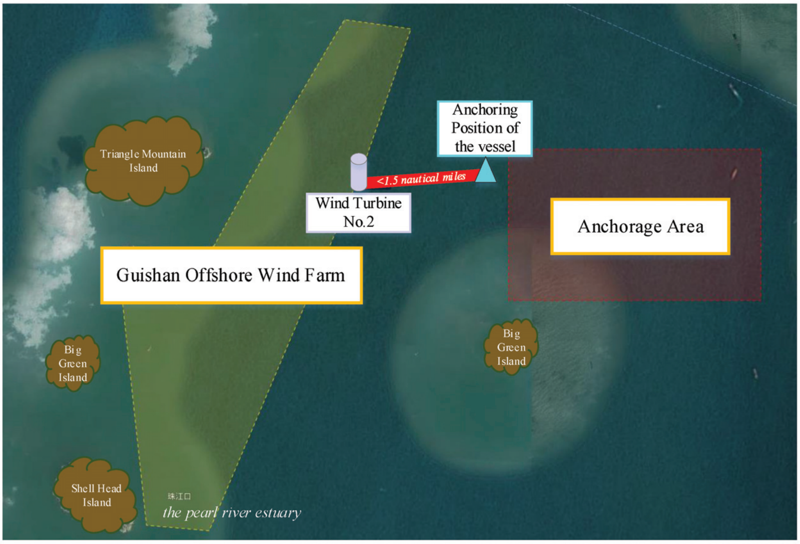


Figure 3. Anchor position of Rongxiang 66 and base position of wind turbine #02.

3.1.2. System, Hazard, and Constraint Definition

In accordance with the basic steps of the application of CAST, the first step is system definition and system hazard identification [46].

1. The traffic system in offshore wind farm water (TSOWF)

Based on the research results of maritime traffic engineering scholars around the world, maritime traffic can be defined as the combination of vessel movements and the overall behavior of vessels in a designated area. Therefore, the maritime traffic system is defined as a dynamic and complex technology environment system involving human control behavior and organizational management roles. The TSOWF can be defined as a collection of various vessel movements and various factors acting on it in offshore wind farm waters. These factors include the water environment, infrastructure, vessels, people who control vessels, and management. The factors change dynamically and interact with each other, thereby jointly determining the dynamic change process of the system state. The system consists of natural environment elements such as hydrology, meteorology, and ocean bottom material, navigation environment elements such as channels, anchorages, ports and wharves, the wind turbines of wind farms, booster platforms, submarine cables, and relevant infrastructure elements of ports and wharves, marine transport vessels, fishing vessels, leisure vessels, engineering vessels, and other vessel elements, as well as crew members, operation and maintenance personnel, supervisors, and other personnel elements.

2. System hazard

According to system theory, not all the functional components in the system operate independently; rather, the components generally interact with each other and undergo dynamic changes in the time and space dimensions. Therefore, the state of the system also changes dynamically. When the stable state of the system collapses, accidents occur. The primary cause of a brewing accident is system hazard. Only by accurately identifying the system hazard can the probability of accidents and losses be effectively reduced [47].

The term "hazard" has many definitions in the field of safety science. It usually refers to the source of danger to a person, property, or the environment. From the perspective of system theory [47], this paper defines the system hazard (*SH*) as a set of hazards (*h*) that may lead to a system collapse:

$$System\ Hazard\ (SH) = \{h_1 + h_2 + h_3 + \cdots + h_n\ \} \tag{10}$$

The key to analyzing the traffic accidents of offshore wind farms lies in identifying the system hazards of the traffic system of offshore wind farms, as well as in analyzing the structure and mechanism of the system hazards with the help of systematic analysis methods in order to find countermeasures. Based on the direct cause, allision is divided into two types: dynamic allision and nondynamic allision. The former is VTA caused only by human error during navigation [48]. The latter refers to an accident in which the vessel loses control due to equipment failure or the impacts of wind, waves, and currents. Rongxiang 66 VTA can be considered a nondynamic allision. In this case study, two system hazards are mainly identified by the Rongxiang 66 VTA accident report.

System Hazard in Rongxiang 66 VTA accident:

SH1: Vessel anchoring failure caused by wind disasters.
SH2: The vessel is unable to be aware of the OWT on time.

A vessel's stability will decline because of strong wind, waves, heavy rainfall, and other natural environmental factors in the water area, which will result in the vessel going out of control. Because the site selection waters of offshore wind farms are mostly located in areas with abundant wind energy, strong winds and massive waves often occur in these areas. In the Rongxiang 66 VTA accident, Rongxiang 66 was anchored at the arriving route of Super Typhoon Hato. Therefore, one of the system hazards was vessel anchoring failure and dragging caused by strong wind and waves. However, due to the proximity of the wind farm to the anchorage, the TSOWF itself carried a high allision risk. It was too late for Rongxiang 66 to discover the wind farm until it was close enough to collide with the wind turbine. Thus, another system hazard was that the vessel could not be aware of the OWT promptly.

The corresponding system safety constraints are:

SC1: Rongxiang 66 should adopt correct anchoring measures when encountering typhoons.
SC2: Rongxiang 66 could be aware of the position of the Guishan OWF earlier.

3.2. Stage 2: Microanalysis

In Stage 2, a hierarchical safety control structure (HSCS) is designed first to depict and code both the system components and their relationships. A detailed analysis of all the system components is conducted based on the proximate events in the VTA accident and the coded HSCS. Then a V-T network/matrix is constructed and weighted according to the HSCS and the components analysis results. Finally, the multiple network eigenvalues, including *PR*, *CC*, *BC*, *NI*, and *CI*, are computerized to attain the critical components in the TSOWF [49].

3.2.1. Hierarchical Safety Control Structure Design and Coding

The most important step in the STAMP/CAST modeling process is to build a safety control structure, which requires combining expert experience with accident investigation reports to comprehensively display accident-related information as much as possible. In this study, based on the system hazards which are identified in Stage 1, the HSCS is optimized to clearly show the control relationships between system components. First, the system components are classified. The system components are divided into three layers from top to bottom according to man and management (MM), machines and facilities (MF), and environment (E), represented by yellow, blue, and green rectangles. Second, the representation of relationships between components is optimized. Control or feedback relationships are represented by implementation arrows, which are always top-down or left-to-right, while the feedback is the opposite. A dashed line represents a communication relationship between two components, which generally exists only between two components that are not in control of each other. Third, dotted arrows are used to indicate physical impact. The impact can be unidirectional or bi-directional. For example, environmental factors have unilateral effects on the vessel. The allision between the vessel and the wind turbine has a two-way impact, as shown in Figure 4. In order to facilitate component analysis and subsequent complex network analysis, we number both components and control relationships. To facilitate component analysis and subsequent complex network analysis, both components and control relationships are coded. Since only 25 system components are involved in Rongxiang 66 VTA, we code these in capital letters and mark the failure control relationship on the HSCS diagram (Figure 5). The detailed coding matrix can be found in Appendix A.

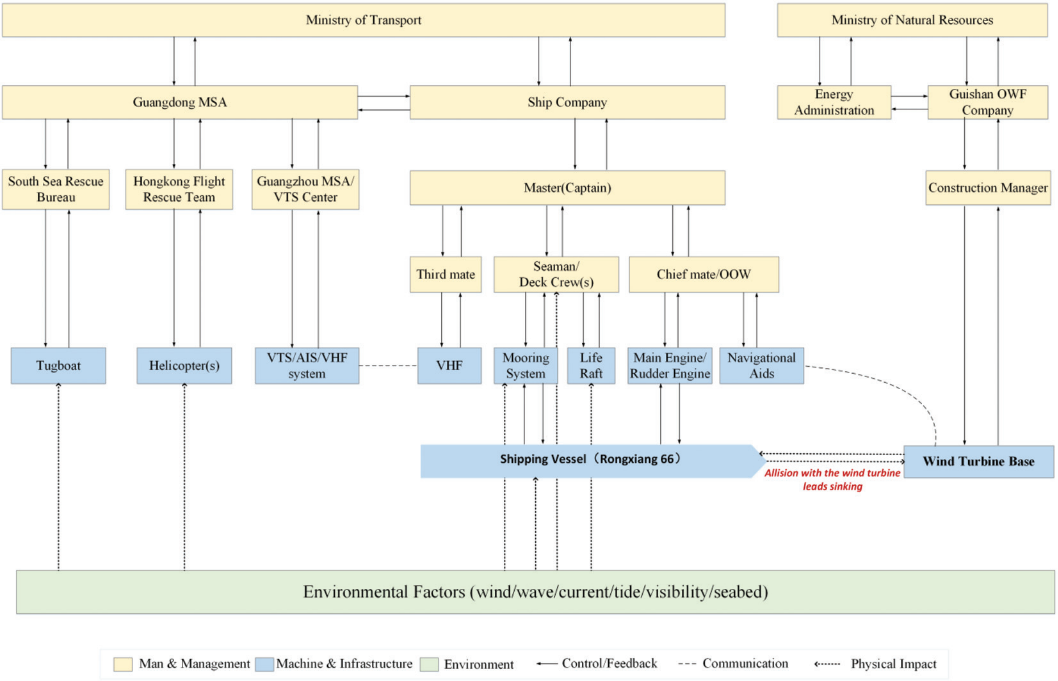


Figure 4. Basic optimized HSCS model.

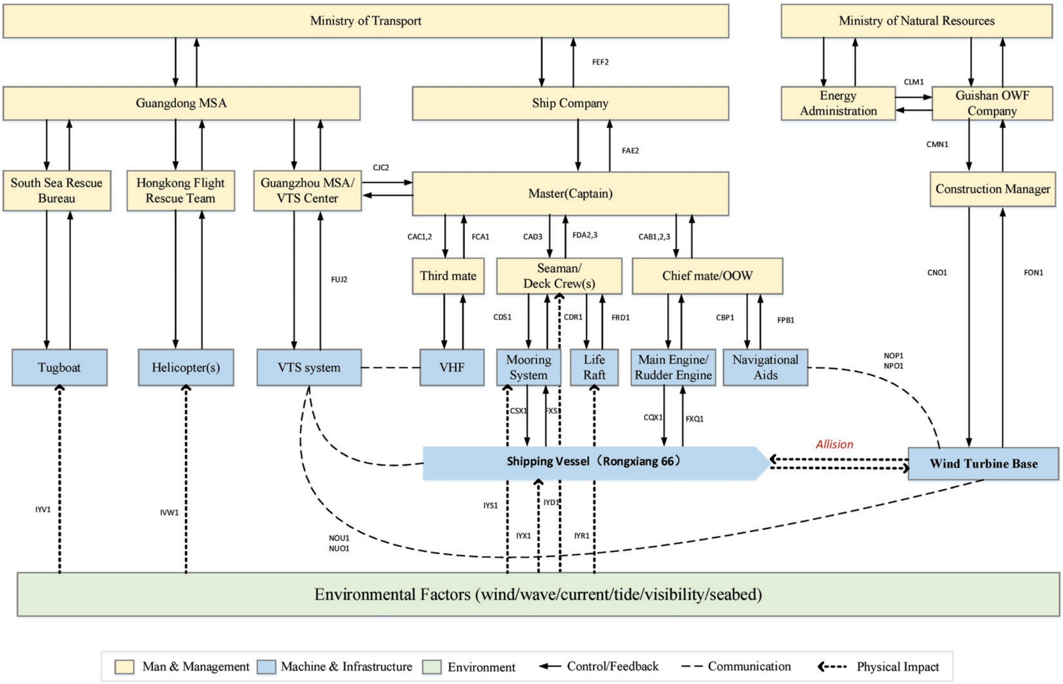


Figure 5. Marked HSCS model.

3.2.2. Proximate Events and System Components Analysis

In this study, the accident proximity events recorded in the accident report are sorted out. Table 2 lists the events on the day of the accident, starting with Hato affecting vessel Rongxiang 66′s stable anchorage and ending with the sinking of the vessel [32]. Based on this detailed analysis of the accident process, we combine Figure 5 to make a detailed analysis of each component of the system, including safety responsibility/basic information, inadequate control/feedback actions, context, and mental/process model flaws, as detailed in Appendix A.

Table 2. Proximate events.

Time	Proximate Events
7:52	Rongxiang 66 began to drag its anchor and move in the southwest direction, drifting at a speed of 1.5 knots.
8:00	Anchor dragging continued. The vessel's drift speed was 1.6 knots, while the northeast wind was at level 8 and the wave height was 2 m. The master and chief mate were on duty at the bridge and began to start the main engine and rudder main to head against the wind.
8:31	Anchor dragging continued. The vessel's drift speed was 1.1 knots, while the northeast wind was at level 9 and the wave height was 3 m.
8:58	The vessel continued to drag anchor, with a drift speed of 1.1 knots, moderate rain, an east wind force of 10, and a wave height of 4 m.
9:38	The vessel continued to drag anchor, with a drift speed of 1.1 knots, heavy rain, an east wind of 11, and a wave height of 4 m.
10:11	The vessel continued to drag anchor, with a drift speed of 2.8 knots, heavy rain, an east wind force of 12, and a wave height of 5 m.
10:28	The vessel continued to drag anchor, with a drift speed of 0.3 knots, a rainstorm, an east wind of 13, and a wave height of 6 m. The master requested to cast the right anchor, but the crew was afraid to go to the bow due to the strong wind and waves
11:00	The vessel continued to drag anchor, with a drift speed of 3.3 knots, a rainstorm, an east wind of 14, and a wave height of 8 m. The master found the vessel approaching the #02 wind turbine base and ordered the crew to report the danger to the shipping company. He ordered the third officer to report the danger to Guangzhou VTS using VHF, and Guangzhou VTS instructed the vessel to take self-rescue measures.
11:05	A VTA occurred and the hull was damaged and flooded. The master reported to Guangzhou VTS and announced the abandonment of the vessel, ordering the first mate to release the life raft, but the life raft was soon blown away by the strong wind after entering the water. The master asked all the crew to assemble at the stern wearing life jackets.
11:10	Rongxiang 66 sank

3.2.3. V-T Network Construction

Based on the component analysis and HSCS diagram, the V-T network and the weighted adjacency matrix (Appendix B) are constructed. The node ID in the V-T network is the same as that in HSCS. The system component label is shown on each node in the diagram in Figure 6. The higher the proportion of incorrect actions is, the lower the weight is and the thinner the lines on the chart are, implying that the link is more fragile. The diagram shows that the master is in the central position with a higher error rate, followed by the chief mate and third mate, which also corresponds to the conclusions of the accident investigation report. The OWF construction managers also show a high error rate. Regarding marine management, there are some problems in the communication and coordination of the Guangzhou VTS center. Moreover, the vessel and all crews were heavily affected by typhoon Hato, while neither the vessel nor the VTS center detected the wind turbine.

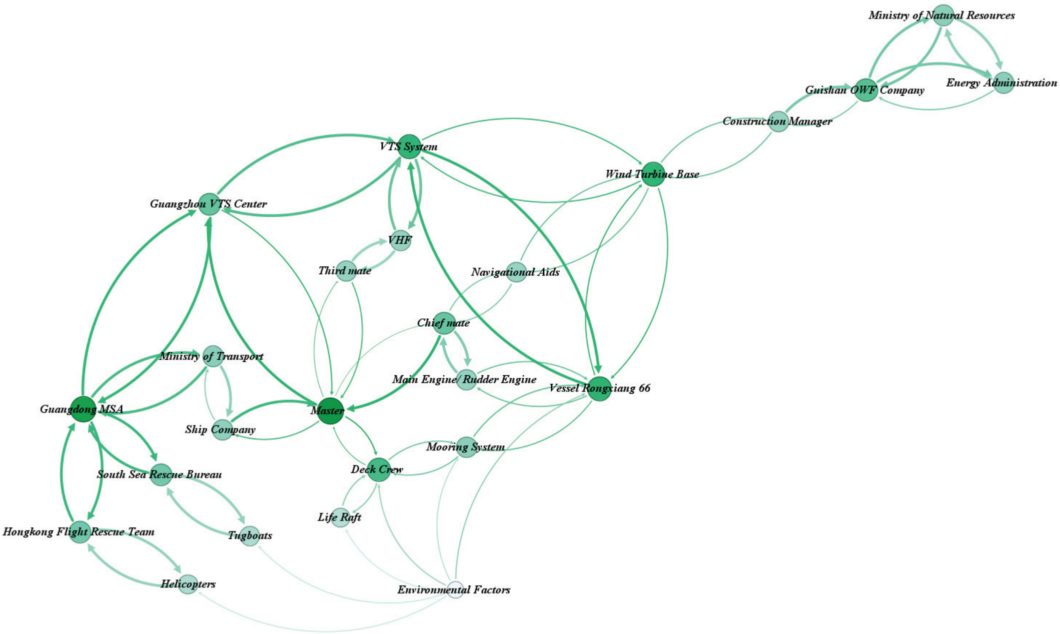


Figure 6. Diagram of the V-T network.

3.2.4. Critical Node Analysis

Based on the V-T network/matrix, the PR, CC, and BC of the network nodes are calculated. The results are displayed in Appendix C. The PR (0.0745) of the master ranks first, indicating that the master has connected more important nodes in the whole system. As the first person in charge of ship safety, the master is indeed the most important role and the most important component of the system in this accident. The highest CC (0.4) ranking is the environmental factor, indicating that the node affected by the environmental factor has the most problems. This also corresponds to the actual situation in that typhoon Hato causes a series of events after Rongxiang 66's anchoring. The highest BC (0.3496)-ranked Guangzhou VTS center is the front-line unit of traffic management because the VTS center should coordinate vessels and report to senior management in a timely fashion. Based on the above three eigenvalues, the normalized network importance (NI) is calculated. The results of NI uncover that the maritime management departments, including the MSA, VTS center, and VTS system, have a high degree of importance in the network model, which indicates the 24/7 vessel traffic service the departments provide and the important command and coordination role they play in emergency response after accidents occur. The calculation results of component impact obtained by superimposing the network importance and component failure ratio are shown in Figure 7. The top ten components account for 71.5% of all component impacts on this VTA accident, so they are regarded as critical components. The master, wind turbine base, environment, deck crew, and VTS center rank higher. The master holds a high level of importance in the network because he undertakes the safety management, typhoon prevention, and emergency evacuation decision-making functions of the vessel and crew. Obviously, in this accident, the master also has an unshirkable responsibility. Meanwhile, of environmental factors having a wide and serious impact on vessels and crew, almost all were triggered by super typhoon Hato. The analysis result of above system components or factors is basically consistent with the causation analysis in the accident investigation report.

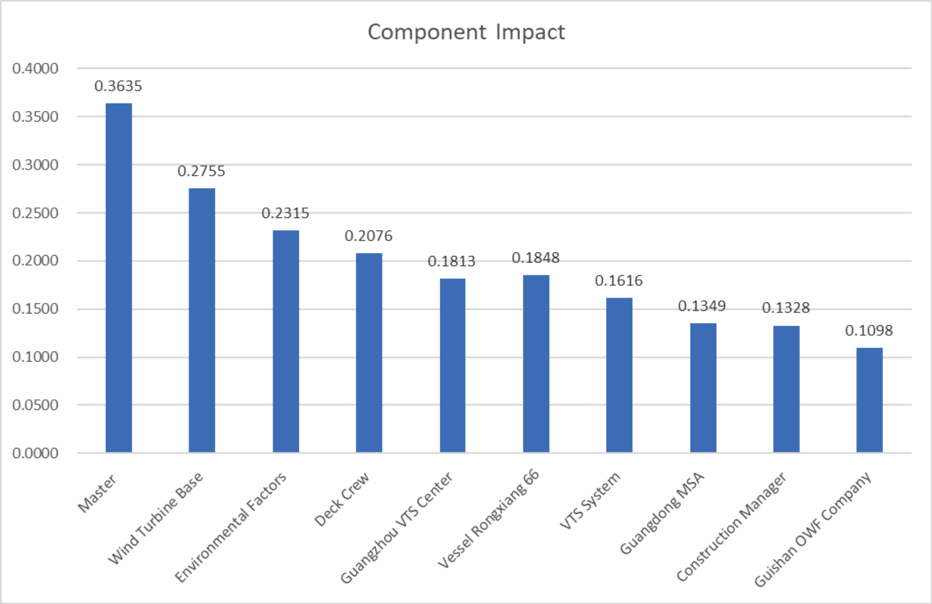


Figure 7. Rank of the component impact.

However, components related to OWF and MSA, which have no fault according to the accident report, play a significant role in the accident impact in this study. For instance, the components related to wind farms, such as wind turbine base, construction managers, and operating companies, played critical roles in this VTA accident, but the accident investigation report did not conduct an in-depth analysis of them, only introducing basic information about the wind farm. The results of this analysis confirm the core philosophy of Leveson’s design of STAMP and CAST, which is not to pursue accountability but to improve system safety. This brings more enlightenment in that the dynamic process and safety of the VTA system need to be analyzed from a higher level.

3.3. *Stage 3: Macroanalysis*

In Stage 3, the system deficiencies in the VTA accident are identified, and improvement recommendations are proposed.

3.3.1. System Deficiency Identification

At the macro level, both internal and external factors of the system will affect each system component [31]. Among them, the internal factors from the surface to the core of a system can be divided into four levels, which are communication and coordination, the safety information system, safety management, and safety culture, as shown in Figure 8. These internal factors are influenced layer by layer. Meanwhile, external factors, including the economy, policy, and environment, are independent of each other and simultaneously generate impact to the system, leading to the occurrence of system deficiencies.

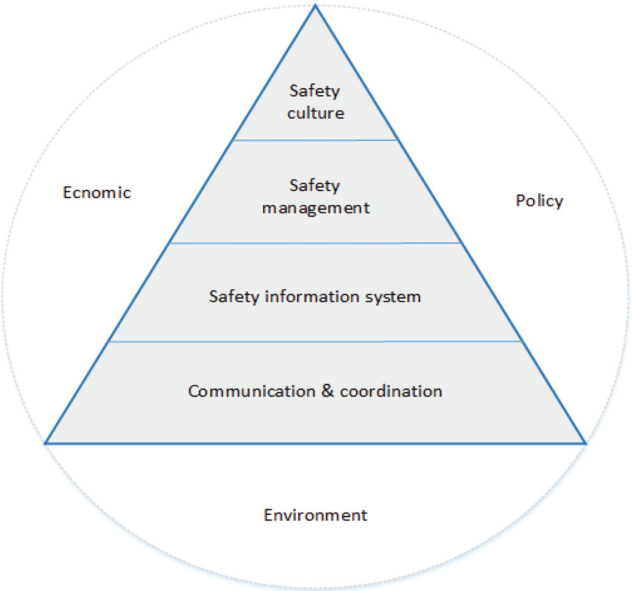


Figure 8. Overall system safety architecture on the macro scale.

Through a CAST-CN analysis of the Rongxiang 66 accident, system defects can be identified based on the above aspects. The limited system defects are explained in the accident report. This study will be further improved or corrected.

1. Internal factors

Communication and coordination: In this case, Figure 5 shows the physical communication or sensing links established at four system levels based on communication and sensing devices, such as VHF, AIS, radar, and CCTV, with the following dotted lines:

NUX (NXU): Sensing link between vessels and VTS system.
NTU (NUT): Communication link between the VTS system and VHF on the vessel.
NOP (NPO): Sensing link between shipborne navigation system and turbine.
NUO (NOU): Sensing link between VTS system and turbine.

Ideally, these links should remain unblocked. Communication between the VHF and VTS systems on board is normal, and Rongxiang 66 can also be found by the VTS system in the accident report; however, there is no reference as to whether the VTS and on-board auxiliary navigation systems can detect wind turbines in wind farms. However, in this accident, the master or any other crew did not perceive the existence of any facility in the OWF through any measure prior to the appearance of the wind turbine in the field of vision, and the VTS center did not indicate the existence of the OWF. Therefore, it can be inferred that the two links of NOP and NUO are abnormal. The abnormal link is probably caused by either no AIS terminal being installed in the infrastructure of the wind farm or the AIS terminal being incorrectly used or faulty after installation, which means that the vessel or VTS system fails to receive AIS signals. It is also possible that the inadequate precision of the VTS system's radar and ship-borne radar or the existence of blind areas leads to the inability of vessels or the VTS system to actively detect wind farm infrastructure.

Hence, at the system level, effective communication and coordination between the wind farm, ship, and VTS center are missing, which is one of the most significant causes of accidents but is not analyzed in the accident report.

Safety information system: A safety information system is an important human–machine interaction intermediary in a complex system. In this case, both shipping companies and offshore wind farm operators have safety information systems, but there is very little information sharing between these systems. As can be seen from Figure 5, there is no communication channel between them, especially between the Ministry of Natural Resources, which is the top management department of the offshore wind farm company, and the Ministry of Transport, which is the top management department of the shipping company. Shipping companies lack clarity about the construction of offshore wind farms in operating waters and it is difficult for offshore wind farm operators to master the dynamics of nearby vessels. Therefore, there are problems with the safety information system throughout the system. This reason is not analyzed in the accident investigation report.

Safety management: Effective safety management is a guarantee for the implementation of a system safety operation mechanism. In this case, the insufficient guidance of a shipping company regarding the anchorage of a vessel's platform is an obvious safety management problem. The inadequate training of the crew results in various erroneous operations, such as the life raft being blown away by the wind after being released, which also reflects the insufficient safety management of the shipping company. In addition, the location of the Guishan offshore wind farm is less than two miles from the pilot anchorage and five miles from the main channel, which is the busy traffic area in the Pearl River Estuary. The wind power infrastructure is likely to affect navigation vessels or interfere with VTS, CCTV, and other systems of the Maritime Bureau. The problems existing in the site selection also reflect the lack of scientific and accurate demonstration by the energy and transportation departments. Finally, the design of structural toughness, anti-collision facilities, and anchor chain strength of the vessel Rongxiang 66 may not match the power of the super typhoon Hato, which leads to the low fault tolerance rate of the driver once the ship falls into this extreme environment. However, the above problems are safety management problems at the whole scale of complex systems. Although there are some safety management suggestions in the accident report, they are not all discovered, and the deeper reasons are not analyzed.

Safety culture: Safety culture is the core of safety management. Specifically, the defectiveness of safety awareness of some enterprise managers will lead to an unsafe cultural ecology of an entire organization. Leveson lists five elements of a safety culture in the CAST manual: Culture of Risk Acceptance, Culture of Denial, Culture of Compliance, Paperwork Culture, and Culture of Swagger. Through our preliminary analysis, these five unhealthy safety cultures have been reflected to some extent in this case. For example, the safety culture of shipping companies is defective. First of all, the precondition for the accident was extreme weather. Before the typhoon approached, the shipping company did not promptly let ship Rongxiang 66 find a safe harbor to anchor but rather let it wait for work in the harbor area. Secondly, when the vessel left the harbor in search of anchorage platforms under the instructions of the VTS center, the shipping company only sent some typhoon information searched on the internet and the company's defense documents (defective). These aspects demonstrate the company's culture of Risk Acceptance, which values productivity while ignoring safety issues. Similarly, in OWF companies, the lack of site selection considerations, based more on the perspective of their operating interests, does not establish adequate safety awareness systems; it only meets the basic government requirements. These also reflect the existence of the culture of Compliance, Paperwork Culture, and other unsafe cultures in the OWF companies.

2. External factors

Additionally, there are three external factors affecting the system: economy, policy, and environment. These external factors are inherently uncertain and can influence a complex system to produce dynamic changes.

Environmental factors: Uncertainty of the environment is one of the main factors of navigation risk of vessels. The system analyzed in this VTA case is severely affected by environmental uncertainties. The factors such as wind, waves, currents, and visibility can have impacts on the stability of ships and the decision-making ability of crew members. These environmental factors, which are rapidly changeable in extreme weather such as typhoon Hato, are difficult to predict accurately. In addition, the path of the typhoon may reroute, and the impact range may also enlarge over time, which will challenge the prevention plan of the master and ship company. Therefore, if the system resilience is not enough to deal with the uncertainty of environment, it is necessary to keep as far away as possible from waters with typhoon activity during the actual voyage and to take relevant measures as early as possible.

Economic factor: In recent years, the revival of the Chinese shipping industry has made maritime vessel traffic busier. To obtain high profits, many shipping companies have chosen to reduce the safety standards of production and operation. These companies can reduce the cost of investment in safety, and they can loosen production and make it develop quickly without restrictions. This has reduced the requirements for ship quality and crew quality, resulting in higher safety risks.

Policy factor: The offshore wind power subsidy policy issued by the Chinese government lasts until 2022, so a large number of offshore wind farms have been constructed with great haste in recent years, which has resulted in a possible lack of safety assessment from site selection to design of facilities. The guiding force of this policy is intended to help accelerate the development of China's offshore wind power industry. However, this policy has also generated a reacting force that leads OWF companies develop wind power resources by all means covered and has hidden some safety hazards.

3.3.2. Improvement Recommendations

Based on the results of the above analysis, we provide the following system improvement recommendations.

1. Overall planning of marine traffic resources

Conflicts concerning the use of marine resources between marine traffic and wind power generation are one of the primary causes of VTA accidents, and require macro and long-term planning. At the stage of planning and site selection for offshore wind farms, relevant port, shipping, and maritime institutions should intervene in advance, strengthen coordination with development and reform, energy, natural resources, and other departments, and actively participate in the formulation and revision of territorial and spatial planning, such as marine functional zoning, to ensure marine transportation resources.

2. Establishing the OWF traffic safety management coordination mechanism

Following the Chinese safety-relevant laws, regulations, and the division of responsibilities, the transportation department should establish and improve the cooperation mechanism of joint law enforcement, supervision, and management with other competent departments in the industry, clarify the government regulatory responsibilities of the energy and transportation departments, strengthen the safety supervision of the offshore wind farms throughout their life cycle, and construct a dual prevention mechanism for hierarchical risk management and control of OWF safety and the investigation and treatment of potential accidents.

3. Simultaneous construction of sufficient navigational safety facilities

In conformity with relevant Chinese laws and rules on work safety, safety facilities for production, operation, and construction projects must be designed, constructed, and operated simultaneously with the main works. Because of the problems existing in the traffic safety management in OWF waters, the owner of an OWF should strengthen the construction of early-warning safety facilities and carry out special research on the impact of wind farm construction on offshore regulatory facilities. Relying on the infrastructure of an OWF, radar, surveillance cameras, and other sensing equipment for vessels should be constructed to improve the coverage of the maritime safety supervision system.

4. Strengthening the construction of maritime supervision and rescue capacity

An offshore wind farm is usually located in an unshielded sea area rich in wind energy. The navigation environment is relatively harsh, and the requirements for the ability for regulatory search and rescue equipment are high. It is necessary to increase the configuration scale of rescue aircraft and improve equipment performance.

4. Discussion

The innovation of this study mainly lies in applying the method of system theory analysis of accidents to special cases of allisions between vessels and wind turbines in OWF waters, which has been rarely performed in previous research. Second, we optimize the CAST method in STAMP and divide the analysis process into three stages: initialization, microanalysis, and macroanalysis, making the analysis clearer, more targeted, and easier to combine with other methods. In addition, to make up for the shortcomings of the STAMP method in quantitative analysis, we integrate the analysis process of complex networks and establish a network model and matrix based on the HSCS constructed by STAMP. We analyze the PR, CC, BC, NI, and CI of each component in the VT network, and the results meet expectations.

The practical implication of this study is based on an accident in the waters of an offshore wind farm. The analysis results obtained through optimization methods are consistent with the accident report, confirming the scientific nature of the method itself, and can be used as a tool for subsequent analysis of such systems. In addition, other factors not analyzed in the accident report are identified, especially regarding communication perception, safety management, and safety culture in the macroanalysis process, as well as the contribution of external economic and policy factors to the occurrence of accidents. Therefore, the analysis method discussed in this article can serve as a powerful supplement to accident investigation, making the results of an accident investigation more scientific and reversing the long-standing responsibility-oriented analysis approach, providing support for the true improvement of system safety.

Admittedly, there is still considerable scope for improvement in this study. For example, there could be a deeper understanding of the case in terms of data acquisition. Relying solely on accident investigation reports and some data information searched online cannot accurately restore the appearance of the accident itself. If the real-time data from vessel tracking systems, weather monitoring, and offshore infrastructure sensors could be integrated into the analysis, the accuracy and timeliness will be enhanced significantly. Future research could explore methods to collect and analyze such data to improve the understanding of offshore wind farm water transportation system safety. Secondly, the method of complex networks only analyzes some basic indicators, which can be further applied to other analysis methods such as the percolation on complex networks.

5. Conclusions

Given the stimulus of the subsidy policy of the Chinese government, offshore wind power has grown at an incredible speed. Many dynamic complex systems composed of vessels and OWF have been formed in a short time, which has caused new navigational safety problems. Due to the characteristics of rare accident cases, incomplete accident data, and dynamic nonlinearity, it is difficult to use traditional data statistics and event chain analysis methods to find the critical factors of a VTA accident. In this study, CAST, in the STAMP family, is utilized, and the complex network theory and method are combined to analyze the accident case of Rongxiang 66 colliding with a wind turbine qualitatively and quantitatively for the macro and micro levels of the system. The main contributions of this study are as follows:

- A typical VTA accident between a vessel and a wind turbine in the navigation system of offshore wind farm waters is systematically analyzed, which establishes the effectiveness of the STAMP model in solving this kind of complex system. In addition, the model design of the safety control structure is improved, and the CAST-CN analysis model is constructed based on the CN, which creatively compensates for the blanks in the quantitative analysis of the cast, and enhances the scientific quality and accuracy of the accident analysis.

- The combined macro and micro analyses of accidents reveal some key factors not mentioned in the accident report, such as maritime traffic management and OWF operation management, which are of great significance to improve navigational safety.

Nevertheless, the CAST-CN model has great potential for improvement; for example, it can be adapted for various types of accidents and network analysis methods. The analysis of different types of offshore wind farm water accidents based on the CAST-CN model will help maritime departments and relevant research organizations gain a comprehensive understanding of the safety problems and key factors of offshore wind farm water transportation system so as to improve the safety of the system. Future research could delve deeper into understanding the perspectives and roles of various stakeholders involved in these systems, such as vessel operators, wind farm operators, regulatory authorities, and coastal communities. This would allow for a more comprehensive analysis of the factors influencing safety and the development of targeted recommendations for each stakeholder group.

Author Contributions: Conceptualization, K.Y. and Y.W.; data curation, K.Y. and W.W.; investigation, K.Y., C.Q. and B.C.; writing—original draft, K.Y.; methodology, K.Y. and Y.W.; writing—review and editing, K.Y. and Y.W.; supervision, C.Q. and L.J.; visualization, W.W. and B.C.; funding acquisition: Y.W. and L.J. All authors have read and agreed to the published version of the manuscript.

Funding: The research described in this paper is supported by the State Key Laboratory of Advanced Rail Autonomous Operation under grant no. [RCS2021ZZ003].

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors thank the Transport Planning and Research Institute for providing data and experimental environment, and Beijing Jiaotong University for providing financial support through the State Key Laboratory of Advanced Rail Autonomous Operation under grant no. [RCS2021ZZ003].

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Component Analysis Based on HSCS-Marked

Table A1. Component analysis results.

Component	Safety Responsibility/Basic Information	Inadequate Control/Feedback Actions	Context	Mental/Process Model Flaws
Master (Captain)	Superior responsibility for safe ship operation and implementation of the safety management system onboard.	CAB1. Issue an order to proceed to the Guishan Pilot Anchorage for typhoon protection. CAB2. Issue the command to use the rudder to withstand the wind after the vessel loses anchor. CAB3. After discovering the wind power base, order OOW to report the danger to the shipping company. CAC1. Command the third officer to strengthen listening to the typhoon information released by VTS. CAC2. After discovering the wind power base, order the third officer to report the danger to Guangzhou VTS center. CAD3. After the anchor pulling occurs, when the wind reaches levels 12–13 and the wave height is 6 m, the command to add the right anchor is issued. FAE2. Deployment of typhoon prevention does not fully meet the requirements of the plan.	1. Encounter extreme conditions—Typhoon Hato; 2. Have experience as a master; 3. Have a certificate of ship management; 4. Recently, the master received safety training from the company; 5. Time lag.	1. Insufficient safety awareness: a comprehensive typhoon prevention plan was not formulated as early as possible, and even waiting for operations in the open waters of the wharf before the typhoon hit; when the VTS center reminded the ship to come and prepare for typhoon prevention, the wrong typhoon prevention anchorage was selected (located in the center of the typhoon); 2. Poor perception: lack of mental preparation for the impact of typhoons and the uncertainty that may change their path; the situation of the surrounding waters is not fully understood, and it is not clear that there are wind farms. 3. Poor emergency decision-making ability. After the anchor slip occurred, there was no choice but to add dual anchors on time, and only applicable to the rudder power against the wind. When the wind reaches 12–13, it is too late to require sailors to add dual anchors. 4. Poor communication skills, failing to report to the company and VTS on time after a breakout occurred, and not reporting until the risk of collision with the wind turbine was discovered.

Table A1. *Cont.*

Component	Safety Responsibility/Basic Information	Inadequate Control/Feedback Actions	Context	Mental/Process Model Flaws
Chief mate/OOW	The chief mate is second in command to the Master, a head of the deck department and, customarily, an officer in charge of the vessel's cargo and deck crew. Officer of the watch(OOW) steers the vessel/keeps watch on the bridge.	CBP1. It is unknown whether to use radar/AIS or other means to observe wind turbine facilities in nearby waters.	1. Encounter extreme conditions—Typhoon Hato; 2. Have shipmate experience; 3. Have a competency certificate for crew members; 4. Recently received safety training from the company.	1. Poor perception: lack of mental preparation for the impact of typhoons and the uncertainty that may change their path; the situation of the surrounding waters is not fully understood, and it is not clear that there are wind farms.
Third mate	The third mate is a watchstander and, customarily, the vessel's safety officer, focusing on firefighting equipment, lifeboats, and various other emergency communication systems.	FCA1. Failing to report the VTS broadcast typhoon prevention suggestions to the master as soon as possible.	1. Encounter extreme conditions—Typhoon Hato; 2. Have shipmate experience; 3. Have a competency certificate for crew members; 4. Recently received safety training from the company; 5. Time lag.	1. Lack of responsibility and safety awareness: when listening to the VTS broadcast, because the VTS did not mention Rongxiang 66, it did not immediately report the typhoon prevention suggestions to the master.
Seaman/Deck Crew	Crewmembers looking after safety in accommodation, public areas, cargo, and other areas. Guarantee the safety of navigation and the maintenance of vessels.	FDA 2. Due to the strong wind and waves, the master's order to add the right anchor was not implemented. FDA 3. The command to release the lifeboat was not effectively implemented due to the strong wind and waves. CDR1. The lifeboat was not released correctly, resulting in the lifeboat being blown away by the wind. CDS1. After the vessel anchored, due to the strong wind and waves, the right anchor was not added.	1. Encounter extreme conditions—Typhoon Hato; 2. Have crew experience; 3. Have a competency certificate for crew members; 4. Experienced safety training from the company.	Inadequate safety awareness and emergency skills: in the case of heavy winds and waves, the deployment of lifeboats failed.

Table A1. *Cont.*

Component	Safety Responsibility/Basic Information	Inadequate Control/Feedback Actions	Context	Mental/Process Model Flaws
Ship Company	Responsible for ship safety management, organizing regular safety risk assessment and hidden danger inspection for vessels, and regular safety training for crew members.	FEF2. The content of the safety management system document "Instructions for Ship Anti Taiwan" is not standardized.	1. Obtain typhoon information through the Internet; 2. Time lag.	There are loopholes in the safety management system: the formulated typhoon prevention documents do not stipulate that the master should study and deploy the vessel's typhoon prevention work, and do not stipulate that the vessel closely tracks the typhoon dynamics; there is no regulation that companies should also track typhoon dynamics and deploy guidance for ship typhoon prevention work.
Guangzhou MSA/ VTS Center	Unified management of water traffic safety and prevention of ship pollution, management of navigation order, and navigation environment, and is responsible for the management of ship and marine facilities inspection industry, as well as ship seaworthiness and ship technology management, etc.	CJC2. No anchorage suitable for typhoon prevention is recommended.	1. Encounter extreme conditions—Typhoon Hato; 2. Have 24 h duty personnel; 3. Keep a close track of typhoon dynamics.	
Energy Administration	Departments that manage natural energy, responsible for the approval of offshore wind farm site selection and construction.	CLM1. Review of offshore wind farm sites and construction plans.	Unknown.	1. The review of offshore wind farm construction plans was not rigorous.

Table A1. *Cont.*

Component	Safety Responsibility/Basic Information	Inadequate Control/Feedback Actions	Context	Mental/Process Model Flaws
Guishan OWF Company	Responsible for the construction and operation safety management of offshore wind farms.	CMN1. Manage and guide the construction manager of offshore wind farms (inadequate).	1. Obtained a sea area use right certificate; 2. The wind farm built is only 1.5 nautical miles from the east anchorage, only 0.88 nautical miles from the southern anchorage, and only 1.36 nautical miles from the high-speed passenger ship route; 3. LED light warning signs are set according to the Navigation Safety Assessment Report.	1. There are defects in the site selection scheme for offshore wind farms, resulting in significant conflicts with traffic and sea use. 2. Lack of safety awareness. Only the LED warning means recommended in the Navigation Safety Assessment Report are used, without any other supplementary means. When encountering severe weather, this means is insufficient.
Construction Manager	The main responsible person for work safety during the construction of offshore wind farms, fully responsible for the construction safety of wind farms.	CNO1. Ensuring the safe construction and operation of wind turbines (insufficient).	1. Encounter extreme conditions—Typhoon Hato.	1. Lack of safety awareness. Increase safety protection for wind farms before responding to typhoons in advance.
Wind Turbine Base	The base load-bearing structure used to support the wind turbine.	FON1. During the construction process, there are no communication links and technical means to provide feedback; NOP1. Whether it can be detected by radar or AIS means is unknown; NOU1. Whether it can be detected by radar or AIS means is unknown.	1. LED light warning signs are installed.	1. Lack of AIS terminals, unable to be perceived; 2. Lack of ship collision prevention facilities; 3. Lack of early warning systems such as electronic fences.
Navigational Aids	Using radar, ECDIS, AIS, and other techniques, helps the crew to perceive the surrounding obstacles and ships and provide decision-making guidance equipment for navigation.	FPB1. It is unknown whether the information on surrounding ships/offshore structures can be displayed normally; NPO1. It is unknown whether the fan base can be detected.	1. Normal operation.	Lack of a ship wind turbine collision warning model: it is not possible to give an alarm when a ship approaches the waters of an offshore wind farm.

Table A1. *Cont.*

Component	Safety Responsibility/Basic Information	Inadequate Control/Feedback Actions	Context	Mental/Process Model Flaws
Main Engine/Rudder Engine	Provide propulsion power to the vessel and drive the propeller. Provide the vessel with slewing torque to keep the vessel on the course or to turn.	CQX1. Provide ship power (insufficient).	1. Encounter extreme conditions—Typhoon Hato.	
Life Raft	Special boat on board for rescuing people overboard or evacuating people when the vessel is in distress.	FRD1. Loss of control after release into the sea.	1. Encounter extreme conditions—Typhoon Hato.	
Mooring System	Equipment used to fix the vessel when it is moored, including anchor, anchor chain, and anchor machine.	CSX1. Provide anchoring control (insufficient).	1. Encounter extreme conditions—Typhoon Hato.	
VTS System	The system used by the MSA to manage the order of ship navigation.	NUO1. It is unknown whether the fan base can be detected; NUX2. Failed to provide early warning of vessel collision with wind turbines; FXQ1. Not fully controlled by the rudder system; FXS1. Anchor dragging occurs.	1. Normal operation.	1. Lack of intelligent aided decision-making model: unable to provide detailed suggestions on typhoon prevention schemes for each ship; 2. Lack of a ship wind turbine collision warning model: it is not possible to give an alarm when a ship approaches the waters of an offshore wind farm.
Rongxiang 66	The vessel that sank in this case, a dry cargo ship, with a total length of 98.6 m, a total tonnage of 2879 tons, a deadweight of 5385 tons, and a main engine power of 1545 kW, was completed in 2004.		1. Encounter extreme conditions—Typhoon Hato.	

Table A1. *Cont.*

Component	Safety Responsibility/Basic Information	Inadequate Control/Feedback Actions	Context	Mental/Process Model Flaws
Environmental Factors	In this case, the environmental factors are mainly the strong winds and waves caused by typhoon Hato, with the maximum wind force reaching 15, the wave height reaching 10 m, and the moving path changing several times. In addition, the anchorage of the Rongxiang 66 anchor is a pilot anchorage, and the seabed sediment cannot provide sufficient anchor grip.	IYD1. When the wind force of Typhoon Hato reached Force 12, the crew was unable to carry out anchoring operations on the deck; IYR1. The strong wind blew away the rescue boat; IYS1. The seabed bottom material cannot provide sufficient anchor grip; IYV1. Severe weather affects rescue speed; IYX1. Heavy winds and waves caused the vessel to lose control.		

Appendix B. The Weighted Matrix of V-T Network

Table A2. V-T matrix with edge weights.

ID	*A*	*B*	*C*	*D*	*E*	*F*	*G*	*H*	*I*	*J*	*K*	*L*	*M*	*N*	*O*	*P*	*Q*	*R*	*S*	*T*	*U*	*V*	*W*	*X*	*Y*
A		0.25	0.333	0.5	0.5					1															
B	1															0.5	1								
C	0.5																			1					
D	0.333																	0.5	0.5						
E	1					0.5																			
F					1		1																		
G						1		1	1	1															
H							1																1		
I							1															1			
J	0.5						1														1				
K												1	1												
L											1		0.5												
M											1	1		0.5											
N													1		0.5										
O														0.5		0.5					0.5			0.5	
P		0.5													0.5										
Q		1																						0.5	
R				0.5																					
S				0.5																				0.5	
T			1																		1				
U										1					0.5					1				1	
V									1																
W								1																	
X															0.5		0.5		0.5		1				
Y				0.5														0.5	0.5			0.5	0.5	0.5	

Appendix C. Network Analysis Result of CAST-CN

Table A3. Network analysis results (*PR*, *CC*, *BC*, *NI*).

ID	Label	*PR*	*CC*	*BC*	*NI*
A	Master	0.0745	0.3710	0.2681	0.086
B	Chief mate	0.0452	0.3382	0.0857	0.037
C	Third mate	0.0322	0.3067	0.0199	0.018
D	Deck crew	0.0526	0.3151	0.1172	0.045
E	Ship company	0.0335	0.2987	0.0314	0.021
F	Ministry of Transport	0.0349	0.2840	0.0181	0.017
G	Guangdong MSA	0.0690	0.3333	0.3219	0.098
H	South Sea Rescue Bureau	0.0415	0.2584	0.0812	0.033
I	Hong Kong Flight Rescue Team	0.0415	0.2584	0.0812	0.033
J	Guangzhou VTS Center	0.0455	0.3966	0.3496	0.105
K	Ministry of Natural Resources	0.0346	0.1933	0.0000	0.010
L	Energy Administration	0.0346	0.1933	0.0000	0.010
M	Guishan OWF company	0.0491	0.2347	0.1558	0.051
N	Construction manager	0.0321	0.2875	0.2228	0.068
O	Wind turbine base	0.0576	0.3594	0.3110	0.095
P	Navigational aids	0.0310	0.3067	0.0374	0.023
Q	Main engine/rudder engine	0.0314	0.2949	0.0100	0.015
R	Life raft	0.0218	0.2421	0.0000	0.010
S	Mooring system	0.0343	0.2987	0.0531	0.027
T	VHF	0.0318	0.3108	0.0254	0.020
U	VTS system	0.0572	0.3898	0.2859	0.090
V	Helicopters	0.0245	0.2072	0.0033	0.010
W	Tugboats	0.0245	0.2072	0.0033	0.010
X	Vessel Rongxiang 66	0.0592	0.3538	0.1443	0.054
Y	Environmental factors	0.0060	0.4000	0.0000	0.014

References

1. Lee, J.; Zhao, F. *Global Wind Report | GWEC*; Global Wind Energy Council: Brussels, Belgium, 2021; Volume 75.
2. DeCastro, M.; Salvador, S.; Gómez-Gesteira, M.; Costoya, X.; Carvalho, D.; Sanz-Larruga, F.J.; Gimeno, L. Europe, China and the United States: Three different approaches to the development of offshore wind energy. *Renew. Sustain. Energy Rev.* **2019**, *109*, 55–70. [CrossRef]
3. Mujeeb-Ahmed, M.P.; Seo, J.K.; Paik, J.K. Probabilistic approach for collision risk analysis of powered vessel with offshore platforms. *Ocean Eng.* **2018**, *151*, 206–221. [CrossRef]
4. Xie, L.; Xue, S.; Zhang, J.; Zhang, M.; Tian, W.; Haugen, S. A path planning approach based on multi-direction A* algorithm for ships navigating within wind farm waters. *Ocean Eng.* **2019**, *184*, 311–322. [CrossRef]
5. Moulas, D.; Shafiee, M.; Mehmanparast, A. Damage analysis of ship collisions with offshore wind turbine foundations. *Ocean Eng.* **2017**, *143*, 149–162. [CrossRef]
6. Chen, X.; Wu, S.; Shi, C.; Huang, Y.; Yang, Y.; Ke, R.; Zhao, J. Sensing Data Supported Traffic Flow Prediction via Denoising Schemes and ANN: A Comparison. *IEEE Sens. J.* **2020**, *20*, 14317–14328. [CrossRef]
7. Chen, X.; Wang, Z.; Hua, Q.; Shang, W.; Luo, Q.; Yu, K. AI-Empowered Speed Extraction via Port-Like Videos for Vehicular Trajectory Analysis. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 4541–4552. [CrossRef]
8. John, A.; Osue, U.J. Collision risk modelling of supply vessels and offshore platforms under uncertainty. *J. Navig.* **2017**, *70*, 870–886. [CrossRef]
9. Kim, S.J.; Seo, J.K.; Ma, K.Y.; Park, J.S. Methodology for collision-frequency analysis of wind-turbine installation vessels. *Ships Offshore Struct.* **2021**, *16*, 423–439. [CrossRef]
10. Hulme, A.; Stanton, N.A.; Walker, G.H.; Waterson, P.; Salmon, P.M. Are accident analysis methods fit for purpose? Testing the criterion-referenced concurrent validity of AcciMap, STAMP-CAST and AcciNet. *Saf. Sci.* **2021**, *144*, 105454. [CrossRef]
11. Lv, P.L.; Zhen, R.; Shao, Z. A Novel Method for Navigational Risk Assessment in Wind Farm Waters Based on the Fuzzy Inference System. *Math. Probl. Eng.* **2021**, *2021*, 4588333. [CrossRef]
12. Zhen, R.; Lv, P.L.; Shi, Z.Q.; Chen, G.Q. A novel fuzzy multi-factor navigational risk assessment method for ship route optimization in costal offshore wind farm waters. *Ocean Coast. Manag.* **2023**, *232*, 106428. [CrossRef]
13. Rawson, A.; Brito, M. Assessing the validity of navigation risk assessments: A study of offshore wind farms in the UK. *Ocean Coast. Manag.* **2022**, *219*, 106078. [CrossRef]
14. Tsai, Y.M.; Lin, C.Y. Investigation on improving strategies for navigation safety in the offshore wind farm in Taiwan strait. *J. Mar. Sci. Eng.* **2021**, *9*, 1448. [CrossRef]
15. Mou, J.; Jia, X.; Chen, P. Research on Operation Safety of Offshore Wind Farms. *J. Mar. Sci. Eng.* **2021**, *9*, 881. [CrossRef]
16. Yu, Q.; Liu, K.; Yang, Z.S.; Wang, H.; Yang, Z.L. Geometrical risk evaluation of the collisions between ships and offshore installations using rule-based Bayesian reasoning. *Reliab. Eng. Syst. Saf.* **2021**, *210*, 107474. [CrossRef]
17. Yu, Q.; Teixeira, Â.P.; Liu, K.; Rong, H.; Guedes, S.C. An integrated dynamic ship risk model based on Bayesian Networks and Evidential Reasoning. *Reliab. Eng. Syst. Saf.* **2021**, *216*, 107993. [CrossRef]
18. Rawson, A.; Brito, M. A critique of the use of domain analysis for spatial collision risk assessment. *Ocean Eng.* **2021**, *219*, 108259. [CrossRef]
19. Dai, L.; Ehlers, S.; Rausand, M.; Utne, I.B. Risk of collision between service vessels and offshore wind turbines. *Reliab. Eng. Syst. Saf.* **2013**, *109*, 18–31. [CrossRef]
20. Mehdi, R.A.; Baldauf, M.; Deeb, H. A dynamic risk assessment method to address safety of navigation concerns around offshore renewable energy installations. *Proc. Inst. Mech. Eng. Part M J. Eng. Marit. Environ.* **2020**, *234*, 231–244. [CrossRef]
21. Mehdi, R.A.; Overloop, J.V.; Nilsson, H. Improving the coexistence of offshore wind farms and shipping: An international comparison of navigational risk assessment processes. *WMU J. Marit. Aff.* **2018**, *17*, 397–434. [CrossRef]
22. Bela, A.; Sourne, H.L.; Buldgen, L.; Rigo, P. Ship collision analysis on offshore wind turbine monopile foundations. *Mar. Struct.* **2017**, *51*, 220–241. [CrossRef]
23. Puisa, R.; Lin, L.; Bolbot, V.; Vassalos, D. Unravelling causal factors of maritime incidents and accidents. *Saf. Sci.* **2018**, *110*, 124–141. [CrossRef]
24. Mentes, A.; Turan, O. A new resilient risk management model for Offshore Wind Turbine maintenance. *Saf. Sci.* **2019**, *119*, 360–374. [CrossRef]
25. Yu, Q.; Liu, K.; Chang, C.H.; Yang, Z. Realising advanced risk assessment of vessel traffic flows near offshore wind farms. *Reliab. Eng. Syst. Saf.* **2020**, *203*, 107086. [CrossRef]
26. Yu, Q.; Liu, K.; Teixeira, A.P.; Soares, C.G. Assessment of the Influence of Offshore Wind Farms on Ship Traffic Flow Based on AIS Data. *J. Navig.* **2020**, *73*, 131–148. [CrossRef]
27. Naus, K.; Banaszak, K.; Szymak, P. The methodology for assessing the impact of offshore wind farms on navigation, based on the automatic identification system historical data. *Energies* **2021**, *14*, 6559. [CrossRef]
28. Yan, K.; Wang, Y.; Jia, L.; Wang, W.; Liu, S.; Geng, Y. A content-aware corpus-based model for analysis of marine accidents. *Accid. Anal. Prev.* **2023**, *184*, 106991. [CrossRef] [PubMed]
29. Mogles, N.; Padget, J.; Bosse, T. Systemic approaches to incident analysis in aviation: Comparison of STAMP, agent-based modelling and institutions. *Saf. Sci.* **2018**, *108*, 59–71. [CrossRef]
30. Leveson, N. A New Accident Model for Engineering Safer Systems. *Saf. Sci.* **2004**, *42*, 237–270. [CrossRef]

31. Kopsidas, A.; Kepaptsoglou, K. Identification of critical stations in a Metro System: A substitute complex network analysis. *Phys. A Stat. Mech. Its Appl.* **2022**, *596*, 127123. [CrossRef]
32. China Maritime Safety Administration. *Vessel Rongxiang 66 and Wind Turbine Allision Accident Investigation Report*; China Maritime Safety Administration: Beijing, China, 2017; Volume 87, pp. 149–200.
33. Ceylan, B.; Akyuz, E.; Arslan, O. Systems-Theoretic Accident Model and Processes (STAMP) approach to analyse socio-technical systems of ship allision in narrow waters. *Ocean. Eng.* **2021**, *239*, 109804. [CrossRef]
34. Leveson, N. *CAST HANDBOOK: How to Learn More from Incidents and Accidents*; MIT Partnership for a Systems Approach to Safety (PSAS): Cambridge, MA, USA, 2019; pp. 1–148.
35. Zhang, Y.; Dong, C.; Guo, W.; Dai, J.; Zhao, Z. Systems theoretic accident model and process (STAMP): A literature review. *Saf. Sci.* **2021**, 105596. [CrossRef]
36. Kwon, Y.; Leveson, N. System Theoretic Safety Analysis of the Sewol-Ho Ferry Accident in South Korea. *INCOSE Int. Symp.* **2017**, *27*, 461–476. [CrossRef]
37. Shahpari, A.; Khansari, M.; Moeini, A. Vulnerability analysis of power grid with the network science approach based on actual grid characteristics: A case study in Iran. *Phys. A Stat. Mech. Its Appl.* **2019**, *513*, 14–21. [CrossRef]
38. Zhao, C.; Li, N.; Fang, D. Criticality assessment of urban interdependent lifeline systems using a biased PageRank algorithm and a multilayer weighted directed network model. *Int. J. Crit. Infrastruct. Prot.* **2018**, *22*, 100–112. [CrossRef]
39. Tang, B.; Guo, S.; Li, J.; Lu, W. Exploring the Risk Transmission Characteristics Among Unsafe Behaviors Within Urban Railway Construction Accidents. *J. Civ. Eng. Manag.* **2022**, *28*, 443–456. [CrossRef]
40. Du, Z.; Tang, J.; Qi, Y.; Wang, Y.; Han, C.; Yang, Y. Identifying critical nodes in metro network considering topological potential: A case study in Shenzhen city—China. *Phys. A Stat. Mech. its Appl.* **2020**, *539*, 122926. [CrossRef]
41. Maduako, I.; Ebinne, E.; Uzodinma, V.; Okolie, C.; Chiemelu, E. Computing traffic accident high-risk locations using graph analytics. *Spat. Inf. Res.* **2022**, *30*, 497–511. [CrossRef]
42. Fan, B.; Shu, N.; Li, Z.; Li, F. Critical Nodes Identification for Power Grid Based on Electrical Topology and Power Flow Distribution. *IEEE Syst. J.* **2022**, 1–11. [CrossRef]
43. Deng, J.; Liu, S.; Shu, Y.; Hu, Y.; Xie, C.; Zeng, X. Risk evolution and prevention and control strategies of maritime accidents in China's coastal areas based on complex network models. *Ocean Coast. Manag.* **2023**, *237*, 106527. [CrossRef]
44. Wang, W.; Wang, Y.; Wang, G.; Li, M.; Jia, L. Identification of the critical accident causative factors in the urban rail transit system by complex network theory. *Phys. A Stat. Mech. Its Appl.* **2023**, *610*, 128404. [CrossRef]
45. Tang, J.; Li, Z.; Gao, F.; Zong, F. Identifying critical metro stations in multiplex network based on D-S D–S evidence theory. *Phys. A Stat. Mech. Its Appl.* **2021**, *574*, 126018. [CrossRef]
46. Düzgün, H.S.; Leveson, N. Analysis of soma mine disaster using causal analysis based on systems theory (CAST). *Saf. Sci.* **2018**, *110*, 37–57. [CrossRef]
47. Sultana, S.; Okoh, P.; Haugen, S.; Vinnem, J.E. Hazard analysis: Application of STPA to ship-to-ship transfer of LNG. *J. Loss Prev. Process Ind.* **2019**, *60*, 241–252. [CrossRef]
48. Ishimatsu, T.; Leveson, N.; Thomas, J.; Katahira, M.; Miyamoto, Y.; Nakao, H. *Modeling and Hazard Analysis Using Stpa*; Harvard University: Cambridge, MA, USA, 2010.
49. Lü, L.; Chen, D.; Ren, X.L.; Zhang, Q.M.; Zhang, Y.C.; Zhou, T. Vital nodes identification in complex networks. *Phys. Rep.* **2016**, *650*, 1–63. [CrossRef]

Journal of
Marine Science
and Engineering

MDPI

Article

A Ship Tracking and Speed Extraction Framework in Hazy Weather Based on Deep Learning

Zhenzhen Zhou [1], Jiansen Zhao [1,*], Xinqiang Chen [2] and Yanjun Chen [1]

[1] College of Merchant Marine, Shanghai Maritime University, Shanghai 201306, China; 202130110057@stu.shmtu.edu.cn (Z.Z.); 202030110085@stu.shmtu.edu.cn (Y.C.)
[2] Institute of Logistics Science and Engineering, Shanghai Maritime University, Shanghai 201306, China; xqchen@shmtu.edu.cn
* Correspondence: jszhao@shmtu.edu.cn

Abstract: Obtaining ship navigation information from maritime videos can significantly improve maritime supervision efficiency and enable timely safety warnings. Ship detection and tracking are essential technologies for mining video information. However, current research focused on these advanced vision tasks in maritime supervision is not sufficiently comprehensive. Taking into account the application of ship detection and tracking technology, this study proposes a deep learning-based ship speed extraction framework under the haze environment. First, a lightweight convolutional neural network (CNN) is used to remove haze from images. Second, the YOLOv5 algorithm is used to detect ships in dehazed marine images, and a simple online and real-time tracking method with a Deep association metric (Deep SORT) is used to track ships. Then, the ship's displacement in the images is calculated based on the ship's trajectory. Finally, the speed of the ships is estimated by calculating the mapping relationship between the image space and real space. Experiments demonstrate that the method proposed in this study effectively reduces haze interference in maritime videos, thereby enhancing the image quality while extracting the ship's speed. The mean squared error (MSE) for multiple scenes is 0.3 Kn on average. The stable extraction of ship speed from the video achieved in this study holds significant value in further ensuring the safety of ship navigation.

Keywords: ship speed extraction; image dehaze; ship detection; ship tracking

Citation: Zhou, Z.; Zhao, J.; Chen, X.; Chen, Y. A Ship Tracking and Speed Extraction Framework in Hazy Weather Based on Deep Learning. *J. Mar. Sci. Eng.* **2023**, *11*, 1353. https://doi.org/10.3390/jmse11071353

Academic Editor: Claudio Ferrari

Received: 17 May 2023
Revised: 26 June 2023
Accepted: 30 June 2023
Published: 2 July 2023

1. Introduction

Currently, the Automatic Identification System (AIS) serves as the primary platform for exchanging navigation information, including ship speed, between ships and between ships and the shore [1]. However, the rapid growth of the shipping industry has led to an increased number of ships, resulting in AIS signal interference in busy waters. Meanwhile, the system's weak ability to combat data defects, system instability, and environmental interference often causes data delays or losses [2]. Additionally, some ships either lack AIS equipment or turn it off in monitored waters, thereby preventing the maritime supervision department from obtaining timely navigation information [3]. In this situation, both the supervisory authority and ships in the same waters are unable to obtain accurate and timely speed information of other ships, posing a hidden danger to navigation safety. Maritime videos, which provide rich information at a low cost, are widely used in maritime supervision. Techniques such as image processing, target detection, and target tracking are employed to identify obstacles at sea [4] and extract navigation information, such as ship trajectories, from maritime images [5]. These approaches have positive implications for enhancing maritime supervision efficiency and ensuring ship safety.

While several studies have been conducted on ship detection and tracking in maritime images [5–7], the related research has not sufficiently explored the application of ship detection and tracking technology, nor has it fully extracted the navigation information from maritime images. Zhao et al. [8] proposed a ship speed extraction framework based on UAV airborne video. In this study, the advantages of optical image data were fully utilized to realize the visual extraction of ship speed information. However, this study does not consider the influence of complex weather on the accuracy of ship speed extraction and lacks research on processing low-quality marine image data.

In this study, we propose a ship tracking and speed extraction framework based on deep learning under hazy weather conditions. Our approach utilizes cost-effective optical data while considering environmental impacts. We achieve ship speed information extraction from video data using ship tracking algorithms, as illustrated in Figure 1. The contributions of this study can be summarized as follows:

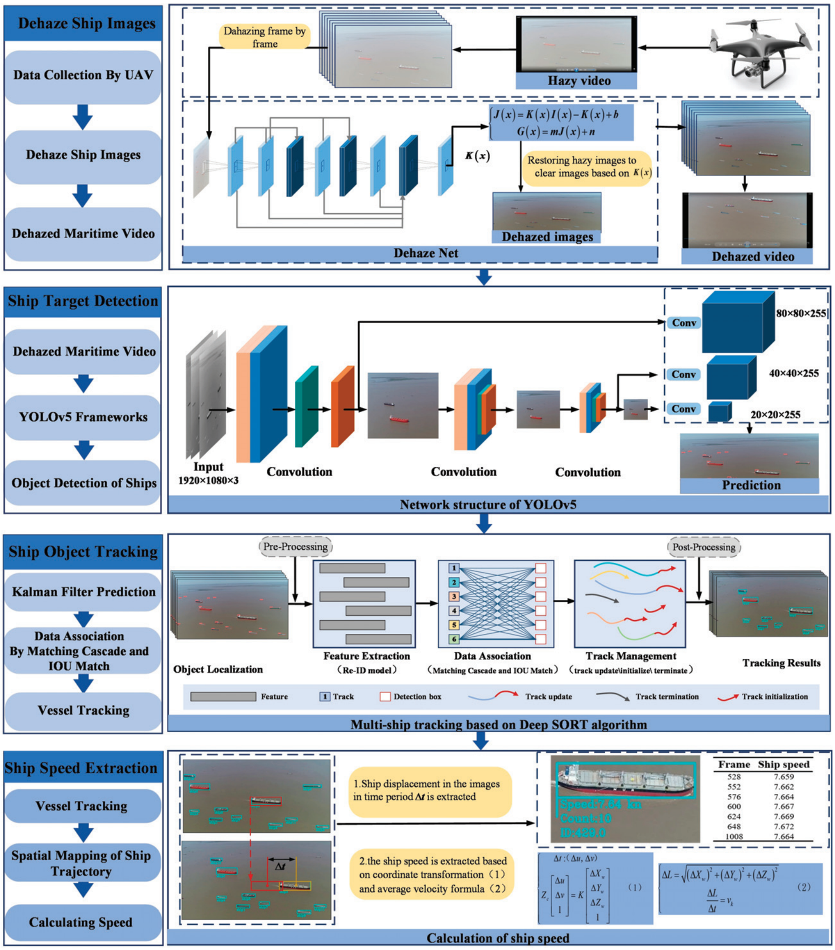


Figure 1. Frame diagram of the method.

- To address the issue of the image becoming dark after haze removal, thereby obscuring the ship's target features, we improved AOD-Net [9] at the pixel level. After haze removal, the mean peak signal-to-noise ratio (PSNR) of multiple maritime scenes reached 23.86, and the mean structural similarity index (SSIM) was 0.96, thus improving the quality of maritime images.
- We extract ship speed from the images based on the image mapping relationship. The average accuracy of ship speed extraction using this framework across multiple scenes is approximately 95%. Furthermore, the mean squared error (MSE) of the speed values

extracted from the dehazed images is approximately 0.3 Kn lower than that extracted from the images before haze removal.

- Provides ideas for the application of advanced vision tasks such as haze removal from maritime haze images and ship tracking in maritime scenarios, improving the efficiency of maritime supervision.

2. Related Work

2.1. Image Haze Removal

Image processing for hazy weather is a significant research direction in the field of computer vision. Haze removal methods based on image enhancement primarily aim to enhance image contrast and highlight image details. These methods include the adaptive histogram equalization method [10], Retinex theory [11], etc. While these methods are simple, easy to implement, and widely applicable, they may lead to loss of details or over-enhancement. He et al. proposed a method of combining the dark channel prior with the atmospheric scattering model for haze removal [12]. The experimental principle of this method is simple and has a good effect on most natural scenes, but it is prone to local coloration or image brightness reduction after removing haze. In recent years, deep learning methods, such as convolutional neural networks (CNN), have been utilized for haze removal, and numerous deep networks have been developed for this purpose [9,13–15]; these haze removal networks have demonstrated improved results in haze removal experiments. However, most of these methods have been applied to land-based scenes, and there is a need for an improved haze removal network specifically tailored for maritime haze videos, considering the differences in sea surface scattering and other imaging characteristics compared to land-based haze. Therefore, this study enhances AOD-Net for maritime haze scenes to more efficiently remove haze from maritime haze images.

2.2. Target Tracking

Current methods of multi-target tracking generally employ the TBD (Tracking-by-Detection) strategy, which involves first detecting the target's position in the image and then establishing associations between frames based on appearance consistency or positional similarity of the same target across frames. In recent years, with the advancement of algorithms such as deep learning, tracking accuracy has been enhanced by utilizing techniques such as neural networks to learn the appearance information of targets across different video frames for precise inter-frame associations [16–18]. However, the study by [19] demonstrates that when different tracking targets share similar appearance features, matching errors in target IDs can occur, making reliance solely on appearance features for inter-frame association unreliable.

Location similarity-based target tracking methods can overcome issues arising from the appearance similarity of tracked targets. Simple online and real-time tracking (SORT) [20] performs data association based on positional similarity and first uses a Kalman filter to predict the position of the track in the next frame and then calculates the Intersection over Union (IoU) between the detected and predicted frames. ByteTrack [21] matches the frames with IoU matching below the threshold twice to improve the tracking performance of the object when it is occluded. Inter-frame matching combining appearance consistency and location similarity can be sufficient to further improve tracking performance [22–24], and Deep SORT [25] uses an independent Re-ID model to extract appearance features from the detected frames to reduce ID matching errors. It is worth stating that the performance of the current multi-target tracking algorithm using the TBD strategy is closely related to the results of the detection model, and the performance of the tracking model can be guaranteed when the detection model reaches high accuracy [20]. In this paper, we adopt Deep SORT [25], a flexible and robust tracking model, after ensuring that yolov5 can detect ships with stable and high accuracy in maritime scenes.

2.3. Techniques for Obtaining Information on Ship Speed

Commonly used technologies for measuring ship speed include AIS [26], radar [27], lasers [28], and video-based speed measurements [29]. The emergence and advancement of the AIS system have provided robust technical support for acquiring ship navigation information [30]. However, as the number of ships at sea continues to increase, AIS signals are prone to interference. Other ships can only obtain ship navigation information [31] if the ship has AIS installed and turned on. Ship speed measurements using laser and radar technologies require specialized and costly equipment. In contrast, marine videos contain a wealth of ship navigation information, which can be easily visualized and processed in real-time. Additionally, visual sensors offer a wide monitoring range and are cost-effective [32], making them ideal for applications in complex marine environments with numerous ships and various influencing factors [33]. With the advancement of visual sensor technology, speed measurement methods based on videos hold promising prospects.

3. Materials and Methods

3.1. Remove Haze in Marine Haze Images Using CNN

The first part of the framework is a lightweight CNN, which is used to remove haze from hazy marine images. To improve the quality of marine images in complex scenes, marine haze images were used to train AOD-Net [9], which can achieve end-to-end dehazing in marine scenes. To avoid the darkening of the maritime images after haze removal by AOD-Net and to solve the problem of not highlighting the structural features of ships in the images after haze removal, Equation (1) was introduced to highlight further the structural features of ships in the images.

$$G(x) = mJ(x) + n \tag{1}$$

where, $J(x)$ is the image before image enhancement, $G(x)$ is the image after image enhancement, m is the gain parameter, and n is the bias parameter, which is used to adjust the contrast and brightness of the marine image to further eliminate the impact of background noise. In this study, the AOD-Net model trained by marine images and can improve the quality of images after haze removal is called e-AOD-Net. The e-AOD-Net uses a CNN to remove haze based on the atmospheric scattering model. The traditional atmospheric scattering model that generates hazy images is described as follows:

$$I(x) = J(x)t(x) + A(1 - t(x)) \tag{2}$$

$$t(x) = e^{-\beta d(x)} \tag{3}$$

In the Equation (2), $I(x)$ is the hazy image, $J(x)$ is the clear image before image enhancement, and $t(x)$ is the medium transmission, describing the light without scattering and transmitted to the visual sensor, which can be expressed by the atmospheric scattering coefficient β and the distance between the field and the visual sensor $d(x)$.

Equations (2) and (3) can be transformed into:

$$J(x) = K(x)I(x) - K(x) + b \tag{4}$$

$$K(x) = \frac{\frac{1}{t(x)}(I(x) - A) + (A - b)}{I(x) - 1} \tag{5}$$

where, b is the deviation value whose default value is 1. Meanwhile, Equation (4) integrates $\frac{1}{t(x)}$ and A into $K(x)$. The e-AOD-Net builds an adaptive depth estimation model based on the physical model of atmospheric scattering and trains the network by minimizing the error between the pixel values of clear and hazy images.

As shown in Figure 2, $I(x)$ is entered into the network to estimate $K(x)$ and then input $K(x)$ into the dehazed image generation module as an adaptive parameter. The function of the $K(x)$ estimation module is to estimate the depth and haze concentration of hazy images. At the same time, Equation (1) is used to reduce the impact of noise on an image. Finally, clear images are synthesized by the multiplication and additional layers, and they can be output directly after haze removal to realize end-to-end haze removal of images.

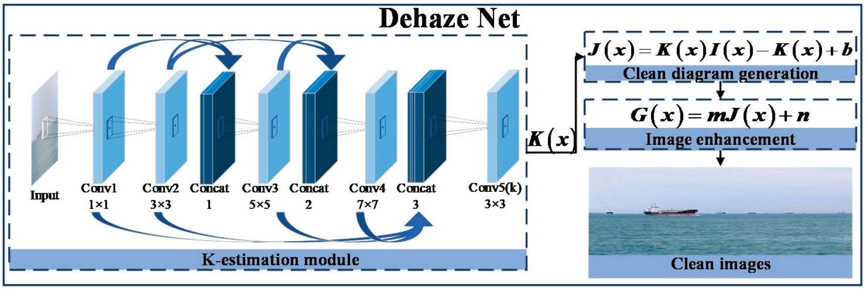


Figure 2. Haze removal network.

3.2. *Marine Shipping Target Detection Using the YOLOv5 Algorithm*

The second part of the framework involves shipping detection. YOLO algorithms are representative of one-stage target detection algorithms, which regard target detection as a regression problem with a simple network flow. Among them, the YOLOv5 network is small, stable, and good in terms of network generalization ability [34], making it an ideal choice for real-time flexible target detection in the offshore environment. Because ships in maritime images are usually small and ship speed extraction has high requirements for the computational speed and flexibility of the detection algorithm, this framework adopts YOLOv5 as the target detection algorithm [35]. Meanwhile, the maritime dataset is used to train the YOLOv5 network to realize the fast and accurate positioning of small and medium-sized ships in maritime images and improve the accuracy of ship speed extraction. The main components of the YOLOv5 network are input, backbone, neck, and prediction, as shown in Figure 3.

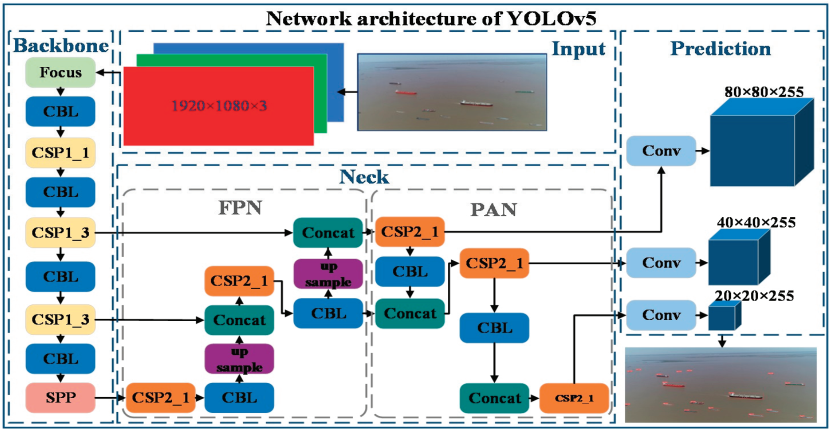


Figure 3. The network structure of YOLOv5.

Among them, mosaic enhancement is used at the input end of the YOLOv5 network to improve the detection accuracy of small ships. The adaptive anchor frame calculation and adaptive scaling for different data are performed to improve the calculation speed of the network. The neck network integrates the information of the upper and lower layers to fully extract the features of the ship. At the same time, the cross stage partial network (CSP-Net) is used to enhance the fusion of the target features of the network and improve the extraction efficiency of the ship's features. In the prediction part of YOLOv5, the anchor frame of the grid is used for target detection on feature graphs of different scales. The complete intersection over union (CIoU) is used as the loss function of the boundary frame, which allows the algorithm to converge quickly and make the prediction frame more consistent with the real frame. Non-maximum suppression (NMS) is used by the neck network to enhance the detection accuracy of multiple ship targets and overlapping ship targets.

3.3. Ship Target Tracking with Deep SORT Algorithm

To obtain the pixel displacement of the ships in the images of continuous frames, the video with the detection frame information is taken as the input of the Deep SORT algorithm [8] in the third part of this framework. The algorithm first predicts the trajectory of the next frame using the Kalman filter. IOU matching and cascade matching are then performed between the predicted value and the detection frame information to track the trajectory of the target ship between the front and back frames of the video [36].

During the prediction process, $x_{k-1} = \left(u, v, r, h, \dot{x}, \dot{y}, \dot{z}, \dot{h}\right)$ represents the motion state of the target in $k-1$ box, where (u, v) is the central point coordinates of the target box, r is the aspect ratio, and h is the height of the detection box. In x_{k-1}, the last four variables are the derivatives of the first four variables, representing the transformation rate of the first four variables. When the standard filter is used to predict the motion state of the target, the last four values are constants. With $x_{k-1} = (u, v, r, h)$ as the prediction result, the motion state prediction of the Kalman algorithm can be expressed as:

$$\hat{x}_k = A\hat{x}_{k-1} \tag{6}$$

where, $\hat{x}_{k-1}$ is the motion state vector of frame $k-1$, and A is the state transition matrix used to predict the motion state $\hat{x}_k$ of frame k.

In the process of data association between continuous frames, Mahalanobis distance, and cosine distance are introduced to conduct the association of data between connected frames, and the thresholds between the observation box and prediction box are set, respectively. In the process of data association, when the Mahalanobis distance and cosine distance are both within the threshold range, the data association of two adjacent frames is considered successful. The comprehensive associated cost equation is as follows:

$$c_{i,j} = \lambda d^{(1)}(i,j) + (1-\lambda)d^{(2)}_{i,j} \tag{7}$$

where, λ is a hyperparameter, and the influence of Mahalanobis distance and cosine distance on the association results can be controlled by controlling λ.

In order to enable the targets blocked for a long time to be continuously tracked, Deep SORT introduced cascade matching to give priority to the targets with more occurrences. Then, the intersection and union (IoU) between the boundary box and the prediction box by the Kalman filter is calculated and detected for the newly emerged targets and the prediction boxes that failed to match. The detection result is recognized if the match value is greater than the minimum IoU value. The equation for calculating IoU is as follows:

$$IoU = \frac{|area(A) \cap area(B)|}{|area(A) \cup area(B)|} \tag{8}$$

where, A is the detection boundary box, and B is the prediction boundary box of the candidate trajectory.

3.4. Space Mapping from 2D to 3D and Speed Extraction

In the fourth part, the mapping relation matrix between the ship's displacement in the images and the actual displacement is obtained by 2D to 3D space mapping. The actual ship displacement is calculated according to the trajectory of the ship in the images obtained by the target tracking algorithm. The ship's movement in a short time is regarded as the uniform motion. The average velocity equation is used to estimate the actual velocity of the ships based on the premise of knowing the time difference and actual displacement of the ship [37].

The process of 2D to 3D space mapping involves solving the mapping relationship between objects from the three-dimensional world and points on the two-dimensional image plane. The process involves four coordinate systems, and the representation methods of coordinate systems and points in coordinate systems are as follows:

(1) World coordinate system. The coordinate system corresponding to the three-dimensional world describes the position of the target in the real world. The unit length of the coordinate axis is m. The points in the world coordinate system are represented by (X_w, Y_w, Z_w).

(2) Camera coordinate system. The origin is located in the optical center of the lens, and its *x*-axes and *y*-axes are parallel to both sides of the phase plane. The *z*-axis is perpendicular to the image plane and is the optical axis of the lens. The unit length of the coordinate axis is m. The points in the camera coordinate system are denoted as (X_c, Y_c, Z_c).

(3) Image coordinate system. The origin is the intersection of the optical axis of the camera and the imaging plane, that is, the midpoint of the imaging plane. The unit length of the coordinate axis is mm. The points in the image coordinate system are represented by (x, y).

(4) Pixel coordinate system: the origin is the top-left corner of the imaging plane in pixels. Points in the pixel coordinate system are represented as (u, v).

The object is transformed from the world coordinate system to the camera coordinate system through translation and rotation, and the transformation equation between the camera coordinate system and the world coordinate system is:

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} R & t \\ \vec{0} & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{9}$$

where R is a 3×3 rotation matrix and t is a 3×1 translation vector.

The transformation of the camera coordinate system into the image coordinate system is based on the projection perspective. The connection between P in space and the camera optical center O is OP, and the intersection point p between OP and phase plane is the projection of point P on the image as shown in Figure 4.

According to the projection perspective, the conversion equation of the camera coordinate system and the image coordinate system is:

$$Z_c \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \tag{10}$$

where Z_c is the scale factor, and f is the focal length. The pixel coordinate system can coincide with the image coordinate system after translation, as shown in Figure 5.

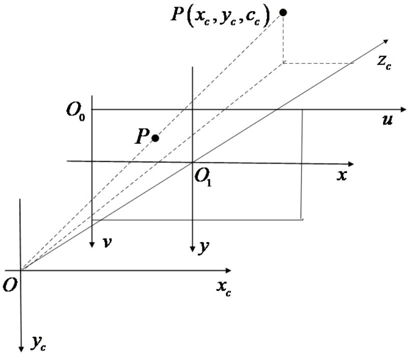


Figure 4. Projection perspective between the camera coordinate system and the image coordinate system.

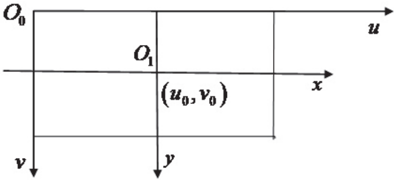


Figure 5. Translation relationship between the image coordinate system and the pixel coordinate system.

Then, the conversion equation of the image coordinate system and pixel coordinate system is:

$$\begin{cases} u = \frac{x}{d_x} + u_0 \\ v = \frac{y}{d_y} + v_0 \end{cases} \tag{11}$$

Represented by the matrix:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{d_x} & 0 & u_0 \\ 0 & \frac{1}{d_y} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{12}$$

where d_x and d_y are the scale factors of the two coordinate systems in the directions of the x-axis and y-axis, and (u_0, v_0) is the coordinate of the origin of the pixel coordinate system.

As can be seen from Equations (9), (10) and (12), the conversion equation between the world coordinate system and the pixel coordinate system in which the image is located can be expressed as:

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} == \begin{bmatrix} \frac{f}{d_x} & 0 & u_0 & 0 \\ 0 & \frac{f}{d_y} & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & t \\ \vec{0} & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{13}$$

If

$$\begin{bmatrix} R & t \\ \vec{0} & 1 \end{bmatrix} = M \tag{14}$$

$$\begin{bmatrix} \frac{f}{d_x} & 0 & u_0 & 0 \\ 0 & \frac{f}{d_y} & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} = K \tag{15}$$

Then

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = KM \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{16}$$

In Equation (16), the element in K is the configuration parameters of the camera, which is called the camera's internal parameter matrix. The elements in M are called the internal parameter matrix of the camera.

In this paper, the plane where the ship is located is set to the $X_wO_wY_w$ plane of the world coordinate system, the direction perpendicular to $X_wO_wY_w$ is the positive direction of the Z_w axes, and the camera coordinate system is set to coincide with the world coordinate system. Under this assumption, the conversion equation of the pixel coordinate system and the world coordinate system can be simplified as:

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{17}$$

In this study, the ship's movement in a very short time is regarded as uniform linear motion. Assuming that the ship displacement in the pixel coordinate system in the period Δt at the moment T can be expressed as $(\Delta u, \Delta v)$, the transformation relationship between the image displacement and actual displacement is as follows:

$$Z_c \begin{bmatrix} \Delta u \\ \Delta v \\ 1 \end{bmatrix} = K \begin{bmatrix} \Delta X_w \\ \Delta Y_w \\ \Delta Z_w \\ 1 \end{bmatrix} \tag{18}$$

Then the actual displacement of the ship in the time period Δt is:

$$\Delta L = \sqrt{(\Delta X_w)^2 + (\Delta Y_w)^2 + (\Delta Z_w)^2} \tag{19}$$

According to the average velocity formula, the velocity of the ship at the moment T can be expressed as:

$$v_T = \frac{\Delta L}{\Delta t} \tag{20}$$

4. Results

The main contents of Section 4 are the experimental details and results. It should be noted that all experiments were carried out on an Intel I7-11800H@4.6 GHz computer with a 6 g memory processor, and the experiments were completed in the Windows10 system using the Pytorch software library.

4.1. Experimental Data

In this study, contrast experiments were conducted on each module of the framework, and simulation experiments were conducted to verify the robustness of the framework in the process of extracting the shipping speed. The experimental data of hazy removal included three shore-based surveillance videos, which contained 5658 images, and three maritime videos with 8000 marine images. The resolution of the shore-based surveillance images is 640 × 386, and that of the self-built images is 1920 × 1080.

In the haze removal experiment of marine images, 5000 synthetic hazy marine images were selected as the training dataset to improve the generalization performance of AOD-Net on marine scenes, and 600 images were used as the non-repetitive test set. During the maritime target detection and tracking experiment, 8000 high-resolution images were divided into 6400 training datasets, 1000 validation datasets, and 600 test datasets to train the YOLOv5 algorithm. It should be noted that the training and test datasets do not overlap. At the same time, the Deep SORT algorithm was selected to combine with the YOLOv5 algorithm to conduct the multi-ship tracking experiment and the simulation experiment of ship speed extraction in marine scenarios. Shore-based scenes include scenes on cloudy days (scene 1 in Figure 6), scenes on sunny days [38] (scene 2 in Figure 6), and scenes with wave disturbance [38] (scene 3 in Figure 6). The self-built dataset includes the scene of many small target ships (scene 4 in Figure 6), the scene with normal light (scene 5 in Figure 6), and the scene with low light (scene 6 in Figure 6).


Scene 1 **Scene 2** **Scene 3**
Scene 4 **Scene 5** **Scene 6**

Figure 6. Experimental scenes.

It should be noted that, when training the haze removal network, this paper [39] is referred to adding synthetic haze images to improve the generalization performance of the haze removal network in marine scenes. We synthesized image datasets with three different haze concentrations, and in this study, T represents the haze concentration. The images with three different haze concentrations are respectively represented as the images with $T = 0.3$, the images with $T = 0.5$, and the image data with $T = 0.7$, among which the images with $T = 0.3$ has the lowest haze concentration, as is shown in Figure 7.

Figure 7. Synthetic haze images.

4.2. Experimental Results and Analysis

4.2.1. Haze Removal of Marine Images

To verify the performance of the framework on haze removal, Retinex [40], Dark Channel Prior [12], Contrast Limited Adaptive Histogram Equalization (CLAHE) [41], AOD-Net [9], and e-AOD-Net adopted in the framework are used to conduct a comparative experiment on haze removal. The experimental results are shown in Figure 8.

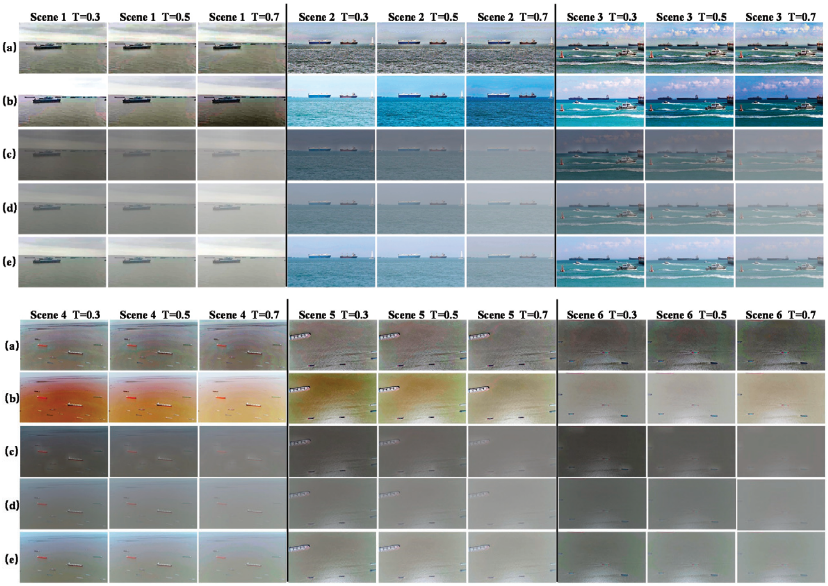


Figure 8. Comparative experiments of haze removal. (**a**) Retinex (**b**) Dark Channel Prior (**c**) CLAHE (**d**) AOD-Net (**e**) e-AOD-Net.

The images generated by the above methods after haze removal are shown in Figure 8. Retinex and Dark Channel Prior usually lead to image distortion, and the color of the images after haze removal is seriously abnormal. CLAHE usually makes the color of images after haze removal too dark, and maritime ship features are not prominent. After haze removal by the AOD-Net network, there is still noise remaining in the images. These phenomena may occur because none of the above competing methods can fully extract the target structural features from ocean images. By contrast, e-AOD-Net can learn more structural features of images in marine scenes after generalization training and adaptive enhancement of marine images. The evaluation results of the dehazed images in multiple scenes using the PSNR and SSIM are presented in Table 1.

Table 1. Evaluation results of dehazed images. (The best results are highlighted in red).

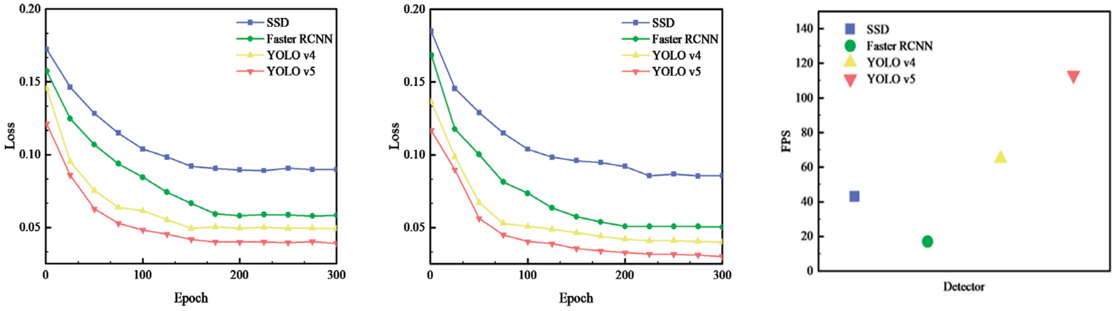
Method	Retinex		CLAHE		Dark Channel Prior		AOD-Net		e-AOD-Net	
	PSNR [1]	SSIM [2]	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Scene 1	10.86	0.70	9.80	0.67	9.72	0.72	22.81	0.94	**23.71**	**0.95**
Scene 2	10.12	0.68	9.06	0.65	9.01	0.67	20.50	**0.95**	**21.86**	0.94
Scene 3	10.10	0.65	9.51	0.63	9.09	0.65	20.06	0.95	**23.84**	**0.95**
Scene 4	10.73	0.55	9.76	0.52	9.48	0.53	20.59	0.92	**21.99**	**0.95**
Scene 5	11.42	0.40	10.46	0.38	11.13	0.39	22.15	0.92	**23.69**	**0.96**
Scene 6	11.79	0.50	10.17	0.43	10.82	0.48	19.68	0.94	**23.87**	**0.96**

[1] PSNR: The peak signal-to-noise ratio [42] is a widely used evaluation index for measuring image quality. [2] SSIM: The structural similarity index measure [43] can objectively determine the structural similarity of images based on the human visual system.

As shown in Table 1, the e-AOD-Net adopted in this study has stable and good performance in multiple marine scenes, indicating that e-AOD-Net achieves better image enhancement performance. Images after haze removal can highlight more ship information, which is the basis of ship detection and tracking.

4.2.2. Multi-Ship Detection and Tracking Experiment after Images Enhancement

In this part, in order to verify the detection performance of the algorithm, SSD [44], Faster RCNN [45], and YOLO v4 [46] are compared with the YOLOv5 algorithm [47] adopted. The training comparison chart (Figure 9a), the verification comparison chart (Figure 9b), the frames per second (FPS) comparison chart (Figure 9c), the training index table (Table 2), the verification index table (Table 3), the test index table (Table 4) are drawn respectively.



(**a**) (**b**) (**c**)

Figure 9. Comparison of detection methods. (**a**) Training carve; (**b**) Validation carve; (**c**) FPS of multiple detection algorithms.

Table 2. Train metrics.

	p	R	mAP_0.5	mAP_0.5:0.95
Faster RCNN	0.9446	0.9195	0.9297	0.8835
SSD	0.9154	0.8974	0.9065	0.8548
YOLO v4	0.979	0.9701	0.972	0.8903
YOLO v5	0.9929	0.973	0.989	0. 920

Table 3. Validation metrics.

	p	R	mAP_0.5	mAP_0.5:0.95
Faster RCNN	0.927	0.907	0.916	0.764
SSD	0.90.1	0.884	0.863	0.718
YOLO v4	0.973	0.982	0.995	0.845
YOLO v5	0.989	0.990	0.993	0.880

Table 4. Test metrics.

	p	R	mAP_0.5	mAP_0.5:0.95
Faster RCNN	0.932	0.895	0.937	0.775
SSD	0.903	0.874	0.895	0.738
YOLO v4	0.983	0.979	0.991	0.80
YOLO v5	0.993	0.984	0.994	0.83

As can be seen from the comparison curve, under the same training conditions, the convergence speeds of the YOLOv4 and YOLOv5 algorithms are fast, and the YOLOv5 algorithm has a faster image-processing speed (in Figure 9c). The test and evaluation parameters show that the trained YOLOv5 algorithm also performs well in detecting accuracy in ocean scenes. To verify the performance of images after haze removal by e-AOD-Net in ship detection tasks, we adopted the stable YOLOv5 algorithm to detect ships in the synthesized hazy images and images after haze removal and compared the detection results. The detection results are presented in Figure 10.

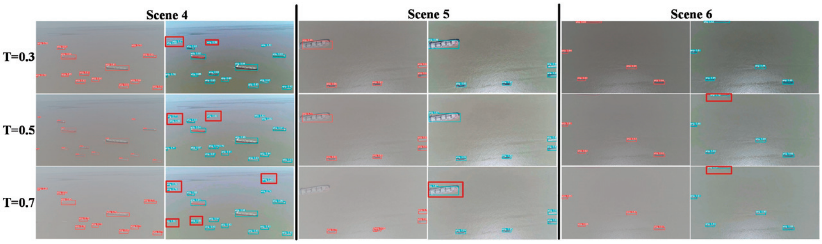


Figure 10. Comparison of detection effects of YOLOv5 on images before and after haze removal (Detection details are highlighted with red boxes).

Figure 10 shows that YOLOv5 has high-precision detection performance in multiple ocean scenarios. However, the degree of recovery of the images after haze removal is high, and the structural features of the ships are prominent, making it easy for the ship target in the image to be detected by the YOLOv5 algorithm, such as the small ships in the red boxes in scene 4. Haze noise in images reduces the accuracy of the target detection. In the case of high haze concentrations, some ships were not detected, such as those in the red boxes in scenes 5 and 6.

Considering the accuracy, detection speed, and detection stability of the algorithm in synthetic haze scenes, the YOLOv5 algorithm is ideal for ship detection in maritime scenarios. In the evaluation of the tracking algorithms, the YOLOv5 algorithm was used as the detector in scenarios 4, 5, and 6. In the three scenarios, multi-objective tracking evaluation parameters were introduced to evaluate the tracking performance of the SORT and Deep SORT algorithms for ships in ocean scenarios. The evaluation results are presented in Tables 5–7. In the tables, parameters with upward arrows indicate that the evaluated method performs better when the evaluated value is larger; those with downward arrows indicate that the evaluated method performs better when the evaluated value is smaller. And the optimal evaluation values when the hazy concentration is T = 0.3, T = 0.5, and T = 0.7 have been highlighted in red, yellow, and green, respectively in Tables 5–7.

As shown in Tables 5–7, the Deep SORT algorithm using YOLOv5 as a detector has higher MOTA and MOTP values as well as lower IDS and ML values in the above scenes. This indicates that Deep SORT can track ships stably while avoiding the number of ID transitions. It should be noted that in the same scenario, the evaluation results of tracking algorithms that use images after haze removal are usually the optimal values, indicating that images after haze removal can effectively improve the robustness of target tracking algorithms in maritime scenarios. It is worth noting that the YOLOv5 algorithm combined with the Deep SORT target tracking algorithm adopted in the framework can maintain high detection accuracy and stable tracking performance in multi-ship tracking, which is the basis for accurate ship speed extraction in this study.

4.2.3. Ship Speed Extraction

In this section, AIS data are considered as the ground truth of the ship speed values. The AIS data of Baosteel Wharf on 5 April 2021 is downloaded from the website http://www.shipxy.com. To make it easier to compare the ground truth of the ship speed values with the shipping speed extracted from marine images, the AIS data were linearly interpolated to match the video image frame by frame after extracting the speed data. To highlight the effect of this framework on hazy images and the performance of ship speed extraction, this section selects a ship in scenes 4, 5, and 6 for the speed extraction simulation experiment and compares the ground truth with the extracted speed of each ship in the scene with different haze concentrations. The extraction and comparison results for the ship speed are shown in Figures 11–13. The mean speed of each ship and the MSE and MAE values compared to the ground truth of the speed are listed in Tables 8–10. In the Tables 8–10, the highlighted in yellow indicates the speed results and speed evaluation results extracted from the haze videos, the highlighted in green indicates the speed results and speed evaluation results extracted from the videos after haze removed, and the highlighted in red indicates the true value of speed extracted from the AIS data.

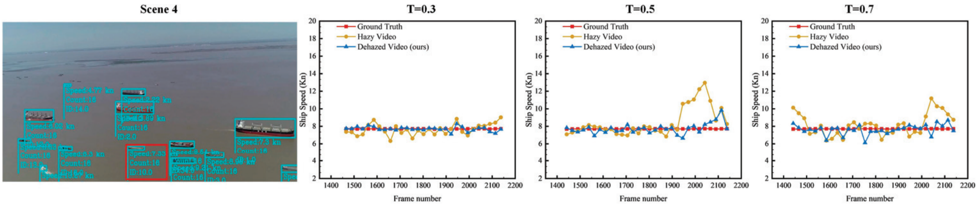
Table 5. Evaluation results of multi-ship tracking in scene 4.

Scene 4										
Data Type	**Tracker**	**Haze Concentration**	**IDF1[1] ↑**	**MOTA[2] ↑**	**MOTP[3] ↑**	**MT[4] ↑**	**ML[5] ↓**	**FP[6] ↓**	**FN[7] ↓**	**IDS[8] ↓**
Hazy video	SORT	T = 0.3	97.6%	96.7%	85.4%	100.0%	0.0%	18	72	1
		T = 0.5	97.3%	95.1%	85.4%	100.0%	12.5%	20	114	2
		T = 0.7	89.3%	88.7%	84.2%	75.0%	12.5%	24	292	3
	Deep SORT	T = 0.3	98.5%	97.1%	85.2%	100.0%	0.0%	15	57	0
		T = 0.5	97.5%	95.6%	85.5%	100.0%	0.0%	20	92	1
		T = 0.7	96.8%	93.8%	84.7%	75.0%	12.5%	23	155	3
Dehazed video	SORT	T = 0.3	98.6%	97.3%	**85.8%**	100.0%	0.0%	14	58	0
		T = 0.5	97.7%	96.8%	86.0%	100.0%	0.0%	13	70	1
		T = 0.7	97.5%	95.9%	85.5%	100.0%	0.0%	17	89	2
	Deep SORT	T = 0.3	**98.7%**	**97.5%**	85.7%	**100.0%**	**0.0%**	**3**	**44**	**0**
		T = 0.5	**98.7%**	**97.3%**	**86.9%**	**100.0%**	**0.0%**	**9**	**51**	**0**
		T = 0.7	**98.3%**	**96.7%**	**86.1%**	**100.0%**	**0.0%**	**14**	**71**	**0**

[1] IDF1: The ratio of correctly identified detections over the average number of ground-truth and computed detections; [2] MOTA (multi-object tracking accuracy): This measure combines three error sources: false positives, missed targets, and identity switches; [3] MOTP(Multiple Object Tracking Precision): The misalignment between the annotated and the predicted bounding boxes; [4] MT: The ratio of ground-truth trajectories that are covered by a track hypothesis for at least 80% of their respective life span; [5] ML: The ratio of ground-truth trajectories that are covered by a track hypothesis for at most 20% of their respective life span; [6] FP: The total number of false positives; [7] FN: The total number of false negatives (missed targets); [8] IDS: The total number of identity switches. (The meaning of the evaluation parameters in Tables 6 and 7 is the same as described above.)

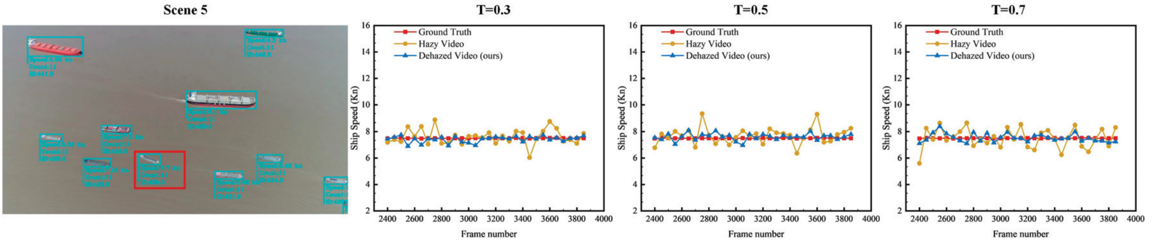
Table 6. Evaluation results of multi-ship tracking in scene 5.

Scene 5										
Data Type	**Tracker**	**Haze Concentration**	**IDF1[1] ↑**	**MOTA[2] ↑**	**MOTP[3] ↑**	**MT[4] ↑**	**ML[5] ↓**	**FP[6] ↓**	**FN[7] ↓**	**IDS[8] ↓**
Hazy video	SORT	T = 0.3	92.7%	85.0%	80.4%	87.5%	0.0%	82	84	2
		T = 0.5	90.2%	81.2%	80.8%	87.5%	12.5%	106	248	1
		T = 0.7	88.4%	80.3%	77.6%	62.5%	25.0%	631	516	3
	Deep SORT	T = 0.3	**98.0%**	90.0%	81.5%	75.0%	0.0%	14	191	0
		T = 0.5	95.9%	94.0%	81.2%	87.5%	12.5%	39	263	1
		T = 0.7	95.3%	89.8%	80.7%	75.0%	12.5%	39	373	2
Dehazed video	SORT	T = 0.3	97.6%	93.2%	**81.8%**	87.5%	0.0%	54	149	0
		T = 0.5	95.1%	86.6%	81.4%	87.5%	12.5%	299	187	0
		T = 0.7	94.7%	88.1%	81.6%	87.5%	0.0%	456	201	0
	Deep SORT	T = 0.3	**98.8%**	**95.7%**	**83.5%**	**87.5%**	**0.0%**	**16**	**50**	**0**
		T = 0.5	**98.0%**	**94.4%**	**81.6%**	**87.5%**	**0.0%**	**20**	**151**	**0**
		T = 0.7	**98.0%**	**94.0%**	80.9%	**87.5%**	**12.5%**	**23**	**177**	**0**



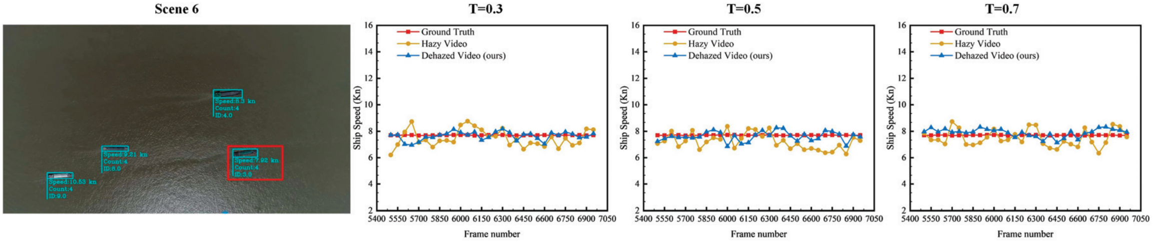
(**a**) (**b**) (**c**) (**d**)

Figure 11. Speed extraction results for ship 1 in scene 4. (**a**) Scene image; (**b**) Comparison of speed measurements (T = 0.3); (**c**) Comparison of speed measurements (T = 0.5); (**d**) Comparison of speed measurements (T = 0.7).



(**a**) (**b**) (**c**) (**d**)

Figure 12. Speed extraction results for ship 2 in scene 5. (**a**) Scene image; (**b**) Comparison of speed measurements (T = 0.3); (**c**) Comparison of speed measurements (T = 0.5); (**d**) Comparison of speed measurements (T = 0.7).



(**a**) (**b**) (**c**) (**d**)

Figure 13. Speed extraction results for ship 3 in scene 6. (**a**) Scene image; (**b**) Comparison of speed measurements (T = 0.3); (**c**) Comparison of speed measurements (T = 0.5); (**d**) Comparison of speed measurements (T = 0.7).

Table 7. Evaluation results of multi-ship tracking in scene 6.

Scene 6										
Data Type	**Tracker**	**Haze Concentration**	**IDF1** [1] ↑	**MOTA** [2] ↑	**MOTP** [3] ↑	**MT** [4] ↑	**ML** [5] ↓	**FP** [6] ↓	**FN** [7] ↓	**IDS** [8] ↓
Hazy video	SORT	T = 0.3	85.7%	78.9%	71.9%	85.7%	14.3%	183	3187	11
		T = 0.5	85.6%	75.0%	70.7%	85.7%	14.3%	226	4381	14
		T = 0.7	80.5%	70.2%	63.5%	71.4%	28.6%	318	4869	20
	Deep SORT	T = 0.3	86.2%	78.8%	74.4%	100.0%	0.0%	179	1356	9
		T = 0.5	85.4%	78.1%	72.4%	85.7%	28.6%	186	1267	12
		T = 0.7	82.7%	71.5%	65.6%	57.1%	28.6%	295	3741	18
Dehazed video	SORT	T = 0.3	86.8%	79.7%	75.5%	85.7%	14.3%	164	3171	12
		T = 0.5	85.7%	78.2%	71.9%	85.7%	14.3%	189	3469	14
		T = 0.7	83.2%	74.8%	68.5%	71.4%	28.6%	233	4715	14
	Deep SORT	T = 0.3	92.7%	82.1%	76.7%	100.0%	0.0%	120	1224	7
		T = 0.5	89.6%	79.9%	75.6%	85.7%	0.0%	163	1267	9
		T = 0.7	88.2%	79.3%	68.7%	85.7%	14.3%	204	1334	11

As shown in Figures 11–13, the fold line representing the speed value of the ship extracted from the AIS data is set to red in the figure; the fold line graph representing the speed value of the ship extracted directly from the haze video is set to yellow, and the fold line representing the speed value of the ship extracted by our framework after removing the haze from the maritime haze image is set to blue. According to the extraction results and the mean ground truth of the ship speed, the speeds of the three ships were at 7.71 Kn, 7.50 Kn, and 7.70 Kn, respectively. For ship No. 1 in Figure 11, the accuracy of the speed extracted is easily affected by noise in the images owing to the small sizes of the ships. When T = 0.3, the MSE and MAE values of the speed are 0.37 Kn and 0.49 Kn due to the slight noise in the images. After haze removal, the fluctuation of the velocity image improved. At this time, the values of MSE are 0.12 Kn, the values of MAE are 0.21 Kn, and the ship's average speed is improved from 7.54 Kn to 7.73 Kn, which is closer to the average speed of the ground truth of the ship speed.

Table 8. Mean ship speeds.

	Haze Concentration	**Mean Speed of Ship (Kn)**		
		Hazy Video	**Dehazed Video**	**Ground Truth**
Ship 1	T = 0.3	7.54	**7.73**	**7.71**
	T = 0.5	8.55	**7.75**	
	T = 0.7	8.15	**7.62**	
Ship 2	T = 0.3	7.59	**7.50**	**7.50**
	T = 0.5	7.71	**7.54**	
	T = 0.7	7.50	**7.58**	
Ship 3	T = 0.3	7.31	**7.63**	**7.70**
	T = 0.5	7.19	**7.63**	
	T = 0.7	7.29	**7.96**	

Table 9. MSE values of ship speed.

	Haze Concentration	**MSE Values of Ship Speed (Kn)**	
		Hazy Video	**Dehazed Video**
Ship 1	T = 0.3	0.37	**0.12**
	T = 0.5	1.71	**0.33**
	T = 0.7	3.57	**0.41**
Ship 2	T = 0.3	0.32	**0.13**
	T = 0.5	0.43	**0.13**
	T = 0.7	0.57	**0.18**
Ship 3	T = 0.3	0.47	**0.14**
	T = 0.5	0.58	**0.16**
	T = 0.7	0.65	**0.22**

Table 10. MAE values of ship speed.

	Haze Concentration	**MAE Values of Ship Speed (Kn)**	
		Hazy Video	**Dehazed Video**
Ship 1	T = 0.3	0.49	**0.21**
	T = 0.5	1.03	**0.41**
	T = 0.7	1.14	**0.43**
Ship 2	T = 0.3	0.44	**0.23**
	T = 0.5	0.50	**0.26**
	T = 0.7	0.54	**0.30**
Ship 3	T = 0.3	0.60	**0.27**
	T = 0.5	0.60	**0.31**
	T = 0.7	0.66	**0.38**

When T = 0.5, due to the influence of haze noise in the images, the curve chart of ship velocity fluctuates wildly, especially in the late video period, and the MSE of ship velocity is 1.71 Kn, and MAE is 1.03 Kn. Although the extraction value of the velocity after haze removal still fluctuated, it was significantly improved compared with that before haze removal. After removing the haze, the MSE and MAE of the ship speed extracted from the image are 0.33 Kn and 0.41 Kn, and the average ship speed was 7.75 Kn.

When T = 0.7, the velocity fluctuation was more prominent. Currently, the MSE and MAE of ship velocity are 3.57 Kn and 1.14 Kn. After haze removal, the fluctuation of the velocity curve chart decreased. Both the MSE and MAE of the ship velocity decreased, and the mean value of the ship velocity was closer to the ground truth.

The same situation appeared in ship No. 2 in Figure 12. It can be seen from the truth line chart is approximately 7.5 Kn. According to the MSE and MAE of ship No. 2, the accuracy of the ship speed extracted can be improved by removing haze.

For ship No. 3 in scenario 6, the curve chart of ship speed fluctuates greatly because the image brightness is low, and the accuracy of the ship speed extracted is reduced after the haze noise is superimposed. The MSE of ship speed under different haze concentration environments was 0.47 Kn, 0.58 Kn, and 0.65 Kn, respectively. The MAE of speed is 0.60 Kn, 0.60 Kn, and 0.66 Kn. The MSE of speed extracted after removing haze is 0.14 Kn, 0.16 Kn, and 0.22 Kn. The MAE is 0.27 Kn, 0.31 Kn, and 0.38 Kn, respectively. After haze removal, the average ship speed extracted from the images was closer to the average value of the ground truth. It shows that the framework adopted in this paper can effectively enhance the quality of haze images in ocean scenes with low brightness and improve the accuracy of ship speed extracted from the images.

5. Conclusions

In this study, a framework for ship detection and ship speed extraction from maritime haze images using deep learning methods is proposed. First, a lightweight CNN was used to remove haze from hazy marine images. Second, YOLO v5 is used to accurately detect ships in marine images after haze removal. Moreover, the Deep-SORT target tracking algorithm is used to track ships. Finally, the ship motion pixels are calculated according to the trajectory information of the ship between adjacent image frames, and the ship speed is estimated and extracted based on the mapping relationship between the image space and the actual space.

Experimental results demonstrate that the proposed framework effectively enhances the clarity and contrast of marine haze images, as indicated by the mean peak signal-to-noise ratio (PSNR) and mean structural similarity index (SSIM) values of 23.86 and 0.96, respectively. The framework achieves high accuracy in extracting ship speed in multiple marine scenes, with an average accuracy above 95% and strong stability. The proposed speed extraction framework significantly improves the accuracy of ship speed extraction in hazy environments, with the mean squared error (MSE) values of ship speed extracted from the images after haze removal averaging 0.3 Kn lower than those from the images before haze removal.

In future studies, additional marine scenarios will be considered to further verify the practicality of this framework in real-world scenarios.

Author Contributions: Conceptualization, J.Z., X.C. and Z.Z.; methodology, Z.Z. and Y.C.; software, Z.Z. and Y.C.; validation, Z.Z., Y.C. and J.Z.; investigation, Z.Z. and X.C.; data curation, Z.Z., Y.C. and X.C.; writing—original draft preparation, Z.Z.; writing—review and editing, J.Z. and X.C.; supervision, J.Z. and X.C.; funding acquisition, J.Z. and X.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was jointly funded by the National Key Research and Development Program of China under Grant 2021YFC2801004 and 2021YFC2801003, National Natural Science Foundation of China under Grant 52102397, 51709167 and 52071199, Shanghai Science and Technology Innovation Action Plan under Grant Nos. 22DZ1204503, China Postdoctoral Science Foundation under Grant 2021M700790.

Data Availability Statement: Data available on request due to restrictions eg privacy or ethical.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Qiao, D.; Liu, G.; Lv, T.; Li, W.; Zhang, J. Marine Vision-Based Situational Awareness Using Discriminative Deep Learning: A Survey. *J. Mar. Sci. Eng.* **2021**, *9*, 397. [CrossRef]
2. Chen, S.; Xiong, X.; Wen, Y.; Jian, J.; Huang, Y. State Compensation for Maritime Autonomous Surface Ships' Remote Control. *J. Mar. Sci. Eng.* **2023**, *11*, 450. [CrossRef]
3. Chen, X.; Wang, Z.; Hua, Q.; Shang, W.L.; Luo, Q.; Yu, K. AI-Empowered Speed Extraction via Port-Like Videos for Vehicular Trajectory Analysis. *IEEE Trans. Intell. Transp. Syst.* **2022**, *24*, 4541–4552. [CrossRef]
4. Kang, B.S.; Jung, C.H. Detecting Maritime Obstacles Using Camera Images. *J. Mar. Sci. Eng.* **2022**, *10*, 1528. [CrossRef]
5. Chen, X.; Chen, W.; Yang, Y.; Li, C.; Han, B.; Yao, H. High-Fidelity Ship Imaging Trajectory Extraction via an Instance Segmentation Model. In Proceedings of the 2022 International Symposium on Sensing and Instrumentation in 5G and IoT Era (ISSI), Shanghai, China, 17–18 November 2022; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2022; pp. 165–169.
6. Chen, X.; Xu, X.; Yang, Y.; Wu, H.; Tang, J.; Zhao, J. Augmented Ship Tracking under Occlusion Conditions from Maritime Surveillance Videos. *IEEE Access* **2020**, *8*, 42884–42897. [CrossRef]
7. Chen, Z.; Chen, D.; Zhang, Y.; Cheng, X.; Zhang, M.; Wu, C. Deep Learning for Autonomous Ship-Oriented Small Ship Detection. *Saf. Sci.* **2020**, *130*, 104812. [CrossRef]
8. Zhao, J.; Chen, Y.; Zhou, Z.; Zhao, J.; Wang, S.; Chen, X. Extracting Vessel Speed Based on Machine Learning and Drone Images during Ship Traffic Flow Prediction. *J. Adv. Transp.* **2022**, *2022*, 3048611. [CrossRef]
9. Li, B.; Peng, X.; Wang, Z.; Xu, J.; Feng, D. AOD-Net: All-in-One Dehazing Network. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4780–4788. [CrossRef]
10. Ling, Z.; Liang, Y.; Wang, Y.; Shen, H.; Lu, X. Adaptive Extended Piecewise Histogram Equalisation for Dark Image Enhancement. *IET Image Process* **2015**, *9*, 1012–1019. [CrossRef]
11. Oishi, S.; Fukushima, N. Retinex-Based Relighting for Night Photography. *Appl. Sci.* **2023**, *13*, 1719. [CrossRef]
12. He, K.; Sun, J.; Tang, X. Single Image Haze Removal Using Dark Channel Prior. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 2341–2353. [CrossRef]
13. Yeh, C.H.; Huang, C.H.; Kang, L.W. Multi-Scale Deep Residual Learning-Based Single Image Haze Removal via Image Decomposition. *IEEE Trans. Image Process.* **2020**, *29*, 3153–3167. [CrossRef]
14. He, L.; Bai, J.; Ru, L. Haze Removal Using Aggregated Resolution Convolution Network. *IEEE Access* **2019**, *7*, 123698–123709. [CrossRef]
15. Qin, M.; Xie, F.; Li, W.; Shi, Z.; Zhang, H. Dehazing for Multispectral Remote Sensing Images Based on a Convolutional Neural Network with the Residual Architecture. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2018**, *11*, 1645–1655. [CrossRef]
16. Bergmann, P.; Meinhardt, T.; Leal-Taixe, L. Tracking without Bells and Whistles. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019. [CrossRef]
17. Chu, P.; Wang, J.; You, Q.; Ling, H.; Liu, Z. TransMOT: Spatial-Temporal Graph Transformer for Multiple Object Tracking. *arXiv* arXiv:2104.00194, 2021.
18. Brasó, G.; Leal-Taixé, L. Learning a Neural Solver for Multiple Object Tracking. *arXiv* **2019**, arXiv:1912.07515.
19. Yang, F.; Odashima, S.; Masui, S.; Jiang, S. Hard to Track Objects with Irregular Motions and Similar Appearances? Make It Easier by Buffering the Matching Space. *arXiv* **2022**, arXiv:2211.14317.
20. Bewley, A.; Ge, Z.; Ott, L.; Ramos, F.; Upcroft, B. Simple Online and Realtime Tracking. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 3464–3468. [CrossRef]
21. Zhang, Y.; Sun, P.; Jiang, Y.; Yu, D.; Weng, F.; Yuan, Z.; Luo, P.; Liu, W.; Wang, X. ByteTrack: Multi-Object Tracking by Associating Every Detection Box. *arXiv* **2021**, arXiv:2110.06864.
22. Pang, J.; Qiu, L.; Li, X.; Chen, H.; Li, Q.; Darrell, T.; Yu, F. Quasi-Dense Similarity Learning for Multiple Object Tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Online, 13–19 June 2020.
23. Wang, Z.; Zheng, L.; Liu, Y.; Li, Y.; Wang, S. Towards Real-Time Multi-Object Tracking. *arXiv* **2019**, arXiv:1909.12605.
24. Zhang, Y.; Wang, C.; Wang, X.; Liu, W.; Zeng, W. VoxelTrack: Multi-Person 3D Human Pose Estimation and Tracking in the Wild. *arXiv* **2021**, arXiv:2108.02452. [CrossRef]
25. Wojke, N.; Bewley, A.; Paulus, D. Simple Online and Realtime Tracking with a Deep Association Metric. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 3645–3649.
26. Abebe, M.; Shin, Y.; Noh, Y.; Lee, S.; Lee, I. Machine Learning Approaches for Ship Speed Prediction towards Energy Efficient Shipping. *Appl. Sci.* **2020**, *10*, 2325. [CrossRef]
27. Electronics, A.; Electric, M.; Kamakura, C.; Kamimachiya, W.; Kanagawa, K. Radar Speed Monitoring System. In Proceedings of the VNIS'94-1994 Vehicle Navigation and Information Systems Conference, Yokohama, Japan, 31 August–2 September 1994; pp. 89–93. [CrossRef]
28. Musayev, E. Laser-Based Large Detection Area Speed Measurement Methods and Systems. *Opt. Lasers Eng.* **2007**, *45*, 1049–1054. [CrossRef]

29. Li, J.; Chen, S.; Zhang, F.; Li, E.; Yang, T.; Lu, Z. An Adaptive Framework for Multi-Vehicle Ground Speed Estimation in Airborne Videos. *Remote Sens.* **2019**, *11*, 1241. [CrossRef]
30. Chen, X.; Liu, S.; Liu, R.W.; Wu, H.; Han, B.; Zhao, J. Quantifying Arctic Oil Spilling Event Risk by Integrating an Analytic Network Process and a Fuzzy Comprehensive Evaluation Model. *Ocean Coast. Manag.* **2022**, *228*, 106326. [CrossRef]
31. Wolsing, K.; Roepert, L.; Bauer, J.; Wehrle, K. Anomaly Detection in Maritime AIS Tracks: A Review of Recent Approaches. *J. Mar. Sci. Eng.* **2022**, *10*, 112. [CrossRef]
32. Chen, X.; Xu, X.; Yang, Y.; Huang, Y.; Chen, J.; Yan, Y. Visual Ship Tracking via a Hybrid Kernelized Correlation Filter and Anomaly Cleansing Framework. *Appl. Ocean Res.* **2021**, *106*, 102455. [CrossRef]
33. Ren, Y.; Yang, J.; Zhang, Q.; Guo, Z. Ship Recognition Based on Hu Invariant Moments and Convolutional Neural Network for Video Surveillance. *Multimed. Tools Appl.* **2021**, *80*, 1343–1373. [CrossRef]
34. Suliva, R.S.S.; Valencia, C.A.A.; Villaverde, J.F. Classification and Counting of Ships Using YOLOv5 Algorithm. In Proceedings of the 2022 6th International Conference on Communication and Information Systems (ICCIS), Chongqing, China, 14–16 October 2022; pp. 153–158. [CrossRef]
35. Jia, Z.; Su, X.; Ma, G.; Dai, T.; Sun, J. Crack Identification for Marine Engineering Equipment Based on Improved SSD and YOLOv5. *Ocean Eng.* **2023**, *268*, 113534. [CrossRef]
36. Jie, Y.; Leonidas, L.; Mumtaz, F.; Ali, M. Ship Detection and Tracking in Inland Waterways Using Improved Yolov3 and Deep Sort. *Symmetry* **2021**, *13*, 308. [CrossRef]
37. Zhang, Z. Flexible Camera Calibration by Viewing a Plane from Unknown Orientations. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, 20–27 September 1999; Volume 1, pp. 666–673. [CrossRef]
38. Prasad, D.K.; Rajan, D.; Rachmawati, L.; Rajabally, E.; Quek, C. Video Processing from Electro-Optical Sensors for Object Detection and Tracking in a Maritime Environment: A Survey. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 1993–2016. [CrossRef]
39. Liu, R.W.; Yuan, W.; Chen, X.; Lu, Y. An Enhanced CNN-Enabled Learning Method for Promoting Ship Detection in Maritime Surveillance System. *Ocean Eng.* **2021**, *235*, 109435. [CrossRef]
40. Wei, C.; Wang, W.; Yang, W.; Liu, J. Deep Retinex Decomposition for Low-Light Enhancement. *arXiv* **2019**, arXiv:1808.04560.
41. Yadav, G.; Maheshwari, S.; Agarwal, A. Contrast Limited Adaptive Histogram Equalization Based Enhancement for Real Time Video System. In Proceedings of the 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Delhi, India, 24–27 September 2014; pp. 2392–2397. [CrossRef]
42. Wang, Z.; Bovik, A.C. Mean Squared Error: Love It or Leave It? A New Look at Signal Fidelity Measures. *IEEE Signal Process. Mag.* **2009**, *26*, 98–117. [CrossRef]
43. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef]
44. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. *SSD: Single Shot Multibox Detector*; Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Springer: Cham, Switzerland, 2016; 9905 LNCS; pp. 21–37.
45. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef]
46. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection 2020. *arXiv* **2020**, arXiv:2004.10934.
47. Kim, J.H.; Kim, N.; Park, Y.W.; Won, C.S. Object Detection and Classification Based on YOLO-V5 with Improved Maritime Dataset. *J. Mar. Sci. Eng.* **2022**, *10*, 377. [CrossRef]

Journal of
Marine Science and Engineering

Article

Research on Multi-Port Ship Traffic Prediction Method Based on Spatiotemporal Graph Neural Networks

Yong Li [1], Zhaoxuan Li [1], Qiang Mei [2,3,*], Peng Wang [2,4,*], Wenlong Hu [5], Zhishan Wang [1], Wenxin Xie [1], Yang Yang [3] and Yuhaoran Chen [1]

[1] Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China; 13521988159@163.com (Z.L.)
[2] Merchant Marine College, Shanghai Maritime University, Shanghai 201306, China
[3] Navigation College, Jimei University, Xiamen 361021, China
[4] Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100086, China
[5] School of Computer Science, University of Auckland, Auckland 1010, New Zealand
* Correspondence: meiqiang@jmu.edu.cn (Q.M.); wangp@ict.ac.cn (P.W.)

Abstract: The intelligent maritime transportation system has emerged as a pivotal component in port management, owing to the rapid advancements in artificial intelligence and big data technology. Its essence lies in the application of digital modeling techniques, which leverage extensive ship data to facilitate efficient operations. In this regard, effective modeling and accurate prediction of the fluctuation patterns of ship traffic in multiple port regions will provide data support for trade analysis, port construction planning, and traffic safety management. In order to better express the potential interdependencies between ports, inspired by graph neural networks, this paper proposes a data-driven approach to construct a multi-port network and designs a spatiotemporal graph neural network model. The model incorporates graph attention networks and a dilated causal convolutional architecture to capture the temporal and spatial dimensions of traffic variation patterns. It also employs a gated-mechanism-based spatiotemporal bi-dimensional feature fusion strategy to handle the potential unequal relationships between the two dimensions of features. Compared to existing methods for port traffic prediction, this model fully considers the network characteristics of the overall port and fills the research gap in multi-port scenarios. In the experiments, real port ship traffic datasets were constructed using data from the Automatic Identification System (AIS) and port geographical information data for model validation. The results demonstrate that the model exhibits outstanding robustness and performs well in predicting traffic in multiple sub-regional port clusters.

Keywords: spatiotemporal graph neural network; traffic flow prediction; ship big data; AIS; port traffic prediction

Citation: Li, Y.; Li, Z.; Mei, Q.; Wang, P.; Hu, W.; Wang, Z.; Xie, W.; Yang, Y.; Chen, Y. Research on Multi-Port Ship Traffic Prediction Method Based on Spatiotemporal Graph Neural Networks. *J. Mar. Sci. Eng.* **2023**, *11*, 1379. https://doi.org/10.3390/jmse11071379

Academic Editor: Mihalis Golias

Received: 3 June 2023
Revised: 3 July 2023
Accepted: 4 July 2023
Published: 6 July 2023

1. Introduction

Maritime transportation, characterized by its substantial carrying capacity, cost-effectiveness, and remarkable adaptability to diverse environmental conditions, plays a significant role in international trade as an important mode of transportation. The accurate prediction of ship traffic flow holds immense significance, as it offers invaluable data support for the advancement of the national maritime industry and the strategic planning of international trade initiatives. Furthermore, it plays a crucial role in facilitating port layout planning, thereby addressing the prevalent challenge of aligning the port capacity with the escalating number of vessels arising from the rapid growth of the maritime industry. Additionally, this predictive capability contributes to the reduction of traffic congestion and the mitigation of accidents within maritime areas, consequently enhancing the overall efficiency of port infrastructure utilization [1].

This paper primarily concentrates on the prediction of maritime traffic flow in multi-port scenarios. Traffic flow prediction entails a typical task of time series forecasting, wherein the objective is to capture inherent patterns within traffic flow and speed data, characterized by their temporal sequences. By effectively discerning such patterns, it becomes feasible to infer future traffic flow states [2]. The primary challenge in time series prediction lies in accurately extracting a consistent temporal pattern. In practice, numerous factors, including environmental fluctuations, seasonal variations, and unforeseen accidents, can disrupt these theoretically stable patterns, thereby adding a considerable level of complexity to time series prediction problems. Consequently, time series prediction methods necessitate the incorporation of diverse influencing factors [3].

In the past, linear methods such as Autoregressive Integrated Moving Average (ARIMA) models and historical average analysis were limited in handling complex external factors. Scholars have attempted to use nonlinear machine learning methods such as backpropagation networks [4] and support vector machines [5] to uncover hidden patterns in time series data. However, these methods often suffer from slow convergence and algorithmic incompleteness, resulting in unsatisfactory solutions to time series prediction problems.

Consequently, researchers have turned to the application of deep learning techniques as a means to tackle practical time series prediction challenges. Deep neural networks exhibit the inherent capability to capture and model nonlinear relationships. By leveraging the stacking of multiple layers, these models can effectively capture a multitude of hidden factors and intricate variables within the data. Recurrent neural networks (RNN) [6], long short-term memory (LSTM) [7], and subsequent models such as gated recurrent units (GRU) [8], WaveNet [9], and Transformer [10,11], have demonstrated outstanding performance in the field of time series prediction.

With the rapid development of graph neural networks in recent years, spatial-convolution-based graph neural network models led by graph convolutional networks (GCN) [12] and graph attention networks (GAN) [13] have gained significant attention. In graph neural networks, a graph is composed of a set of vertices (nodes) that are interconnected by a set of edges. By performing convolutions on graphs with arbitrary structures, graph neural networks can learn rich spatial features. In recent years, they have been successfully applied in various domains, such as trajectory prediction [14,15] and traffic flow prediction [16].

In traffic flow prediction methods, conventional approaches often consider traffic intersections or sensors as entities, represented as individual nodes. Simultaneously, the road network, which accommodates these traffic entities, serves as the edges that symbolize the relationships between these nodes. Graph neural-network-based traffic flow prediction methods aggregate traffic flow information from neighboring nodes within a local spatial range, thereby predicting the potential information of nodes in the traffic subnetwork.

Nevertheless, relying solely on spatial information to predict traffic flow states is not rigorous and can lead to a significant loss of temporal information. To address this limitation, researchers have extended their efforts by incorporating temporal prediction learning methods into graph neural networks [17]. By leveraging techniques such as gate units and recurrent neural networks (RNN), they have introduced “spatiotemporal graph” models, which effectively capture both the spatial and temporal features of traffic flow prediction [18,19]. These spatiotemporal graph models take into account both temporal patterns and the spatial correlation structure of traffic entities. They can effectively capture the underlying meanings and interactions of nodes and edges within complex systems, resulting in superior prediction performance in traffic flow prediction.

In recent years, numerous excellent spatiotemporal graph models have been proposed and applied in the field of urban traffic prediction. Representative works include Multivariate Time Series Forecasting with Graph Neural Networks (MTGNN) [20], Diffusion Convolutional Recurrent Neural Networks (DCRNN) [21], Spatiotemporal Graph Convolutional Networks (STGCN) [18], and others.

However, the application traffic flow prediction methods in the maritime domain still faces two major challenges. Firstly, there is a scarcity of public datasets that can be directly used for analysis. The process of constructing maritime traffic flow datasets is arduous and complex. Secondly, the maritime traffic flow prediction scenarios do not possess the same traffic network structures as urban settings, rendering it difficult to directly apply spatiotemporal graph methods developed for urban traffic flow prediction [22]. Existing port flow prediction methods typically focus on single-port scenarios and do not fully consider the characteristics of port networks. They fail to consider the patterns of flow changes from the perspective of the overall port distribution structure, resulting in a relative scarcity of research in multi-port scenarios [23].

The scarcity of maritime traffic datasets stems primarily from the extensive geographical distribution, intricate structural arrangements, and challenges in regulating the maritime networks comprising ports across diverse countries. The widespread adoption of the Automatic Identification System (AIS) has significantly addressed the data gap in the maritime domain. The AIS is an onboard broadcast response system that has been gradually deployed on international vessels since 2002 and now has achieved widespread coverage through satellite networks. Vessels equipped with AIS devices regularly transmit navigational status data such as position points, speed, heading, and identity to ground-based stations and satellite receivers [24]. This enables data exchange between vessels and assists in navigation.

Despite the development of AIS, there are significant challenges related to the noise contamination in the raw AIS data, which limits their application in maritime traffic systems [25]. In this study, we address this issue by combining AIS data with spatial information from various ports worldwide and employing various big data processing techniques, similar to [26]. Through these approaches, we are able to effectively filter out noisy AIS data to the maximum extent possible and reconstruct realistic maritime traffic flow scenarios. As a result, we have constructed a multi-port flow dataset that can assist in conducting in-depth research on ship traffic flow prediction and validating the effectiveness of models under fair conditions.

The second challenge is that most existing research focuses on specific and single-port traffic flow scenarios, while studies on wide-scale, networked, and multi-port ship traffic flow prediction are relatively scarce. For example, ref. [27] combines Kalman filtering with regression analysis to improve short-term ship traffic flow prediction performance. Ref. [28] proposes a multi-variable extended CNN model based on convolutional methods, which specifically considers the impact of extreme weather events on ship traffic flow changes. However, the experimental settings in [28] involve geographically close ship traffic statistical areas, failing to consider the complex correlation structures among multiple regions. Ref. [29] uses AIS data to analyze the hourly ship traffic volume in a specific area near Ningbo and achieves ship traffic flow prediction for multiple time periods using an improved GRU-based time series prediction model. Nevertheless, it is also limited to the task of predicting traffic flow in a single maritime area.

This article tackles the challenge of ship traffic forecasting in a scenario involving multiple ports, employing a spatiotemporal graph model. Our proposed approach adopts a data-driven methodology that leverages actual data to construct a comprehensive multi-port graph structure, thereby establishing a realistic representation of the traffic network for the accurate prediction of maritime traffic flow. Subsequently, we utilize graph neural networks to capture the intricate message-passing patterns occurring between nodes within the port network. This enables the timely identification of abnormal traffic fluctuations in neighboring port nodes, thus enhancing the model's capacity to capture and exploit complex temporal dependencies. The incorporation of these techniques leads to improved prediction performance and effectively addresses the need for coordinated management in the context of multiple ports.

In summary, this paper contributes to the research in the following ways:

- It proposes a data-driven method for the construction of a multi-port network based on historical data and creates a realistic port ship traffic dataset using AIS data.
- It presents a traffic prediction model specifically designed for maritime scenarios, utilizing a spatiotemporal graph neural network. The novel model addresses the issue of imbalanced temporal and spatial features in the spatiotemporal dataset.

- The improved model's effectiveness is validated through experiments conducted on multiple port group datasets.

2. Methods

This study focuses on the task of forecasting ship traffic flow in multiple ports, which is a typical time series prediction problem. This problem can be formulated as follows:

$$(X_{t-\tau+1}, X_{t-\tau+2}, \ldots, X_t) \xrightarrow{f(*)} (X_{t+1}, X_{t+2}, \ldots, X_{t+n}) \tag{1}$$

As mentioned above, in this task, the model is given historical data of τ consecutive time steps as input. After being transformed through the function $f(*)$, the model predicts the future states of port traffic data for n consecutive time steps. Here, N represents the number of ports, C represents the feature dimension of the data, and $X_t \in \mathbb{R}^{N \times C}$ represents the traffic state of each port at time t. Since the purpose of this study is to forecast the inbound and outbound traffic of ports, the initial value of C is 2 (for outbound traffic and inbound traffic).

2.1. Spatiotemporal Blocks

In the context of multi-port traffic flow prediction, we consider each individual port as a distinct node and employ a data-driven methodology to establish edge relationships between ports, thereby constructing the port network based on historical data. The collection of nodes and edges represents the graph structure of the multi-port network.

The previous study [30] presented in this paper demonstrated that existing spatiotemporal graph models do not exhibit significant advantages over traditional time series prediction models when applied to multi-port traffic flow prediction. The spatial exploration capability of the spatiotemporal graph network was found to be underutilized. We propose two potential explanations for these experimental results. Firstly, the multi-port network structure in this scenario may inadequately represent the connectivity and spatial relationships between ports. Secondly, there may exist an inherent imbalance between the temporal and spatial features, with the roles of time and space varying across different datasets. To address the first conjecture, we optimize the existing data-driven method for the construction of the multi-port network structure. Additionally, we conduct theoretical research based on the second conjecture and make targeted improvements to the existing spatiotemporal graph framework.

Firstly, this paper summarizes the existing general framework of spatiotemporal graphs, as shown in Figure 1. The combination of the temporal layer and the spatial layer is referred to as a "spatiotemporal block." In the spatiotemporal block, the temporal layer captures the temporal patterns of historical traffic flow data for each port, while the spatial layer handles the relationships among port nodes across different time dimensions. The sequential arrangement of multiple spatiotemporal blocks forms a deep spatiotemporal graph neural network model.

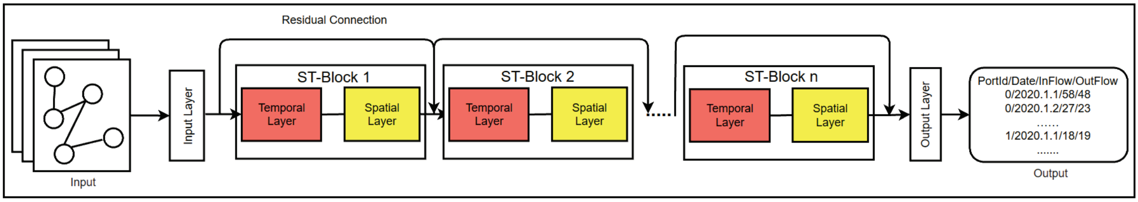


Figure 1. General framework of spatiotemporal graph model.

From a holistic perspective, the framework directly utilizes the output of the temporal layer as the input for the subsequent spatial layer, and the output of the spatial layer becomes the input for the next temporal layer. After several iterations of these spatiotemporal blocks, the predicted results are generated through the output layer. There is no specific handling between the temporal and spatial layers in this architecture. The advantage of this approach lies in its ability to fully preserve the features processed by the temporal and spatial layers. However, it overlooks the potential master–slave relationship between the temporal and spatial layers.

Inspired by [31], this paper employs a gating mechanism to determine the propagation and forgetting of information, in order to simulate the imbalance between time and space. As shown in Figure 2, the model first duplicates the input tensor into two identical copies. One copy is processed through the temporal layer and then passed through a *tanh* activation function for output. The other copy is processed through the spatial layer and mapped to the range of 0 to 1 using a *sigmoid* function, treating it as a filtering net for spatial information. Finally, the two parts of the output are combined using element-wise multiplication (Hadamard product). In other words, this structure treats the spatial hidden features as a filter, using spatial features to filter important temporal features, which are then outputted in the form of temporal features to enter the next hidden layer. This structure enhances the model's sensitivity to temporal flow data while reducing the influence of the graph structure on feature information. It can effectively simulate the imbalance between time and space.

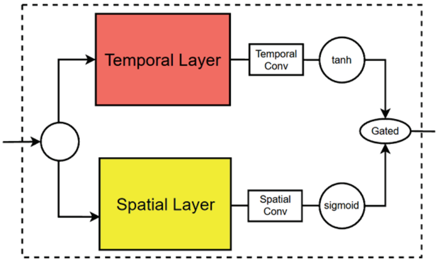


Figure 2. Structure of a spatiotemporal block.

In summary, the spatiotemporal graph framework used in this paper is illustrated in Figure 3.

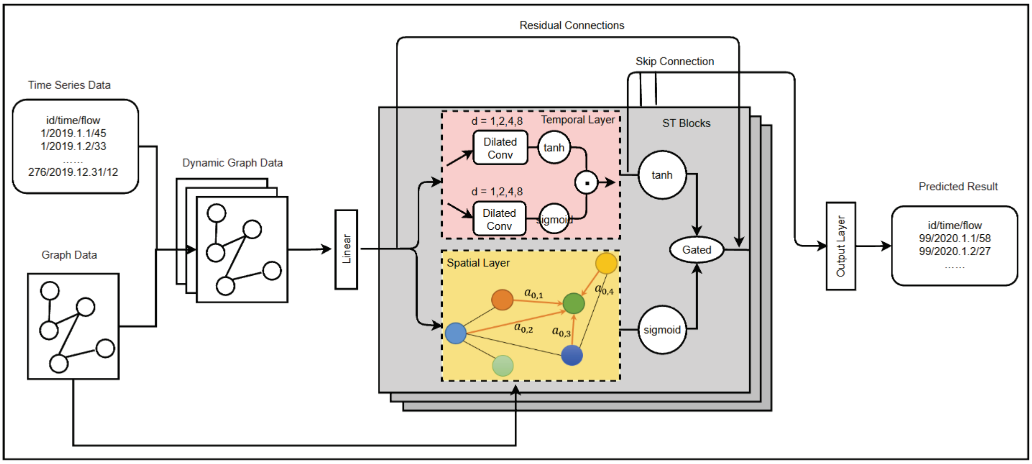
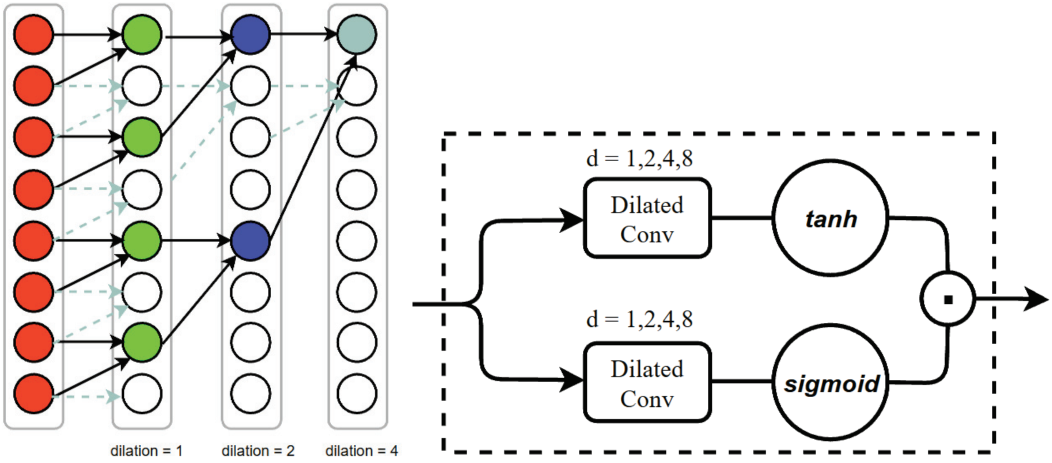


Figure 3. Overall framework of the Gated Attention Graph WaveNet model.

2.2. Temporal Layer

The fundamental concept behind the temporal layer is based on dilated convolution. By incorporating dilated convolution, the model can effectively expand its receptive field while maintaining the sequential nature of data modeling. This allows the model to capture longer temporal feature patterns with fewer computational costs, thereby capturing temporal dependencies over a larger time span.

Dilated convolution is a convolutional operation. As shown in Figure 4a, during the convolution process, fixed-sized "holes" are introduced between the elements of the convolution kernel, expanding the kernel. This means that, with the same computational cost, a larger effective filter size is used for convolution. Compared to a convolutional kernel of the same size, dilated convolution offers higher computational efficiency. In particular, dilated convolution with a dilation factor of 1 is equivalent to a regular convolution.



(**a**) (**b**)

Figure 4. Structure of temporal layer. (**a**) Stacked dilated convolutional layers; (**b**) WaveNet structure with gating mechanism.

To accommodate different time step lengths, this paper adopts a dilation factor pattern of "1, 2, 4, 8" as the base and cycles through it as the model's hidden layers increase. Let us apply this to the scenario of port traffic flow prediction, assuming that the input historical flow data have a time step length of 14 and the convolution kernel size is 3.

In the first layer of the temporal layer, the dilation factor is 1, which means that fine-grained time feature processing is performed at the level of 1 time step. As the network goes deeper, the second layer of the temporal layer has a dilation factor of 2; the process of applying time convolutions involves making leapfrog-like strides over the input sequence. This approach results in less fine-grained time feature processing, but, correspondingly, the same-sized convolutional kernel can handle twice the length of time steps.

By repeating this process, the temporal convolutional layer with a dilation factor of 8 can cover the entire input sequence, thereby extracting long-term historical features. This enables the model to capture temporal dependencies over a larger time span.

On the other hand, as a key to the success of recurrent neural networks, gating mechanisms preserve the non-linear capabilities while addressing the vanishing gradient problem, leading to better performance in tasks involving long-term dependencies. WaveNet [9] incorporates gating activation structures into causal convolutions, proposing the WaveNet structure shown in Figure 4b. The formula can be represented as

$$h_l(X) = tanh(X * W) \odot \delta(X * V) \tag{2}$$

where X represents the feature tensor, and l denotes the current layer. W and V represent two dilated convolution kernels, $\delta(*)$ represents the *sigmoid* function, and $\odot$ denotes the element-wise multiplication operator.

Based on this gating structure, the temporal layer can easily perform feature selection on the input data. Compared to linear processing methods, this nonlinear structure possesses stronger modeling capabilities for time series features.

2.3. Spatial Layer

The non-Euclidean nature, irregularity, and sparsity of graphs pose challenges in directly applying convolutional neural networks (CNNs) to process graph features. This limitation has impeded progress in graph representation within the field of machine learning. However, in recent years, researchers have made strides by combining spectral graph theory with Fourier transforms, allowing the definition of convolutional kernels in the spectral domain. This breakthrough has enabled graph convolutions in the spectral domain and has laid the foundation for graph convolutional networks (GCN) [12]. By simplifying the computation process, GCNs have facilitated rapid advancements in graph neural network models.

Subsequently, the graph attention network (GAT) [13] was proposed as a graph neural network model based on attention mechanisms. GAT controls the aggregation of information from nodes and edges by assigning different learning weights to their neighbors. This allows the model to extract more valuable hidden features. Compared to GCN, GAT offers a more flexible node feature aggregation process, addressing the limitation of GCN in which the fixed adjacency matrix prevents the graph structure from being expandable.

GAT can be divided into two stages: Stage 1 involves computing global similarity coefficients, while Stage 2 focuses on computing node features based on local attention coefficients.

Take a graph G = (V, E) with N port nodes as an example, where the historical flow features of each port can be represented as $X_i \in \mathbb{R}^{T\times C}$, with T being the time dimension and C being the feature dimension. In the graph attention layer, a trainable shared weight matrix W is first used to linearly transform the initial features of all nodes. Based on the edge relationships, the transformed node features are then used to calculate the similarity coefficients e_{ij} between adjacent nodes on each edge. The formula for this stage is as follows, where N is the set of first-order neighboring nodes of the target node i, l is the current neural network layer, and the mapping function $a(*)$ is used to compute the similarity between the flow features of port nodes i and j.

$$e_{ij}^{(l)} = a\left(\left[W^{(l)}X_i^{(l)} \parallel W^{(l)}X_j^{(l)}\right]\right), j \in \mathcal{N}_i \tag{3}$$

In the aggregation stage, the model uses $softmax$ to normalize the similarity coefficients of all nodes within the first-order neighborhood of the target node i. The normalized coefficients represent the attention coefficients for each node within the neighborhood. The specific formula for the calculation of the attention coefficients is as follows:

$$\alpha_{ij}^{(l)} = \frac{\exp\left(LeakyReLU\left(e_{ij}^{(l)}\right)\right)}{\sum\limits_{k\in\mathcal{N}_i} exp\left(LeakyReLU\left(e_{ik}^{(l)}\right)\right)} \tag{4}$$

Subsequently, the features of each node are weighted and aggregated based on the calculated attention coefficients, resulting in the new feature $X_i^{(l)}$ of the central node i at layer l. Here, $\sigma(*)$ represents an optional activation function.

$$h_l(X_i) = \sigma\left(\sum_{j\in\mathcal{N}_i} \alpha_{ij}^{(l)} W^{(l)} X_j^{(l)}\right) \tag{5}$$

Repeating Formulas (4) and (5), we perform feature aggregation for all nodes in the graph, completing the processing of a GAT layer.

In summary, the overall calculation formula for the spatiotemporal block of the model can be summarized as follows:

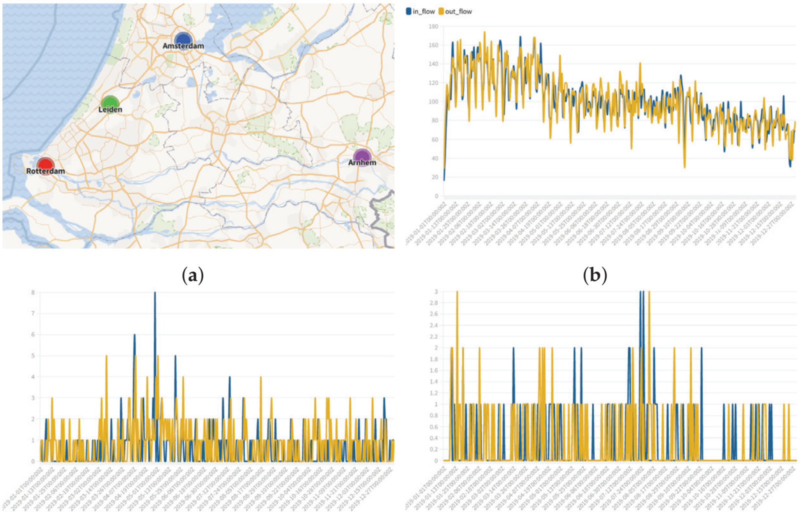
$$h_l\left(X^{(l-1)}\right) = tanh\left(WaveNet\left(X^{(l-1)}\right) * W + b\right) \odot \delta\left(GAT\left(X^{(l-1)}\right) * V + c\right) \quad (6)$$

3. Scenario and Data Sources

The experimental dataset in this study consists of 2019 AIS data and global port geospatial data.

Firstly, we employed big data techniques to process the vast amount of raw AIS data. To ensure data accuracy, we cross-referenced the information with vessel Lloyds Register profiles and applied data cleaning techniques to eliminate erroneous data. Subsequently, we integrated the trajectory data with spatial boundary information from global port geospatial data. This integration enabled us to infer the departure and arrival details of vessels, resulting in the creation of a global vessel port origin–destination (OD) dataset for the year 2019. The dataset comprised 154,205 vessels, 2697 ports, and a total of 5,275,645 records. Finally, leveraging the origin and destination port IDs, along with the recorded arrival and departure times in the OD dataset, we calculated the daily inflow and outflow of vessels for each port. This enabled us to quantify the daily traffic volume of ships entering and leaving each port.

Following the acquisition of historical port traffic data, the creation of a multi-port spatial graph structure emerged as a pressing research challenge. Figure 5 illustrates that employing the physical distance alone as the basis for constructing the port graph structure would result in a significant loss of information. For instance, in the case of the Netherlands, Rotterdam Port exhibits considerably different average daily vessel traffic in comparison to nearby ports such as Leiden Port and Arnhem Port. This difference is also reflected in the scale of the ports, where Rotterdam Port is classified as a large-sized port with a maximum draught of 18 m, whereas Leiden Port and Arnhem Port fall under the category of medium-sized ports. Such disparities would introduce substantial errors during the feature aggregation stage within the graph convolutional layer.



(**a**) (**b**)

(**c**) (**d**)

Figure 5. Fluctuation in ship flow between different ports in the Netherlands. (**a**) The locations of four closely located ports within the Netherlands; (**b**) Rotterdam Port 2019 daily inbound and outbound traffic curve; (**c**) Arnhem Port traffic curve; (**d**) Leiden Port traffic curve.

To tackle this problem, the present study introduces a methodology for the construction of a dynamic port correlation graph utilizing historical data. The fundamental concept behind this approach is to employ actual maritime traffic patterns as an evaluative criterion. By quantifying the frequency of vessel movement between two ports, a spatial interconnection relationship is established, thereby creating an extensive multi-port correlation network.

Specifically, the initial step involves utilizing the previously generated 2019 global vessel port OD dataset, which is derived from the flow data calculation procedure, as the foundation for the construction of the graph. The direct count of vessel traffic between each pair of ports is computed on a monthly basis, encompassing global coverage. For the purpose of experimentation, three months of OD data are randomly selected for the computation process. A unidirectional edge relationship is established between two ports if the vessel traffic count surpasses a predefined threshold. In the experiment, this threshold was set to 10 to distinguish the level of closeness in the navigational relationship between two ports. To prevent the graph structure from becoming overly dense or sparse, we comprehensively consider the port scale and previous experiments, and determine the optimal range for this threshold as 5 to 10, guided by prior knowledge, which allows us to achieve the optimal solution in terms of computational costs and experimental effectiveness.

Moreover, taking into account the extensive number of ports globally and the substantial volume of OD data involved, we adopt a strategy to mitigate the computational overhead. This involves constructing independent sub-networks comprising port clusters, with several prominent ports across the world serving as central hubs. In our experiment, the central hubs selected were Rotterdam Port, Shanghai Port, Singapore Port, Boston Port, Antwerp Port, Hong Kong Port, and Incheon Port. This approach allows for the more efficient processing of the data while still capturing the essential connectivity among ports.

Using Rotterdam Port as an illustration, we initially filter the OD dataset to include only maritime shipping data where Rotterdam Port is either the origin or destination. We then identify the destination (or origin) ports that exhibit a mutual traffic count that satisfies the threshold constraint, thereby indicating direct connections to Rotterdam. Through this procedure, we identify a total of 250 port nodes that are directly linked to Rotterdam Port.

Subsequently, we shift our attention to these 251 ports and proceed to compute the direct vessel traffic count between each pair of ports. Employing a similar methodology as before, if the traffic count surpasses the predefined threshold, a unidirectional edge relationship is established between the two ports.

Finally, we obtain a sub-network structure comprising multiple ports, with Rotterdam Port as the central hub. The corresponding adjacency matrix is generated as a result. Figure 6 provides a visual representation of the schematic diagram depicting the multi-port graph structure centered around Rotterdam Port.

The geographical coordinates and location information of the seven central ports are displayed in Figure 7. The comprehensive data attributes of the port cluster network scene, centered around these seven ports, are recorded in Table 1.

Table 1. Dataset attributes for seven port subnetworks.

Central Port Name	Port Nodes	Edges	Data Size
Rotterdam	251	3036	91,615
Shanghai	241	10,876	87,965
Singapore	388	10,017	141,620
Boston	121	1054	44,165
Antwerp	217	2706	79,205
Hong Kong	182	7731	66,430
Incheon	86	2733	31,390

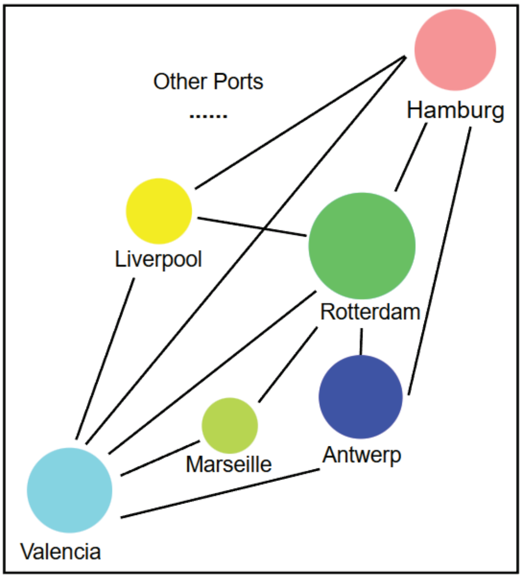


Figure 6. A multi-port graph structure with Rotterdam as the central port.

Figure 7. Distribution of locations of central ports.

4. Results

4.1. Experimental Environment

This study utilized a 2080ti GPU to conduct the experiments. To ensure a fair and consistent testing environment, the experiments were conducted on the Libcity platform [32], utilizing standardized hyperparameter configurations, with the hidden feature dimensions uniformly set to 32. The input window for time series data was defined as 21, while the output window was set to 7, meaning that the models aimed to predict the future 7-day traffic changes based on the preceding 21 days of data.

Specifically, the models employed in the study consisted of eight hidden layers. For models incorporating the multi-head attention mechanism, eight heads were used. In the case of models employing the diffusion convolution mechanism, the dilated factors were cyclically set as "1, 2, 4, 8".

4.2. Baseline and Metric

In the experiment, the baseline models were divided into two major categories, spatiotemporal graph models and temporal models, based on whether they handled spatial features. RNN, AE [33], Seq2Seq [34], WaveNet [9], and Transformer [10] are classical temporal prediction methods that have proven effective in various domains, such as NLP and traffic prediction. STGCN [18], AGCRN [35], and the proposed model GAGW in this paper belong to spatiotemporal graph models. On the other hand, based on different time layer processing strategies, models can be further categorized into RNN-based prediction models led by RNN and DCRNN [21], TCN models such as GWNET [36] and STMGAT [37], and self-attention-mechanism-based models such as STTN [38].

The evaluation metrics used in the experiment include commonly employed measures in traffic flow prediction, MAE, MAPE, and RMSE, as depicted in Equations (7)–(9), respectively. MAE, known as the mean absolute error, is a widely used performance metric in regression tasks, directly quantifying the average difference between predicted values and actual values. MAPE, built upon MAE, measures the average percentage error of the experimental results. RMSE, or the root mean square error, in comparison to MAE, is more sensitive to large errors, thus challenging the stability and robustness of the predictive model.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i| \tag{7}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right| * 100\% \tag{8}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \tag{9}$$

4.3. Experimental Results

In the experiment, we extracted the consecutive 21-day inflow and outflow data for each port from the test set as input and expected the model to generate predictions for the next 7 days of each port in a single output. Taking the Rotterdam sub-scenario as an example, which comprises 251 port nodes, the input should include an adjacency matrix with 251 nodes and an initial tensor of shape [32, 251, 21, 2]. Here, "32" represents the specified batch size, and "2" indicates that the initial features only include inflow and outflow. Correspondingly, the model's output should be a tensor of shape [32, 251, 7, 2]. By comparing the predicted 7-day traffic variations with the ground truth values, the predictive capability of the model can be evaluated.

As presented in Table 2, cross-validation experiments were conducted across seven major multi-port network scenarios centered around the seven central ports. The GAGW model proposed in this study incorporates a weighted relationship to address the imbalance between spatial and temporal features. Consequently, the model demonstrates consistent and superior predictive performance across all scenarios, achieving the best results in the majority of experimental scenarios.

Through a comprehensive comparison of the experimental results, it has been observed that the majority of the existing spatiotemporal graph models outperform the temporal prediction models. This indicates that the incorporation of graph neural networks indeed enhances the performance of predictive models in various scenarios and effectively captures intricate traffic flow patterns. Among the temporal prediction models, Transformer and WaveNet exhibit significantly superior performance compared to traditional temporal prediction models and even outperform certain spatiotemporal graph models in specific scenarios. This highlights the significance of capturing temporal patterns in maritime traffic flow prediction scenarios, thereby affirming the importance of time features in the field of traffic prediction.

Table 2. Results of multi-port dataset.

	Rotterdam			Boston			Antwerp			Hong Kong		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE
RNN	4.5813	0.6493	23.1354	3.2588	0.6722	6.4718	2.5007	0.6392	4.9709	16.3516	1.0782	33.7049
Auto Encoder	4.084	1.1835	11.4571	3.2034	0.625	6.3242	2.3923	0.6687	4.4949	18.3066	1.7544	32.9607
Seq2Seq	4.8228	0.6932	24.695	3.3392	0.6943	6.7335	2.4829	0.6209	5.0688	16.6819	0.9056	33.5273
WaveNet	2.5465	**0.4315**	9.4552	2.2362	0.4396	4.1007	1.8745	0.5385	3.2237	8.6431	**0.432**	18.5065
Transformer	2.5101	0.4891	**8.401**	2.1902	0.473	3.79	2.0819	0.5873	3.8074	8.5305	0.4564	17.6556
STGCN	3.2307	0.7186	9.5492	2.2647	0.498	3.9539	2.1825	0.5937	3.6725	13.1973	0.9862	27.4965
AGCRN	3.2396	0.5869	13.6182	2.5493	0.5111	4.7692	2.0179	0.5207	3.6471	9.8796	0.5501	21.5886
DCRNN	2.8718	0.616	9.8986	2.1044	0.4462	3.8234	1.7907	0.4909	3.191	8.9693	0.5549	18.5956
STTN	3.8664	0.6449	12.9845	2.2696	0.4827	4.0493	2.3098	0.5858	4.1153	16.9211	0.9443	33.6544
STMGAT	2.762	0.5231	8.6171	2.2309	0.4816	3.8571	1.7959	0.5038	3.0626	9.0416	0.6927	17.7191
GWNET	2.4831	0.4498	9.0111	2.1861	0.4421	4.0143	1.9133	0.5216	3.4256	8.5496	0.4489	17.6672
GAGW	**2.3964**	0.4485	8.514	**2.0983**	**0.434**	**3.7663**	**1.7669**	**0.4719**	**3.0236**	**8.216**	0.4436	**17.2679**

	Shanghai			Singapore			Incheon		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE
RNN	18.1033	1.8698	36.8206	10.1521	0.8326	31.2282	18.4991	1.3141	37.6816
Auto Encoder	17.5539	1.8645	35.2272	8.5357	0.7448	24.0477	19.6817	1.8012	37.5785
Seq2Seq	14.8196	1.1846	29.9094	10.1913	0.8657	31.0303	19.0404	1.3452	40.7782
WaveNet	7.4423	0.4437	15.98	5.1619	0.5057	14.8807	8.7669	**0.4229**	19.9683
Transformer	7.5331	0.4567	15.9231	**4.3012**	0.4288	**11.5323**	**8.5554**	0.4245	**18.5527**
STGCN	11.8722	0.6802	25.1572	7.9983	1.8568	16.0688	11.8275	0.783	26.6319
AGCRN	8.5064	0.6398	17.7152	4.7981	0.5475	12.9965	9.6283	0.5119	20.6766
DCRNN	7.6933	0.6046	16.1944	4.8215	0.5592	12.6221	8.9704	0.5052	19.4997
STTN	14.6455	1.1332	30.7543	9.279	0.8025	25.8419	19.6088	1.2163	40.3185
STMGAT	7.6263	0.4518	15.718	4.9113	0.65	12.3215	9.7628	0.7354	19.699
GWNET	7.525	0.5259	16.2727	4.426	0.5077	11.8064	9.4297	0.4956	22.2984
GAGW	**7.306**	**0.4418**	**15.6874**	4.3781	**0.428**	11.8228	8.7403	0.4672	19.8435

In the comparison of spatiotemporal graph models, the overall predictive performance of the STGCN model is relatively poor, which can be attributed to its use of a coarse-grained temporal processing strategy. The relatively simple gate convolution structure of STGCN is unable to fully capture the complex traffic flow variations in maritime scenarios, resulting in its performance being inferior to that of more sophisticated temporal prediction models such as Transformer.

In this experiment, the performance of the STTN model, which integrates self-attention mechanisms in both the temporal and spatial layers, did not meet our expectations. Despite adjusting certain hyperparameters, we were unable to obtain satisfactory experimental results. We speculate that the underlying reason for this lies in the spatial feature capturing strategy based on self-attention mechanisms. In the maritime context of this study, this strategy fails to effectively represent the data-driven, cross-spatial graph correlation structures, thereby leading to a decline in model performance.

The STMGAT, GWNET, and the proposed model in this paper all employ a time-convolution-based temporal layer processing strategy. The experimental results demonstrate that gate-based processing structures centered around causal convolutions can better capture the long-term variations in traffic flow data, enabling the fine-grained analysis of temporal features and obtaining more accurate prediction results.

The training curves of the top-performing three spatiotemporal graph models in the traffic flow scenario centered around the Rotterdam Port cluster are shown in Figure 8. The proposed model in this paper achieves more accurate predictive performance by utilizing a multi-head attention mechanism similar to STTN and STMGAT in the graph convolutional layer. However, this improvement in accuracy comes at the expense of sacrificing some training speed, resulting in a convergence speed that is not state-of-the-art. As illustrated in Figure 9, our model's traffic flow predictions exhibit better adherence to the actual flow variation patterns compared to other baseline models and demonstrate good fitting capability for significant flow fluctuation curves.

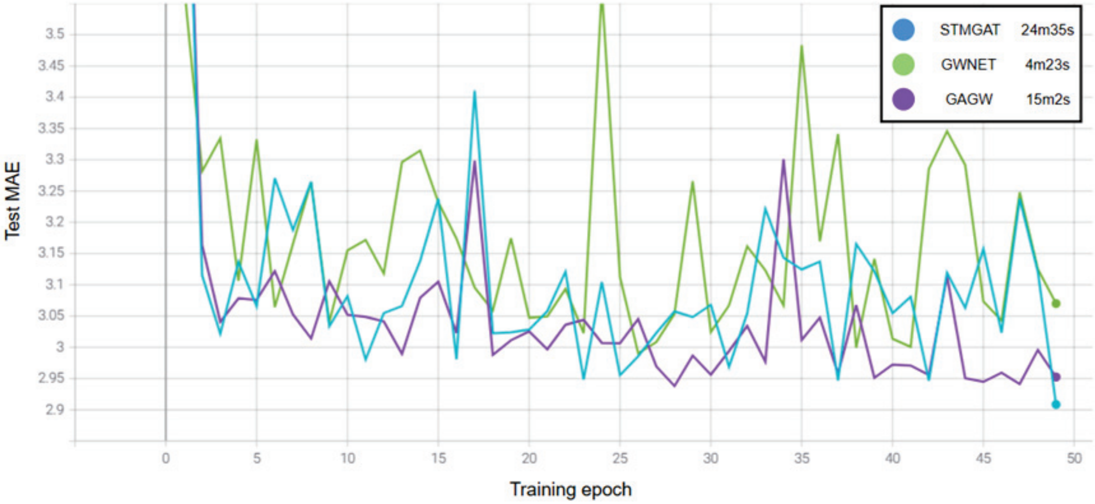


Figure 8. Comparison of MAE training curves for top three models.

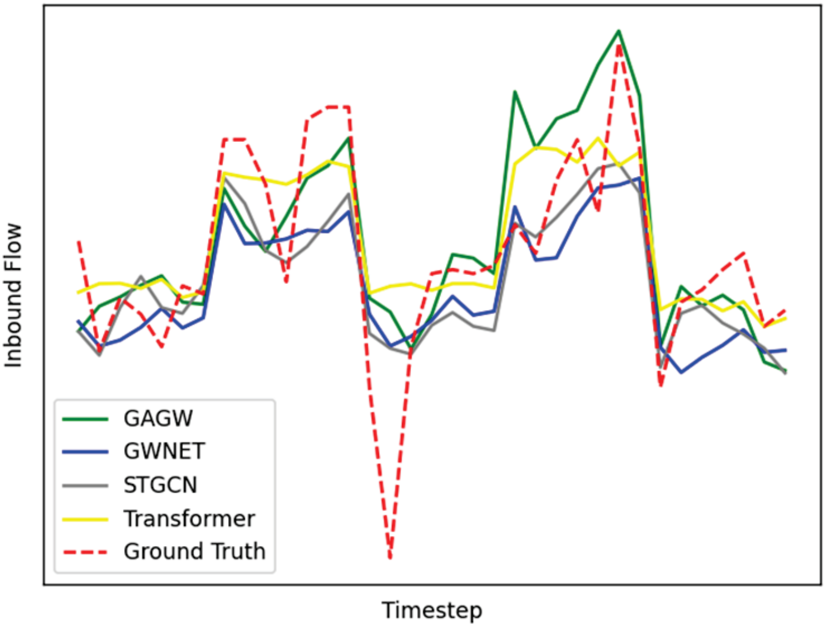


Figure 9. Comparison of GAGW and other models with ground truth inbound flow.

To further investigate the stability of the model in the small-sample maritime test scenario, we extracted the traffic flow data from June to September as a specific monthly dataset for independent testing (Table 3). These four months are characterized by substantial fluctuations in port traffic, without a discernible pattern in the context of international shipping. Moreover, the navigation relationships among ports during this period are more intricate, placing higher demands on the predictive capabilities of the models. In this challenging scenario, the GAGW model stands out among numerous spatiotemporal prediction models and exhibits exceptional adaptability to the limited coverage of the specific monthly dataset. This further confirms the effectiveness of the new architecture.

To validate the issue of the unequal treatment of spatial and temporal features, we conducted ablation experiments by modifying some existing spatiotemporal graph model architectures to incorporate the proposed gate-based mechanism for spatiotemporal feature fusion. The experimental results, as shown in Table 4, indicate that the spatiotemporal graph models with the inclusion of the feature fusion structure generally outperform the models with the original architectures in the port traffic prediction scenario of this paper. Among them, since the baseline models AGCRN and DCRNN both belong to iterative architectures based on RNN, it is difficult to incorporate the fusion module. Therefore, we introduced the ASTGCN [39] model, which simultaneously uses attention mechanisms and temporal convolution strategies, as the new baseline model.

To validate the relationship weights between time and spatial features, we conducted ablation experiments on the temporal and spatial layers of the GAGW model. In Table 5, "w/o T" represents the removal of all temporal hidden layers in the model, while "w/o S" indicates the removal of all graph structure processing methods. The experimental results demonstrate that ablating the temporal layer has a more severe impact on spatiotemporal graph models, leading to a significant decrease in predictive performance. In comparison, the performance decline is relatively limited when the spatial layer is ablated, further confirming the inequality between time and spatial features.

Table 3. Results of specific month dataset.

	Rotterdam			Boston			Antwerp			Hong Kong		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE
STGCN	10.0471	3.3148	19.1571	2.7766	0.558	5.0216	2.2405	0.5603	3.7479	21.7833	1.7334	44.5805
AGCRN	4.0285	0.5375	23.3363	2.7057	0.5186	5.118	**2.0334**	0.5032	3.7522	24.765	1.4236	53.009
DCRNN	6.7308	0.7575	36.2379	3.6936	0.5506	9.0707	3.5909	0.7169	9.1206	23.4573	1.6995	45.0641
STTN	7.3341	0.7816	37.1314	2.8498	0.508	5.6422	3.4785	0.6061	8.2283	25.1063	1.6549	50.5299
STMGAT	4.0649	0.5334	20.0261	2.7224	0.5165	5.0336	2.0679	**0.4968**	**3.6653**	18.2011	1.5736	39.1712
GWNET	3.62	**0.5294**	19.4332	2.7968	0.4875	5.3694	2.2134	0.5158	4.1839	14.0173	0.8861	31.1028
GAGW	**3.3688**	0.579	**15.5115**	**2.6928**	**0.4753**	**5.0081**	2.1446	0.5008	4.0861	**13.3074**	**0.8242**	**29.6616**

	Shanghai			Singapore			Incheon		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE
STGCN	21.8422	1.7012	45.8386	17.122	3.6693	36.7725	26.8699	1.8365	60.105
AGCRN	23.038	1.5282	47.7465	12.0759	0.9366	34.2086	21.9169	1.3855	44.5575
DCRNN	21.6196	1.8093	41.1723	14.5029	1.2927	39.0149	29.8365	1.6589	65.2048
STTN	22.5262	1.9444	44.5484	16.9381	1.2286	44.5798	23.6124	1.1277	59.0685
STMGAT	24.5565	2.6933	46.0391	6.9351	0.6712	**19.7745**	**13.7049**	0.6775	**32.9118**
GWNET	13.5886	**0.9734**	29.5113	6.9078	0.797	20.0405	15.5624	**0.6453**	40.8259
GAGW	**11.9513**	1.0716	**26.2667**	**6.7402**	**0.6319**	20.937	14.9141	0.6781	39.1262

Table 4. Ablation experiment on spatiotemporal feature fusion module.

	***w/o* Feature Fusion**			***w/* Feature Fusion**		
	MAE	**RMSE**	**R^2**	**MAE**	**RMSE**	**R^2**
STGCN	13.7484	23.7532	0.9145	**11.9819**	**21.2891**	**0.9319**
STTN	21.6699	36.0566	0.8089	**18.8857**	**33.1819**	**0.8384**
GWNET	**10.2954**	20.1326	0.9402	10.3233	**19.8169**	**0.943**
ASTGCN	13.0184	22.8163	0.9229	**11.3801**	**21.2428**	**0.9346**
GAGW	10.9752	20.2516	0.9385	**10.194**	**19.691**	**0.9428**

Table 5. Ablation experiment of temporal and spatial layers.

	MAE	**RMSE**	**R^2**
***w/o* T**	12.5244	21.3558	0.9323
***w/o* S**	10.2882	20.0338	0.9406
GAGW	**10.1879**	**19.9007**	**0.9412**

To demonstrate the superior performance of the proposed model in traffic flow prediction, we also conducted experiments on METR_LA, which is a renowned dataset comprising urban highway traffic speed data. As shown in Table 6, the GAGW model performs at an advanced level in the urban traffic dataset as well. This confirms that the phenomenon of unequal treatment between time and spatial features also exists in urban traffic scenarios.

Table 6. Results of METR_LA.

METR_LA	**RNN**	**AE**	**Seq2Seq**	**WaveNet**	**Transformer**	**STGCN**	**AGCRN**	**DCRNN**	**STTN**	**STMGAT**	**GWNET**	**GAGW**
MAE	4.11	4.53	3.71	3.98	3.81	3.86	3.19	3.12	3.49	4.15	3.17	**3.12**
RMSE	8.17	8.77	7.19	8.98	8.73	8.34	6.33	6.19	7.92	8.24	6.39	**6.11**

In summary, the proposed model has demonstrated strong predictive capabilities in various experiments conducted in the multi-port traffic flow prediction scenario.

5. Conclusions

In this study, a comprehensive dataset of real port traffic and a port topology structure were constructed using global AIS data, Lloyd's ship archive, and port geospatial data. To explore different modeling approaches, a variety of comparative models were reproduced using an open-source spatiotemporal graph model training framework [32].

Subsequently, a novel fine-grained multi-port traffic flow prediction spatiotemporal graph network model called GAGW was designed and implemented. This new model aimed to address the challenge of imbalanced temporal and spatial hidden features. Drawing inspiration from gate-based structures, the GAGW model processed temporal and spatial features in parallel, departing from the traditional sequential "time–space–time" processing pattern. The model incorporated distinct mapping functions for temporal and spatial feature vectors, placing greater emphasis on the influence of temporal features on prediction outcomes while reducing the sensitivity to spatial features.

To evaluate the effectiveness of the GAGW model, comparative experiments were conducted using maritime datasets, special monthly datasets, and urban road datasets. The results demonstrated the reliability and robustness of the proposed model. The GAGW model achieved advanced levels of prediction performance and robustness, among other evaluation metrics.

Overall, this research contributes to the field of port traffic flow prediction by introducing a novel spatiotemporal graph model that effectively handles imbalanced temporal and spatial features. The proposed GAGW model demonstrates superior prediction performance and adaptability, making it a promising approach for multi-port traffic forecasting applications.

Although the new framework has demonstrated the inequality of spatiotemporal features in experiments, there is room for further improvement in the construction of dynamic port graph structures. The proposed method for dynamic graph construction solely considers the spatial correlation between ports based on ship traffic, without incorporating the objective geographical relationships that exist between ports. In future work, the paper aims to focus on constructing flow datasets that encompass multi-port scenarios of practical significance, such as Europe and coastal regions of China.

Additionally, this paper only utilized AIS source data from the year 2019, which may be considered insufficient in capturing the full variability in port traffic patterns at a daily level. In future research, the study intends to incorporate AIS data from multiple consecutive years to augment the volume of data available for model training.

In summary, the future research endeavors of this paper will concentrate on two main aspects: optimizing the design of dynamic graph structures and refining the strategies for the processing of temporal layers in the model.

Author Contributions: Conceptualization, Y.L. and Z.L.; methodology, Z.L. and Q.M.; software, W.H., Z.W. and W.X.; validation, Y.L., Z.L. and Q.M.; formal analysis, Q.M.; investigation, Y.Y. and Y.C.; resources, Y.L. and P.W.; data curation, Z.L., W.H. and Z.W.; writing—original draft preparation, Z.L.; writing—review and editing, Y.L. and Q.M.; visualization, W.X.; supervision, Y.L. and P.W.; project administration, Y.L.; funding acquisition, Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China: [Grant Number 71804059]; Natural Science Foundation of Fujian Province: [Grant Number 2021J01821]; Shanghai Science and Technology Committee: [Grant Number 18DZ1206300]; the National Key Research and Development Program of China: [Grant Number 2018YFC1407400].

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are included within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Liu, R.W.; Chen, J.; Liu, Z.; Li, Y.; Liu, Y.; Liu, J. Vessel traffic flow separation-prediction using low-rank and sparse decomposition. In Proceedings of the 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), Yokohama, Japan, 16–19 October 2017; pp. 1–6.
2. Lara-Benítez, P.; Carranza-García, M.; Riquelme, J.C. An experimental review on deep learning architectures for time series forecasting. *Int. J. Neural Syst.* **2021**, *31*, 2130001. [CrossRef] [PubMed]
3. Yin, J.; Rao, W.; Yuan, M.; Zeng, J.; Zhao, K.; Zhang, C.; Li, J.; Zhao, Q. Experimental study of multivariate time series forecasting models. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management, Beijing, China, 3–7 November 2019; pp. 2833–2839.
4. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536. [CrossRef]
5. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
6. Nguyen, D.; Vadaine, R.; Hajduch, G.; Garello, R.; Fablet, R. A multi-task deep learning architecture for maritime surveillance using AIS data streams. In Proceedings of the 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA), Turin, Italy, 1–3 October 2018; pp. 331–340.
7. Gao, M.; Shi, G.; Li, S. Online prediction of ship behavior with automatic identification system sensor data using bidirectional long short-term memory recurrent neural network. *Sensors* **2018**, *18*, 4211. [CrossRef] [PubMed]
8. Shi, J.h.; Liu, Z.j. Deep learning in unmanned surface vehicles collision-avoidance pattern based on AIS big data with double GRU-RNN. *J. Mar. Sci. Eng.* **2020**, *8*, 682. [CrossRef]
9. Oord, A.v.d.; Dieleman, S.; Zen, H.; Simonyan, K.; Vinyals, O.; Graves, A.; Kalchbrenner, N.; Senior, A.; Kavukcuoglu, K. Wavenet: A generative model for raw audio. *arXiv* **2016**, arXiv:1609.03499.
10. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems 30, Proceedings of the Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017*; Curran Associates, Inc.: Red Hook, NY, USA, 2017.
11. Nguyen, D.; Fablet, R. TrAISformer-a generative transformer for ais trajectory prediction. *arXiv* **2021**, arXiv:2109.03958.
12. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv* **2016**, arXiv:1609.02907.
13. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph attention networks. *arXiv* **2017**, arXiv:1710.10903.
14. Gouareb, R.; Can, F.; Ferdowsi, S.; Teodoro, D. Vessel destination prediction using a graph-based machine learning model. In Proceedings of the Network Science: 7th International Winter Conference, NetSci-X 2022, Porto, Portugal, 8–11 February 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 80–93.
15. Mohamed, A.; Qian, K.; Elhoseiny, M.; Claudel, C. Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 14424–14432.
16. Huang, R.; Huang, C.; Liu, Y.; Dai, G.; Kong, W. LSGCN: Long Short-Term Traffic Prediction with Graph Convolutional Networks. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20), Online, 7–15 January 2020; Volume 7, pp. 2355–2361.
17. Eljabu, L.; Etemad, M.; Matwin, S. Anomaly detection in maritime domain based on spatio-temporal analysis of ais data using graph neural networks. In Proceedings of the 2021 5th International Conference on Vision, Image and Signal Processing (ICVISP), Kuala Lumpur, Malaysia, 18–20 December 2021; pp. 142–147.
18. Yu, B.; Yin, H.; Zhu, Z. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In Proceedings of the 27th International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 13–19 July 2018; pp. 3634–3640.
19. Seo, Y.; Defferrard, M.; Vandergheynst, P.; Bresson, X. Structured sequence modeling with graph convolutional recurrent networks. In Proceedings of the International Conference on Neural Information Processing, Siem Reap, Cambodia, 13–16 December 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 362–373.
20. Wu, Z.; Pan, S.; Long, G.; Jiang, J.; Chang, X.; Zhang, C. Connecting the dots: Multivariate time series forecasting with graph neural networks. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual, 6–10 July 2020; pp. 753–763.
21. Li, Y.; Yu, R.; Shahabi, C.; Liu, Y. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv* **2017**, arXiv:1707.01926.
22. Liang, M.; Liu, R.W.; Zhan, Y.; Li, H.; Zhu, F.; Wang, F.Y. Fine-Grained Vessel Traffic Flow Prediction with a Spatio-Temporal Multigraph Convolutional Network. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 23694–23707. [CrossRef]
23. Yan, R.; Wang, S.; Zhen, L.; Laporte, G. Emerging approaches applied to maritime transport research: Past and future. *Commun. Transp. Res.* **2021**, *1*, 100011. [CrossRef]
24. Qiang, M.; Lin, W.; Peng, P.; Peng, Z.; Jinhai, C. Typical Spatial Distribution of Merchant Vessels and Trade Flow in South China Sea. *J. Geo-Inf. Sci.* **2018**, *20*, 8.
25. Ribeiro, C.V.; Paes, A.; de Oliveira, D. AIS-based maritime anomaly traffic detection: A review. *Expert Syst. Appl.* **2023**, *231*, 120561. [CrossRef]
26. Iphar, C.; Ray, C.; Napoli, A. Data integrity assessment for maritime anomaly detection. *Expert Syst. Appl.* **2020**, *147*, 113219. [CrossRef]

27. He, W.; Zhong, C.; Sotelo, M.A.; Chu, X.; Liu, X.; Li, Z. Short-term vessel traffic flow forecasting by using an improved Kalman model. *Clust. Comput.* **2019**, *22*, 7907–7916. [CrossRef]
28. Wang, X.; Li, J.; Zhang, T. A machine-learning model for zonal ship flow prediction using ais data: A case study in the south atlantic states region. *J. Mar. Sci. Eng.* **2019**, *7*, 463. [CrossRef]
29. Xu, X.; Bai, X.; Xiao, Y.; He, J.; Xu, Y.; Ren, H. A Port Ship Flow Prediction Model Based on the Automatic Identification System and Gated Recurrent Units. *J. Mar. Sci. Appl.* **2021**, *20*, 572–580. [CrossRef]
30. Zhaoxuan, L.; Qiang, M.; Yong, L.; Peng, W.; Yang, Y.; Wenlong, H. Prediction and analysis of ship traffic flow based on a space-time graph traffic computing framework. In Proceedings of the 2022 IEEE 20th International Conference on Embedded and Ubiquitous Computing (EUC), Wuhan, China, 9–11 December 2022; pp. 28–35.
31. Dauphin, Y.N.; Fan, A.; Auli, M.; Grangier, D. Language modeling with gated convolutional networks. In Proceedings of the International Conference on Machine Learning, PMLR, Sydney, Australia, 6–11 August 2017; pp. 933–941.
32. Wang, J.; Jiang, J.; Jiang, W.; Li, C.; Zhao, W.X. Libcity: An open library for traffic prediction. In Proceedings of the 29th International Conference on Advances in Geographic Information Systems, New York, NY, USA, 2–5 November 2021; pp. 145–148.
33. Lv, Y.; Duan, Y.; Kang, W.; Li, Z.; Wang, F.Y. Traffic flow prediction with big data: A deep learning approach. *IEEE Trans. Intell. Transp. Syst.* **2014**, *16*, 865–873. [CrossRef]
34. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27, Proceedings of the 28th Annual Conference on Neural Information Processing Systems 2014, Montreal, QC, Canada, 8–13 December 2014*; Curran Associates, Inc.: Red Hook, NY, USA, 2014.
35. Bai, L.; Yao, L.; Li, C.; Wang, X.; Wang, C. Adaptive graph convolutional recurrent network for traffic forecasting. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 17804–17815.
36. Wu, Z.; Pan, S.; Long, G.; Jiang, J.; Zhang, C. Graph WaveNet for Deep Spatial-Temporal Graph Modeling. *arXiv* **2019**, arXiv:1906.00121.
37. Tian, K.; Guo, J.; Ye, K.; Xu, C.Z. St-mgat: Spatial-temporal multi-head graph attention networks for traffic forecasting. In Proceedings of the 2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI), Baltimore, MD, USA, 9–11 November 2020; pp. 714–721.
38. Xu, M.; Dai, W.; Liu, C.; Gao, X.; Lin, W.; Qi, G.J.; Xiong, H. Spatial-Temporal Transformer Networks for Traffic Flow Forecasting. *arXiv* **2020**, arXiv:2001.02908.
39. Guo, S.; Lin, Y.; Feng, N.; Song, C.; Wan, H. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. *Proc. AAAI Conf. Artif. Intell.* **2019**, *33*, 922–929. [CrossRef]

Journal of
Marine Science
and Engineering

Article

Ship Autonomous Berthing Simulation Based on Covariance Matrix Adaptation Evolution Strategy

Guoquan Chen, Jian Yin * and Shenhua Yang

Navigation College, Jimei University, Xiamen 361021, China; cgq0802@foxmail.com (G.C.);
yangshh@163.com (S.Y.)
* Correspondence: yinjian1182@jmu.edu.cn

Abstract: Existing research on auto-berthing of ships has mainly focused on the design and implementation of controllers for automatic berthing. For the real automatic docking processes, not only do external environmental perturbations need to be taken into account but also motion paths, docking strategies and ship mechanical constraints, which are important influential factors to measure autonomous docking methods. Through a literature review of ship path planning and motion control for automatic berthing, it is found that many studies ignore the interference of the actual navigational environment, especially for ships sailing at slow speed when berthing, or do not consider the physical constraints of the steering gear and the main engine. In this paper, we propose a hybrid approach for autonomous berthing control systems based on a Linear Quadratic Regulator (LQR) and Covariance Matrix Adaptation Evolution Strategy (CMA-ES), which systematically addresses the problems involved in the berthing process, such as path planning, optimal control, adaptive berthing strategies, dynamic environmental perturbations and physically enforced structural constraints. The berthing control system based on the LQR and modified LQR-CMA-ES have been validated by simulation work. The simulation results show that the proposed method is able to achieve the automatic docking of the ship well and the system is robust and well adapted to environmental disturbances at slow speed when docking.

Keywords: autonomous berthing; CMA-ES; LQR; berthing strategy

Citation: Chen, G.; Yin, J.; Yang, S. Ship Autonomous Berthing Simulation Based on Covariance Matrix Adaptation Evolution Strategy. *J. Mar. Sci. Eng.* **2023**, *11*, 1400. https://doi.org/10.3390/jmse11071400

Academic Editor: Fausto Pedro García Márquez

Received: 15 June 2023
Revised: 8 July 2023
Accepted: 9 July 2023
Published: 11 July 2023

1. Introduction

Accelerating the construction of intelligent, efficient and green marine transportation vessels is one of the methods for developing the marine economy. Unmanned Surface Vehicle (USV) refers to an intelligent ship that can sail autonomously on the water without the participation of crew. Elevated velocity, diminutive dimensions, reduced cost and the ability for autonomous navigation constitute the salient attributes possessed by this kind of watercraft, rendering USVs preferred in manifold scenarios such as open sea search and rescue and military maneuvers. In virtue of the intrinsic properties surrounding the hull, watercraft of such small size are readily maneuvered yet easily affected by disturbances from ambient conditions, thereby imposing relatively strict requirements on the robustness of the kinetic control system.

In 1987, Kose et al. [1] first realized the automatic berthing and unberthing of ships by means of computer-aided control and tug assistance. In 2015, Mizuno et al. [2] proposed a quasi-real-time automatic berthing control method based on a multiple shooting method, which can give a solution in a short time. In 2017, Zhang et al. [3] promulgated a nonlinear feedback algorithm predicated upon a bipolar S-PID, based on the MMG vessel model, which suppressed perturbations and extraneous disturbances during the docking process to maintain the ideal heading for the watercraft and the safety of the berthing procedure. In 2020, Xu et al. [4] proposed a robust adaptive control algorithm to address the perturbations engendered by the bank effect on the berthing control of a vessel and eliminate the oscillatory manifestations in the control outputs. Han [5] promulgated a disturbance rejection control algorithm based on neural networks to accomplish the heading control of vessels with minimal overshoot and effectuate berthing control maneuvers. In 2021, Jia et al. [6] retrained the controller by using an ANN, which improved the accuracy of the output and solved the problem of yaw deviation during berthing. Wang et al. [7] designed the Linear Quadratic Regulator (LQR) controller combined with a GA, which reduced the additional resistance of the system caused by pitch and heave motion.

In 2019, Yang [8] proposed an accelerated evolutionary algorithm based on the Covariance Matrix Adaptation Evolution Strategy (CMA-ES), which reduced the time complexity of the algorithm and solved the issue of premature convergence of multi-objective optimization. In 2020, Maki et al. [9] applied CMA-ES to the optimization of control parameters of an automatic berthing system, and the deviations between the actual trajectory and the predetermined reference path during the berthing maneuver were minimized. In 2021, Maniyappan et al. [10] used the CMA-ES to carry out PD optimal control on the time series of rudder angle change, and the experiments showed that the optimized controller improved the yaw check effectively. In 2022, Miyauchi et al. [11] used the CMA-ES algorithm to optimize the parameters of the Nomoto model which described the ship's kinematic behavior in the process of berthing and obtained a better control effect by enhanced accuracy.

In Table 1, a synopsis of the research on kinetic control mechanisms and optimization thereof for watercraft, undertaken by a selection of scholars, has been compiled, to serve as the theoretical foundation for subsequent study in this paper.

Automated berthing of vessels has always been an important area in the research of ship navigation automation. Since the automatic berthing of a ship is low-speed maneuvering motion in restricted waters, the interference from the external environment on the maneuver control system cannot be ignored. Although a series of automated measures have been basically realized, including autonomous collision avoidance and route planning, autonomous ship berthing remains a major challenge in the development of intelligent ships at present. The "final one mile" problem of autonomous ships has not yet been well solved, and much research work remains to be carried out.

According to the summation shown in Table 1, LQR demonstrates propitious compatibility which has great facilities in the MIMO control domain, providing a new way to control the complicated docking process of ships. Moreover, the CMA-ES has exhibited efficacious effects on model parameter and control law optimization. The essence of the research is combining a controller with effective compatibility and a robustly effectual optimizer to accomplish autonomous berthing of ships.

The autonomous berthing system architecture proposed in this paper based on the current status of research on automatic berthing of ships is depicted in Figure 1. It encompasses the path planning, optimal control, power distribution and other indicators required for ship intellectualization and can achieve the "final one mile" motion control objective of automatic ship berthing. In Table 2, the problems that have been disregarded or not thoroughly explored in the studies above are summarized, and the corresponding solutions are provided in this paper.

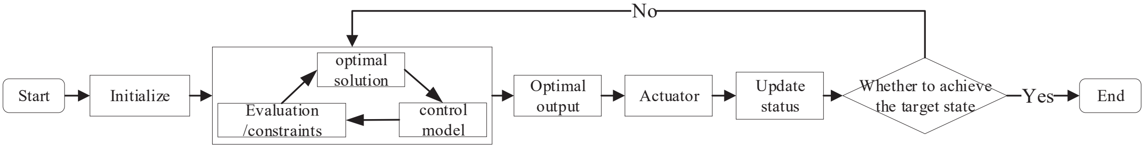


Figure 1. Architecture of ship autonomous berthing system.

Table 1. Comparison of research on berthing-related papers in recent years.

Core Method	Researchers	Outcome	Experiment
LQR control	Yazdanpanah et al. [12]	The feedback gain of the system was calibrated through the combination of LQR and fuzzy logical control.	Simulation
	Brasel [13]	The heading angle and velocity of the vessel were modulated under disparate operating conditions via LQR.	Simulation
	Shao [14]	The kinetics of an oil tanker was governed within the limited boundary of the channel through an LQR controller.	Simulation
	Farzanegan et al. [15]	The heaving motions of a ship were damped using an LQR regulator.	Simulation
	Tian [16]	The yaw-checking maneuver was optimized under the straight sailing condition by an LQR controller.	Simulation
	Zhao et al. [17]	Based on the LQR, a controller without static error was utilized to achieve heading stabilization.	Simulation
CMA-ES optimization	Chen [18]	The ambulatory kinetics of the robot were optimized.	Simulation
	Maki et al. [9,19,20]	The frequent rudder, propeller operation and control objective function during the berthing process were optimized.	Simulation
	Maniyappan et al. [10]	Global optimization of the rudder control for the yaw checking.	Simulation
	Liu et al. [21]	A synopsis of the technology for automated docking and undocking at the current stage was given.	Simulation
	Akimoto et al. [22]	Antagonistic CMA-ES was used for automatic berthing control with uncertainty.	Simulation
	Miyauchi et al. [23]	A systematic berthing model was constructed and the berthing trajectory was optimized.	Simulation
Other	Jia et al. [6]	The issue of excessive heading deviation was addressed through neural networks by berthing training data.	Simulation
	Homburger et al. [24]	MPPI methodology was utilized to implement berthing control and achieve optimal control in a nonlinear system.	Simulation
	Sawada et al. [25]	An automated berthing control system was propounded and trajectory tracking was performed through numerical simulation.	Simulation
	Zhang et al. [26]	Reinforcement learning based on demonstration data was used for auto-berthing control.	Simulation
	Kamil et al. [27]	An ANN-based controller was proposed to simulate a human brain's activity during the execution phase.	Simulation
	Xu et al. [28]	A three-phase guidance algorithm was devised to ensure stable operation of the vessel when transient and during berthing.	Simulation

Table 2. Inadequacies in current research and corresponding solutions.

Problems in Current Research	Solutions Given in This Paper
The disturbances of wind and currents are not considered.	The interference with the berthing process is considered.
The mechanical constraints of ship motion are not considered.	The mechanical constraint of rudder and propeller is added to the whole process of ship motion.
PID, robust and fuzzy control are inconspicuous in the MIMO field.	The LQR control method with good performance in MIMO is selected.
The difference in ship mathematical motion models (large drift angle) in the low-speed domain is not considered.	The low-speed large drift angle motion correction model is introduced in the berthing stage.
The overall optimization of the berthing process is not considered.	An architecture of the berthing control system is proposed.

In order to solve the problems existing in the current research, this paper uses the Linear Quadratic Regulator (LQR) as the fundamental control model for the controller according to the requirements of the ship's auto-berthing in the "***Guide to Autonomous Cargo Ships***" [29] issued by the China Classification Society in 2018. The output of an LQR controller and the path of ship berthing are optimized by using the Covariance

Adaptive Adjustment Evolutionary Strategy (CMA-ES) algorithm. The kinematic modeling of the vessel included the low-speed and high-drift angle ship maneuvering emendation model proposed by Yoshimura Yasuo [30]. To fulfill the real situation during the berthing process, disturbances caused by the wind and current were introduced in the experiments in this paper and a simple navigational chart with isometric scaling was introduced as the test environment.

The organizational structure of this paper is as follows: The first section delineates the relevant research on autonomous ship berthing technology. In the second part, the fundamental ship handling maneuvering model used in this paper is characterized. The third section articulates the LQR controller and CMA-ES optimization strategy, introducing the berthing approach based on the CMA-ES and LQR algorithms. This section also introduces the modified model for low-speed and large drift angle motility to suit authentic simulation experiments on a vessel. In the fourth section, several simulation experiments are described, implementing and validating the efficiency of the proposed methodology and analyzing control effects under different simulation scenarios, while two berthing strategies for different berthing conditions are proposed. Finally, in the fifth part, an encapsulation of the research expounded is summarized.

2. Ship Kinematics Model

According to the ship state space model proposed by Fossen [31], the dynamic model of a USV is described in Equation (1), where $\boldsymbol{\nu}$ and η are defined as state vectors describing the linear velocity (or angular velocity) and position (or Euler angle) of the ship in three degrees of freedom, $\tau = [X, Y, N]^T$ is the force or moment acting on the ship, ω is the force or moment acting on the ship by wind and current induced by the environment and $J_\Theta(\eta)$ is the Euler angle rotation matrix. The parameters relevant to this model have been explicated in detail in [31].

$$\begin{aligned} \boldsymbol{M}\dot{\boldsymbol{\nu}} + \boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{G}(\eta) = \tau + \omega(t) \\ \dot{\eta} = J_\Theta(\eta)\boldsymbol{\nu} \end{aligned} \tag{1}$$

For the facilitation of the research endeavors herein, the conventional three degrees of freedom (surge, sway and yaw) ship handling maneuvering model was adopted. Figure 2 delineates the schematic representation of the three degrees of freedom model of the ship utilized. The simplified three degrees of freedom do not consider the restoring force, thus,$G(\eta) = 0$. At the same time, the influence of wind and current is added to the model, and the three degrees of freedom ship model is finally expressed as Equation (2), where $\boldsymbol{\tau}_{\text{wind}}$ and $\boldsymbol{\tau}_{\text{current}}$ correspond to the force or moment of wind and current acting on the ship, respectively, $\boldsymbol{V}_c = J_\Theta(\eta)\boldsymbol{\nu}_c$ is the current speed in the navigation coordinate system (NED coordinate system), $\boldsymbol{\nu}_r$ is the ship's Speed Through Water (STW) and $\boldsymbol{\nu} = \boldsymbol{\nu}_r + \boldsymbol{\nu}_c$ is the Speed Over Ground (SOG).

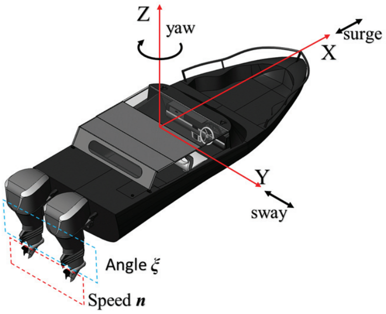


Figure 2. A schematic of USV's three degrees of freedom.

$$\begin{array}{r} \boldsymbol{M}\dot{\boldsymbol{\nu}}_r + \boldsymbol{C}(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r + \boldsymbol{D}(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r = \boldsymbol{\tau} + \boldsymbol{\tau}_{\text{wind}} + \boldsymbol{\tau}_{\text{current}} \\ \dot{\eta} = J_{\Theta}(\eta)\boldsymbol{\nu}_r + \boldsymbol{V}_c \end{array} \tag{2}$$

$$v = [u, \nu, r]^T, \eta = [x, y, \varphi]^T \tag{3}$$

The propellers p1 and p2 constitute the twin actuators upon the vessel utilized in this work, which can be controlled by revolution n and angle ξ. The thrust generated by the propeller for the USV is expressed as Equation (4) and, for simplicity, the heaving motion of the ship is not considered in this paper, thus $F_\omega = 0$.

Equation (5) is obtained by decomposing the thrust generated by propeller P in the normal and tangent direction separately, where t_p is the thrust generated by each propeller, F_p is the resultant force and ξ_p represents the instant angle of each propeller.

$$F = [F_u, F_\nu, F_\omega]^T \tag{4}$$

$$F_p = \begin{bmatrix} \cos(\xi_p) \\ \sin(\xi_p) \\ 0 \end{bmatrix} \cdot t_p = \begin{bmatrix} \cos(atan2(F_\nu, F_u)) \\ \sin(atan2(F_\nu, F_u)) \\ 0 \end{bmatrix} \cdot \parallel F_p \parallel \tag{5}$$

The force τ acting on the ship is decomposed into linear force and torque, where τ_{linear} is the total translational force generated by each propeller, τ_{torque} is the cross product sum of the translational force and the corresponding torque of each propeller, where r_p represents the position vector of the propeller and $S(r_p)$ is the skew symmetric matrix of the vector.

$$\tau = \begin{bmatrix} \tau_{\text{linear}} \\ \tau_{\text{torque}} \end{bmatrix} = \left[\sum^{P} F_p, \sum^{P} S(r_p) F_p \right]^T = \begin{bmatrix} F_1 + F_2 \\ S(r_1)F_1 + S(r_2)F_2 \end{bmatrix} \tag{6}$$

According to Fossen [31], Formula (7), of the thrust generated by the propeller and its rotational speed, is derived, where the propeller has the physical parameters T_{nn} and $T_{n\nu}$, the corresponding rotational speed is defined as n_p and the localized velocity to the near propulsive is defined as V_A. Combined with Formula (5), the control outputs of the two propellers are calculated and expressed as Equation (8), where $\boldsymbol{\xi}$ is the angle of the propeller and the rotational speed of the propeller is defined as $\boldsymbol{n}$.

$$\begin{array}{c} t_p = T_{nn} n_p^2 + T_{n\nu} V_A n_p \ , \ T_{nn} > 0 > T_{n\nu} \\ n_p = \operatorname{sgn}(t_p) \frac{\sqrt{Tn\nu^2 V_A^2 + 4T_{nn}|t_p|} - T_{n\nu} V_A}{2T_{nn}} \end{array} \tag{7}$$

$$\boldsymbol{\xi} = [\xi_1, \xi_2]^T, \boldsymbol{n} = [n_1, n_2]^T \tag{8}$$

Ship motion is affected by wind and current when sailing. According to "***the Manual of Ship Fluid Dynamics and Motion Control***" [31], the superimposed force τ_{wind} of wind is expressed as Equation (9) and the superimposed force τ_{current} of current is described in Equation (10).

$$\tau_{\text{wind}} = \frac{1}{2}\rho_a V_a^2 \begin{bmatrix} C_X(\mu_{wd})A_T \\ C_Y(\mu_{wd})A_L \\ LC_N(\mu_{wd})A_L \end{bmatrix} \tag{9}$$

$$\tau_{\text{current}} = \frac{1}{2}\rho V_{cd}^2 \begin{bmatrix} C_{XC}(\mu_{cd})A_{TC} \\ C_{YC}(\mu_{cd})A_{LC} \\ C_{NC}(\mu_{cd})A_{LC} \end{bmatrix} \tag{10}$$

3. Mathematical Modeling

3.1. LQR Control Model

Some researchers [14] implemented ship motion control of a large cruise ship based on the LQR, and the simulation experiments show that the ship handling experimental effect is good under certain wind conditions. The advantages of the LQR in ship motion control is confirmed by simulation experiments in some studies [13,15–17]. Liu et al. summarized the main automatic berthing control algorithms at the current stage and showed the inadequacies of various algorithms in actual control. For instance, algorithms such as neural networks and fuzzy logic for the precise control of berthing often do not work well anymore. Meanwhile, rather high requirements are placed on both the computational real-time capability and model accuracy by the Model Predictive Control (MPC) controller [21].

However, these disadvantages mentioned above can be overcome by the LQR which in fact is the core of MPC. Otherwise, excellent generalization capability over disparate models is possessed by the LQR controller, without particular utility functions needing to be specified for different scenario requirements. In this paper, a Linear Quadratic Regulator (LQR) controller is chosen for the motion control of a USV.

In order to realize the ship LQR control, firstly, it is necessary to determine the state space model of the ship and, secondly, design the control gain of the controller according to the corresponding performance index. According to the control model of Micha Brasel [13], the state space equation of the system is expressed as Equation (11), and the performance index J is defined as Equation (12).

The system state and control vector, respectively, are defined as x and u, $\boldsymbol{A}$ is the state matrix of the controlled object, $\boldsymbol{B}$ is the linear control matrix representing u to x, $\boldsymbol{H}$ is the interference matrix of the system, $\boldsymbol{C}$ is the system output matrix, $\boldsymbol{Q}$ is a semi-positive definite symmetric weighted matrix of x and $\boldsymbol{R}$ is a weighted positive definite symmetric matrix of u.

$$\begin{aligned} \dot{x} &= \boldsymbol{A}x + \boldsymbol{B}u + \boldsymbol{H} \\ y &= \boldsymbol{C}x \end{aligned} \tag{11}$$

$$J = \frac{1}{2}\int_0^T \left(x^{\mathrm{T}}\boldsymbol{Q}x + u^{\mathrm{T}}\boldsymbol{R}u\right)dt \tag{12}$$

Equation (12) shows that when the performance index J reaches the minimum value, the corresponding $u(t)$ is the optimal control $u^*(t)$ of the controller output. According to the "***the optimal control theory***" [32], it is proved that $u^*(t)$, that enables J to reach the minimum value, can be obtained by Equation (13), where $\boldsymbol{K}$ represents the optimal feedback coefficient matrix and $\boldsymbol{P}$ is the solution to the *Riccati* equation.

$$\begin{aligned} u^* &= -\boldsymbol{R}^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P}x = -\boldsymbol{K}x \\ 0 &= \boldsymbol{P}\boldsymbol{A} + \boldsymbol{A}^{\mathrm{T}}\boldsymbol{P} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P} + \boldsymbol{Q} \end{aligned} \tag{13}$$

The optimal solution of the LQR has a standard analytical form, whereby optimal feedback control can be conveniently obtained. The LQR controller possesses excellent stability performance when the control system is able to maximize both system robustness and responsiveness to the greatest extent possible, subject to various disturbances.

3.2. CMA-ES Algorithm

The CMA-ES, an evolutionary computation method for global optimization, was given by Hansen et al. [33] from evolutionary strategy algorithms, which can find the global optimal solution for both nonlinear and nonconvex problems. CMA-ES algorithm has good search capability. Compared with the traditional linear programming method, no gradient calculation is needed by this method, thereby computational cost is reduced.

Some studies have achieved good experimental results in the optimization of ship lateral motion control by using CMA-ES [9,10,19,20]. The CMA-ES optimization method is adopted for berthing control under model uncertainty [22]. This method has also been used to optimize the berthing trajectory and improve the robustness and flexibility of navigation control [23]. The CMA-ES method can automatically optimize the search space by adaptively adjusting the covariance matrix, thus achieving an efficient global search, which can play a good role in path planning and motion control.

The computational process of the CMA-ES algorithm for auto-berthing is as the following:

Step 1. Parameter initialization. According to the search space, the evolutionary path is set, and the initial algebra is defined as g = 0.

Step 2. The population mutation is controlled by using the mean value m^g, the step size σ^g and the covariance matrix C^g in Equation (14), where g represents the population algebra.

$$x_k^{g+1} = m^g + \sigma^g N(0, C^g), k = 1, \ldots, \lambda \tag{14}$$

Step 3. According to the fitness function, the offspring are selected, and the first μ individuals with the smallest fitness value are regarded as the new generation population. Equation (15) realizes truncated selection by $\mu < \lambda$ and takes different weights as the selection mechanism.

$$\begin{gathered} m^{g+1} = m^g + \sum_{i=1}^{\mu} \omega_i (x_{i:\lambda}^{g+1} - m^g) \\ \sum_{i=1}^{\mu} \omega_i = 1, \omega_1 \geq \omega_2 \geq \cdots \geq \omega_\mu \geq 0 \end{gathered} \tag{15}$$

Step 4. The parameter update method is described as Equation (15), while the definition of the covariance matrix, the explanation of the evolution path and the description of the control step are given in Equation (16) to Equation (18), respectively. The proportion coefficient of new population individuals is represented by ω_i. The learning rate and effective selection quality of covariance matrix C are represented as c_μ and μ_{eff}. The learning rate of the step length is represented by c_σ. p_c and p_σ are conjugated and the damping factor is represented by d_σ.

$$\begin{gathered} C^{g+1} = (1 - c_\mu) C^g + c_\mu \sum_{i=1}^{\mu} \omega_i y_{i:\lambda}^{g+1} \left(y_{i:\lambda}^{g+1}\right)^T \\ y_{i:\lambda}^{g+1} = \frac{\left(x_{i:\lambda}^{g+1} - m^g\right)}{\sigma^g}, c_\mu = \min(1, \frac{\mu_{eff}}{n^2}) \end{gathered} \tag{16}$$

$$\begin{gathered} p_c^{g+1} = (1 - c_c) p_c^g + \sqrt{c_c (2 - c_c) \mu_{eff}} \frac{m^{g+1} - m^g}{\sigma^g} \\ p_\sigma^{g+1} = (1 - c_\sigma) p_\sigma^g + \sqrt{c_\sigma (2 - c_\sigma) \mu_{eff}} C^{-\frac{1}{2}} \frac{m^{g+1} - m^g}{\sigma^g} \end{gathered} \tag{17}$$

$$\sigma^{g+1} = \sigma^g \exp\left(\frac{c_\sigma}{d_\sigma} \left(\frac{\parallel p_\sigma^{g+1} \parallel}{E(\parallel N(0, I) \parallel)} - 1 \right) \right) \tag{18}$$

Step 5. Conditional judgment. When the set threshold condition is satisfied, the iterative output result is obtained.

The CMA-ES algorithm is suitable for auto-berthing control for a USV owing to its inherent advantage. The basic architecture of a ship's autonomous berthing system including path planning and the LQR combined with the CMA-ES is shown in Figure 1.

3.3. Autonomous Berthing Model of USV Based on CMA-ES

For ship motion control, in fact, it is difficult to accurately establish the ship mathematical model as it is a strong coupling nonlinear system. Fossen [34], Zhang [35] and other scholars have made some achievements in large ship motion control. As the ship usually navigates at a slow speed with a large drift angle when moving alongside the berth, there is a great difference from a ship sailing at normal speed. The current research concentrates primarily upon the minimization of path length and time expenditure. External environmental perturbations, physical constraints and restrictions to ship maneuvering are neglected although such factors would also exert great influence upon the ship handling characteristics of a vessel.

Aiming at the inadequacies of the algorithms employed in the current phase of ship control, the optimization of the berthing control system in this paper can primarily encompass two aspects: on the one hand, the constrained optimization of the ship's navigation path is undertaken. According to the relevant navigation rules and constraints, a search and evaluation of all viable paths are performed and the optimal initialization path is selected, as delineated in Figure 3a. On the other hand, the optimization of the performance indices of the LQR controller is carried out, such as the optimization of the motion or amplitude stability indices.

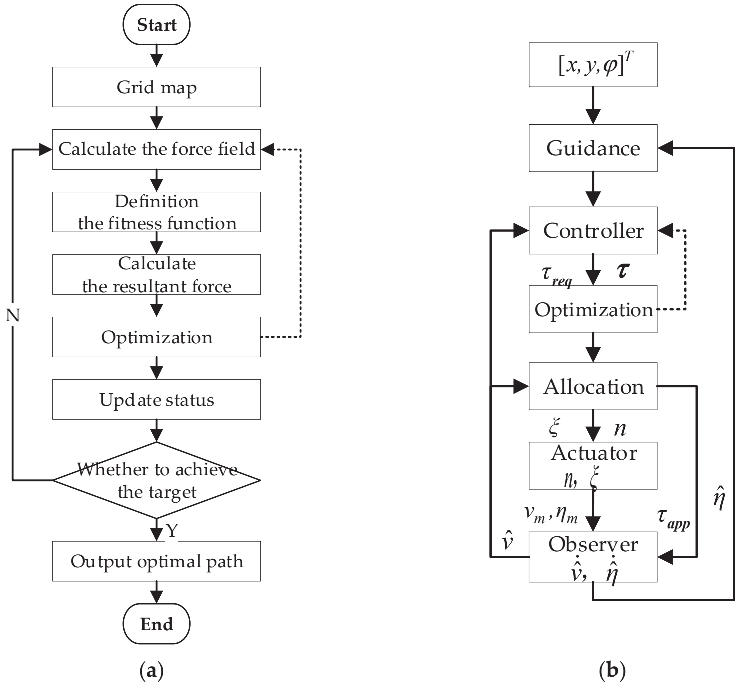


Figure 3. System control optimization strategy. (**a**) Path planning; (**b**) LQR controller.

As shown in Figure 3b, based on the CMA-ES algorithm, the objective function and constraints considering ship actuators like engine and steering gear for the optimizer are constructed and the improved LQR controller is given for the auto-berthing control problem.

The framework of the path planning algorithm selected for this work, the Artificial Potential Field (APF) method, is depicted in Figure 3a. The crux of the APF path planning method constitutes the construction of potential fields by computing the gravitation of the destination and the repulsion of obstacles, with the potential field gradient descent direction harnessed to instigate updates in movement. The process of CMA-ES optimization is to calculate whether the fitness function belongs to the threshold interval, otherwise, the weight is changed again to calculate the gravitation and repulsion. In the process of path planning using the APF algorithm [36], the resultant field is represented by $U(q)$, $U_{att}(q)$ is the gravitational field and $U_{rep}(q)$ is the repulsive field.

$$U(q) = U_{att}(q) + U_{rep}(q) \tag{19}$$

The framework schematic of the LQR control algorithm employed within this paper is delineated in Figure 3b. The optimizer CMA-ES calculates and finds the optimal value that meets the constraint conditions by adjusting the weight matrix parameters of state variables and control variables.

According to Formulas (2) and (11), Formula (20) is obtained, where the state matrix is $\boldsymbol{A}(\nu_r) = -\boldsymbol{M}^{-1}(\boldsymbol{C}(\nu_r) + \boldsymbol{D}(\nu_r))$ and the control matrix is $\boldsymbol{B} = \boldsymbol{M}^{-1}$.

$$\begin{aligned} \dot{\nu}_r &= \boldsymbol{A}(\nu_r)\nu_r + \boldsymbol{B}(\tau + \tau_{wind} + \tau_{current}) \\ \dot{\eta} &= J_{\Theta}(\eta)\nu_r + V_c \end{aligned} \tag{20}$$

According to the hydrodynamic model with large drift angle proposed by Yoshimura Yasuo, the ship model navigating at slow speed, especially when moving alongside the berth, is revised as (21), and the detailed derivation process and parameter explanation are shown in references [30,31]. The modified model is decomposed into (22) after being brought into the damping matrix $\boldsymbol{D}$, where d is the original static constant, $\Phi(\nu_r)$ is a regression function and ϑ is an error parameter.

$$\begin{cases} X_{\mathrm{H}} = X_{\mathrm{H}}(r=0) + X_{\nu r}\nu r + X_{rr}r^2 \\ Y_{\mathrm{H}} = Y_{\mathrm{H}}(r=0) + Y_r|u|r + \frac{1}{2}\rho d C_d\left(L\nu|\nu| - \int_{-L/2}^{L/2}(\nu + C_{ry}xr)|\nu + C_{ry}xr|\mathrm{d}x\right) \\ N_{\mathrm{H}} = N_{\mathrm{H}}(r=0) + N_r|u|r + \frac{1}{2}\rho d C_d\int_{-L/2}^{L/2}(\nu + C_{rr}xr)|\nu + C_{rn}xr|x\mathrm{d}x \end{cases} \tag{21}$$

$$\begin{aligned} \boldsymbol{D}(\nu_r)\nu_r &= -d\nu_r - \Phi(\nu_r)\vartheta : \\ d &= -diag([X_u, Y_\nu, N_r]) \\ \vartheta &= [X_{|u|u}, X_{uuu}, Y_\nu, Y_r, Y_{|\nu|\nu}, Y_{|r|\nu}, Y_{|\nu|r}, Y_{|r|r}, N_\nu, N_{|\nu|\nu}, N_{|r|\nu}, N_{|\nu|r}, N_{|r|r}]^T \end{aligned} \tag{22}$$

Combining Equations (12) and (22), the control law, Equation (23), is obtained, where $R(\psi)$ is the rotation matrix about the Z axis.

$$\begin{aligned} \dot{\nu}_r &= \boldsymbol{M}^{-1}(\tau + \tau_{wind} + \tau_{current} - \boldsymbol{C}(\nu_r)\nu_r + d\nu_r + \Phi(\nu_r)\vartheta) \\ \dot{\eta} &= \boldsymbol{R}(\psi)\nu_r + V_c \end{aligned} \tag{23}$$

$$\boldsymbol{R}(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{24}$$

3.3.1. Constraint Condition

Basic state constraints (25) are to be satisfied when the ship berths, where $P_1 = (x_1, y_1, \psi_1)^T$ denotes the prescribed desired berthing state, $P1 = (x1, y1, \psi1)^T$ is the motion state of the ship in actual motion control, d_ref and ψ_ref are the maximum allowable errors.

$$P1 - P_1 \le [d\_ref, d\_ref, \psi\_ref]^T \tag{25}$$

Position constraint is to reduce the optimized search space and stabilize the ship's docking position more efficiently. The set berth area is II_C when the ship is within II_C, and it is considered that the ship has entered the berth area. p_i is the four points of the ship's safety rectangle, and C_{berth} represents the berth area.

As shown in Figure 4, The waters proximate to the berth are demarcated into three zones: Zone I (the outer stabilizing region beyond the berth), Zone II (the berth per se) and Zone III (the quayside wharf). The berthing process is likewise demarcated into three stages: Stage 1 (the approaching phase, arriving at the outer stabilizing region far from the berth), Stage 2 (outer stabilizing, adjusting the orientation of the ship's head) and Stage 3 (berthing).

$$II_C = \sum_{i=1}^{4}\int_{p_i \in C_{\mathrm{berth}}} |Y_i - Y_{\mathrm{berth}}|\mathrm{d}t \tag{26}$$

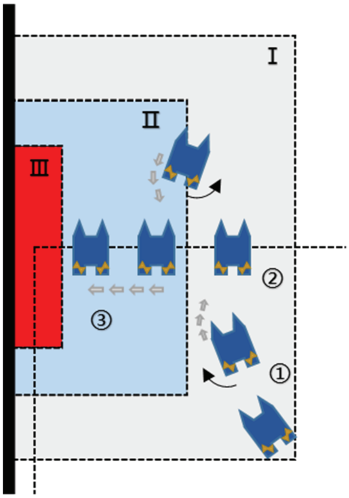


Figure 4. Berthing area.

The control mechanism of a USV is composed of propellers and steering gear system and is limited by its own physical characteristics. This paper mainly considers the mechanical constraint $[C_{min}, C_{max}]$ composed of rudder angle δ and propeller speed n of the USV. The constraint $[C'_{min}, C'_{max}]$ of rudder angle change rate δ' and rotational speed change rate n' are also considered.

$$\begin{array}{c} C_{min} = [\delta_{\min}, n_{\min}]^T, C'_{min} = [\delta'_{\min}, n'_{\min}]^T \\ C_{max} = [\delta_{\max}, n_{\max}]^T, C'_{max} = [\delta'_{\max}, n'_{\max}]^T \end{array} \tag{27}$$

The ship berthing speed is restricted as per Equation (28), and u_t and v_t represent the longitudinal and lateral velocity of the ship at time t

$$|u_t| \le u_{\max}, |v_t| \le v_{\max} \tag{28}$$

The constraint on the ship's course is described in Equation (29), ψ_t is the ship's heading at time t and *rot* is the ship's turn rate, that is, the course change rate.

$$|rot| = \left| \frac{\Delta\psi}{\Delta t} * \frac{180}{pi} \right| \le ROT_{\max} \tag{29}$$

3.3.2. Objective Function

The initial state of the system is defined as $x(0)$. The optimal control vector u^* can be obtained by the LQR, and the state vector x at the next moment can be obtained by motion control of the ship. As the initial sample of the CMA-ES, the optimal state x_i at the next moment can be obtained.

Suppose that the system state vector $x = (x, y, \psi)^T$ and the control vector $\dot{u} = (n, \delta)^T$, then Formula (23) is simplified to obtain the system state described in Equation (30), where the interference matrix $\boldsymbol{H} = \boldsymbol{M}^{-1}(\tau_{\text{wind}} + \tau_{\text{current}})$, $\boldsymbol{f}(\cdot)$ is the system state matrix.

$$\begin{aligned} \dot{x} &= \boldsymbol{M}^{-1}(\boldsymbol{d}x - \boldsymbol{C}x) + \boldsymbol{M}^{-1}(u + \Phi(x)\vartheta) + \boldsymbol{M}^{-1}(\tau_{\text{wind}} + \tau_{\text{current}}) \\ &= \boldsymbol{f}(x) + \boldsymbol{M}^{-1}(u + \Phi(x)\vartheta) + \boldsymbol{H} \\ y &= \boldsymbol{R}(\psi)x + V_c \end{aligned} \tag{30}$$

The desired ideal output is set to $\eta^*(t)$ and $y(t)$ is the actual control output. $e(t) = y(t) - \eta^*(t)$ is the deviation, and the index functional of the control system is expressed as J.

$$J = \frac{1}{2}e^T(T)\mathbf{S}e(T) + \frac{1}{2}\int_0^T \left(e^T(t)\mathbf{Q}(t)e(t) + u^{\mathrm{T}}(t)\mathbf{R}(t)u(t)\right)dt \tag{31}$$

In order to simplify the study of energy consumption of the ship power system, within this work, the control variables for the rudder and propellers during the ship berthing process are adopted as the energy consumption indices for the system. n_t and δ_t are the sampling values at the corresponding time, λ_1 and λ_2 are the weight coefficients of the corresponding control variables. According to Formula (32), $F(n, \delta)$ [37] is calculated as the energy consumption index during the ship's navigation.

$$F(n, \delta) = \sum_{t=0}^{T-1}(\lambda_1 n_t^2 + \lambda_2 \delta_t^2) \tag{32}$$

The control objective function of the system can be written as Equation (33), where K_J and K_F are the corresponding weight parameters, the value is determined according to the actual situation and the value is [0.75, 0.25] in the experiment. Combined with Formula (19), the objective function of path planning is expressed as Equation (34), H_U and H_F are the corresponding weight functions and the values in the experiment are [0.25, 0.75].

$$K_J * J + K_F * F \tag{33}$$

$$H_U * U + H_F * F \tag{34}$$

In summary, the optimization objective function and constraint conditions of the berthing system can be described as Formulas (35) and (36).

$$\min\begin{cases} H_U * U + H_F * F \\ K_J * J(P, C) + K_F * F(n, \delta) \end{cases} \tag{35}$$

$$s.t.\begin{cases} P1 - P_1 \leq [d\_ref, d\_ref, \psi\_ref]^T \\ |u_t| \leq u_{\max} \\ |\nu_t| \leq \nu_{\max} \\ |rot| \leq ROT_{\max} \\ C_{\min} \leq C \leq C_{\max} \\ C'_{\min} \leq C' \leq C'_{\max} \end{cases} \tag{36}$$

Upon the actual berthing process, external environmental disturbance and the intrinsic physical maneuverability properties of the hull exert considerable influence on the behaviors of the vessel. In accordance with the actual hydrodynamic environments and maneuvering constraints inherent to the ship itself, corresponding constraints have been incorporated into the overall control process. An optimization strategy predicated upon the CMA-ES effects adaptive optimization of the safety and smoothness of the ship's berthing trajectory as well as the outputs of the controller.

4. Simulation Analysis

The ship adopted in this experiment is a fully actuated ship, and the hydrodynamic coefficient of the model refers to the Otter ship model given by Fossen [31]. In this paper, two groups of simulation experiments are set up: group 1 implements automatic berthing of the vessel in external environments with no consideration of wind and current through employment of the LQR. The second group is the automatic berthing simulation experiment with consideration of wind and current environments by using the LQR and the integrated control system proposed in this paper.

No dynamic obstacles are set in the two sets of experiments in this paper. The configuration of the simulation environment of the experiment is shown in Table 2, the parameters of the simulated ship model used in the experiment are shown in Table 3 and the values of the relevant parameters of the experiment are shown in Tables 4 and 5.

Table 3. Experimental environment.

Parameters	Configuration
System environment	Win11 Intel (R) Core (TM) i5-9500 CPU @3.00GHz 16G RAM

Table 4. USV experimental ship model parameters.

Parameters	Value
Length (m)	2.00
Width B (m)	1.08
Draft (m)	0.1951
Mass m (kg)	55
Rotation radius (m)	0.432
Square coefficient	0.4
Rudder angle δ range (°)	$[-35, 35]$
Propeller speed n range (r/s)	$[-50, 50]$

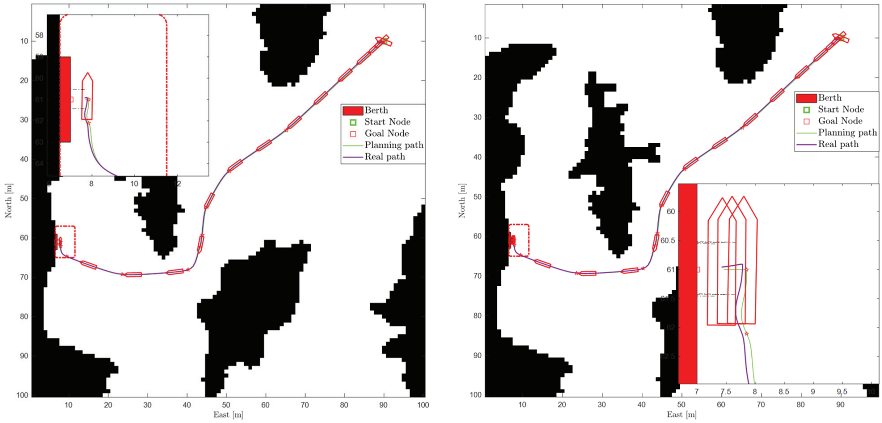
Table 5. USV experimental simulation parameters.

Parameters	Value
Simulation map	100 m × 100 m
Initial point state	[90, 10, 120°]
Berthing point status	[7, 61, 0°]
$[u_{approah}, u_{dock}]$	[3 m/s, 0.075 m/s]
$[d_{ref}, \psi_{ref}]$	[0.01 m, 3°]
$[u_{\max}, v_{\max}]$	[0.075 m/s, 0.075 m/s]
$ROT_{\max}$	$2.5\ ^\circ/\mathrm{s}^2$
$[K_J, K_F, H_U, H_F]$	[0.75, 0.25, 0.25, 0.75]

4.1. Automatic Berthing of USV Based on LQR

In circumstances where the waters proximate to the berth do not show disturbances from a northerly or easterly wind or current, a static calm water condition is assumed, $\boldsymbol{\tau}_{\text{wind}} = [0, 0]$, $\boldsymbol{\tau}_{\text{current}} = [0, 0]$ (the unit is m/s). Granted the port berth articulates no particular stipulations regarding ship berthing, the testing shows that the vessel can be capable of directly affecting berthing. Through experimental simulation, the berthing process is delineated in Figure 5.

For the secondary berthing phase depicted in Figure 5b, with the ship stabilized outside of the berth, the lateral berthing maneuver is executed based on distance detection data from shipborne sensors indicating proximity to the berth terminal. In this process, the ship's posture must adjust flexibly according to the optimal control outputs generated by the LQR controller to facilitate lateral berthing of the ship and, ultimately, automatic berthing is completed.



(**a**) Approaching stage

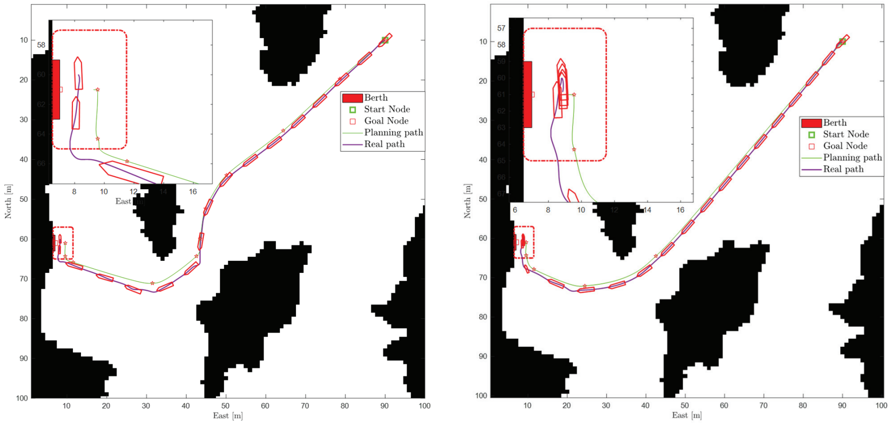
(**b**) Berthing stage

Figure 5. USV auto-berthing process based on LQR without wind or current. (Black indicates the non-navigable area of the map, and the red dashed line indicates the berth area I mentioned in Figure 4, the same as below).

4.2. Automatic Berthing of USV Based on CMA-ES Optimization

The USV approaches the berth with perturbations from northerly and easterly winds and currents, $\boldsymbol{\tau}_{\text{wind}} = [1.5, -2.5]$, $\boldsymbol{\tau}_{\text{current}} = [-0.2572, -0.0514]$ (the unit is m/s). Compared with the ideal environment of group 1, group 2 considers the environmental perturbations from wind and current, which shows effective validation of the algorithm's robustness under simulation conditions and they are also much closer to the real situation during ship berthing.

In accordance with the framework of the auto-berthing control system shown in Figure 3, during the path planning phase the CMA-ES algorithm optimizes the APF berthing path planning, rendering the ship's motion path safer and smoother. By fully considering the maneuvering constraints intrinsic to the ship itself and the frequency and amplitude of changes in the ship's rotational velocity and rudder angle, all those constraints ensure the actual ship motion dynamics are as close as possible to the real berthing process. In Figure 6, the simulation results with CMA-ES optimization at the initial berthing stage are illustrated.

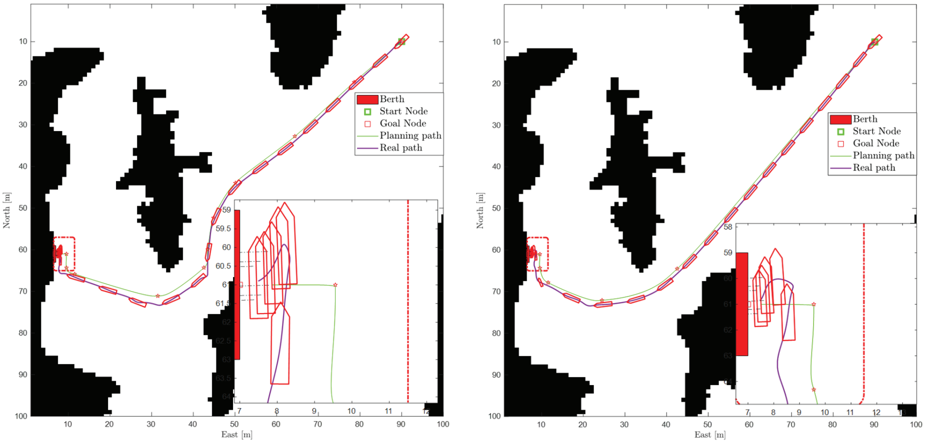


(**a**) Before optimization

(**b**) After optimization

Figure 6. The approaching process before and after optimization with wind and current.

For the second berthing stage illustrated in Figure 7, motion constraints imposed on the ship's steering gear and propeller aim to exclude high-frequency control inputs to the steering gear during the berthing process. The performance metrics of the controller are optimized via the CMA-ES method, enabling its rapid and stable convergence to the desired state and completing the berthing task.

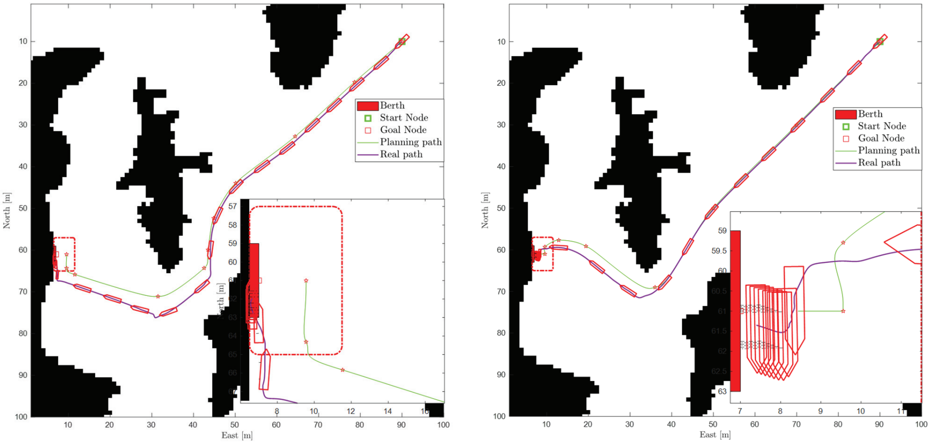


(**a**) Before optimization

(**b**) After optimization

Figure 7. The auto-berthing process before and after optimization with wind and current.

Ships get as close to the berth against the wind and current as possible in tidal harbors of China, as mentioned in reference [38]. In actual berthing operations, choosing a strategy of berthing against the wind and current can prevent the ship from being pushed away by the current and wind. The experimental water area is subject to wind and current interference in the north and east directions ($\tau_{\text{wind}} = [2, -3.5]$,$\tau_{\text{current}} = [-1.0289, 0.514]$, the unit is m/s), as shown in Figure 8.



(**a**) Before optimization

(**b**) After optimization

Figure 8. Two berthing methods with wind and current.

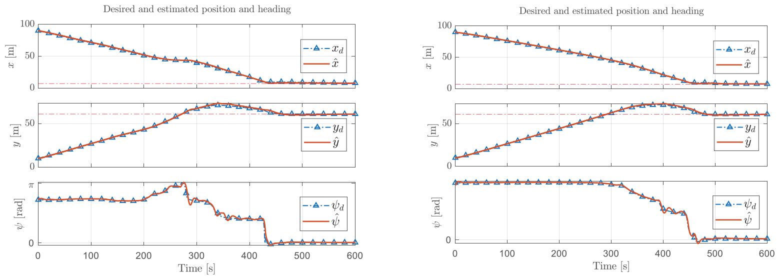
Hence, Figure 8a delineates the outcome of the berthing methodology being adopted under the identical experimental conditions. Owing to the substantial current, the drifting motion of the vessel is rather conspicuous. A considerably large inertial residual velocity is still maintained adjacent to the berth, which poses a grave peril to the execution of the vessel's berthing operation. By executing a turning maneuver, the collision risk imposed by the drifting motion when the vessel comes to a stop adjacent to the berth is averted, as shown in Figure 8b.

4.3. Result Analysis

By comparing Figures 5–7, it is shown that the wind and current have a great influence on the ship's motion. From the berthing stages in Figures 5b and 7a, we can find the influence of wind and current on the ship's low-speed motion: the wind interference will cause ships, especially small ships, to deviate greatly from the planned trajectory at low speed.

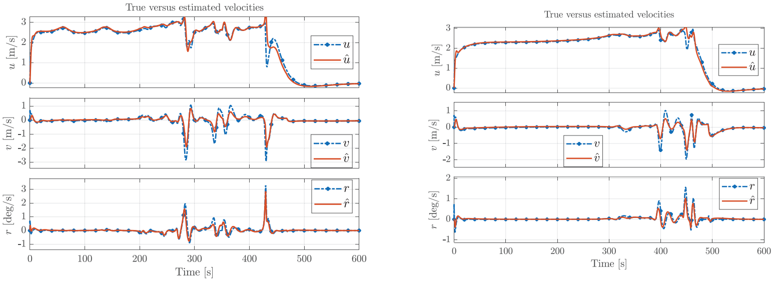
Figures 6 and 7 delineate the autonomous berthing process of a USV under specific wind and current interference conditions (stabilization outside the berth followed by parallel berthing). Figures 6a and 7a represent the nonoptimized scenarios. Figures 6b and 7b display the optimal control profiles obtained through the CMA-ES optimization approach. A comparative analysis between Figure 7a,b reveals that even with wind and current, the CMA-ES optimized control system still shows competence in mitigating abrupt fluctuations in the ship's motion and amplitude.

Specifically, in the berthing path planning in Figure 6, the left side of Figure 6a is without optimization processing. In the middle part of the figure, the path planned by the APF algorithm has many waypoints. After optimization processing, Figure 6b has much less twisting and turning, and the overall path becomes smoother. This smoothed path can effectively reduce the rudder angle changes of the ship during sailing and reduce the number of manipulations, and this change can be verified in the curve of the ship's rudder angle changes in Figure 9.



(**a**) Curves of position and heading changes before optimization

(**b**) Curves of position and heading changes after optimization

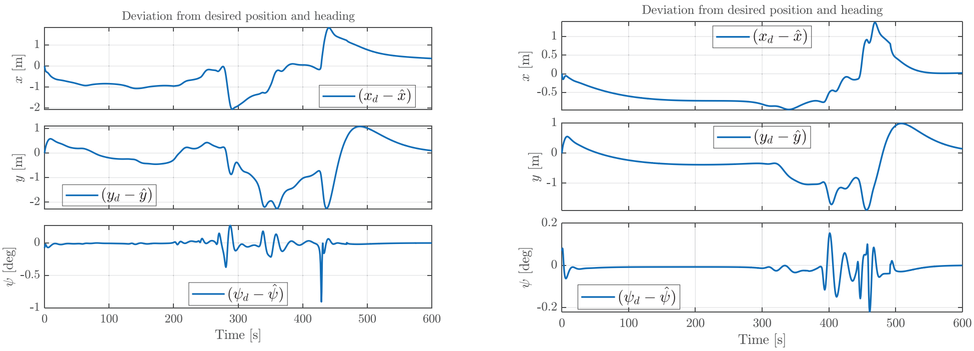


(**c**) Velocity profile before optimization

(**d**) Velocity profile after optimization

Figure 9. Comparison of motion parameters in the approaching stage. (The red represents the actual, and the blue curve is the predicted).

The enlarged part of Figure 7 shows the final berthing process of the ship. The left figure shows the result of LQR control without optimization, and the right figure shows the result after optimization. Comparison shows that the process shown in the optimized Figure 7b is more acceptable. From the evaluation of the berthing effect, the berthing error after optimization is better, and the error curve in Figure 10 can support this result.



(**a**) Before optimization (**b**) After optimization

Figure 10. Location and heading error values.

Through the comparison of the optimization of the whole control process in Figures 6 and 7, it is found that the optimized effect of the CMA-ES is significant.

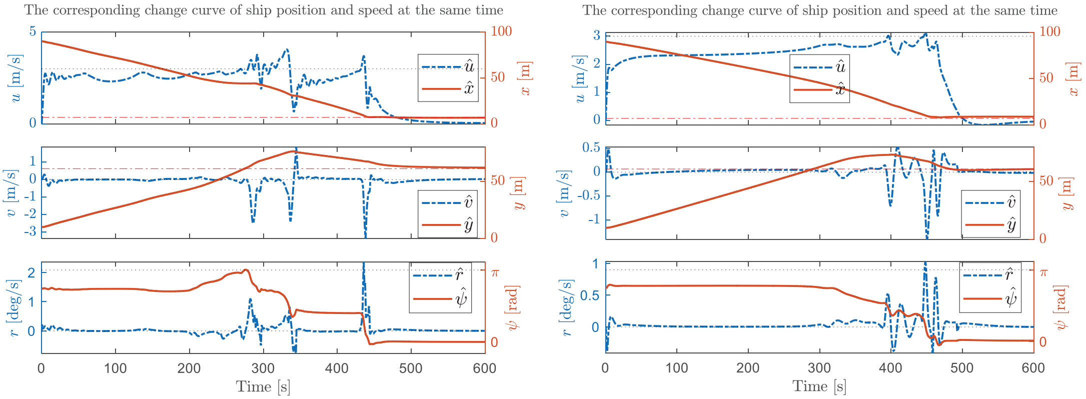
A comparison between Figure 9a,b demonstrates that the incorporation of path optimization significantly alleviates the abrupt changes in the ship's heading during navigation. Within the simulation time span of 0 to 400 s, the optimized path results in smoother motion. During the 400 to 600 s period when the ship proceeds to the second docking stage, the path optimization enables the ship to make minimal attitude adjustments to satisfy the target state with smoother changes in heading angle and position, eliminating any instantaneous mutations. Meanwhile, it can be observed that the optimized path is more amenable as the input for the controller, yielding higher consistency between the actual ship motion and the planned path.

A comparison between Figure 9c,d indicates that the path optimization augments the fault tolerance of the ship's output control during the first docking stage. The speed profile in Figure 9c exhibits frequent fluctuations with evident instantaneous reversals in polarity, which are undoubtedly deleterious to the ship's steering control. In contrast, Figure 9d does not show such issues. Meanwhile, it can be observed that incorporating the optimization function in the second docking stage renders the output of the LQR controller more stable. Compared to the sharp increases and decreases in Figure 9c, the curve in Figure 9d demonstrates that the overall process does not incur rapid changes in either speed or rudder angle, cohering with the actual operation of ship berthing maneuvers.

Obviously, it can also be found from Figure 9c,d that even if the frequency and amplitude of the rudder angle change are reduced as a whole, the optimized rudder angle is more frequent than before in the docking stage during the 400 to 600 s, which may be caused by the high accuracy setting.

In contrast, the control system optimized by the CMA-ES demonstrates higher accuracy in control effect and greater energy saving of the system, as evidenced by the heading variation frequency and amplitude illustrated in Figure 10.

According to the simulation results of the two berthing strategies under the same environment in Figure 8, it can be analyzed that: when there are strong wind and current disturbances to the ship's berthing motion, the control difficulty of the ship increases in Figure 8a when berthing directly, and the probability of collision with the berth is increased; the turning strategy in Figure 8b can effectively offset the strong wind and current interference to the ship and ensure the safety of ship berthing. Choosing a suitable berthing strategy is also a way to reduce the risk of berthing collision and ensure the safety of ship berthing. The motion process curves of the two different berthing strategies are shown in Figure 11.



(**a**) Berth alongside (**b**) Turning around and berthing

Figure 11. Comparison of motion process parameters of different berthing strategies. (The blue dotted line on the left axis represents the speed change curve of the movement, and the red solid line on the right axis represents the position change curve.)

Analysis of Figure 12 indicates that the optimized rudder angle profile displays significantly reduced fluctuations with lower magnitude of changes; the abrupt shifts in heading angle have been eliminated, resulting in a smoother curve. By comparing the simulation results of the two berthing strategies, the optimized path (we understand it as strategy 2, that is, turning around is required under the current environment) appears generally smoother, decreasing the frequency of steering gear operation during ship berthing. Within the experimental setting of this study and given the berthing maneuver strategy employed, it warrants attention that the conditions should entail a relatively stable state of wind and current; more substantial marine environmental interferences can compromise the stability of the ship's autonomous berthing process.

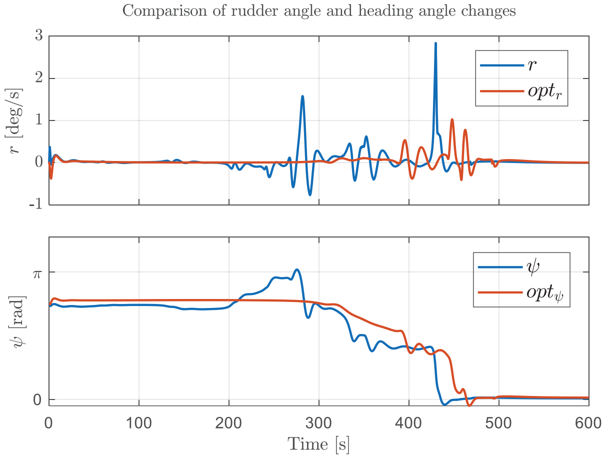


Figure 12. Changes in ship heading and heading angle before and after optimization.

5. Conclusions

In this paper, a complete framework of a berthing control system is proposed, and the berthing control system based on an LQR controller is optimized by the CMA-ES. During the experiment, the dynamic factors of the ship, the interference of the natural environment and the energy consumption of motion are fully considered. The results show that the berthing process of the ship can be well controlled, which is in line with the experimental expectation. The simulation results show that the automatic berthing system based on the CMA-ES proposed in this paper is feasible. However, in the current experimental process, there are still some shortcomings, such as the shallow water bank effect of the ship not being considered and the gain parameters of the controller cannot fully adapt to the model changes. In the future, we will continue to carry out in-depth research on the autonomous berthing of USVs and complete the algorithm test in a real environment in combination with a real testing ship.

Author Contributions: Conceptualization, G.C.; methodology, J.Y.; validation, G.C., J.Y. and S.Y.; writing—review and editing, J.Y. and S.Y.; funding acquisition, G.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (No. 51879119), the Key Projects of the National Key R & D Program (No. 2021YFB390150), the Natural Science Project of Fujian Province (No. 2022J01323, 2021J01822), the Science and Technology Plan Project of Fujian Province (No. 3502ZCQXT2021007).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kose, K.; Fukudo, J.; Sugano, K.; Akagi, S.; Harada, M. Study on a Computer Aided Manoeuvring System in Harbours. *Nav. Archit. Ocean. Eng.* **1987**, *25*, 105–113.
2. Mizuno, N.; Uchida, Y.; Okazaki, T. Quasi Real-Time Optimal Control Scheme for Automatic Berthing. *IFAC Pap. Line* **2015**, *48*, 305–312. [CrossRef]
3. Zhang, Q.; Zhang, X.; Im, N. Ship nonlinear-feedback course keeping algorithm based on MMG model driven by bipolar sigmoid function for berthing. *Int. J. Nav. Archit. Ocean. Eng.* **2017**, *9*, 525–536. [CrossRef]
4. Xu, H.X.; Zhu, M.F.; Yu, W.Z. Robust adaptive control for automatic berthing of intelligent ships. *J. Huazhong Univ. Sci. Technol. (Nat. Sci. Ed.)* **2020**, *48*, 25–29.
5. Han, Z.Z. Simulation Study on Automatic Berthing of Ships Based on Active Disturbance Rejection Neural Network Control. Master's thesis, Dalian Maritime University, Dalian, China, 2020.
6. Jia, Y.P.; Shen, H.L.; Yin, Y.; Zhang, X.F. Simulation of autonomous berthing of unmanned ships based on neural network. *China Navig.* **2021**, *44*, 107–111.
7. Wang, S.; Jin, H.; Meng, L.; Li, C. Optimize motion energy of AUV based on LQR control strategy. In Proceedings of the 2016 35th Chinese Control Conference (CCC), Chengdu, China, 27–29 July 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 4615–4620.
8. Yang, S.F. Research on Multi-Objective Optimization Based on Adaptive Learning Mechanism of Covariance Matrix. Master's thesis, Guizhou University, Guiyang, China, 2019.
9. Maki, A.; Sakamoto, N.; Akimoto, Y.; Nishikawa, H.; Umeda, N. Application of optimal control theory based on the evolution strategy (CMA-ES) to automatic berthing. *J. Mar. Sci. Technol.* **2020**, *25*, 221–233. [CrossRef]
10. Maniyappan, S.; Umeda, N.; Maki, A.; Akimoto, Y. Effectiveness and mechanism of broaching-to prevention using global optimal control with evolution strategy (CMA-ES). *J. Mar. Sci. Technol.* **2021**, *26*, 382–394. [CrossRef]
11. Miyauchi, Y.; Maki, A.; Umeda, N.; Rachman, D.M.; Akimoto, Y. System parameter exploration of ship maneuvering model for automatic docking/berthing using CMA-ES. *J. Mar. Sci. Technol.* **2022**, *27*, 1065–1083. [CrossRef]
12. Yazdanpanah, R.; Mahjoob, M.J.; Abbasi, E. Fuzzy LQR controller for heading control of an unmanned surface vessel. In Proceedings of the International Conference in Electrical and Electronics Engineering, Selangor, Malaysia, 4–5 December 2013; pp. 73–78.
13. Brasel, M. Adatpive LQR control system for the nonlinear 4-DoF model of a container vessel. In Proceedings of the 2013 18th International Conference on Methods & Models in Automation & Robotics (MMAR), Miedzyzdroje, Poland, 26–29 August 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 711–716.

14. Shao, C. Research on Large Ship Maneuvering Motion Control Based on LQR. Master's thesis, Shanghai Jiao Tong University, Shanghai, China, 2017.
15. Esmailian, E.; Farzanegan, B.; Malekizadeh, H.; Ghassemi, H.; Ardestani, M.F.; Menhaj, M.B. Control System Design for a Surface Effect Ship by Linear-Quadratic Regulator Method. *J. Mar. Eng.* **2017**, *13*, 47–56.
16. Tian, T. Research on Rudder Roll Stabilization Control System Based on Robust Optimal Control. Master's thesis, Harbin Engineering University, Harbin, China, 2021.
17. Zhao, Y.; Zhang, Z.; Wang, J.; Wang, H. Fin-Rudder Joint Control Based on Improved Linear-Quadratic-Regulator Algorithm. *IEEE Access* **2022**, *10*, 111105–111114. [CrossRef]
18. Chen, J.Y. Walking Optimization of Simulated Soccer Robot Based on Improved CMA-ES. Master's thesis, Nanjing University of Posts and Telecommunications, Nanjing, China, 2020.
19. Maki, A.; Sakamoto, N.; Akimoto, Y.; Banno, Y.; Maniyappan, S.; Umeda, N. On broaching-to prevention using optimal control theory with evolution strategy (CMA-ES). *J. Mar. Sci. Technol.* **2021**, *26*, 71–87. [CrossRef]
20. Maki, A.; Akimoto, Y.; Naoya, U. Application of optimal control theory based on the evolution strategy (CMA-ES) to automatic berthing (part: 2). *J. Mar. Sci. Technol.* **2021**, *26*, 835–845. [CrossRef]
21. Liu, Z.L.; Yuan, S.Z.; Zheng, L.H.; Jiang, J.Y.; Sun, Y.X. The development status and trend of ship automatic berthing technology. *China Shipbuild.* **2021**, *62*, 293–304.
22. Akimoto, Y.; Miyauchi, Y.; Maki, A. Saddle Point Optimization with Approximate Minimization Oracle and Its Application to Robust Berthing Control. *ACM Trans. Evol. Learn. Optim.* **2022**, *2*, 1–32. [CrossRef]
23. Miyauchi, Y.; Sawada, R.; Akimoto, Y.; Umeda, N.; Maki, A. Optimization on planning of trajectory and control of autonomous berthing and unberthing for the realistic port geometry. *Ocean. Eng.* **2022**, *245*, 110390. [CrossRef]
24. Homburger, H.; Wirtensohn, S.; Reuter, J. Docking control of a fully-actuated autonomous vessel using model predictive path integral control. In Proceedings of the 2022 European Control Conference (ECC), London, UK, 12–15 July 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 755–760.
25. Sawada, R.; Hirata, K.; Kitagawa, Y.; Saito, E.; Ueno, M.; Tanizawa, K.; Fukuto, J. Path following algorithm application to automatic berthing control. *J. Mar. Sci. Technol.* **2021**, *26*, 541–554. [CrossRef]
26. Zhang, H.; Yin, C.; Zhang, Y.; Jin, S.; Li, Z. Reinforcement Learning from Demonstrations by Novel Interactive Expert and Application to Automatic Berthing Control Systems for Unmanned Surface Vessel. *arXiv* **2022**, arXiv:2202.113252022.
27. Kamil, A.; Melhaoui, Y.; Mansouri, K.; Rachik, M. Artificial neural network and mathematical modeling of automatic ship berthing. *Commun. Math. Biol. Neurosci.* **2022**, *2022*, 113.
28. Xu, Z.; Galeazzi, R. Guidance and Motion Control for Automated Berthing of Twin-waterjet Propelled Vessels. *IFAC-Pap. Line* **2022**, *55*, 58–63. [CrossRef]
29. China Classification Society. Guidelines for Autonomous Cargo Ships (2018) Released. *Ship Stand. Eng.* **2018**, *51*, 53.
30. Yoshimura, Y. Study on Mathematical Model of Steering Motion in Shallow Water (Part 2): Fluid Forces Acting on Main Hull during Low-speed Steering. *J. Kansai Shipbuild. Ocean. Eng.* **1988**, 77–84.
31. Fossen, T.I. *Handbook of Marine Craft Hydrodynamics and Motion Control*; John Wiley & Sons: Hoboken, NJ, USA, 2011.
32. Sun, B.D. *Theoretical Basis of Modern Control, 4rd ed*; Mechanical Industry Press: Beijing, China, 2018.
33. Hansen, N.; Kern, S. Evaluating the CMA evolution strategy on multimodal test functions. In Proceedings of the PPSN 2004: Parallel Problem Solving from Nature—PPSN VIII, Birmingham, UK, 18–22 September 2004; Springer: Berlin/Heidelberg, Germany, 2004; Volume 8, pp. 282–291.
34. Fossen, T.I. A nonlinear unified state-space model for ship maneuvering and control in a seaway. *Int. J. Bifurc. Chaos* **2005**, *15*, 2717–2746. [CrossRef]
35. Zhang, X.K.; Zhang, G.Q. Simple and Robust Control of Ship Maneuvering in Port. *China Navig.* **2014**, *37*, 31–34.
36. Khatib, O. Real-Time Obstacle Avoidance for Manipulators and Mobile Robots. *Int. J. Robot. Res.* **1986**, *5*, 90–98. [CrossRef]
37. Liu, X.Z. Research on Automatic Berthing Control of Ships Based on Optimal Trajectory Planning. Master's thesis, Dalian Maritime University, Dalian, China, 2020.
38. Fu, Z.-Y. Self-reliant berthing and unberthing operation of small and medium-sized ships in restricted waters of tidal ports. *China. Water Transport. (First Half Mon.)* **2019**, 56–58. [CrossRef]

Journal of
Marine Science and Engineering

MDPI

Article

An Improved A-Star Ship Path-Planning Algorithm Considering Current, Water Depth, and Traffic Separation Rules

Rong Zhen *, Qiyong Gu, Ziqiang Shi and Yongfeng Suo

Navigation College, Jimei University, Xiamen 361021, China
* Correspondence: zrandsea@163.com; Tel.: +86-15080331250

Abstract: The influence of the maritime environment such as water currents, water depth, and traffic separation rules should be considered when conducting ship path planning. Additionally, the maneuverability constraints of the ship play a crucial role in navigation. Addressing the limitations of the traditional A-star algorithm in ship path planning, this paper proposes an improved A-star algorithm. Specifically, this paper examines the factors influencing ship navigation safety, and develops a risk model that takes into account water currents, water depth, and obstacles. The goal is to mitigate the total risk of ship collisions and grounding. Secondly, a traffic model is designed to ensure that the planned path adheres to the traffic separation rules and reduces the risk of collision with incoming ships. Then, a turning model and smoothing method are designed to make the generated path easy to track and control for the ship. To validate the effectiveness of the proposed A-star ship path-planning algorithm, three cases are studied in simulations and representative operational scenarios. The results of the cases demonstrate that the proposed A-star ship path-planning algorithm can better control the distance to obstacles, effectively avoid shallow water areas, and comply with traffic separation rules. The safety level of the path is effectively improved.

Keywords: ship global path planning; A-star algorithm; navigational safety; path optimization

Citation: Zhen, R.; Gu, Q.; Shi, Z.; Suo, Y. An Improved A-Star Ship Path-Planning Algorithm Considering Current, Water Depth, and Traffic Separation Rules. *J. Mar. Sci. Eng.* **2023**, *11*, 1439. https://doi.org/10.3390/jmse11071439

Academic Editor: Markel Penalba

Received: 30 June 2023
Revised: 14 July 2023
Accepted: 17 July 2023
Published: 18 July 2023

1. Introduction

Ships are essential carriers in maritime transportation and play a crucial role in the transportation network [1]. Additionally, the integration of intelligent technologies, such as autonomous systems and advanced data analytics, into ships has become an inevitable advancement in maritime transportation [2]. In recent years, there has been rapid development in the technology of intelligent ships, which has garnered unprecedented attention in both military and commercial sectors [3,4]. The utilization of intelligent ships offers numerous benefits, such as reduced labor costs, energy savings, and fewer accidents [5]. Furthermore, these ships can effectively perform complex and hazardous engineering tasks in specific waters [6]. Research on intelligent ship autonomy has gained increasing attention as they heavily rely on highly autonomous systems.

Path planning plays a vital role in the development of autonomous systems for ships. It serves as the foundation for ship systems [7]. A safe and efficient path is essential for ships to ensure routine safe navigation, dynamic collision avoidance, and avoidance of grounding areas [8]. Path planning entails determining a secure and effective route from the starting point to the destination while considering specific requirements such as path length, risk factors, and rule constraints [9,10]. Path planning can be broadly classified into global path planning and local (real-time) path planning [11]. Global path planning calculates paths in advance in a static environment with stationary obstacles, whereas local path planning computes navigational paths in real time in a dynamic environment with both moving and stationary obstacles. Research on path planning is crucial for the advancement of autonomous systems in ships and serves to enhance their autonomy.

Despite the numerous proposed improvements in ship path-planning algorithms, many of these methods fail to sufficiently account for the impact of the marine environment on ship navigation [12,13]. The majority of existing path-planning algorithms prioritize enhancing algorithm performance and broadening their application scope. While safety-oriented path planning for unmanned vehicles has placed emphasis on assessing obstacle hazards, only some studies have thoroughly evaluated the risks associated with diverse factors in marine environments. As a result, there is a pressing need to develop a path-planning algorithm that considers various navigation factors to comprehensively enhance path safety.

The A-star algorithm has been extensively studied and utilized in the field of path planning due to its high performance and efficiency in most cases. At the same time, the A-star algorithm offers the advantage of heuristic search, which reduces the number of search nodes while ensuring an optimal path, thus improving search efficiency. However, the conventional A-star algorithm only takes into account the path length, making it unsuitable for ship navigation in complex sea environments. To effectively utilize the A-star algorithm for ship path planning, it is crucial to expand its functionality to incorporate multiple navigation safety factors, thus enhancing path safety. Additionally, the planned path must accommodate the dynamic motion characteristics of the ship. In global path planning, the minimum turning radius of the ship holds significant importance as a parameter of the ship's dynamic motion characteristics; therefore, it must be given due consideration. This paper aims to propose an improved A-star algorithm that enhances the security of global path planning. The contributions of this paper are as follows:

(1) By analyzing the key factors that impact the safe navigation of ships, this paper establishes a risk model that comprehensively considers factors such as water current, water depths, and obstacle distances. The model aims to reduce the risk of collision with obstacles and prevent grounding.
(2) This paper quantifies the traffic separation rules to establish a traffic model. The model enables the ship to adhere to traffic separation rules, reducing the risk of collision with incoming ships.
(3) This paper proposes a turn model and a smooth method to enhance the smoothness of the path. The model optimizes the path on the basis of the ship's minimum turning radius to make it easier for the ship to track.

The remainder of this article is structured as follows: Section 2 introduces recent research on path-planning algorithms, including a detailed overview of the A-star algorithm. Section 3 explains the established methods used in the risk, traffic, and turn models. Section 4 presents the principles and specific methods for improving the A-star algorithm. Section 5 showcases three case studies. Lastly, Section 6 summarizes the conclusions and proposes directions for future research.

2. Literature Review

Ship path planning differs from path planning for robots, roads, and other applications because it considers the influence of the water environment and the maneuvering restrictions of the ship [14,15]. This section offers a concise overview of recent research advancements in ship path-planning algorithms and provides a detailed examination of the research conducted on the A-star algorithm in the context of ship path planning.

2.1. Research Progress of Ship Path Planning

Ship path planning plays a crucial role in achieving autonomous navigation [16]. Over the years, global path-planning methods have been primarily categorized into four groups: search algorithms based on existing map information, random expansion algorithms, intelligent bionic algorithms, and deep reinforcement learning algorithms. The Dijkstra algorithm [17] is a typical search algorithm that utilizes existing map information to determine the shortest path between two points. The A-star algorithm, which enhances the efficiency of the Dijkstra algorithm by incorporating a heuristic function, has been widely employed in ship path planning [18].

The most commonly used random expansion algorithms in ship path planning are the rapidly exploring random trees (RRT) algorithm and the probabilistic road map (PRM) algorithm [19]. These algorithms obtain feasible paths by randomly sampling path nodes, which allows them to effectively solve path-planning problems with complex constraints without requiring accurate environmental modeling [20]. However, paths planned by these algorithms may not be suitable for tracking, and the convergence speed of the algorithms can be slow. To address the slow convergence of the RRT algorithm, Dong et al. [21] developed an environment framework that provides an initial path and guides the algorithm's expansion. Furthermore, Cao et al. [22] proposed an RRT algorithm enhanced with path shearing and smoothing modules to mitigate navigational risks in inland rivers. However, these algorithms do not consider the specific navigation characteristics of the ship and cannot guarantee that the path is the optimal solution.

Intelligent bionic algorithms, derived from bionic research, are employed to address path-planning challenges in complex environments. Commonly used algorithms include genetic algorithm (GA), ant colony optimization (ACO), and particle swarm optimization (PSO) [23]. Although these algorithms are robust, they often get stuck in local optima and are slow in planning. Thus, they are frequently used in combination with other algorithms [24]. To overcome the problem of GA getting trapped in the local optimum, Long et al. [25] introduced a new population initialization method with adaptive mutation and crossover probabilities to escape from the local optimal solution. Zhang et al. [26] suggested a differential evolution particle velocity approach, which effectively resolved the issue of PSO falling into the local optimum. Implementing these algorithms typically requires extensive environmental modeling, substantial computational resources, and parameter tuning. These requirements can pose challenges when applying them to practical path-planning scenarios.

The rapid advancement of autonomous ship navigation has rendered traditional algorithms insufficient to meet the demands of ship operations. As a result, researchers have increasingly turned to deep learning and reinforcement learning techniques for ship path planning [27]. Li et al. [28] enhanced the action space and reward function by incorporating marine collision avoidance rules into the reward function, and by introducing an artificial potential field. Simulation experiments demonstrated that the trained ship model can autonomously avoid collisions. Liu et al. [29] improved the efficiency and generalization capability of an algorithm by utilizing continuous multi-time target ship information and a redesigned reward function. Their approach, based on the TD3 algorithm, resulted in a smoother and more stable path. However, it is important to note that reinforcement learning algorithms require extensive training of ships to effectively plan paths. In reality, ship accidents are relatively rare, and available data are limited. Therefore, the application of such algorithms to real-world scenarios remains challenging.

2.2. Research Progress of A-Star Algorithm

The A-star algorithm is widely utilized in path planning due to its high level of completeness and optimality. However, when used in isolation, it can lead to paths that overlook environmental factors, exhibit poor algorithm efficiency, and do not align with real conditions [30]. To tackle these challenges, researchers have put forward various enhancements to the A-star algorithm.

Complex environments and large maps exponentially increase the time complexity of the A-star algorithm [31]. To tackle this challenge, researchers have made improvements to the search method of the A-star algorithm. Duchon et al. [32] enhanced the algorithm's efficiency by modifying the traditional eight-neighbor search to an omnidirectional search and incorporating the hop search algorithm into the path-planning process. Chen et al. [33] introduced a bidirectional A-star algorithm that utilizes a dynamic window to search for paths simultaneously from both the start and the goal positions. The dynamic window significantly improved the algorithm's efficiency, resulting in a substantial reduction in search time. Fernandes et al. [34] improved the algorithm's efficiency by limiting node expansion and constructing 16 directional layers that only visit adjacent layers when searching for the lowest cost. Similarly, Zhang et al. [35] proposed an expansion method that prioritizes nodes in the same direction as the target node, reducing the number of expanded nodes in the A-star algorithm.

The A-star algorithm often produces paths with many turns and lacks smoothness [36]. Researchers have proposed various cost functions to smoothen paths and several post-smoothing algorithms for generated paths to address these issues. Thaker et al. [37] tackled the problem of excessive turning points in A-star algorithm paths by increasing the offset distance of obstacles using a buffer area centered on the robot. Experimental results confirmed that this method resulted in smoother paths. Lu et al. [38] introduced a path-planning algorithm that combined the A-star algorithm with the Floyd algorithm to reduce the sharpness of breakpoints in the planned path. Gunawan et al. [39] addressed the issue of large turning point angles in the traditional A-star algorithm paths by incorporating angle information between nodes to minimize the turning angle, thereby avoiding potential safety hazards associated with sudden turns of unmanned vehicles. Sun et al. [40] combined the A-star algorithm with the variable weight evaluation function of the artificial potential field (APF) method and optimized the relevant parameters using the PSO algorithm, resulting in smoother planned paths and enabling dynamic obstacle avoidance.

Environmental factors have a significant impact on global route planning for ships. Recognizing the limitation of the A-star algorithm in considering only path length, researchers have endeavored to incorporate various environmental factors in the maritime area that affect navigation. To address the issue of the A-star algorithm's path being too close to obstacles, Shu et al. [41] proposed the utilization of an obstacle detection method to select safer nodes. In their study, Liu et al. [42] improved the algorithm's cost function by considering the risk of water currents, presenting an A-star algorithm that addresses the combination problem of the normal path and the berthing path. Additionally, Liu [43] integrated environmental water depth interpolation into the algorithm and analyzed the ship motion characteristics to devise an A-star algorithm that takes into account water depth risk, effectively reducing the risk associated with water depth along the path.

In summary, researchers have made significant improvements to the A-star algorithm, resulting in increased efficiency and practicality. However, these improved algorithms often fail to guarantee safe navigation due to their limited consideration of environmental factors, ship turning restrictions, and traffic separation rules. While these algorithms may offer shorter paths, they also introduce higher navigation risks, making them unsuitable for intelligent navigation [44]. To address these limitations, this paper proposes an improved A-star algorithm that incorporates various navigation factors. The algorithm comprises risk model, traffic model, and turn model. The risk model takes into account the influence of current, water depth, and obstacle distance on navigation risks, thereby avoiding grounding areas and maintaining a safe distance from obstacles. The traffic model ensures that ships comply with maritime rules, reducing the risk of collision with incoming ships. Furthermore, to better align with the dynamic motion characteristics of the ship during the path-planning stage, the algorithm utilizes a geometric smoothing method. This method optimizes the path to accommodate the ship's turning radius, ensuring that it is in line with the ship's maneuverability. By considering these factors, the proposed algorithm enhances both the safety and the practicality of ship navigation.

3. Model Design

3.1. Overview of the Model

When planning a ship's path, it is crucial to consider the length of the path and various risks that can impact navigation safety [45]. Turbulent currents, grounding, ship encounters, and improper maneuvering are among the factors that can compromise the safety of navigation [46,47]. The influence of water currents on ship movement is particularly significant in the ocean. The speed and direction of currents can have a profound impact on a ship's course. Neglecting the influence of water currents during a voyage can lead to deviations from the intended path, resulting in unnecessary delays or increased risks. Water depth is another critical factor that affects navigation safety. Insufficient water depth can lead to grounding or damage to the ship. Therefore, when planning a ship's path, it is essential to consider the limitations imposed by water depth to avoid potential safety issues. Furthermore, adherence to traffic separation rules is necessary in busy waterways to ensure safe and orderly navigation. Failure to comply with these rules can result in hazardous situations such as traffic congestion or collisions. Considering these factors, path planning should incorporate risk assessment to enhance navigation safety. By accounting for turbulent currents, water depth limitations, and traffic separation rules, the path can be optimized to minimize potential risks and ensure a safe and efficient voyage.

To focus on the given problem, certain assumptions were made in this paper. The overall map was assumed to represent a confined ocean environment near a harbor, resulting in a relatively short total travel distance and no significant environmental changes during the ship's journey. Assuming a static environment is reasonable for short-duration travel of 1–2 h. This is because weather information is typically forecasted and updated on an hourly basis, with minimal changes expected during this period. Therefore, if the travel time is <1 h, it can be safely assumed that the environment is static.

To tackle these challenges, three models were developed: the risk model, the traffic model, and the turn model. Figure 1 illustrates the relationship between the risk factors and the respective models. By integrating these models into the path-planning process, we can prioritize safety and ensure the utmost security for ship navigation.

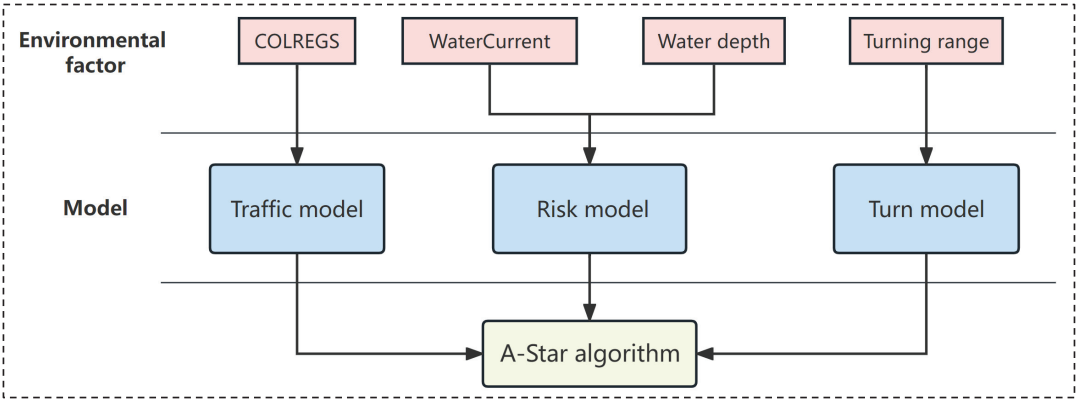


Figure 1. Correspondence between risk factors and models.

3.2. Risk Model

Although the traditional A-star algorithm can identify the shortest path, the generated path often remains very close to obstacles. This approach significantly increases the navigation risk due to the inherent time lag and inertia in ship movements. During navigation, if the water currents push the ship toward obstacles, the risk of collision further escalates. Furthermore, water depth plays a crucial role in safe navigation as it directly affects the risk of grounding. Therefore, in path planning, it is essential to simultaneously consider the proximity to obstacles, water depth, and the influence of water and wind currents.

On the basis of the analysis conducted earlier, this paper presents the following definition of the risk model:

$$r_s(m,n) = r_{obs}(m,n) + r_{depth}(m), \tag{1}$$

where $r_{obs}(m,n)$ denotes the risk from obstacles, and $r_{depth}(m)$ denotes the stranding risk from shallow water area. The purpose of designing the model $r_{obs}(m,n)$ is to maintain a certain safe distance from obstacles in the environment and prevent collisions between the ship and obstacles caused by wind, water currents, and other factors. The purpose of designing the model $r_{depth}(m)$ is to ensure that the ship can avoid grounding or potentially grounding areas. The expression for $r_{obs}(m,n)$ is as follows:

$$r_{obs}(m,n) = e^{-d}(1 + v_{cur}h_c), \tag{2}$$

where d is the distance from a navigable node $N[m]$ to the obstacle node $O[n]$. Furthermore, v_{cur} is the current velocity. It should be noted that this article replaced obstacle expansion with the distance function in Equation (2). This was to prevent the algorithm from being trapped in a locally optimal solution in a narrow area. Lastly, h_c denotes the direction coefficient, expressed as follows:

$$h_c = \begin{cases} -v_U\cos(\theta_{cur} - \theta_{mn}), & \cos(\theta_{cur} - \theta_{mn}) < 0 \\ 0, & \cos(\theta_{cur} - \theta_{mn}) \geq 0 \end{cases}, \tag{3}$$

where v_U is the ship velocity, and θ_{cur} represents the angle between the water current v_{cur} and the horizontal direction. As shown in Figure 2, θ_{mn} is the angle between vector $\overrightarrow{N[m]O[n]}$ and the horizontal direction, while $\overrightarrow{N[m]O[n]}$ represents the angle formed between the current position of the ship $N[m]$ and the obstacle $O[n]$.

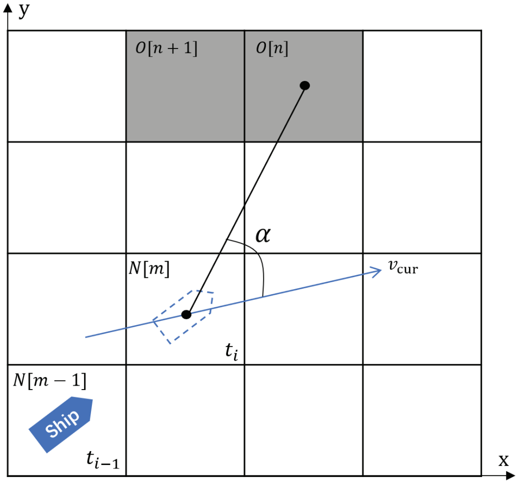


Figure 2. Schematic diagram of calculating risk.

In Figure 2, assuming that node $N[m]$ is the node where the ship sails at time t_i, it can be observed that at this moment the ship has collision risks with obstacles $O[n]$ and $O[n+1]$. α represents the angle between the vector $\overrightarrow{N[m]O[n]}$ and the water current v_{cur}. By calculating α, the risk value at node $N[m]$ can be computed using Equation (2).

To ensure that the ship can avoid grounding areas, we define $r_{depth}(m)$ as follows:

$$r_{depth}(m) = \begin{cases} \frac{S_{min}}{D(m)}, & S_{min} < D(m) \\ inf, & D(m) \leq S_{min} \end{cases}, \tag{4}$$

where $D[m]$ is the water depth at node $N[m]$, and S_{min} is the maximum draft of the ship, expressed as follows:

$$S_{min} = z_{max} + 0.5Ltan\theta_{max} + \overline{T} + e_{enc}, \tag{5}$$

where z_{max} is the maximum settlement amplitude of ships at different velocity, L is the length of ships, θ_{max} is the maximum pitch angle, $\overline{T}$ is the average draft under the mission load, and e_{enc} is the calculation error.

According to Equation (1), there will be k obstacle nodes near a feasible node $N[m]$, corresponding to k different risk values. In this case, the maximum value is taken as the final risk degree.

$$r_s(m,n) = \max\{r_s(m,1), r_s(m,2), r_s(m,3), \ldots, r_s(m,k)\}. \tag{6}$$

In addition, a local search is used instead of a global search to save time. The search range is set to d_g, as shown in Figure 3.

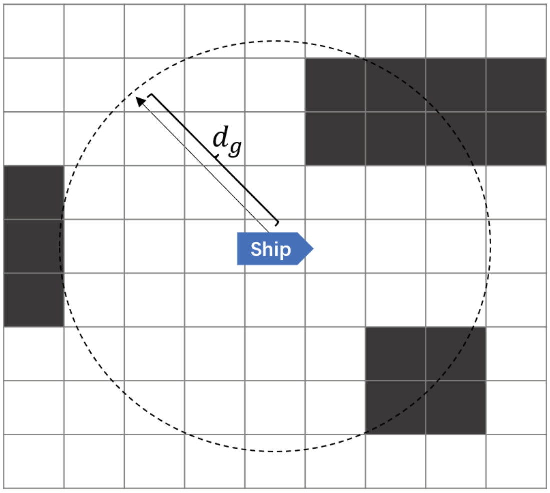


Figure 3. Schematic diagram of local search.

3.3. *Traffic Model*

Similar to road traffic, ship navigation is also governed by traffic rules at sea. The International Regulations for Preventing Collisions at Sea (COLREGS) mandate that ships should cross separation zones in a direction perpendicular to the separation zone whenever possible. This rule aims to clarify the intention of crossing, minimize the risk of collision with other ships within the separation zone, and enhance navigation efficiency. However, the conventional A-star algorithm does not inherently ensure that the planned path aligns with the specified direction required by the traffic separation rules. To address this issue, we define the following traffic model on the basis of the rules:

$$r_{tra}(m) = \begin{cases} 1 - \cos\left(\theta_t - \theta_{ship}\right), & if\ \cos\left(\theta_t - \theta_{ship}\right) \geq 0 \\ math.inf, & if\ \cos\left(\theta_t - \theta_{ship}\right) < 0 \end{cases}, \tag{7}$$

where $r_{tra}(m)$ is the traffic separation cost at node $N[m]$, θ_t is the direction of traffic rules toward the true north, $N[m].f$ is the parent node of $N[m]$, and θ_{ship} is the direction of path vector $\overrightarrow{N[m]O[m].f}$ toward the true north, which is the ship's driving direction. As shown in Figure 4, the orange horizontal line denotes the traffic separation zone. The orange arrow is the driving direction specified by the traffic rules, where $\alpha = \theta_t - \theta_U$ is used to calculate whether the currently planned path conforms to the specified direction. For example, if the ship continues to follow the path in Figure 4 at time t_{i+1}, the ship will violate the traffic rules at sea.

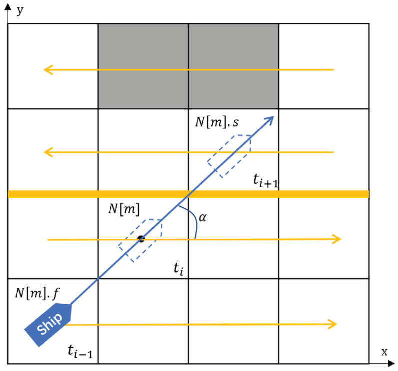


Figure 4. Schematic diagram of the traffic model.

3.4. Turn Model

In actual ship navigation, frequent turning can increase the difficulty of the captain's maneuvering and pose a risk to navigation. However, traditional A-star path-planning algorithms only consider the path length and often include many turning nodes. To address this issue, this paper presents a turn model to reduce the number of turns in the planned path. The turn model is defined as follows:

$$r_{turn}(m) = arcos\left(\frac{\overrightarrow{N[m]N[m].s}}{\overrightarrow{N[m].fN[m]}}\right), \tag{8}$$

where $r_{turn}(m)$ is the turning cost at node $N[m]$, $N[m].s$ is the child node of node $N[m]$, and $N[m].f$ is the parent node of node $N[m]$. As shown in Figure 5, the ship is at node $N[m]$ at time t_i. The turning cost here is estimated by calculating the angle β_i. β_i is the angle of vector $\overrightarrow{N[m]N[m].s_i}$ toward the vector $\overrightarrow{N[m].fN[m]}$. A greater turning range at node $N[m]$ attracts a greater turning cost. This model not only reduces the number of turns but also limits the scope of turns to a certain extent.

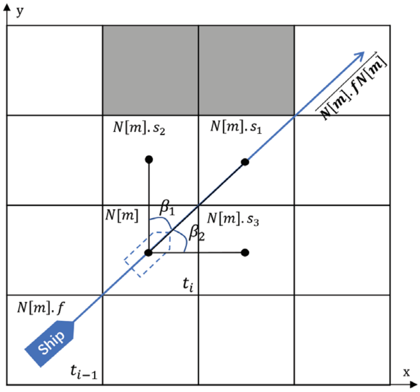


Figure 5. Schematic diagram of the turn model.

4. A-Star Algorithm and Improvements

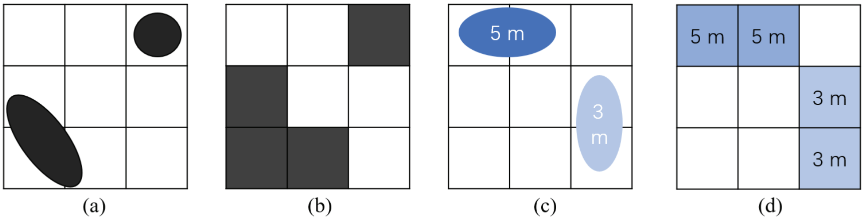
This section begins by introducing the rules of environmental modeling, which can help to ensure that the A-star algorithm plans a path that is safer, better suits ship tracking, and adheres to traffic separation rules. Next, we propose an improved A-star algorithm based on the model presented in Section 3.

4.1. Environment Modeling

To accurately represent the path-planning process, it is necessary to create a two-dimensional environment map. There are several conventional methods for environmental modeling, including the grid, geometric information, and view methods [48]. The grid method involves dividing a planar map into a series of grids to create a grid map. This method is efficient in representing the characteristics of the actual environment while optimizing time and space consumption. It is also simple and direct, reducing the path search time and simplifying programming. Therefore, the grid method is utilized in this study to model the navigation environment.

(1) Water area and water depth division standard

The division of grids into navigable and non-navigable areas can be based on the boundaries of the environment. A grid that contains objects is considered non-navigable and denoted in black, representing non-navigable waters. On the other hand, a grid that does not contain any objects is considered navigable and denoted in white, representing navigable water areas, as illustrated in Figure 6a. Obstructions that occupy less than one grid are expanded to ensure grid regularity and facilitate subsequent simulation implementation. The navigable and non-navigable waters after expansion are depicted in Figure 6b. The method for establishing the water depth environment follows the same process as above, as shown in Figure 6c,d.



(a) (b) (c) (d)

Figure 6. Obstacles division and water depth standards.

(2) Movement rules of ships in the grid environment

Ships can only move within the white grids representing feasible water areas and cannot cross or appear in the black grids. At the same time, the ship can move in eight directions within the grid environment, as illustrated in Figure 7.

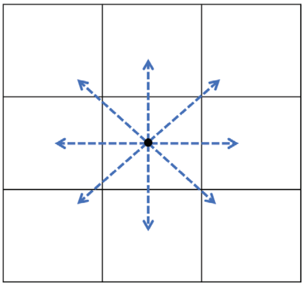


Figure 7. Schematic diagram of eight neighborhoods.

It is worth noting that the algorithm proposed in this paper is also applicable to the Electronic Nautical Chart (ENC) environment. The specific steps are to import the ENC data into the computer for preprocessing and establish the environmental model of the electronic chart on the basis of the processed ENC data. The environmental model is established using the grid map method, with each grid containing data on water depth, water currents, wind currents, and traffic separation rules. Lastly, path planning is performed using the constructed environment and the improved algorithm, and the resulting path nodes are displayed on the electronic chart [49].

4.2. Improved A-Star Algorithm

Given the limitations of traditional A-star algorithms in ship path planning, this section explores an improved A-star algorithm based on the risk models developed in Section 3.

4.2.1. Traditional A-Star Algorithm

The A-star algorithm is a heuristic algorithm based on the graph method, and it is also the most effective direct search algorithm for finding the shortest path in static road networks. Due to its high accuracy and efficiency, it is widely used in global path planning. The algorithm searches for the path by calculating the cost function of each node in the field, which has the following form:

$$F(N[i]) = g(N[i]) + h(N[i]), \tag{9}$$

where $g(N[i])$ is the actual cost function from $N[start]$ to $N[i]$, $h(N[i])$ is the cost estimation function from $N[i]$ to $N[target]$, and $F(N[i])$ is the cost estimation function from $N[start]$ to $N[target]$ through $N[i]$. Here, $h(N[i])$ has several forms, which are expressed by Euclidean distance in this paper, as follows:

$$h(N[i]) = \sqrt{(x_i - x_g)^2 + (y_i - y_g)^2}, \tag{10}$$

where (x_i, y_i) and (x_g, y_g) are the current node $N[i]$ and target node $N[target]$ position coordinates, respectively.

4.2.2. Improved A-Star Algorithm

To incorporate the risk model, traffic model, and turn model into the A-star algorithm, this paper introduces a redesigned cost function. The improved cost function, denoted as $F(N[i])$, is defined as follows:

$$F(N[i]) = g(N[i])' + \varepsilon h(N[i]), \tag{11}$$

$$g(N[i])' = g(N[i]) + \pi r_n(N[i]), \tag{12}$$

where $\varepsilon > 0$ is a constant coefficient. ε is used to balance the weight between $g(n)'$ and $h(n)$. $\pi > 0$ is a constant coefficient. $g(N[i])$ is defined as

$$g(N[i]) = \sum_{i=0}^{t-1} d(k), \tag{13}$$

where t is the node amount from $N[start]$ to $N[i]$. $r_n(N[i])$ is the risk function, which is defined as follows:

$$r_n(N[i]) = r_s(N[i]) + r_{tra}(N[i]) + r_{turn}(N[i]), \tag{14}$$

where $r_s(n), r_{tra}(n),$ and $r_{turn}(n)$ were defined in Equations (1), (7) and (8).

The improved A-star algorithm was designed to consider various factors that affect navigation safety so that the planned path can meet several key characteristics. Firstly, it can maintain a safe distance from obstacles in the environment. Secondly, it can avoid shallow water areas and reduce the risks associated with currents. Thirdly, it can comply with traffic separation rules. Lastly, it can minimize the number of turns required. The pseudocode of Algorithm 1 demonstrates the implementation of the improved A-star algorithm.

Algorithm 1: Improved A-star algorithm

```
Mark $N[start]$ as openlist
if not traffic separation rule then
  $r_{tra}$ = 0 at any time
While $openlist \neq \varnothing$ do
  Select openlist $N[i]$ whose value of $F_n[i]$ is the smallest
  if $N[i]$=$N[goal]$ then
    return "path $P_n$ is found"
  else
    Mark $N[i]$ as closelist
    if successor $N_i[j]$ of $N[i]$ not in closelist or openlist then
      Mark $N_i[j]$ as openlist
      Calculate $r_s(j)$, $r_{tra}(j)$, $r_{turn}(j)$ by (1), (7), and (8), respectively.
      if $N_i[j]$ in openlist and $F_{new}(N_i[j])$ is smaller than $F_{old}(N_i[j])$ then
        $F_{old}(N_i[j]) = F_{new}(N_i[j])$ and set parent node of $N[j]$ as $N[i]$
return "path $P_n$ is not found"
```

4.2.3. Smooth Paths with Geometry

The map is modeled using a grid-based approach, and the planned path consists of line segments formed by the grid. However, the vertices of this path are not conducive to the tracking and smooth navigation of the ship. Since the dynamic motion characteristics of the ship play a crucial role in path planning, and the turning radius of the ship is an important parameter of its dynamic motion characteristics in global path planning, the planned path must be within the range of the ship's maneuverability. To address this issue, this paper adopts the geometric smoothing path method to replace the vertices of the planned path with curve segments, taking into account the ship's minimum turning radius as an important reference parameter. The optimization process is shown in Figure 8.

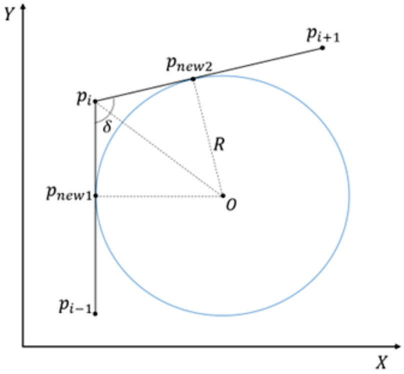


Figure 8. Principle of geometrically smooth paths.

As shown in Figure 8, $p_{i-1}(x_{i-1}, y_{i-1}), p_i(x_i, y_i),$ and $p_{i+1}(x_{i+1}, y_{i+1})$ are polyline segments on the planned path. We use a circle of radius R such that $p_{i-1}p_i$ and $p_i p_{i+1}$ are tangents to the circle, intersecting at $p_{new1}(x_{new1}, y_{new1})$ and $p_{new2}(x_{new2}, y_{new2})$. The calculation process of p_{new1}, p_{new2} is as follows:

$$\delta = arcos\left(\frac{\overrightarrow{\mathrm{p_{i-1}p_i}}}{\overrightarrow{\mathrm{p_{i+1}p_i}}}\right), \tag{15}$$

$$p_i p_{new1} = p_i p_{new2} = R \times \cot\left(\frac{\delta}{2}\right). \tag{16}$$

The radius of the circle is set to $R = p_{new1}O$, and the coordinate scale coefficient φ of the tangent point is set as follows:

$$\varphi_1 = \frac{p_i p_{new1}}{p_{i-1} p_i},\ \varphi_1 = \frac{p_i p_{new2}}{p_i p_{i+1}}. \tag{17}$$

From $p_{i-1}(x_{i-1}, y_{i-1}), p_i(x_i, y_i), p_{i+1}(x_{i+1}, y_{i+1})$, and the coordinate scale coefficient φ, we get

$$p_{new1} = (\varphi_1 \times x_{i-1} + (1-\varphi_1) \times x_i,\ \varphi_1 \times y_{i-1} + (1-\varphi_1) \times y_i), \tag{18}$$

$$p_{new2} = (\varphi_2 \times x_{i+1} + (1-\varphi_2) \times x_i,\ \varphi_2 \times y_{i+1} + (1-\varphi_2) \times y_i). \tag{19}$$

The slopes and formulas of the straight line $p_{i-1}p_i$ and the straight line $p_i p_{i+1}$ are

$$k_1 = \frac{y_i - y_{i-1}}{x_i - x_{i-1}}, \quad y_1 = k_1 \cdot x - \frac{x_{i-1}y_i - x_i y_{i-1}}{x_i - x_{i-1}}, \tag{20}$$

$$k_2 = \frac{y_{i+1} - y_i}{x_{i+1} - x_i}, \quad y_2 = k_2 \cdot x - \frac{x_i y_{i+1} - x_{i+!} y_i}{x_{i+1} - x_i}. \tag{21}$$

From the slope k_1 and the point p_{new1}, the vertical straight line y_1 and the straight line $y_1\prime$ passing through the point p_{new1} can be obtained. In the same way, $y_2\prime$ can be obtained ($y_2\prime$ passes through p_{new2} and is perpendicular to y_2). The intersection of the straight lines $y_1\prime$ and $y_2\prime$ is the center of the circle, and the center of the circle is set to $O(x_o, y_o)$; then, the function expression of the curve after smooth geometric optimization is as follows:

$$f(x) = \begin{cases} y_1, & | \, x_{i-1} \le x \le x_{new1} \\ \pm\sqrt{R^2 - (x - x_o)^2} + y_o, & | \, x_{new1} < x < x_{new2} \\ y_2, & | \, x_{new2} \le x \le x_{i+1} \end{cases}. \tag{22}$$

In practical applications, we can adjust the curvature of the smooth curve by changing the size of R, so that the curvature of the optimized curve satisfies the minimum turning radius of ships.

5. Case Study

Simulation experiments were conducted in this section to validate the effectiveness of the proposed improved A-star path-planning method. First, the path-planning performance of the proposed A-star method was compared with traditional A-star methods, considering obstacles, water depth, water currents, and traffic separation rules. Then, the proposed improved A-star method was tested in real scenes in Zhoushan and Hainan ports. It should be noted that the environmental data for Case 2 and Case 3 were obtained from shipxy.com. In case 2, nautical charts were used for path planning instead of satellite maps to illustrate the bathymetric boundaries and marine traffic diversion areas. In case 1 and case 2, the China-made autonomous cargo ship, 'Jindouyun 0', was selected for the simulation experiment. The ship's key parameters are shown in Table 1. The minimum safe navigation depth was calculated on the basis of the ship's key parameters.

Table 1. Ship key parameters.

Parameters	Value
Ship length	12.86 m
Ship width	3.8 m
Design maximum draft	1 m
Minimum turning radius R_0	36 m
Design velocity	8 knots

5.1. Case 1: Path Planning in Complex Simulation Environment

For this study, we conducted simulation experiments on the Python 3.8 platform and used a 50 × 40 grid map for the experiment. In order to simulate realistic environmental conditions, we created a complex environment that included current velocity, shallow water areas, and traffic separation zones.

5.1.1. Setup

The relevant model parameters need to be determined to establish the proposed model. The velocity of the current is set to $v_m = 2\text{m/s}$. The direction of the current is due east. The ship velocity is set to $v_U = 4\text{m/s}$. $N[start]$ is set to (5, 8), and $N[target]$ is set to (40, 35). The grid length is set to 20 m. The depth of shallow water is set to 1 m. Other parameters of the simulation experiment are shown in Table 2. In order to verify the effectiveness of each model, this section designs a path for each model. Using the model established in Section 3, the following four different cost functions are used for simulation experiments: (1) Path1 is the traditional A-star algorithm; (2) Path2 verifies the validity of the risk model, where $r_n(N[i]) = r_s(N[i]) + r_{turn}(N[i])$; (3) Path3 verifies the validity of the traffic model, where $r_n(N[i]) = r_{tra}(N[i]) + r_{turn}(N[i])$; (4) Path4 simultaneously considers all models in Section 3, i.e., $r_n(N[i]) = r_s(N[i]) + r_{tra}(N[i]) + r_{turn}(N[i])$.

Table 2. Parameter initialization.

Parameters	Value
Node range of x-axis	[1, 50]
Node range of y-axis	[1, 40]
Grid length l	20 m
Minimum radius of ship R_0	36 m
π	0.2
ε	0.5

5.1.2. Results

Figure 9 compares the four different path-planning results. Path1 is located close to the obstacle and passes through a shallow water area, violating traffic rules and increasing the risk of grounding. Moreover, Path1 has many sharp turns that could lead to ship collisions and groundings. Path2 starts planning the path toward the upper left direction from the starting point, aiming to move away from obstacles and reduce the risk of water currents pushing the ship toward them. This demonstrates the effectiveness of the proposed risk model in this paper. However, Path2 does not comply with the rule of separated traffic, which could increase the risk of collision with oncoming ships when navigating in the opposite direction. Path3 follows the separation rule by driving to the right in the lower part of the map and crossing the separation zone perpendicularly in the upper part of the map to reach the destination. This demonstrates the effectiveness of the traffic model proposed in this paper. However, Path3 is still too close to the obstacle, which is due to the lack of consideration of the risk model. Path4 is a path planned using the improved algorithm proposed in this paper that considers multiple risk factors. This path keeps a safe distance from the obstacle, avoids shallow water areas, and reduces the risk of ship navigation. Path4 also adheres to traffic separation rules, with fewer turns and a smoother trajectory.

With the aid of geometrically smooth curves, the optimized path ensures that the curvature of turns remains within the maneuverability range of the ship, making it more suitable for safe navigation and tracking. The proposed algorithm comprehensively improves the safety of path planning in complex marine environments, making the planned route highly suitable for ship navigation, and demonstrating its superiority over the traditional methods.

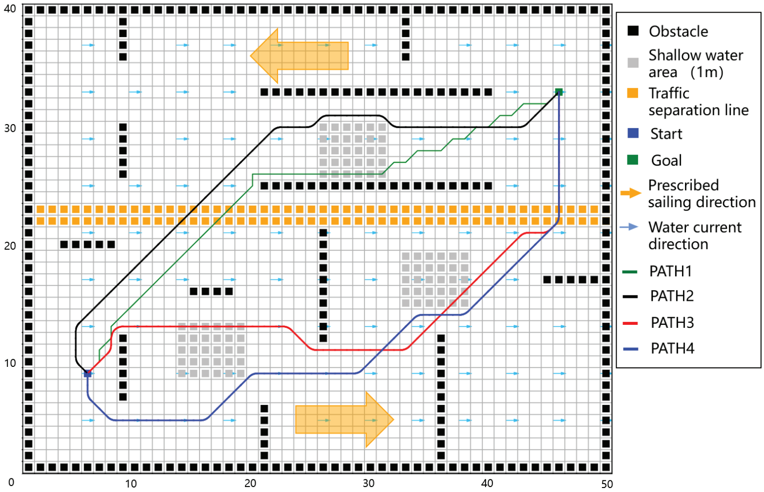


Figure 9. Simulation results of four different cost functions.

To further demonstrate the superiority of the improved algorithm, we calculated risk indicators for two paths: navigation risk $r_s = r_s(m, n)$. We compared path length, navigation risk, number of turns, maximum turn angle, and adherence to traffic separation rules among the four paths. The risk calculation was based on Equation (1) of the risk model.

The simulation results of the statistical experiment are presented in Table 3. Comparing the data, we can observe that Path1 planned by the traditional A-star algorithm has the shortest length. However, it has significantly higher risks of ship grounding and collision, as well as more turns, indicating a higher navigation risk compared to the algorithm proposed in this paper. Path2 also considers the risk model and has a navigation risk close to Path4. Path3 complies with the traffic model and conforms to traffic rules, but its proximity to the obstacle increases the risk of ship navigation. Paths2–4 have fewer turns compared to Path1, as they all consider the turn model. Path 4 significantly reduces various risks at the expense of a certain path length. Meanwhile, the path is smooth and conforms to traffic separation rules. The turning radius at the path's bends is within the ship's handling capabilities. In summary, the advantages of Path4 prove the effectiveness of the improved A* algorithm.

Table 3. Comparison of experimental simulation data.

	Path Length (m)	**Collision Risk** r_s	**Turn Times (*n*)**	**Compliance with Traffic Separation Rules**
Path1	516.9	30.11	19	No
Path2	545.2	13.53	7	No
Path3	592.1	18.58	8	Yes
Path4	608.7	9.39	8	Yes

5.2. Case 2: Path Planning in Real Scenes in Zhoushan Port

In this scenario, the effectiveness of the improved A-star method is verified using a real scene of the hidden reef in Zhoushan Harbor. The ship intends to travel from a starting point (29°56.551 N, 122°13.826 E) to a destination point (29°56.96 N, 122°13.60 E). The area is characterized by numerous hidden reefs that obstruct ships from passing through. In addition, several shallow areas in this range affect navigation, as depicted in Figure 10. The grid length is set to 25 m, and the velocity of the water current is set to $v_c = 1\ \mathrm{m/s}$. The direction of the water current is northwest. The geometric smooth path radius R is set to 50 m. It should be noted that the reef area does not adhere to traffic separation regulations due to the low volume of ship traffic in the area, and traffic separation rules are not taken into account in this case. The remaining parameter settings are the same as in Case 1.

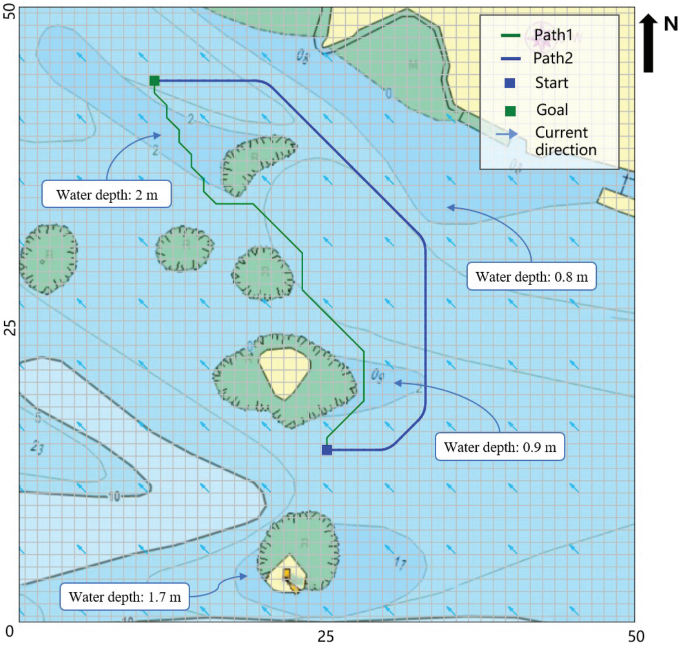


Figure 10. The path planned by the proposed A-star algorithm in Zhoushan port, China.

In Figure 10, Path1 and Path2 represent the paths generated by the traditional and improved A-star algorithms, respectively. From Figure 10, it can be observed that the path distance planned by the traditional A-star algorithm is close to the reefs. In complex marine environments, there is a high probability of collision with reefs due to the influence of water currents, which undermines the assurance of safe ship arrival at the destination. Furthermore, Path1 passes through two shallow water areas comparable to the ship's maximum draft, posing a high risk of grounding. Compared with Path1, Path2 avoids all shallow water areas and keeps a certain distance from obstacles to ensure that the ship will not collide with obstacles due to water currents. Table 4 compares various metrics for the paths. According to the experimental data, Path1 traverses shallow water areas below the maximum draft of the ship, resulting in an infinite risk of grounding. The navigation risk associated with Path1 is extremely high, making it unsuitable for ship tracking. On the other hand, Path2 takes into account multiple safety factors during navigation, exhibiting a higher level of safety and thus being more suitable for ship operations. Moreover, Path2 has a turning radius of 50 m (2 grids), well within the maneuverable range of the ship, ensuring a smooth trajectory. In summary, the improved A-star algorithm significantly reduces ship risks while slightly increasing the path length. Case 2 demonstrates the effectiveness of the proposed model and highlights its practical significance in the context of complex shallow water ship path planning.

Table 4. Comparison of experimental simulation data.

	Path Length (miles)	Risk r_s	Turn Times (n)
Path1	0.52	inf	16
Path2	0.65	21.8	4

5.3. Case 3: Path Planning in Real Scenes in Hainan Port

In this case study, we choose the real scene of Hainan Port to verify the effectiveness of the improved A-star method. There are multiple traffic separation zones in this area, as shown in the rectangular area circled by the pink dotted line in Figure 11. The sides of the traffic separation zone dictate the opposite direction of travel, and ships that violate these rules run the risk of colliding with incoming ships. Therefore, it is crucial to ensure that planned routes respect traffic segregation regulations. In order to verify the effectiveness of the algorithm in this case, the "Yude Ship" is selected for simulation experiments. The "Yude Ship" has a length of 199 m, a full load draft of 12 m, and a minimum turning radius of twice the length of the ship. The planning start point (20°4.755 N, 110°8.501 E) and end point (20°15.392 N, 110°22.11 E) are set. The area is divided into 100 × 100 grids, and the length of each grid is set to 300 m. The water current velocity v_c is 2 m/s, and the direction is 10° northeast. The smooth path radius R is set to two meshes.

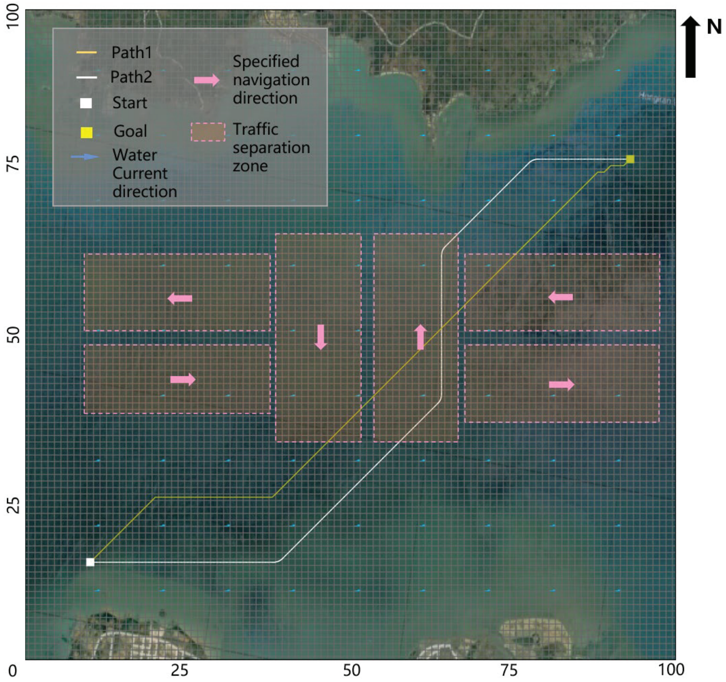


Figure 11. The path planned by the proposed A-star algorithm in Hainan port, China.

In Figure 11, Path1 and Path2 represent paths generated by the traditional A-star algorithm and the improved A-star algorithm, respectively. The area framed by the dotted line of the pink rectangle indicates the traffic control area where ships are required to obey the traffic separation rules. Pink arrows indicate the navigation direction for each area. As shown in Figure 11, although both paths avoid all navigation markers and reach the destination safely, their planned paths are quite different. Path2 obeys the traffic separation rules, while Path1 violates the rules, and there is a path traveling in the opposite direction. Table 5 compares the experimental result data of the two paths. According to the experimental data, the reverse driving of Path1 violates the traffic rules, resulting in an infinite risk value. Path2 sacrifices a certain path length to make the planned path comply with traffic separation rules. The path planned by the improved A* algorithm improves the safety of the path and enables the ship to follow the prescribed channel. In summary, Case 3 verifies the effectiveness of the improved A-star algorithm.

Table 5. Comparison of experimental simulation data.

	Path Length (miles)	**Risk r_s**	**Turn Times (n)**	**Compliance with Traffic Separation Rules**
Path1	17.38	inf	6	No
Path2	19.54	25.8	4	Yes

6. Conclusions

This paper presented a ship path-planning approach that considers multiple safety factors. The proposed algorithm aims to enhance ship navigation safety by considering environmental effects, traffic regulations, and ship maneuvering constraints. It considers environmental factors such as water currents and water depth, as well as traffic regulations and the minimum turning radius of the ship. The effectiveness of the proposed A-star algorithm was demonstrated through three cases. The simulation results showed that the algorithm effectively considers multiple risk factors during navigation, maximizing the safety of the voyage. The planned paths not only comply with traffic regulations but also remain within the ship's maneuvering capabilities, ensuring safe and efficient navigation. Additionally, the algorithm strikes a balance between path length and navigation safety, reducing the risks of ship collisions and groundings. These improvements to the A-star algorithm have significant potential for enhancing path-planning safety during ship navigation. The findings contribute to the field of ship navigation safety, benefiting the maritime industry and mitigating the risks associated with ship collisions and groundings.

With the rapid development of meteorology and measurement technology, more accurate environmental information can be forecasted. By utilizing this information, safer and more precise paths can be generated. The proposed path-planning system is generally applicable to ships of any size, as most parameters such as ship dimensions, maneuvering constraints, and water depth are used as input parameters.

A major drawback of the proposed path-planning method is the increased time complexity due to the increased number of nodes and computational burden. This can be reduced through code optimization, and the time used for a priori path planning before actual navigation would not substantially affect the operation of the ship. Second, some assumptions and simplifications in this article may differ from reality. In future research, the consideration of factors such as weather can be explored to plan paths that avoid adverse weather conditions.

Author Contributions: Conceptualization, R.Z. and Q.G.; methodology, Q.G.; software, Z.S.; validation, Q.G. and Z.S.; formal analysis, Y.S.; resources, R.Z.; data curation, Q.G.; writing—original draft preparation, Q.G.; writing—review and editing, R.Z.; visualization, Y.S.; project administration, R.Z.; funding acquisition, R.Z. All authors read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (grant number 52001134), the Fujian Provincial Natural Science Foundation (grant numbers 2020J01661, 2020J01660 and 2023J01326), and the Fuzhou–Xiamen–Quanzhou Independent Innovation Region Cooperated Special Foundation (grant number 3502ZCQXT2021007).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created in this study.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Silveira, P.; Teixeira, A.; Figueira, J.R.; Soares, C.G. A multicriteria outranking approach for ship collision risk assessment. *Reliab. Eng. Syst. Saf.* **2021**, *214*, 107789. [CrossRef]
2. Liu, J.; Yan, X.; Liu, C.; Fan, A.; Ma, F. Developments and Applications of Green and Intelligent Inland Vessels in China. *J. Mar. Sci. Eng.* **2023**, *11*, 318. [CrossRef]
3. Jorge, V.A.M.; Granada, R.; Maidana, R.G.; Jurak, D.A.; Heck, G.; Negreiros, A.P.F.; dos Santos, D.H.; Goncalves, L.M.G.; Amory, A.M. A Survey on Unmanned Surface Vehicles for Disaster Robotics: Main Challenges and Directions. *Sensors* **2019**, *19*, 702. [CrossRef] [PubMed]
4. Wang, L.; Li, S.; Liu, J.; Hu, Y.; Wu, Q. Design and implementation of a testing platform for ship control: A case study on the optimal switching controller for ship motion. *Adv. Eng. Softw.* **2023**, *178*, 103427. [CrossRef]
5. Felski, A.; Zwolak, K. The ocean-going autonomous ship—Challenges and threats. *J. Mar. Sci. Eng.* **2020**, *8*, 41. [CrossRef]
6. Zhen, R.; Shi, Z.; Liu, J.; Shao, Z. A novel arena-based regional collision risk assessment method of multi-ship encounter situation in complex waters. *Ocean Eng.* **2022**, *246*, 110531. [CrossRef]
7. Wang, N.; Sun, Z.; Yin, J.; Zou, Z.; Su, S.-F. Fuzzy unknown observer-based robust adaptive path following control of underactuated surface vehicles subject to multiple unknowns. *Ocean Eng.* **2019**, *176*, 57–64. [CrossRef]
8. Zhen, R.; Shi, Z.; Shao, Z.; Liu, J. A novel regional collision risk assessment method considering aggregation density under multi-ship encounter situations. *J. Navig.* **2022**, *75*, 76–94. [CrossRef]
9. Zhou, C.; Gu, S.; Wen, Y.; Du, Z.; Xiao, C.; Huang, L.; Zhu, M. The review unmanned surface vehicle path planning: Based on multi-modality constraint. *Ocean Eng.* **2020**, *200*, 107043. [CrossRef]
10. Li, S.; Xu, C.; Liu, J.; Han, B. Data-driven docking control of autonomous double-ended ferries based on iterative learning model predictive control. *Ocean Eng.* **2023**, *273*, 113994. [CrossRef]
11. Liu, J.; Wang, J. Overview of obstacle avoidance path planning algorithms for unmanned surface vehicle. *Comput. Appl. Softw.* **2020**, *37*, 1–10+20.
12. Cheng, T.; Utne, I.B.; Wu, B.; Wu, Q. A novel system-theoretic approach for human-system collaboration safety: Case studies on two degrees of autonomy for autonomous ships. *Reliab. Eng. Syst. Saf.* **2023**, *237*, 109388. [CrossRef]
13. Chen, X.; Liu, S.; Liu, R.W.; Wu, H.; Han, B.; Zhao, J. Quantifying Arctic oil spilling event risk by integrating an analytic network process and a fuzzy comprehensive evaluation model. *Ocean Coast. Manag.* **2022**, *228*, 106326. [CrossRef]
14. Liu, Z.; Zhang, Y.; Yu, X.; Yuan, C. Unmanned surface vehicles: An overview of developments and challenges. *Annu. Rev. Control* **2016**, *41*, 71–93. [CrossRef]
15. Chen, X.; Wu, S.; Shi, C.; Huang, Y.; Yang, Y.; Ke, R.; Zhao, J. Sensing data supported traffic flow prediction via denoising schemes and ANN: A comparison. *IEEE Sens. J.* **2020**, *20*, 14317–14328. [CrossRef]
16. Liu, Y.C.; Bucknall, R. Path planning algorithm for unmanned surface vehicle formations in a practical maritime environment. *Ocean Eng.* **2015**, *97*, 126–144. [CrossRef]
17. Singh, Y.; Sharma, S.; Sutton, R.; Hatton, D.; Khan, A. Feasibility study of a constrained Dijkstra approach for optimal path planning of an unmanned surface vehicle in a dynamic maritime environment. In Proceedings of the 2018 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), Torres Vedras, Portugal, 25–27 April 2018; pp. 117–122.
18. Liang, C.L.; Zhang, X.K.; Watanabe, Y.; Deng, Y.J. Autonomous Collision Avoidance of Unmanned Surface Vehicles Based on Improved a Star and Minimum Course Alteration Algorithms. *Appl. Ocean Res.* **2021**, *113*, 102755. [CrossRef]
19. Gu, Q.; Zhen, R.; Liu, J.; Li, C. An improved RRT algorithm based on prior AIS information and DP compression for ship path planning. *Ocean Eng.* **2023**, *279*, 114595. [CrossRef]
20. Chi, W.; Ding, Z.; Wang, J.; Chen, G.; Sun, L. A Generalized Voronoi Diagram-Based Efficient Heuristic Path Planning Method for RRTs in Mobile Robots. *IEEE Trans. Ind. Electron.* **2021**, *69*, 4926–4937. [CrossRef]
21. Dong, Y.; Camci, E.; Kayacan, E. Faster RRT-based nonholonomic path planning in 2D building environments using skeleton-constrained path biasing. *J. Intell. Robot. Syst.* **2018**, *89*, 387–401. [CrossRef]
22. Cao, S.; Fan, P.; Yan, T.; Xie, C.; Deng, J.; Xu, F.; Shu, Y. Inland waterway ship path planning based on improved RRT algorithm. *J. Mar. Sci. Eng.* **2022**, *10*, 1460. [CrossRef]
23. Öztürk, Ü.; Akdağ, M.; Ayabakan, T. A review of path planning algorithms in maritime autonomous surface ships: Navigation safety perspective. *Ocean Eng.* **2022**, *251*, 111010. [CrossRef]
24. Lin, H.X.; Xiang, D.; Ou, Y.J.; Lan, X.D. Review of Path Planning Algorithms for Mobile Robots. *Comput. Eng. Appl.* **2021**, *57*, 38–48.
25. Long, Y.; Su, Y.; Zhang, H.; Li, M. Application of improved genetic algorithm to unmanned surface vehicle path planning. In Proceedings of the 2018 IEEE 7th Data Driven Control and Learning Systems Conference (DDCLS), Enshi, China, 25–27 May 2018; pp. 209–212.

26. Zhang, J.; Chen, X.; Hu, H.; Zhang, J.; Wang, L. Global path planning algorithm based on the IPSO algorithm for USVs. In Proceedings of the 2018 Chinese Control and Decision Conference (CCDC), Shenyang, China, 9–11 June 2018; pp. 4901–4906.
27. Woo, J.; Kim, N. Collision avoidance for an unmanned surface vehicle using deep reinforcement learning. *Ocean Eng.* **2020**, *199*, 107001. [CrossRef]
28. Li, L.; Wu, D.; Huang, Y.; Yuan, Z.-M. A path planning strategy unified with a COLREGS collision avoidance function based on deep reinforcement learning and artificial potential field. *Appl. Ocean Res.* **2021**, *113*, 102759. [CrossRef]
29. Liu, Z.; Zhou, Z.Z.; Zhang, M.Y.; Liu, J.X. A Twin Delayed Deep Deterministic Policy Gradient Method for Collision Avoidance of Autonomous Ships. *J. Transp. Inf. Saf.* **2022**, *40*, 60–74.
30. Ayawli, B.B.K.; Chellali, R.; Appiah, A.Y.; Kyeremeh, F. An Overview of Nature-Inspired, Conventional, and Hybrid Methods of Autonomous Vehicle Path Planning. *J. Adv. Transp.* **2018**, 8269698. [CrossRef]
31. Guo, B.; Kuang, Z.; Guan, J.; Hu, M.; Rao, L.; Sun, X. An Improved A-Star Algorithm for Complete Coverage Path Planning of Unmanned Ships. *Int. J. Pattern Recognit. Artif. Intell.* **2022**, *36*, 2259009. [CrossRef]
32. Duchon, F.; Babinec, A.; Kajan, M.; Beno, P.; Florek, M.; Fico, T.; Jurisica, L. Path planning with modified A star algorithm for a mobile robot. In Proceedings of the 6th Conference on Modelling of Mechanical and Mechatronic Systems (MMaMS), Vysoke Tatry, Slovakia, 25–27 November 2014; pp. 59–69.
33. Chen, D.T.; Liu, X.D.; Liu, S.W. Improved A* algorithm based on two-way search for path planning of automated guided vehicle. *J. Comput. Appl.* **2021**, *41*, 309–313.
34. Fernandes, E.; Costa, P.; Lima, J.; Veiga, G. Towards an Orientation Enhanced A star Algorithm for Robotic Navigation. In Proceedings of the IEEE International Conference on Industrial Technology (ICIT), Seville, Spain, 17–19 March 2015; pp. 3320–3325.
35. Zhang, J.G.; Zhang, F.; Chen, L.G.; Jiang, Q.; Zhu, W. Path planning of automatic guided vehicle based on improved A* algorithm. *Foreign Electron. Meas. Technol.* **2022**, *41*, 123–128. [CrossRef]
36. Zhang, Y.; Li, L.-l.; Lin, H.-C.; Ma, Z.; Zhao, J. Development of Path Planning Approach Based on Improved A-Star Algorithm in AGV System. In Proceedings of the 3rd European-Alliance-for-Innovation (EAI) International Conference on IoT as a Service (IoTaaS), Taichung, Taiwan, 20–22 September 2017; pp. 276–279.
37. Nayl, T.; Mohammed, M.Q.; Muhamed, S.Q. Obstacles Avoidance for an Articulated Robot Using Modified Smooth Path Planning. In Proceedings of the International Conference on Computer and Applications (ICCA), Doha, United Arab Emirates, 6–7 September 2017; pp. 185–189.
38. Lu, W.L.; Lei, J.S.; Shao, Y. Path Planning for Mobile Robot Based on an Improved A* Algorithm. *J. Ordnance Equip. Eng.* **2019**, *40*, 197–201.
39. Gunawan, S.A.; Pratama, G.N.; Cahyadi, A.I.; Winduratna, B.; Yuwono, Y.C.; Wahyunggoro, O. Smoothed A-star algorithm for nonholonomic mobile robot path planning. In Proceedings of the 2019 International Conference on Information and Communications Technology (ICOIACT), Yogyakarta, Indonesia, 24–25 July 2019; pp. 654–658.
40. Sun, J.G.; Sun, T. Research on UAV path planning based on fusion A* algorithm. *Electron. Meas. Technol.* **2022**, *45*, 82–91. [CrossRef]
41. Shu, W.N.; Zhao, J.S.; Xie, Z.X.; Zhang, X.S.; Ma, X. Path planning for unmanned surface vessels based on improved A* algorithm. *J. Shanghai Marit. Univ.* **2022**, *43*, 1–6. [CrossRef]
42. Liu, C.G.; Mao, Q.Z.; Chu, X.M.; Xie, S. An Improved A-Star Algorithm Considering Water Current, Traffic Separation and Berthing for Vessel Path Planning. *Appl. Sci.* **2019**, *9*, 1057. [CrossRef]
43. Liu, S. Study on the Path Planning Algorithm of Unmanned Surface Vehicles in Complex Marine Environment. Master's Thesis, Tianjin University, Tianjin, China, 2019.
44. Lee, T.; Kim, H.; Chung, H.; Bang, Y.; Myung, H. Energy efficient path planning for a marine surface vehicle considering heading angle. *Ocean Eng.* **2015**, *107*, 118–131. [CrossRef]
45. Razgallah, I.; Kaidi, S.; Smaoui, H.; Sergent, P. The impact of free surface modelling on hydrodynamic forces for ship navigating in inland waterways: Water depth, drift angle, and ship speed effect. *J. Mar. Sci. Technol.* **2019**, *24*, 620–641. [CrossRef]
46. Zhang, X.; Zhang, Q.; Yang, J.; Cong, Z.; Luo, J.; Chen, H. Safety Risk Analysis of Unmanned Ships in Inland Rivers Based on a Fuzzy Bayesian Network. *J. Adv. Transp.* **2019**, *2019*, 4057195. [CrossRef]
47. Andersen, P. Ship motions and sea loads in restricted water depth. *Ocean Eng.* **1979**, *6*, 557–569. [CrossRef]
48. Bai, X.E.; Jiang, M.Z.; Xu, X.F.; Sun, D.Y. Ship route planning based on improved ant colony algorithm of two-way search. *China Navig.* **2022**, *45*, 13–20.
49. Wang, Y.; Liang, X.; Li, B.; Yu, X. Research and implementation of global path planning for unmanned surface vehicle based on electronic chart. In *Recent Developments in Mechatronics and Intelligent Robotics: Proceedings of the International Conference on Mechatronics and Intelligent Robotics (ICMIR2017)*; Springer: Berlin/Heidelberg, Germany, 2018; Volume 1, pp. 534–539.

Journal of
Marine Science and Engineering

MDPI

Article

Identification of Spoofing Ships from Automatic Identification System Data via Trajectory Segmentation and Isolation Forest

Hailin Zheng [1,2], Qinyou Hu [1,*], Chun Yang [1], Qiang Mei [1,3], Peng Wang [1,4] and Kelong Li [2]

[1] Merchant Marine College, Shanghai Maritime University, Shanghai 201306, China; hlzhzjou@126.com (H.Z.); chunyang@shmtu.edu.cn (C.Y.); meiqiang@jmu.edu.cn (Q.M.); wangp@ict.ac.cn (P.W.)
[2] School of Naval Architecture and Maritime, Zhejiang Ocean University, Zhoushan 316022, China; 18158057677@163.com
[3] Navigation Institute, Jimei University, Xiamen 361021, China
[4] Institute of Computing Technology, Chinese Academy of Sciences, Beijing 101408, China
* Correspondence: qyhu@shmtu.edu.cn

Abstract: Outliers of ship trajectory from the Automatic Identification System (AIS) onboard a ship will affect the accuracy of maritime situation awareness, especially for a regular ship trajectory mixed with a spoofing ship, which has an unauthorized Maritime Mobile Service Identification code (MMSI) owned by a regular ship. As has been referred to in the literature, the trajectory of these spoofing ships would simply be removed, and more AIS data would be lost. The pre-processing of AIS data should aim to retain more information, which is more helpful in maritime situation awareness for the Maritime Safety Administration (MSA). Through trajectory feature mining, it has been found that there are obvious differences between the trajectory of a regular ship and that of a regular ship mixed with a spoofing ship, such as in terms of speed and distance between adjacent trajectory points. However, there can be a long update time interval in the results of severe missing trajectories of a ship, bringing challenges in terms of the identification of spoofing ships. In order to accurately divide the regular ship trajectory and spoofing ship trajectory, combined with trajectory segmentation by the update time interval threshold, the isolation forest was adopted in this work to train the labeled trajectory point of a regular ship mixed with a spoofing ship. The experimental results show that the average accuracy of the identification of spoofing ships using isolation forest is 88.4%, 91%, 93.1%, and 93.3%, corresponding to different trajectory segmentation by update time intervals (5 h, 10 h, 15 h, and 20 h). The research conducted in this study can almost eliminate the outliers of ship trajectory, and it also provides help for maritime situation awareness for the MSA.

Keywords: automatic identification system; spoofing ship; missing points; jumping points; trajectory segmentation; isolation forest

Citation: Zheng, H.; Hu, Q.; Yang, C.; Mei, Q.; Wang, P.; Li, K. Identification of Spoofing Ships from Automatic Identification System Data via Trajectory Segmentation and Isolation Forest. *J. Mar. Sci. Eng.* **2023**, *11*, 1516. https://doi.org/10.3390/jmse11081516

Academic Editor: Sergei Chernyi

Received: 29 June 2023
Revised: 19 July 2023
Accepted: 28 July 2023
Published: 29 July 2023

1. Introduction

On 1 June 2020, the special rectification of national maritime communication order for radio equipment on board ship began, such as AIS, very-high-frequency communication (VHF), and so on. The China MSA at all levels have concentrated on the monitoring of maritime communication order and improving the ability of maritime communication supervision and maritime service support in China. The special rectification focuses on the rectification of outstanding problems, such as irregularly authorized ship MMSI, one MMSI owned by several ships, several MMSI owned by one ship, the illegal occupation of channels, and the violation of communication order.

Among the violations of maritime communication order mentioned above, the irregular use of AIS may have a significant impact on the quality of AIS data [1]. The quality of AIS data is a subject of interest for many researchers [2–5], but published research on pre-processing raw data to improve quality is limited. Shelmerdine took the development of a vessel database as the key to managing AIS data and for quality control [6]. All fields were checked for obvious outliers. If it was not possible to correct an outlier, it was removed. The common method to filter inaccurate single position points is the threshold of position, speed, and course [7]. To solve the problem of sharing MMSI numbers, a method of elimination was applied by Pallotta et al. [8]. Mazzarella et al. proposed a nearest-neighbor approach to assign AIS messages to the right tracks, but there was no detailed experimental method, performance, or results [9]. Wu et al. created a simple algorithm to calculate the likelihood of an association between an AIS message and each candidate vessel [10]. It is used for processing massive data on a global scale, but it cannot be applied in a small region where AIS messages are sampled at a high rate. The reason is that the algorithm is unable to handle an association in the case where there are at least three consecutive abnormal trajectory points. A similar method with speed threshold was proposed by Greidanus et al. [11]. Given that none of these techniques are universally applicable, it is necessary to propose a method with general applicability.

Wei et al. observed an abnormal 'jumping point' in ship trajectory by calculating the speed between adjacent ship trajectory points and setting the speed threshold according to ship maneuvering characteristics, which meant that abnormal trajectory points were identified [12]. Since the constant velocity threshold method does not consider the change in the motion state of a moving ship at different times, it can only detect some abnormal points whose velocity exceeds the specified threshold, and the robustness of this method is poor. Han et al. proposed a novel trajectory outlier detection algorithm based on the adaptive threshold, designing a local threshold window and mean filter window, and calculated the local speed (acceleration) threshold and global speed (acceleration) threshold, and found three classes of abnormal trajectory points, including isolated outliers, continuous outliers, and obvious outliers [13]. Zhang et al. calculated the bow deflection rate between adjacent trajectory points by counting the speed distribution of ship trajectory points, setting the speed threshold of trajectory points according to the probability of the speed distribution of adjacent trajectory points, setting the threshold of the bow deflection rate according to the characteristics of ship cycle, and identified abnormal trajectory points [14]. Liu et al. converted ship speed, heading, and position from AIS data into evidence reliability and used evidence reasoning rule synthesis to detect three classes of trajectory points, referring to the manual identification method of abnormal AIS data adopted by the MSA [15,16]. Chen et al. and Guo et al. cleaned ship AIS data in these three rules: abnormal ship position (the longitude and latitude of ship are beyond the scope of study area), abnormal speed (the difference between adjacent trajectory points exceeds speed threshold), and abnormal rate of turning (the course difference between adjacent trajectory segments exceeds rate of turning threshold) [17–20]. Data derived from AIS plays a key role in water traffic data mining. However, there are various errors regarding time and space. To improve availability, AIS data quality dimensions are presented by Zhao et al. to detect errors of AIS tracks, including physical integrity, spatial logical integrity, and time accuracy [21–23]. After systematic summary and analysis, algorithms for error pre-processing are proposed. In the aspect of abnormal AIS data identification, combined with the characteristics of adjacent trajectory points in a period of time, an abnormal AIS data identification model based on BP neural network was constructed by Wang et al. [24].

Zhang et al. designed an MMSI spoofing detection algorithm based on the spatiotemporal data provided by AIS and radar. When a ship is monitored by AIS and radar before and after MMSI spoofing, both monitoring processes continue for a period of time, meaning the MMSI spoofing algorithm demonstrates a good performance [25]. Iphar et al. propose a rule-based method for data integrity assessment, with rules built from the system technical specifications and by domain experts, formalized by a logic-based framework, resulting in the triggering of situation-specific alerts [26–28]. Jeong et al. provided an automatic shipping route construction method using functional data analysis (FDA), and the proposed approach includes two steps: outlier detection and shipping route construction [29]. Huang et al. proposed a new method for detecting anomalous vessel dynamics using functional data analysis. Empirical investigations of this approach demonstrate the effective detection of outlier flows in terms of ship traffic volume [30]. In summary, researchers regard outliers in trajectory points as random error values and delete them in order to clean the trajectory. However, these outliers may be the trajectory point of another ship, that is, the trajectory of a spoofing ship sharing the same MMSI with other regular ship. Moreover, due to a large number of missing trajectories, the speed between adjacent trajectory points cannot be used as a basis for distinguishing the trajectory points of a spoofing ship and regular ship.

In this paper, we aim to propose a novel spoofing ship identification framework with the support of trajectory segmentation. Our main contributions can be summarized as follows: (1) We mined the trajectory feature of a regular ship and spoofing ship and obtained the correlation between the time interval and distance and average sailing speed between the adjacent trajectory points for a regular ship and spoofing ship. (2) We segmented ship trajectory by considering the ratio of missing trajectory points and distribution of the time interval between adjacent trajectory points, and obtained the trajectory segment with a low missing points ratio. (3) Considering the low ratio of jumping trajectory points in the data sample and the higher identification efficiency of isolation forest, we adopted isolation forest to identify a spoofing ship and testified the proposed framework performance on 20 regular ship mixed with spoofing ship trajectories. We aim for this study to be able to help the MSA identify spoofing ship trajectories and thus take early warning measurements to enhance maritime traffic efficiency and safety. The remainder of this paper is organized as follows. We introduce the data source used in our study in Section 2. After that, the methodology details about trajectory feature mining of the AIS data are illustrated in Section 3, and then isolation forest used for identifying spoofing ships is presented, combined with trajectory segmentation. The experimental results are shown in Section 4. Section 5 briefly discusses the study and illustrates future work.

2. Data

Shanghai Meili Shipbuilding Technology Co., Ltd. (Shanghai, China) provides large-scale AIS data, which benefit many AIS-relevant studies due to their public accessibility (https://www.hifleet.com/, accessed on 1 June 2023). The hifleet has online access to over 50 AIS satellites and over 3000 AIS base stations, receiving 150 million AIS data per day, as well as purchase Lloyd's ship archives and access global electronic chart data and ocean meteorological data. The original AIS dataset includes both kinematic and static information for a ship, which contains the MMSI, latitude, longitude, speed over ground (SOG), heading, course over ground (COG), timestamp, call sign, port of call, and so on.

When collecting records of the port of call for container ships, it was found that some ships continuously call at ports that are far apart within a short time interval. It was found that there were jumping points in the trajectories of these ships when selecting the real-time trajectories of these container ships within the corresponding statistical time. Due to the presence of jumping points in these trajectories, the trajectories of these container ships also exhibit jumping characteristics, rather than showing continuity like those of regular ships. Consequently, we collected the AIS data of container ships which have trajectory jumping points and labeled them with different colors for different ships. We collected 20 container ships and 52,538 AIS data samples from 1 January 2017 to 31 December 2017 (see Figure 1), and the average time interval for sampling the AIS data was 1 h.

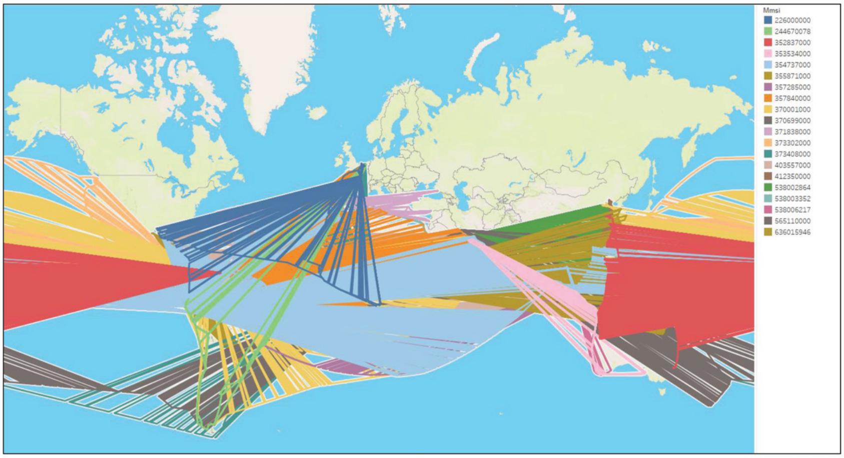


Figure 1. Raw AIS data of regular container ships mixed with spoofing ships.

3. Methodology

We were affected by the limited coverage of AIS base stations, limited AIS communication capacity, and the illegal use of unauthorized ship-borne AIS, such as a ship leaving the coverage area of AIS base stations, the quantity of ships exceeding AIS communication capacity, the deliberate closure of ship-borne AIS, the use of the same MMSI for multiple ships or the use of multiple MMSI for one ship, resulting in large quantities of missing or jumping trajectories. To address this issue, we firstly implemented trajectory characteristics mining and trajectory segmentation to obtain the distribution of speed and distance between adjacent trajectory points and then identified spoofing ships using isolated forest. The schematic overview for the proposed framework is shown in Figure 2. In order to accurately describe the ship motion pattern, ship trajectory is defined as follows:

$$S = \{\{S_1\}, \{S_2\}, \ldots, \{S_i\}, \ldots, \{S_m\}\} \tag{1}$$

$$S_i = \left\{s_i^1, s_i^2, \ldots, s_i^k, \ldots, s_i^n\right\} \tag{2}$$

$$s_i^k = (m_i^k, t_k, \lambda_i^k, \phi_i^k, v_i^k, c_i^k) \tag{3}$$

In Equation (1), $\{Si\}$ represents the trajectory of ship i. In Equation (2), s_i^k represents the trajectory point of ship i at time t_k. In Equation (3), m_i^k represents the MMSI of ship i, t_k represents the update time of AIS data, λ_i^k and ϕ_i^k represents the ship longitude and latitude at time t_k, v_i^k represents ship speed at time t_k, and c_i^k represents ship course at time t_k.

3.1. Trajectory Feature Mining

According to the trajectory point distribution for missing and jumping ship trajectory, ship trajectory points are divided into four categories: regular ship trajectory points (Normal_Point, abbreviated as N_P), spoofing ship trajectory points (Spoofing_Point, abbreviated as S_P), and confusion points (No Labeled_Point, abbreviated as NL_P).

In order to accurately detect spoofing ship trajectory points, the distance between adjacent trajectory points and average sailing speed are two important parameters. Generally speaking, average sailing speed between adjacent trajectory points is consistent with the ship maneuvering performance for regular ship. Taking the cargo ship as an example, the speed of this ship would not exceed 50 knots. Therefore, two adjacent trajectory points whose average sailing speed exceeds the speed threshold must not belong to the same ship;

accordingly, the trajectory points of regular ships mixed with a spoofing ship can be effectively identified. The average speed between adjacent trajectory points is closely related to the update time interval of trajectory points, as well as the distance between adjacent trajectory points. The distance between adjacent trajectory points could be calculated by spherical distance (namely, Great Circle distance), as calculated in Equation (4), and speed ${v_{avg}}_i^{k,\,k+1}$ between adjacent trajectory points was calculated, as in Equation (5):

$$d_i^{k,k+1} = a\cos\left(sin(\phi_i^{k+1}) * \sin(\phi_i^k) + cos(\phi_i^{k+1}) * \cos(\phi_i^k) * \cos(\lambda_i^{k+1} - \lambda_i^k)\right) \tag{4}$$

$${v_{avg}}_i^{k,k+1} = \frac{d_i^{k,k+1}}{t_{(k+1)} - t_k} = \frac{d_i^{k,k+1}}{\Delta t} \tag{5}$$

In Equation (5), Δt represents the update time interval of the trajectory points. If the update time interval of the trajectory point is not affected by the working performance of the AIS base station and traffic density, and is only related to ship speed, the trajectory point would update more frequently. At this time, the distance between adjacent trajectory points and the speed of navigation have good discrimination between regular ship trajectory and regular ship trajectory mixed with spoofing ship. The average sailing speed between regular ship trajectories is within the speed threshold, as shown in Equation (6), while the average sailing speed among a regular ship trajectory mixed with a spoofing ship is beyond the speed threshold, as shown in Equation (7).

$$\Delta t < \Delta t_{(th)}, d_i^{k,k+1} < d_{th} \wedge {v_{avg}}_i^{k,k+1} < v_{avg(th(\min))} \tag{6}$$

$$\Delta t < \Delta t_{(th)}, d_i^{k,k+1} \geq d_{th} \wedge {v_{avg}}_i^{k,k+1} \geq v_{avg(th(\max))} \tag{7}$$

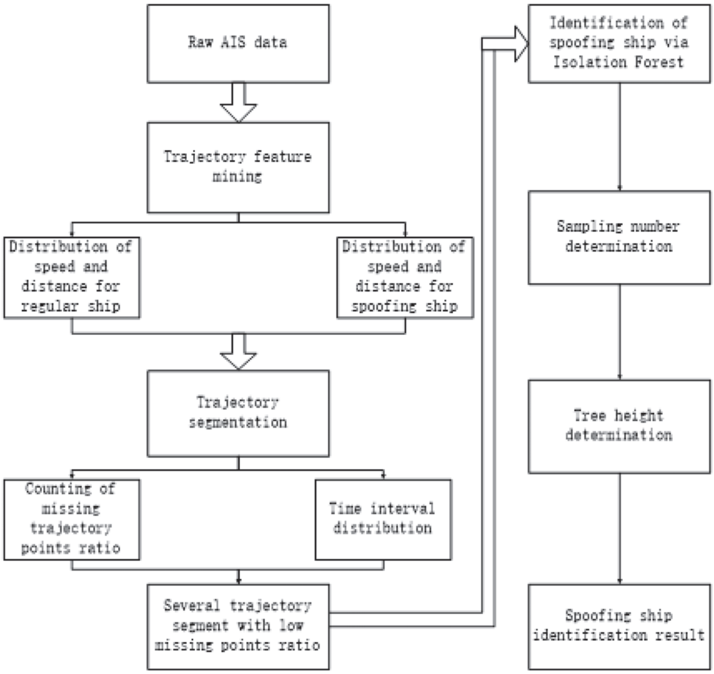


Figure 2. Overview for the proposed spoofing ship identification frame work.

In Equation (6), $\Delta t_{(th)}$ represents the threshold of the update time interval between adjacent trajectory points. $v_{avg(th(\min))}$ represents the threshold of minimum speed between adjacent trajectory points, that is, the normal navigation speed between adjacent trajectory points belonging to the same ship, abbreviated as $v_{th(\min)}$. d_{th} represents the threshold of minimum distance between adjacent trajectory points, corresponding to $v_{th(\min)}$.

In Equation (7), $v_{avg(th(\max))}$ represents the threshold of maximum speed between adjacent trajectory points, which is much larger than the normal navigation speed between the adjacent trajectory points of a regular ship, abbreviated as $v_{th(max)}$.

When the continuity of ship trajectories is good, that is, the time interval between trajectory points is short, there is a clear distinction between the distance and average sailing speed between regular ship trajectory points compared to the regular ships mixed with spoofing ships. However, due to the ship trajectory being missing, the time interval between ship trajectory points becomes longer, and the average sailing speed between them will be confused, making it difficult to identify spoofing ships. Therefore, trajectory segmentation is very necessary as it can convert a poorly continuous ship trajectory into several well continuous trajectory segments, which helps to identify the trajectory points of spoofing ships.

3.2. Trajectory Segmentation

For missing ship trajectory, re-emerged regular ship trajectory points may be identified as the trajectory points of a spoofing ship because the distance exceeds the corresponding threshold. At the same time, the re-emerged trajectory points of a spoofing ship may be misjudged as regular ship trajectory points due to the speed between adjacent trajectory points being within a corresponding speed threshold, as shown in Equation (8).

According to Section 3.1 of the paper, with the increase in the time interval between adjacent trajectory points, the average sailing speed between regular ship trajectory points remains unchanged, but the distance between trajectory points will gradually increase. However, the distance between adjacent trajectory points for the regular ship mixed with spoofing ship remains unchanged, but the average sailing speed between adjacent trajectory points will gradually decrease. Therefore, through trajectory feature mining in Section 3.1, it can be observed that when the time interval between adjacent trajectory points increases to a certain value, the distance between regular ship trajectory points is close to that between regular ship trajectory points mixed with spoofing ships, or the average sailing speed between regular ship trajectory points mixed with spoofing ships is close to that between regular ship trajectory points. Consequently, this time interval can be used as the threshold for trajectory segmentation.

Moreover, the time interval threshold for trajectory segmentation varies due to the distance between the trajectory points of the spoofing ship and the regular ship trajectory points. For spoofing ship trajectory points that are close to regular ship trajectory points, or overlapped with regular ship trajectory, when the time interval between adjacent trajectory points is small, trajectory features for this class of a regular ship mixed with a spoofing ship would be similar to regular ships. Therefore, it is necessary to set a small time interval threshold for trajectory segmentation. For spoofing ship trajectory points that are far away from regular ship trajectory points, these two ship trajectories will not overlap. Only when the time interval between adjacent trajectory points is large will the regular ship trajectory characteristics mixed with a spoofing ship be similar to regular ships. Therefore, a larger time interval threshold can be set for trajectory segmentation.

In order to avoid error identification for missing ship trajectory points, ship trajectory could be segmented according to the threshold of the update time interval, as shown in Equations (9)–(12).

$$\Delta t \geq \Delta t_{(th)}, d_i^{k,k+1} \geq d_{th} \wedge {v_{avg}}_i^{k,k+1} < v_{avg(th(\min))} \tag{8}$$

$$S_i = \left\{TR_i^1, TR_i^2, \ldots, TR_i^j, TR_i^k, \ldots, TR_i^m\right\} \tag{9}$$

$$TR_i^j = \left\{s_i^{j+1}, s_i^{j+2}, \ldots, s_i^k) \middle| t_{k+1} - t_k > \Delta t_{th}\right\} \tag{10}$$

$$TR_i^k = \left\{s_i^{k+1}, s_i^{k+2}, \ldots, s_i^{k+m} \middle| t_{k+(m+1)} - t_{k+m} > \Delta t_{th}\right\} \tag{11}$$

$$flag(s_i^{k+1}) = \begin{cases} NN\_P, \Delta t^1 \geq \Delta t_{(th)} \wedge flag(s_i^k) = N\_P \\ NN\_P, \Delta t^2 \geq \Delta t_{(th)} \wedge flag(s_i^k) = S\_P \end{cases} \tag{12}$$

In Equations (9)–(11), TR_i^j and TR_i^k represent the trajectory segmented by the corresponding time interval threshold. In Equation (12), s_i^k represents the trajectory point of ship i at time t_k, and s_i^{k+1} represents the trajectory point of ship i at time $t_{(k+1)}$. $flag(s_i^k)$ represents the class of trajectory point s_i^k, and $flag(s_i^{k+1})$ represents the class of trajectory point s_i^{k+1}. Δt^1 represents the time difference between the current trajectory point and latest time trajectory point in N_P. Δt^2 represents the time difference between the current trajectory point and latest time trajectory point in S_P. NN_P represents a new class of points derived from the missing ship trajectory, namely the trajectory points of a new regular ship.

3.3. Identification of Spoofing Ship via Isolation Forest

The trajectory segment characteristics between a regular ship and a regular ship mixed with a spoofing ship have an obvious difference, and the proportion of the abnormal trajectory segment is small. In Figure 1, there are only 10 percent of spoofing ship trajectory points included in the overall AIS data sample. Therefore, the isolation forest is applicable for spoofing ship identification for the AIS data sample of the paper. Ship trajectory is divided into a set of trajectory segments composed of adjacent trajectory points. In Equation (13), TR_i^j is a set of trajectory segments composed of the adjacent trajectory points of ship i, defined as follows:

$$TR_i^j = \left\{tr_i^{j+1,j+2}, tr_i^{j+2,j+3}, \ldots, tr_i^{k-1,k} \middle| 0 \leq j \leq k, 1 \leq k \leq n\right\} \tag{13}$$

$$tr_i^{k-1,k} = (d_i^{k-1,k}, v_{avg}{}_i^{k-1,k}) \tag{14}$$

In Equation (13), n is the number of trajectory points of ship i.

A sample with the number of m is selected from the mother sample. A dimension of the sample is randomly selected, and a segmentation value is also selected. The first isolated tree is constructed according to the binary tree method. Samples less than the segmentation value are divided into the left cross tree, and samples greater than the segmentation value are divided into the right cross tree. Then, the first isolated tree would be constructed until the number of segmentations reaches h. The average path length of isolated trees is calculated as in Equation (15):

$$c(m) = 2 * (\ln(m-1) + 0.5772156649) - \frac{2(m-1)}{m} \tag{15}$$

In Equation (15), m represents the number of sub-sampling points.

When the path length of sample tr in j isolated tree is set as h_j^{tr}, the expected path length of sample tr in all isolated trees is calculated as in Equation (16):

$$E(h_j^{tr}) = \frac{\sum\limits_{j=1}^{p} h_j^{tr}}{p} \tag{16}$$

In Equation (16), p represents the number of isolated trees, and h represents the restricted height of isolated trees.

The abnormal score $s_{(tr)}$ of sample tr is the basis for judging whether the sample is an outlier. The calculation method is as follows:

$$s_{(tr)} = 2^{\frac{-E(h_j^{tr})}{c(m)}} \tag{17}$$

The threshold of the abnormal score of sample tr is set to $s_{tr(th)}$, and the discriminant method of the outliers is as follows:

$$label_{(tr)} = \begin{cases} 1, s_{(tr)} > s_{(tr)th} \\ 0, s_{(tr)} \leq s_{(tr)th} \end{cases} \tag{18}$$

In Equation (18), $label_{(tr)}$ is the category labeling of ship trajectory segment tr, and 1 means that the ship trajectory segment belongs to an outlier, that is, that the trajectory segment is composed of two types of ship trajectory points (that is the regular ship trajectory mixed with that of the spoofing ship). 0 indicates that the ship trajectory segment is normal, that is, the trajectory segment is composed of only the regular ship trajectory point. Combined with trajectory segmentation, the outliers of ship trajectory points are identified as follows:

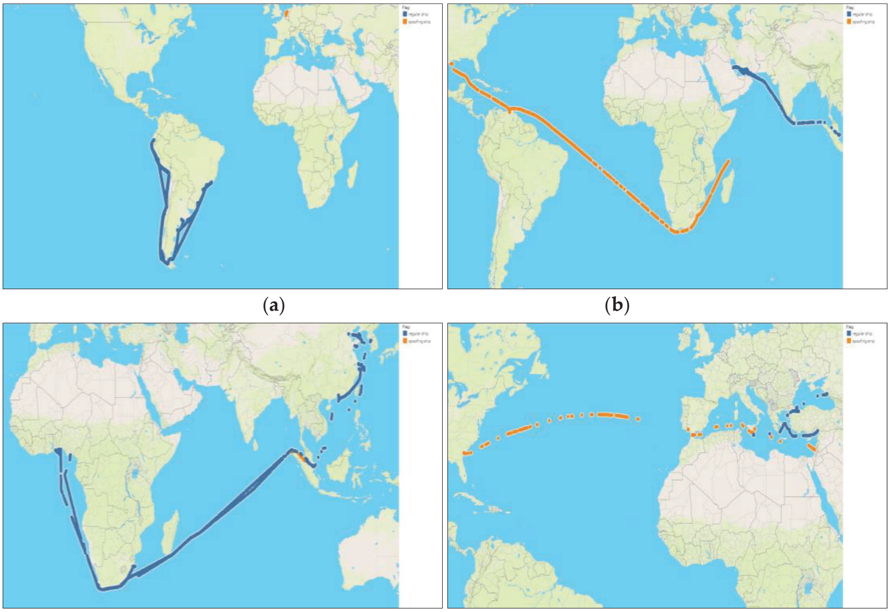
$$flag(s_i^{k+1}) = \begin{cases} S\_P, label_{(tr)} = 1 \wedge flag(s_i^k) = N\_P \\ N\_P, label_{(tr)} = 0 \wedge flag(s_i^k) = N\_P \\ N\_P, label_{(tr)} = 1 \wedge flag(s_i^k) = S\_P \\ S\_P, label_{(tr)} = 0 \wedge flag(s_i^k) = S\_P \end{cases} \tag{19}$$

In Equation (19), if the trajectory segment is labeled as 0, indicating that distance and speed for the trajectory segment tend to be normal, then the trajectory points at the adjacent time have the same category and belong to one ship. Conversely, it shows that the distance and speed for the trajectory segment tend to be abnormal, meaning the trajectory points at adjacent times belong to two different ships.

4. Experiments

According to the trajectory characteristics of regular ships mixed with spoofing ships, these characteristics can be divided into the following four categories: ① the regular ship trajectory is continuous, while the spoofing ship trajectory points are concentrated; both ship trajectories are not overlapped (class I spoofing ship); ② both trajectories are continuous and not overlapped (class II spoofing ship); ③ the regular ship trajectory is continuous, while the spoofing ship trajectory is concentrated, and both ship trajectories are overlapped (class III spoofing ship); and, finally, ④ both trajectories are continuous and overlapped (class IV spoofing ship), as shown in Figure 3. In Figure 3, there appears to be a phenomenon of trajectory point jumping due to the trajectory points of a spoofing ship (labeled with orange dots) mixed in a regular ship trajectory (labeled with blue dots). In Figure 3a,c, there are only some scattered points of the spoofing ship, and they are concentrated in certain areas and taken as some isolated outliers. In Figure 3b,d, there are continuous trajectory points of the spoofing ship, which is obviously the trajectory of another ship that occupies the same MMSI as a regular ship, namely a spoofing ship.

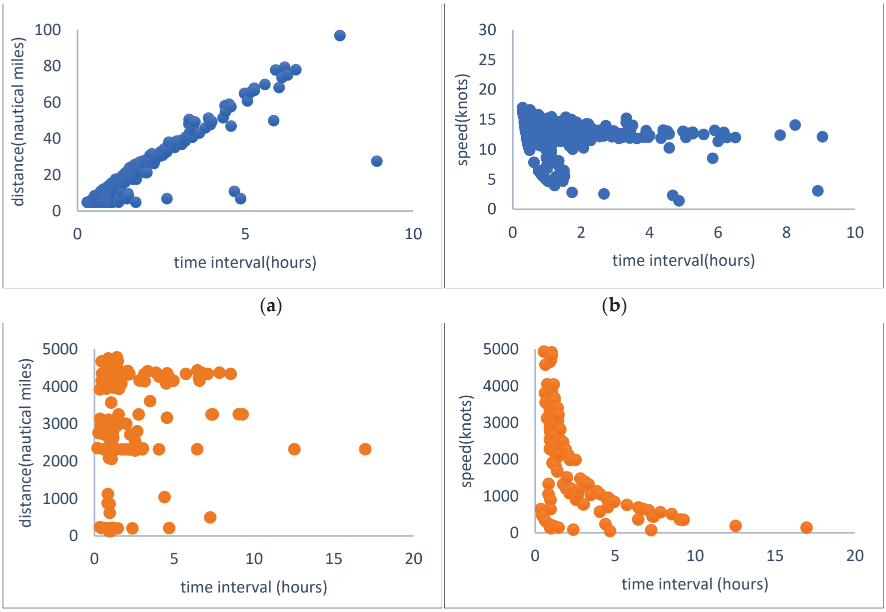
When navigating at sea, the distance between adjacent trajectory points is almost linearly related to the update time interval of trajectory points, as shown in Figure 4a, while the speed between the adjacent trajectory points remains almost unchanged, as shown in Figure 4b. If there are two classes of ships with the same MMSI at sea, that means that the trajectory of a regular ship has been mixed with a spoofing ship. The variation in the trend of average sailing speed and distance between adjacent trajectory points is no longer consistent with Figure 4a,b, as shown in Figure 4c,d. The distance between adjacent trajectory points is large and almost does not change with time, while the speed between trajectory points decreases exponentially with time.



(**a**) (**b**)

(**c**) (**d**)

Figure 3. Distribution of regular ship trajectory points mixed with spoofing ship: (**a**) class I; (**b**) class II; (**c**) class III; (**d**) class IV.



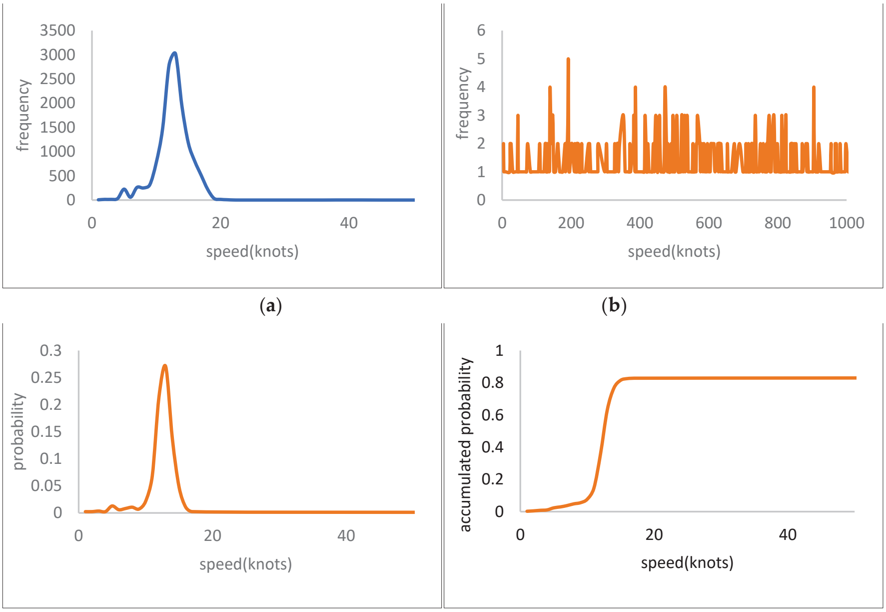
(**a**) (**b**)

(**c**) (**d**)

Figure 4. Distribution of distance and speed between adjacent trajectory points: (**a**) distance distribution of regular ship; (**b**) speed distribution of regular ship; (**c**) distance distribution of regular ship mixed with spoofing ships; (**d**) speed distribution of regular ship mixed with spoofing ships.

4.1. Distribution of Speed and Distance between Adjacent Trajectory Points

In order to set a reasonable threshold of average sailing speed between adjacent trajectory points, it is vital to understand the distribution of the average sailing speed. Through trajectory feature mining, it was found that the average sailing speed among regular ship trajectory points is normally distributed, and the expected value in Figure 5a is 12.5 knots. Average sailing speed between different trajectory points conforms to normal distribution, and expected value in Figure 5b is 2750 knots. Figure 5c shows the probability distribution diagram of average sailing speed among trajectory points, while Figure 5d shows the variation in the trend of the cumulative probability of average sailing speed between trajectory points. Among them, 82.59% of the average sailing speed between trajectory points is less than 16 knots, which can be used as the average sailing speed threshold for identifying spoofing ship trajectory points.

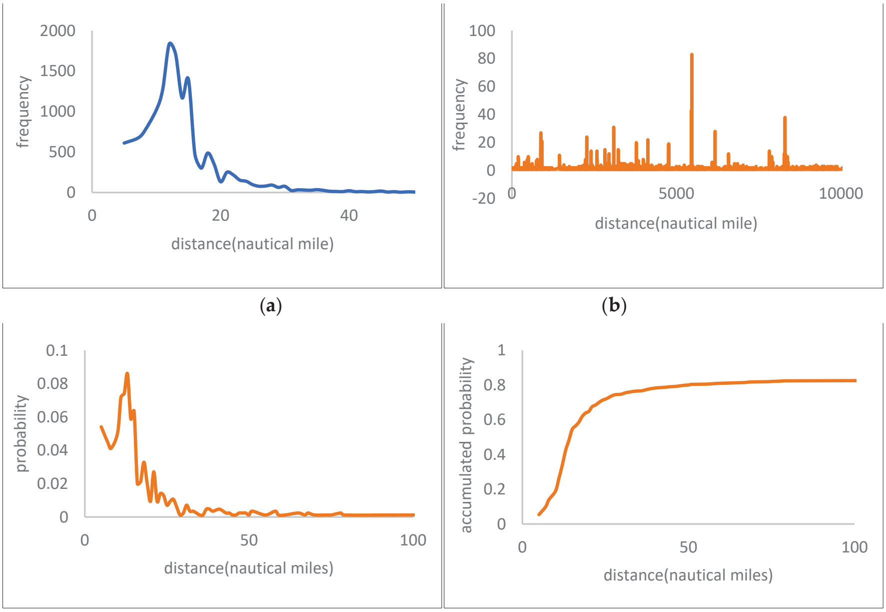


(**a**) (**b**)

(**c**) (**d**)

Figure 5. Speed distribution of trajectory segment: (**a**) speed of trajectory segment for regular ship; (**b**) speed of trajectory segment for regular ship mixed with spoofing ship; and (**c**,**d**) probability distribution of speed among trajectory segments for regular ship mixed with spoofing ship.

Through trajectory feature mining, it was found that the distance between regular ship trajectory points is normally distributed, with the expected value in Figure 6a being 20 nautical miles. The distance between trajectory points mixed with spoofing ships is normally distributed, and the expected value in Figure 6b is 800 nautical miles. Figure 6c shows the probability distribution of the distance between adjacent trajectory points, while Figure 6d shows the variation in the trend of the cumulative probability of the distance between adjacent trajectory points. Among them, 82.42% of the distance between adjacent trajectory points is less than 80 nautical miles, which can be used as the threshold of the distance between adjacent trajectory points for identifying spoofing ship trajectory points.



(**a**) (**b**)

(**c**) (**d**)

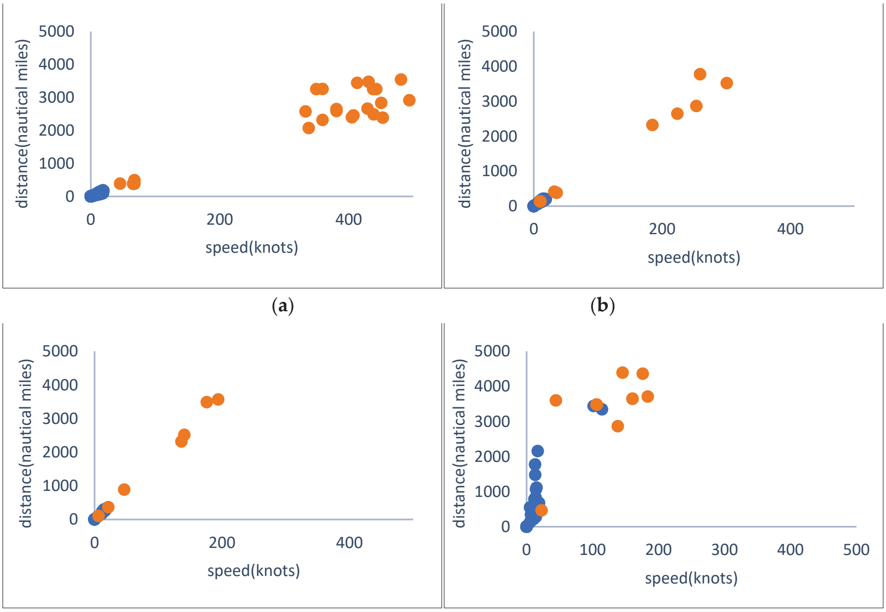
Figure 6. Distance distribution of trajectory segment: (**a**) trajectory segment distance among regular ship; (**b**) trajectory segment distance among regular ship mixed with spoofing ship; and (**c**,**d**) probability distribution of trajectory segment distance among regular ship mixed with spoofing ship.

With regard to time interval between adjacent trajectory points, it is found that time interval of regular trajectory points is normally distributed through statistical learning, and the expected value is 1.5 h. However, the time interval between some adjacent trajectory points is relatively large, but the ratio of these trajectory segments is relatively small. As the time interval between trajectory points increases, the proportion of the trajectory segment gradually decreases. The distribution pattern of the time intervals between adjacent trajectory points is shown in Table 1.

Table 1. Trajectory segment number distribution for various time intervals.

Time Interval (Hours)	Corresponding Number of Trajectory Segment	The Number of Overall Trajectory Segment	Trajectory Segment Ratio (Percent)
≤1	25,009	52,537	47.6
(1, 2)	24,025	52,537	45.73
(2, 3)	1686	52,537	3.2
(3, 4)	598	52,537	1.15
(4, 5)	337	52,537	0.64
(5, 10)	603	52,537	1.15
(10, 15)	142	52,537	0.27
(15, 20)	53	52,537	0.1
>20	84	52,537	0.16

Figure 7 shows the distribution pattern of the distance and average sailing speed between adjacent trajectory points. The blue dots represent the scatter plots of distance and average sailing speed between regular ship trajectory points, while the orange dots represent the ship trajectory mixed with spoofing ship trajectory points. In Figure 7a, when the time interval between adjacent trajectory points is within 5 h, the continuity of the ship's trajectory is good, and the blue and orange points have a good distinguishing ability. When the time interval between ship trajectory points exceeds 5 h, the blue and orange points will overlap, making it difficult to identify the trajectory points of the spoofing ship, as shown in Figure 7b. Moreover, the longer the time interval between adjacent ship trajectory points, the less easily the regular ship trajectory and the ship trajectory mixed with spoofing ships are identified, as shown in Figure 7c,d.



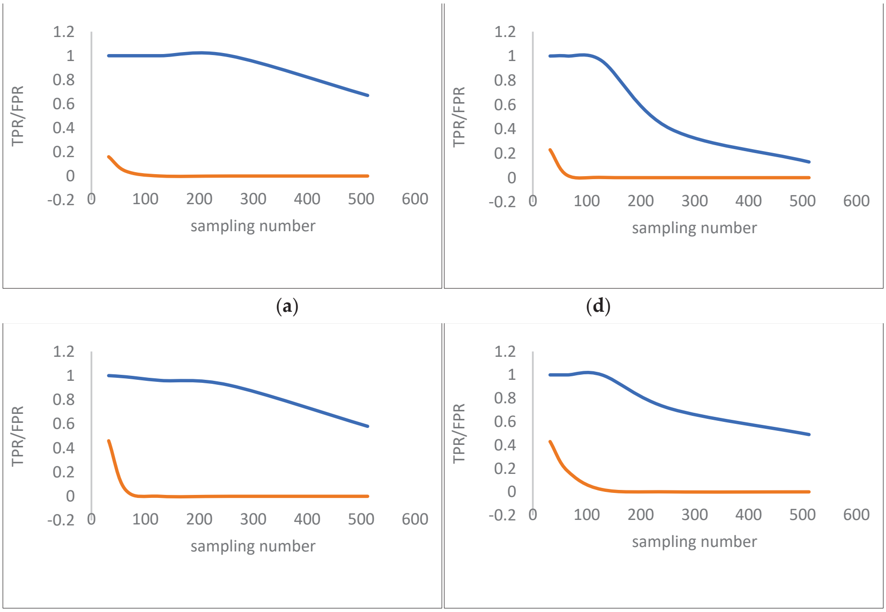
(**a**) (**b**)

(**c**) (**d**)

Figure 7. The distance and average sailing speed distribution of trajectory segment corresponding to various time interval between adjacent trajectory points: (**a**) time interval within 5 h; (**b**) time interval beyond 5 h and within 10 h; (**c**) time interval beyond 10 h and within 15 h; and (**d**) time interval beyond 15 h and within 20 h.

4.2. Identification of Spoofing Ships Based on Trajectory Segmentation and Isolation Forest

For the trajectory of classes I, II, III, and IV of the spoofing ship, the accuracy of identifying outliers of the ship trajectory shows the following trend with the number of sub-sampling points and height of the isolated tree. Figure 8 reflects a correlation between the accuracy of identifying outliers and the number of sub-sampling points. For the trajectory of classes I, II, III, and IV of spoofing ships, the accuracy of identifying outliers of trajectory points gradually decreases, and the error rate within the identification of regular ship trajectory points gradually decreases, with an increase in the number of sub-sampling points. The number of sub-sampling points is one of the important parameters of the isolation forest, which would affect the true positive rate (outliers correctly identified) and false positive rate (trajectory points of regular ship wrongly identified). Generally speaking, the higher the true positive rate is, and the lower the false positive rate is, the more reasonable the number of sub-sampling points is. In Figure 8, when the number of sub-sampling points is about 100, the true positive rate is higher than 0.95, and the false positive rate is lower than 0.05, so the number of sub-sampling points is set as 128.



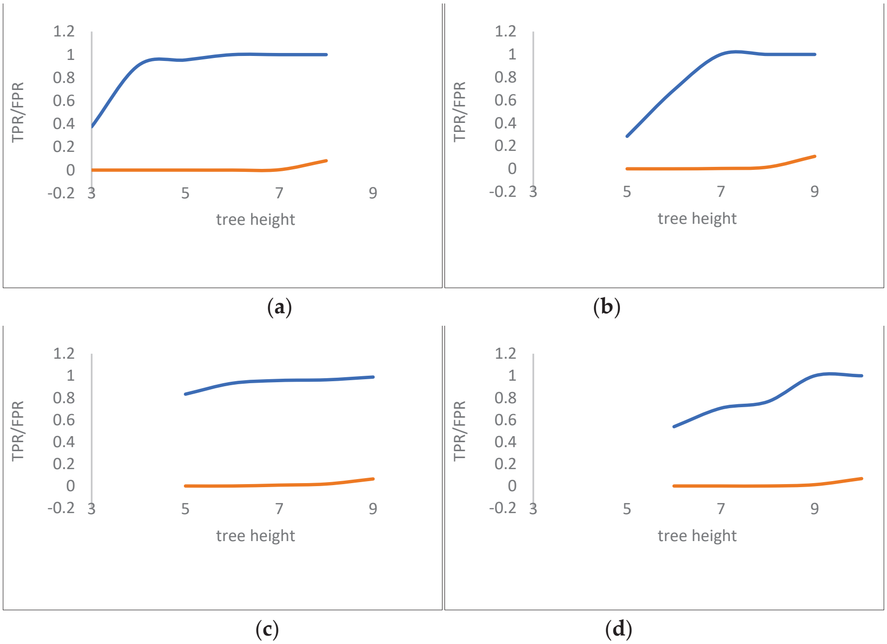
(**a**) (**d**)

(**c**) (**d**)

Figure 8. The correlation between True Positive Rate, False Positive Rate, and the number of sub-sampling points for spoofing ship trajectory identification: (**a**) class I; (**b**) class II; (**c**) class III; (**d**) class IV.

Figure 9 reflects a correlation between the accuracy of identifying outliers and the height of isolated trees. For the trajectory of classes I, II, III, and IV spoofing ships, the accuracy of identifying outliers of trajectory points gradually increases, and the identification error rate of regular ship trajectory points also gradually increases, with an increase in the height of isolated trees. The height of isolated trees is one of the important parameters of isolated forest. The higher the isolated tree height is, the more effectively true positive samples are identified (outliers correctly identified). However, as isolated tree height increases, some false positive samples may also be mistaken for positive samples (regular ship trajectory point wrongly identified). Generally speaking, the higher the true positive rate is, and the lower false positive rate is, the more reasonable the height of the isolated trees is. In Figure 8, when the height of isolated trees is about eight, the true positive rate is higher than 0.95, and the false positive rate is lower than 0.05, so the height of isolated tree is set as eight.

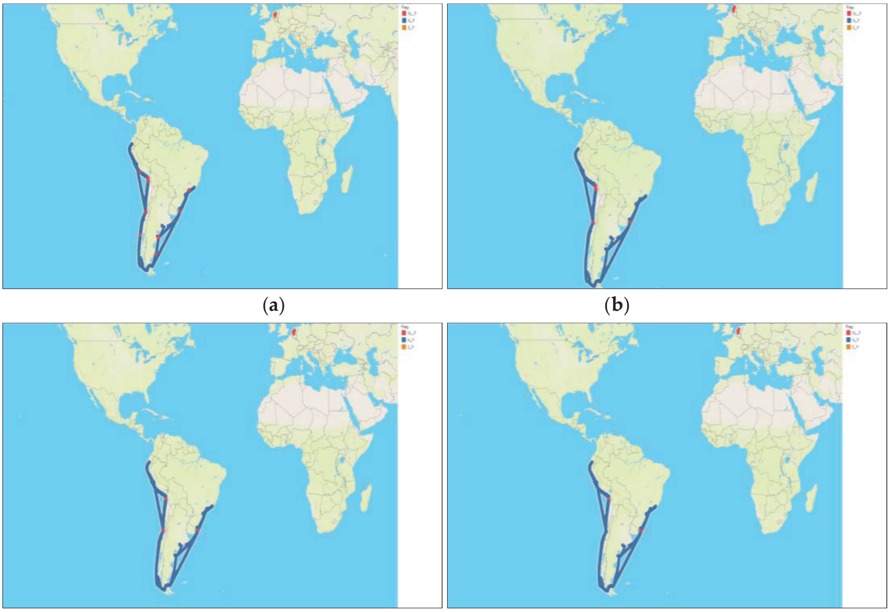
In Figures 10–13, the N_P and S_P of the ship trajectory are labeled with blue and orange dots, and the NL_P of the ship trajectory are labeled with red dots. For outliers of trajectory points that cannot be identified by statistical learning, the number of unidentified trajectory points shows the following trend after adopting the isolated forest algorithm for recognition. The number of unidentified trajectory points for class I spoofing ships gradually decreases with an increase in the trajectory segmentation time; the specific values are listed as 32, 19, 12, and 9, respectively, as shown in Figure 10. The number of unidentified trajectory points for class II spoofing ships gradually decreases with an increase in the trajectory segmentation time, with specific values of 5, 3, 1, and 1, as shown in Figure 11. The number of unrecognized trajectory points for class III spoofing ships gradually decreased with an increase in the trajectory segmentation time; the specific values are listed as 20, 11, 8, and 4, as shown in Figure 12. The number of unrecognized trajectory points has always been 0 for class IV spoofing ships, as shown in Figure 13.



(**a**) (**b**)

(**c**) (**d**)

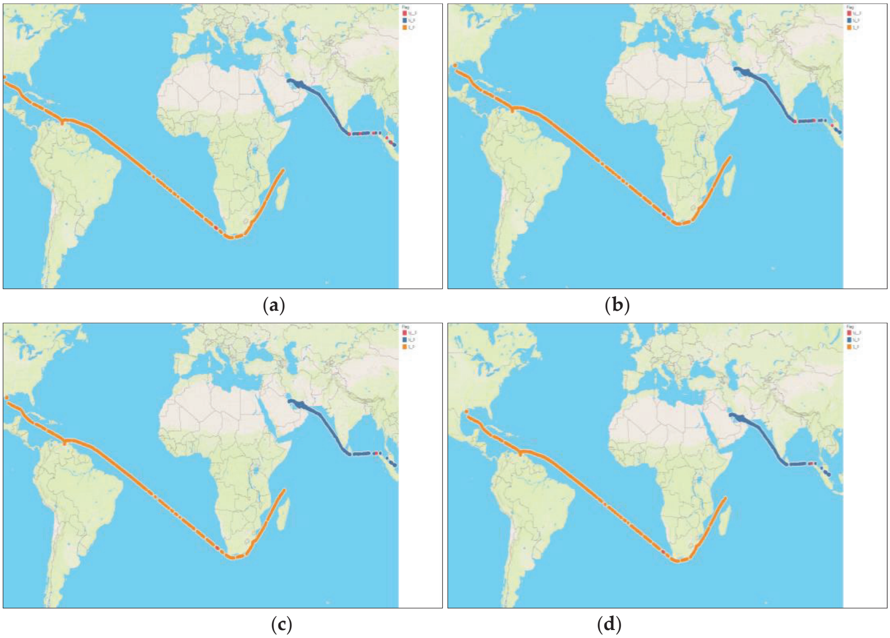
Figure 9. The correlation between true positive rate, false positive rate, and tree height for spoofing ship trajectory identification: (**a**) class I; (**b**) class II; (**c**) class III; and (**d**) class IV.



(**a**) (**b**)

(**c**) (**d**)

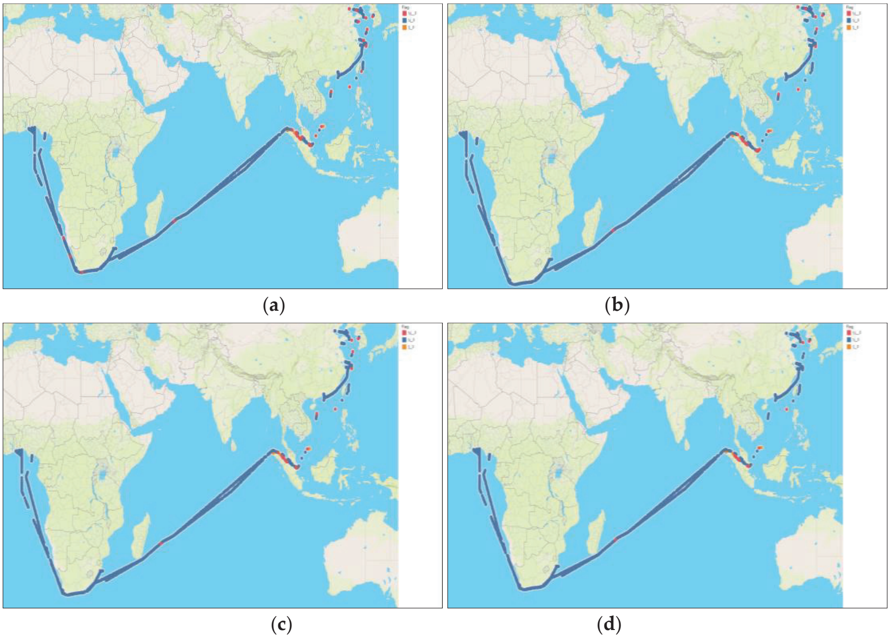
Figure 10. Identification of outliers for class I spoofing ships via isolation forest: (**a**) trajectory segmented by 5 h, (**b**) trajectory segmented by 10 h, (**c**) trajectory segmented by 15 h, and (**d**) trajectory segmented by 20 h.



(a) (b)

(c) (d)

Figure 11. Identification of outliers for class II spoofing ships via isolation forest: (**a**) trajectory segmented by 5 h, (**b**) trajectory segmented by 10 h, (**c**) trajectory segmented by 15 h, and (**d**) trajectory segmented by 20 h.



(a) (b)

(c) (d)

Figure 12. Identification of outliers for class III spoofing ships via isolation forest: (**a**) trajectory segmented by 5 h, (**b**) trajectory segmented by 10 h, (**c**) trajectory segmented by 15 h, and (**d**) trajectory segmented by 20 h.

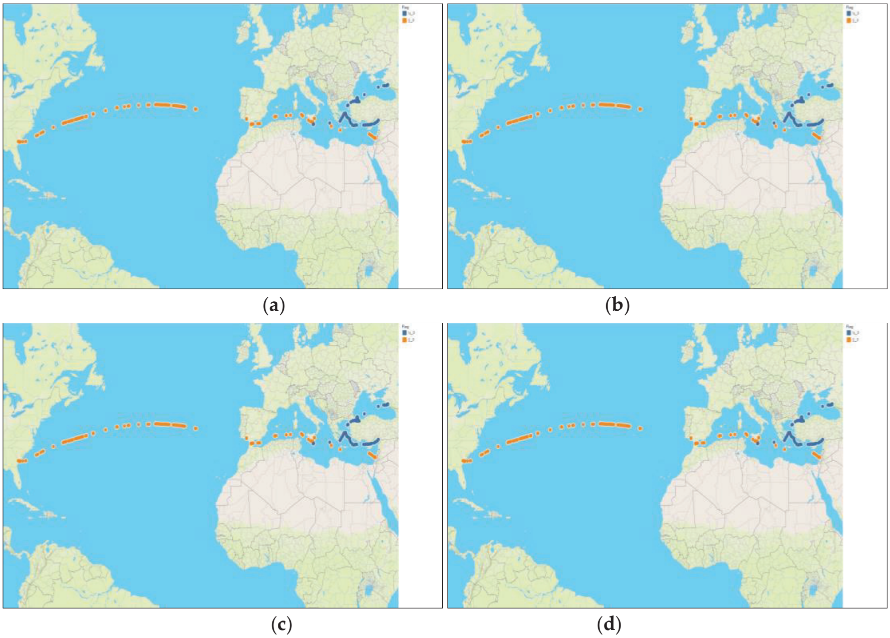

(**a**) (**b**)
(**c**) (**d**)

Figure 13. Identification of outliers for class IV spoofing ships via isolation forest: (**a**) trajectory segmented by 5 h, (**b**) trajectory segmented by 10 h, (**c**) trajectory segmented by 15 h, and (**d**) trajectory segmented by 20 h.

As can be seen in Figure 14, the isolation forest was adopted to continue identifying the outliers of trajectory points that could not be identified through statistical learning. The time interval for trajectory segmentation increased from 5 h to 20 h, and the accuracy of identifying outliers for class I spoofing ships improved first and then decreased, with specific values of 95.7%, 98.3%, 94.5%, and 90.4%, gradually improved specific values of 76.4%, 86.9%, 94.4%, and 94.4% for class II spoofing ships, and gradually improved specific values of 88.2%, 91.9%, 93.4%, and 98.1% for class III spoofing ships. However, the accuracy of identifying outliers for class IV spoofing ships remains high and unchanged, mainly due to the short time interval among trajectory points, with constant values of 100%.

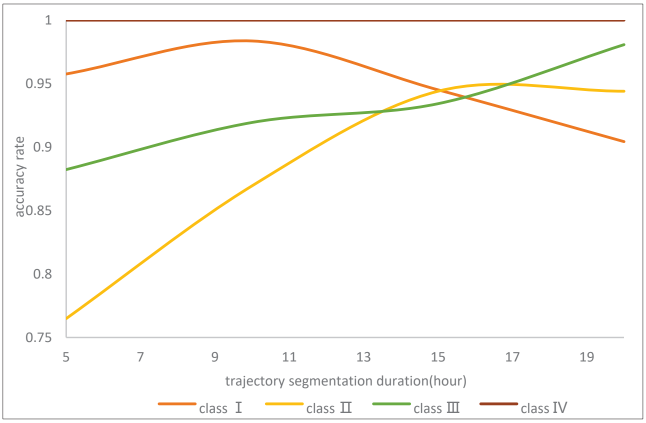


Figure 14. Comparison of the accuracy of identifying outliers for four classes of spoofing ships.

In Figure 14, among the four classes of regular ship trajectories mixed with spoofing ships, the identification accuracy of class IV and class I is relatively high, reaching 100% and 95.7%, respectively, while the identification accuracy of class II and class III is relatively low, at only 76.4% and 88.2%. The segmentation threshold for these four types of ship trajectories is all 5 h, and the reason for the inconsistent identification accuracy is that the ratio of missing trajectories varies. Taking the sampling time of one hour as an example, the complete trajectory points of a ship in one day should be 24. Therefore, the ratio of missing points and jumping points for four classes of ships can be calculated based on the statistical time. The correlation between the ratio of jumping and missing points and identification accuracy for four classes of spoofing ships are shown in Table 2.

Table 2. Correlation between the ratio of jumping and missing points and identification accuracy for four classes of spoofing ships.

Class of Spoofing Ship	Statistical Time (Months)	The Number of Jumping Trajectory Points	The Number of Actual Trajectory Points	Jumping Points Ratio (Percent)	The Number of Complete Trajectory Points	Missing Points Ratio (Percent)	Identification Accuracy (Trajectory Segmented by 5 h)
I	4	18	2197	0.82	2880	23.7	95.7
II	2	528	1085	48.66	1440	24.6	76.4
III	3	58	1629	3.56	2160	24.5	88.2
IV	1	218	690	31.59	720	4.1	100

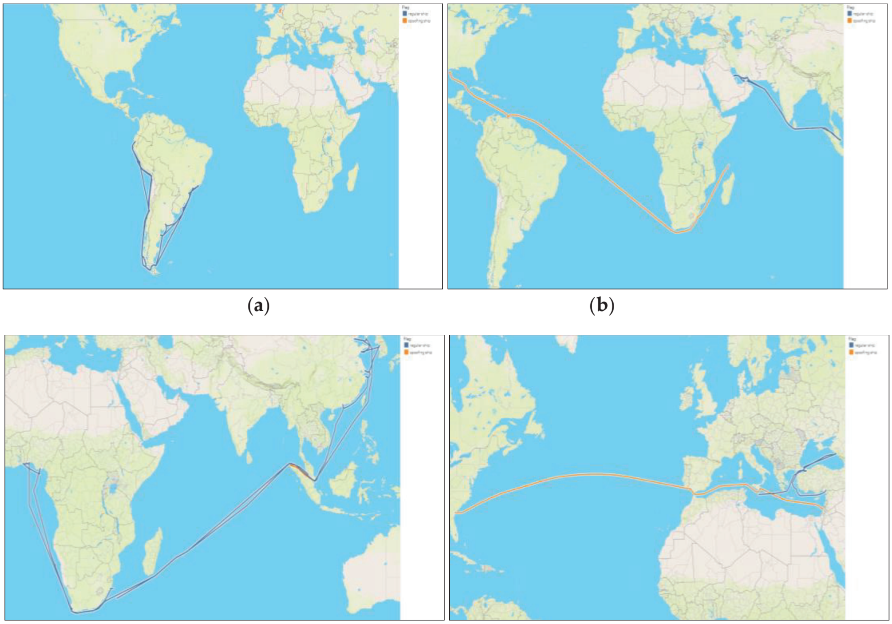
In Table 2, the ratio of missing trajectories points for class IV spoofing ships is only 4.1%, while the ratio for the other three classes of spoofing ships is close to 25%. Therefore, the accuracy of identifying the trajectories of class IV spoofing ships is much higher than that of the other three classes of spoofing ships. For the other three classes of spoofing ships, class I spoofing ships have the smallest ratio of jumping points, class III spoofing ships have a slightly larger ratio of jumping points, and class II spoofing ships have the largest ratio of jumping points. The accuracy of identifying these three classes of spoofing ships is also consistent with the changes in the ratio of jumping points.

In Figure 14, the identification accuracy of the first three classes of spoofing ships did not reach 100%, and the accuracy did not improve with the increase in the time interval threshold, such as class I and II spoofing ships. The applicability of the three parameters and their related thresholds in this paper varies for each class of spoofing ship. For class I and IV spoofing ships, the small time interval threshold brought the best identification effect due to the low ratio of missing trajectory points. For class II and III spoofing ships, the identification effect is the best when the time interval threshold between trajectory points is large due to the high ratio of missing trajectory points. In addition, in order to avoid mistakenly identifying the trajectory point of a spoofing ship as a regular ship, the distance threshold between trajectory points and the average sailing speed threshold are set to be small, which results in some regular ship trajectory points not being recognized and labeled as confusion points. In Table 3, as the distance threshold and average sailing speed threshold change, there are variations in the trend of the identification accuracy of the four classes of spoofing ships.

Table 3. Correlation between various values of threshold and identification accuracy (True Positive Rate, False Positive Rate) for four classes of spoofing ships' trajectory segmented by 5 h.

Minimum Speed Threshold (Knots)	Maximum Speed Threshold (Knots)	Minimum Distance Threshold (Nautical Miles)	TPR and FPR for Class I (Percent)	TPR and FPR for Class II (Percent)	TPR and FPR for Class III (Percent)	TPR and FPR for Class IV (Percent)
20	100	100	91.98, 8.02	76.49, 23.51	88.03, 12.54	100, 0
30	100	150	92, 8.57	76.47, 34.45	88.59, 11.93	100, 0
50	100	250	92.28, 8.16	76.49, 49	89.01, 11.46	39.28, 60.72
80	100	400	91.13, 9.2	76.48, 58.28	89.35, 11.09	81.08, 72.97

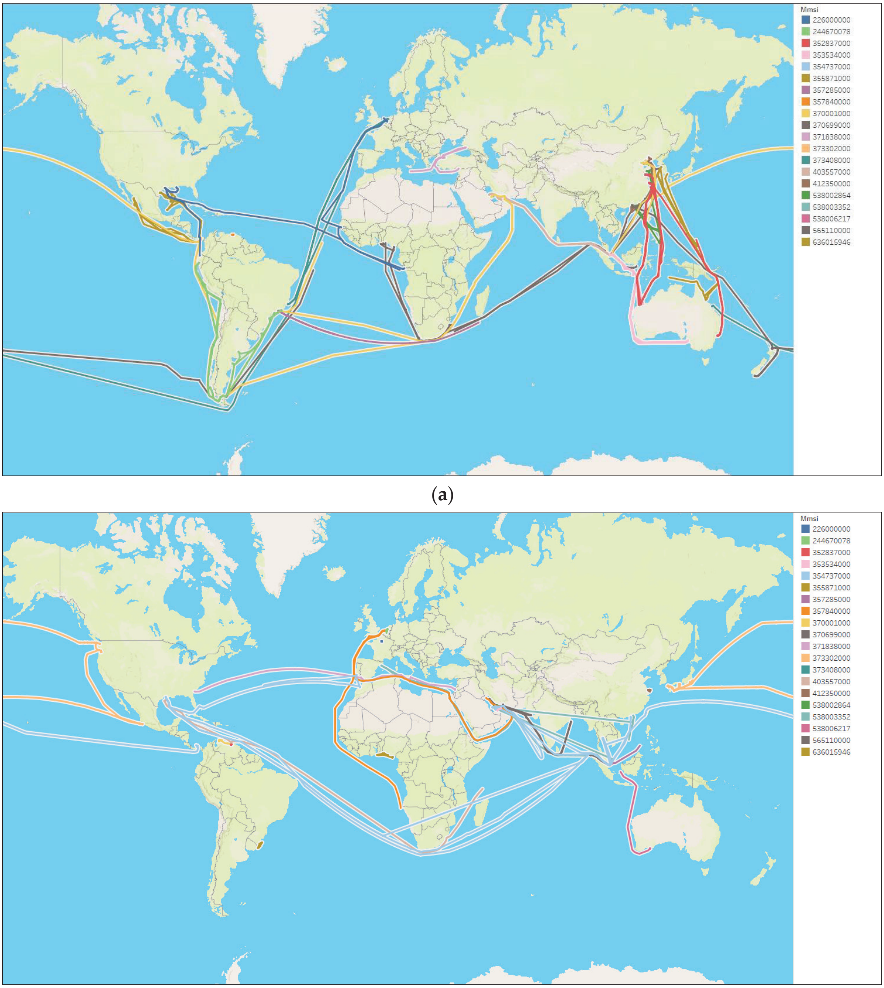
For the spoofing ship of class I and III, the ship trajectory can be accurately displayed without scatter jumping points, as shown in Figure 15a,c. For the spoofing ship of class II and IV, the trajectories of two ships can be accurately displayed, as shown in Figure 15b,d. The blue lines indicate the trajectories of regular ships, and the orange lines indicate the trajectories of spoofing ships. Through the identification of the spoofing ship, the outliers of the trajectory of a container ship in Figure 1 have been almost removed and classified via trajectory segmentation and isolated forest, and the trajectory of a regular ship and spoofing ship are exhibited, respectively, in Figure 16a,b, which can reflect a ship's motion pattern accurately.



(**a**) (**b**)

(**c**) (**d**)

Figure 15. Trajectory of four classes of regular ship trajectory without outlier: (**a**) trajectory of class I exhibited with regular ship (marked by blue line) and spoofing ship (marked by orange line) separately; (**b**) trajectory of class II; (**c**) trajectory of class III; and (**d**) trajectory of class IV.



(**a**)

(**b**)

Figure 16. Trajectory of container ships exhibited in Figure 1 without confusion points between regular ship and spoofing ship: (**a**) regular ship trajectory; (**b**) spoofing ship trajectory.

The framework was implemented on Windows 10 OS with 8 GB RAM and 2.8 GHz CPU. We employed Matlab (2016 version) to perform trajectory segmentation and the spoofing ship identification procedure on the ship trajectory data. With regard to the runtime test, the paper dealt with 52,538 trajectory points owned by 20 container ships via isolation forest, and runtime was 15.77 s.

5. Discussion

Each ship sailing at sea has a unique MMSI, which can be used to extract the complete trajectory of any ship from AIS data. However, some ships have obtained unauthorized MMSI through illegal approaches, which are duplicated with MMSI owned by existing ships. This leads to a trajectory being extracted from one MMSI that actually belong to multiple ships, which poses serious challenges to ship motion pattern identification based on AIS data.

The paper adopts trajectory feature mining to clarify the distribution patterns of distance and average sailing speed between adjacent trajectory points (see Figures 5 and 6 for details). To address the impact of the missing trajectories of ships, updating the time interval threshold is set to segment ship trajectories. For a segmented ship trajectory, outliers can be identified via isolation trees established based on distance and speed among trajectory segments. By observing the trend of changes in the true positive and false positive rates of trajectory point identification, two important parameters of isolation forest are determined, namely the sampling number and isolated tree height. After adopting isolation forest, the identification accuracy of spoofing ships was improved, as shown in Figure 14. The number of unidentified trajectory points for class I, II, and III spoofing ships has gradually decreased with an increase in the trajectory segmentation time, as shown in Figures 10–12. However, the identification accuracy of class IV spoofing ships was not improved, and the reason may be that class IV spoofing ships have good continuity, as shown in Figure 13.

Due to the identification of spoofing ships by isolation forest, the number of trajectories in Figure 16 is almost half more than that in Figure 1. The trajectory of these spoofing ships would just be removed if using the approaches described in the literature, and more information of the AIS data would be lost. The pre-processing of AIS data should aim to retain more information, which is more helpful to the situational awareness of ship motion for the MSA based on AIS data. For the trajectory of a spoofing ship away from a regular ship, it can be identified according to the serial number of the AIS base station included in the AIS data. A part of the ship trajectory in this paper meets such characteristics and would be identified more efficiently by the serial number of the AIS base station. However, many AIS data do not contain information such as the serial number of the AIS base station, so the spoofing ship identification method supplied by this paper is still necessary.

However, some trajectory points are still falsely identified as a regular ship rather than a spoofing ship. The main reason for this is that trajectory points correlation shows a poor performance among some trajectory segments. Future research should focus on the clustering of trajectory points among various trajectory segments, so as to identify the NN_P listed in Equation (12). In addition, this study only set a constant threshold and established isolated trees via trajectory feature mining from the history trajectory of typical cargo ships so as to identify outliers of ship trajectories. Future research should focus on setting an adaptive threshold for speed and distance between adjacent trajectory points based on differences in the maneuvering performance for various types of ships and the speed difference of ships in different navigation stages.

6. Conclusions

A long time interval between adjacent trajectory points results in severe missing trajectories of a ship, and the identification of spoofing ships are not ideal. In order to eliminate the impact of missing trajectory points on the accuracy of identifying spoofing ships, the trajectory is segmented by the time interval threshold. After trajectory segmentation, the trajectory points of each trajectory segment maintain good continuity, that is, the time interval between adjacent trajectory points for each trajectory segment is relatively short. Combined with trajectory segmentation, the isolation forest is efficient at distinguishing between regular ship trajectory points and spoofing ship trajectory points. Consequently, outliers of ship trajectories were almost removed or classified correctly in this work, and the labeled ship trajectory points can reflect a ship’s motion pattern accurately.

Author Contributions: Conceptualization, H.Z.; methodology, C.Y.; software, Q.M.; validation, H.Z. and Q.H.; formal analysis, H.Z.; investigation, H.Z.; resources, C.Y. and P.W.; data curation, H.Z.; writing—original draft preparation, H.Z.; writing—review and editing, Q.H.; visualization, C.Y.; supervision, Q.H.; project administration, Q.H.; funding acquisition, Q.H. and K.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Project of Ministry of Transport, grant number 2020MS6162. Project of Zhoushan science and Technology Bureau, grant number 2021C21010; National innovation and entrepreneurship training program for Zhejiang Ocean University, grant number 202210340043.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Androjna, A.; Perkovič, M.; Pavic, I.; Mišković, J. AIS data vulnerability indicated by a spoofing case-study. *Appl. Sci.* **2021**, *11*, 5015. [CrossRef]
2. Felski, A.; Jaskólski, K.; Banyś, P. Comprehensive assessment of automatic identification system (AIS) data application to anti-collision manoeuvring. *J. Navig.* **2015**, *68*, 697–717. [CrossRef]
3. Wawruch, R. Ability to test shipboard automatic identification system instability and inaccuracy on simulation devices. *Zesz. Nauk. Akad. Morskiej W Szczecinie* **2017**, *52*, 128–134.
4. Jaskólski, K. Two-dimensional coordinate estimation for missing automatic identification system (AIS) signals based on the discrete Kalman filter algorithm and universal transverse mercator (UTM) projection. *Zesz. Nauk. Akad. Morskiej W Szczecinie* **2017**, *52*, 82–89.
5. Peters, D.J.; Hammond, T.R. Interpolation between AIS reports: Probabilistic inferences over vessel path space. *J. Navig.* **2011**, *64*, 595–607. [CrossRef]
6. Shelmerdine, R.L. Teasing out the detail: How our understanding of marine AIS data can better inform industries, developments, and planning. *Mar. Policy* **2015**, *54*, 17–25. [CrossRef]
7. Sang, L.Z.; Wall, A.; Mao, Z.; Yan, X.P.; Wang, J. A novel method for restoring the trajectory of the inland waterway ship by using AIS data. *Ocean. Eng.* **2015**, *110*, 183–194. [CrossRef]
8. Pallotta, G.; Vespe, M.; Bryan, K. Vessel pattern knowledge discovery from AIS data: A framework for anomaly detection and route prediction. *Entropy* **2013**, *15*, 2218–2245. [CrossRef]
9. Mazzarella, F.; Vespe, M.; Damalas, D. Discovering vessel activities at sea using AIS data: Mapping of fishing footprints. In Proceedings of the 17th International Conference on Information Fusion (FUSION), Salamanca, Spain, 7–10 July 2014.
10. Wu, L.; Xu, Y.; Wang, Q. Mapping global shipping density from AIS data. *J. Navig.* **2017**, *70*, 67–81. [CrossRef]
11. Greidanus, H.; Alvarez, M.; Eriksen, T. Completeness and accuracy of a wide-area maritime situational picture based on automatic ship reporting systems. *J. Navig.* **2016**, *69*, 156–168. [CrossRef]
12. Wei, G.; Yang, C. Detection of AIS Data Error. *Navig. China* **2016**, *39*, 11–14.
13. Han, Z.; Xu, G.; Huang, T. vessel trajectory outlier detection algorithm based on adaptive threshold. *Comput. Mod.* **2018**, *9*, 42–47.
14. Zhang, L.; Meng, Q.; Xiao, Z. A novel ship trajectory reconstruction approach using AIS data. *Ocean. Eng.* **2018**, *159*, 165–174. [CrossRef]
15. Liu, X.; Chu, X.; Ma, F. Discriminating method of abnormal dynamic information in AIS messages. *J. Traffic Transp. Eng.* **2016**, *16*, 142–150.
16. Liu, X. Study on the Approach of Ensuring AIS Data Availability in Inland Waterway. Ph.D. Thesis, Wuhan University of Technology, Wuhan, China, 2017.
17. Chen, X.; Ling, J.; Yang, Y.; Zheng, H.; Xiong, P.; Postolache, O.; Xiong, Y. Ship Trajectory Reconstruction from AIS Sensory Data via Data Quality Control and Prediction. *Math. Probl. Eng.* **2020**, *2020*, 7191296. [CrossRef]
18. Chen, X.; Wu, S.; Shi, C.; Huang, Y.; Yang, Y.; Ke, R.; Zhao, J. Sensing Data Supported Traffic Flow Prediction via Denoising Schemes and ANN: A Comparison. *IEEE Sens. J.* **2020**, *20*, 14317–14328. [CrossRef]
19. Chen, X.; Wang, Z.; Hua, Q.; Shang, W.L.; Luo, Q.; Yu, K. AI-Empowered Speed Extraction via Port-Like Videos for Vehicular Trajectory Analysis. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 4541–4552. [CrossRef]
20. Guo, S.; Mou, J.; Chen, L.; Chen, P. Improved kinematic interpolation for AIS trajectory reconstruction. *Ocean. Eng.* **2021**, *234*, 109256.1–109256.15. [CrossRef]
21. Zhao, L. Ship Trajectory Outlier Detection Based on AIS Data and Recurrent Neural Network. Ph.D. Thesis, Dalian Maritime University, Dalian, China, 2019.
22. Zhao, L.; Shi, G. Maritime Anomaly Detection using Density-based Clustering and Recurrent Neural Network. *J. Navig.* **2019**, *72*, 894–916. [CrossRef]
23. Zhao, L.; Shi, G.; Yang, J. Ship trajectories pre-processing based on AIS data. *J. Navig.* **2018**, *71*, 1210–1230. [CrossRef]
24. Wang, Y. Detection and Early Warning Method of Ship Abnormal Behavior Based on Massive AIS Data. Ph.D. Thesis, Dalian Maritime University, Dalian, China, 2020.
25. Zhang, T.; Zhao, S.A.; Cheng, B. Detection of AIS closing behavior and MMSI spoofing behavior of ships based on spatiotemporal data. *Remote Sens.* **2020**, *12*, 702. [CrossRef]

26. Iphar, C.; Ray, C.; Napoli, A. Data integrity assessment for maritime anomaly detection. *Expert Syst. Appl.* **2020**, *147*, 113219. [CrossRef]
27. Feng, C.; Fu, B.; Luo, Y.; Li, H. The design and development of a ship trajectory data management and analysis system based on AIS. *Sensors* **2021**, *22*, 310. [CrossRef]
28. Varlamis, I.; Kontopoulos, I.; Tserpes, K.; Etemad, M.; Soares, A.; Matwin, S. Building navigation networks from multi-vessel trajectory data. *GeoInformatica* **2021**, *25*, 69–97. [CrossRef]
29. Jeong, M.H.; Jeon, S.B.; Lee, T.Y.; Youm, M.K.; Lee, D.H. Vessel trajectory reconstruction based on functional data analysis using automatic identification system data. *Appl. Sci.* **2020**, *10*, 881. [CrossRef]
30. Huang, H.; Qiu, K.; Jeong, M.H.; Jeon, S.B.; Lee, W.P. Detecting anomalous vessel dynamics with functional data analysis. *J. Coast. Res.* **2019**, *91*, 406–410. [CrossRef]

Journal of
Marine Science and Engineering

MDPI

Article

Application of an Encoder–Decoder Model with Attention Mechanism for Trajectory Prediction Based on AIS Data: Case Studies from the Yangtze River of China and the Eastern Coast of the U.S.

Licheng Zhao [1], Yi Zuo [1,2,*], Tieshan Li [3] and C. L. Philip Chen [4]

[1] Navigation College, Dalian Maritime University, Dalian 116026, China
[2] Maritime Big Data & Artificial Intelligent Application Centre, Dalian Maritime University, Dalian 116026, China
[3] School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China
[4] School of Computer Science & Engineering, South China University of Technology, Guangzhou 510006, China
* Correspondence: zuo@dlmu.edu.cn

Abstract: With the rapid growth of shipping volumes, ship navigation and path planning have attracted increased attention. To design navigation routes and avoid ship collisions, accurate ship trajectory prediction based on automatic identification system data is required. Therefore, this study developed an encoder–decoder learning model for ship trajectory prediction, to avoid ship collisions. The proposed model includes long short-term memory units and an attention mechanism. Long short-term memory can extract relationships between the historical trajectory of a ship and the current state of encountered ships. Simultaneously, the global attention mechanism in the proposed model can identify interactions between the output and input trajectory sequences, and a multi-head self-attention mechanism in the proposed model is used to learn the feature fusion representation between the input trajectory sequences. Six case studies of trajectory prediction for ship collision avoidance from the Yangtze River of China and the eastern coast of the U.S. were investigated and compared. The results showed that the average mean absolute errors of our model were much lower than those of the classical neural networks and other state-of-the-art models that included attention mechanisms.

Keywords: ship trajectory prediction; AIS data; neural network; attention mechanism; encoder–decoder model

Citation: Zhao, L.; Zuo, Y.; Li, T.; Chen, C.L.P. Application of an Encoder–Decoder Model with Attention Mechanism for Trajectory Prediction Based on AIS Data: Case Studies from the Yangtze River of China and the Eastern Coast of the U.S.. *J. Mar. Sci. Eng.* **2023**, *11*, 1530. https://doi.org/10.3390/jmse11081530

Academic Editor: Sergei Chernyi

Received: 3 July 2023
Revised: 26 July 2023
Accepted: 29 July 2023
Published: 31 July 2023

1. Introduction

Since 2002, the International Maritime Organization (IMO) has required that all seagoing ships (>300 GT) and passenger ships are equipped with an onboard automatic identification system (AIS) [1]. This is a transmission and communication technology that enables a ship to transmit AIS information to other ships. This information includes the ship identity, location, speed, and course; that is, the ship navigation behavior and status [2–4]. Based on these data, ships can effectively avoid collisions with other ships. Decisions about collision avoidance must comply with the collision avoidance rules formulated by the IMO, which have been noted in the Convention on the International Rules for the Prevention of Collisions at Sea (COLREGs) [5]. The risk of collision is specified in the COLREGs, which is assessed based on the estimated closest point of approach. The distance closest point of approach and time closest point of approach are used as indicators of collision risk. If these two values are less than the threshold, a risk of ship collision is considered [6]. Therefore, it is necessary to accurately predict the trajectory of ships, to help with ship navigation planning and collision warning [7,8].

Ship trajectory prediction methods can be divided into two main categories. In most traditional methods [9–17], ship trajectory prediction usually requires professional knowledge of different application scenarios [18]. In recent methods, machine-learning models are trained based on AIS data, to provide decision-making support to ship navigation and path planning. Support vector machines (SVM) [19], clustering algorithm [20], multi-layer perceptron (MLP) [21], back-propagation neural network (BPNN) [22,23], and long short-term memory (LSTM) [24–26] are widely used in ship trajectory prediction. During the last decade, there has been a dramatic increase in ship trajectory prediction studies based on deep learning [27]. Among them, recurrent neural networks (RNN) [28] and LSTM are particularly popular. However, most studies have considered for route planning macro design. However, studies of collision avoidance from a micro perspective are required.

To address the above issues, this paper proposes an encoder–decoder model for ship trajectory prediction for collision avoidance, which uses a sequence-to-sequence (Seq2Seq) structure and multi-attention mechanism [29–31]. The main advances of this study in the field of machine learning and ship navigation can be divided into two aspects.

First, this study is the first to introduce state-of-the-art attention mechanisms into the field of navigation trajectory prediction, to effectively capture the potential information and correlations in the AIS series data. Second, this study applied the proposed model to case studies of ship collision avoidance, to demonstrate its effectiveness and efficiency. We performed experiments with six case studies of trajectory prediction for ship collision avoidance on the Yangtze River of China and the eastern coast of United States. The results showed that the mean absolute error (MAE) of our model in trajectory prediction was much lower than those of the classical models, such as back-propagation neural networks (BPNN) and LSTM. Furthermore, our model also outperformed other state-of-art models with attention mechanisms for trajectory prediction.

The remainder of this paper is organized as follows: Section 2 reviews related work in the field of ship trajectory prediction. Section 3 summarizes the trajectory prediction model studied in this paper and introduces data preprocessing. In Section 4, we apply the proposed prediction method to real data of AIS and summarize the results. Section 5 discusses our conclusions and future work.

2. Literature Review

2.1. Trajectory Prediction Based on Kinematics Models

Regarding kinematics models, most studies directly used the current position and sailing speed of the ship to estimate its future position and then used the constant speed and ground heading values to predict the future position of the ship [9,10]. These studies also described the uncertainty of the future position of ships based on statistical models [11–13]. On the other hand, ship trajectory prediction can be considered a typical time-series problem; therefore, Kalman filters [14,15] and Markov models [16,17] are used. Perera et al. [15] proposed an extended Kalman filter to formulate the ship position, speed, and acceleration, to predict its trajectory under noisy conditions. Guo et al. [17] divided the designated sea area into grids, with the state of ship position, speed, and direction, and then used a K-order hidden Markov model to establish the state transition matrix for prediction.

2.2. Trajectory Prediction Based on Machine Learning Techniques

Classical machine-learning methods, such as SVM [19] and clustering algorithm [20], are widely used in ship trajectory prediction. They have improved prediction efficiency and accuracy. At the beginning of the 2000s, Hinton et al. proposed a multi-hidden-layer neural-network model [32]; deep-learning methods have shown advanced performance in the field of machine learning. In trajectory prediction, deep-learning methods have achieved higher prediction accuracy than MLP [21] and BPNN [22]. Since RNN [28] and LSTM [33] have become the most representative prediction methods for time-series classification and prediction models, a large number of studies have applied them to ship trajectory prediction [24–26]. Based on a RNN, the encoder–decoder model is considered the standard method for Seq2Seq prediction tasks, because of its excellent performance in machine translation [30] and speech recognition [34], which can also be applied to trajectory prediction [35]. The Seq2Seq model based on an attention mechanism [31,36,37] has proven its effectiveness in a wide range of prediction tasks. Several studies have applied attention-mechanism-based models to the field of ship trajectory prediction [38–41]. Capobianco et al. [41] proposed an attention-based recursive encoder–decoder architecture to solve the trajectory prediction problem of applying uncertainty quantification to a case study in the maritime field.

In comparison with previous studies, the model proposed in this study introduces multiple-attention modules of global attention and multi-head self-attention. The global attention mechanism is used to combine the trajectory history and current state information to obtain hidden information from within sequences, and the multi-head self-attention mechanism can capture the spatiotemporal correlations between sequence-feature data, to perform feature fusion for generating new feature representations, as well as effectively capturing potential information and correlations contained in the ship position sequence.

3. Methodology

3.1. Problem Statement

During marine traffic encounters, an AIS can obtain the state of interaction between the target ship and the surrounding ships. The state of the ship at time t can be expressed as $s_t = (LON_t, LAT_t, SOG_t, COG_t)$, where LON, LAT, SOG and COG represent longitude, latitude, speed, and heading, respectively. $s_{t-1} - s_t$ represents the change in relative position and state information from *t-1* to time t. The spatial position of the encounter ship can be expressed as s'_t, where $s_t - s'_t$ represents the relative position and navigation status information from the encounter ship to the target ship.

As shown in Figure 1, the ship state information at t is used as input to the prediction model, and the location information at *t+1* after t is used as the model output. Therefore, we can formulate $Output = f(Input)$, where $f(.)$ represents the prediction function of the ship trajectory obtained using our model.

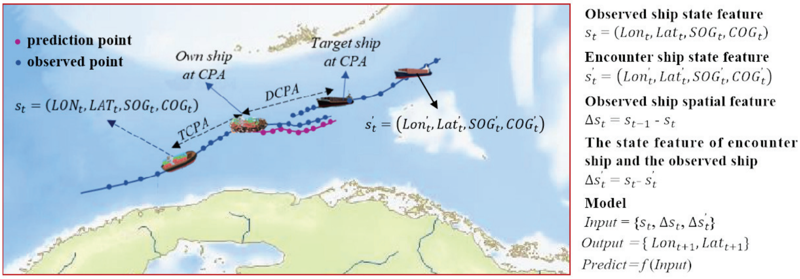


Figure 1. Overview explanation of trajectory prediction.

3.2. Methodology Design of Ship Trajectory Prediction

To solve the trajectory prediction of ships in an encounter situation, we add the relative position and navigation status information of the trajectory of the observed ship and the trajectories of the surrounding ships to our prediction framework. In this study, a sequence model is used to determine the impact of the relative positional changes of the observed ship and the ship sailing on the future navigation trajectory of the observed ship. The attention mechanism is used to dynamically adjust the weight of the sequence information to help the model focus on the important position change information, to dynamically adjust the prediction in the sequence prediction process.

This study proposes a new trajectory prediction structure that uses AIS data to train the model, as shown in Figure 2. The model is composed of three modules: Module 1 is an AIS data processing module, which can effectively improve the data quality and model execution efficiency. Module 2 is the trajectory prediction model developed in this study, which is a deep-learning prediction model with an encoder–decoder structure and an attention mechanism. The encoder structure is a multi-layer LSTM, and the decoder consists of an RNN, a multi-layer LSTM, and a self-attention mechanism, which is described in detail in Section 3.3. The training data from Module 1 are used as input to Module 2 to train the prediction model. Finally, Module 3 is a prediction and validation module. Its main function is to apply the test data of Module 1 to optimize the parameters and verify the prediction model of Module 2.

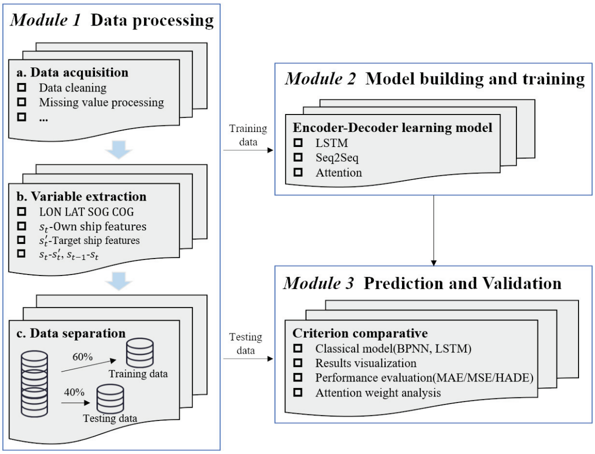


Figure 2. Overview of the proposed trajectory prediction model.

3.3. Design of the Encoder–Decoder Learning Model

The encoder of the proposed model is an LSTM neural network, which maps the input influence onto the sequential context representation. Based on the attention mechanism, the hidden state sequence encoder is combined with the information representation of the context. The decoder of the proposed model is a feature fusion layer that extracts the potential relationships of future ship trajectory state information from the historical and current state information. The weighted representation between the feature vectors of each trajectory is then input into the RNN, so that it can obtain the information representation of the correlation between features in each future prediction step. The RNN in the decoder has a multi-layer structure, to improve the learning ability of the internal sequence information representation. The overall structure is shown in Figure 3.

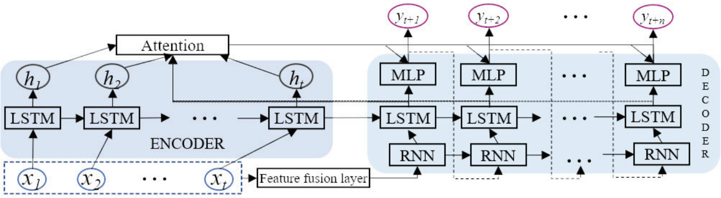


Figure 3. Overview of our model.

3.3.1. LSTM-Based Sequence to Sequence

The Seq2Seq model consists of an encoder and a decoder. The two units use a recursive neural network (RNN or LSTM) to encode the input as a vector representation and then use another sequential network to decode it. The main task of the encoder is to read the sequence and pass the discovered rules to the decoder. The decoder decodes the received rule information to generate an output sequence.

Figure 4 shows the classic Seq2Seq model architecture.

- $x_1$to x_t represent the input sequence characteristic information of the model;
- $h_1$to h_t are the outputs of each circulating neural network cell;
- y_1 and y_2 represent the label sequence of the model output;
- The variable C between the encoder and decoder represents the sequence information representation obtained by passing the input feature sequence information through the encoder.

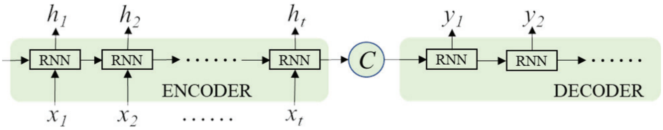
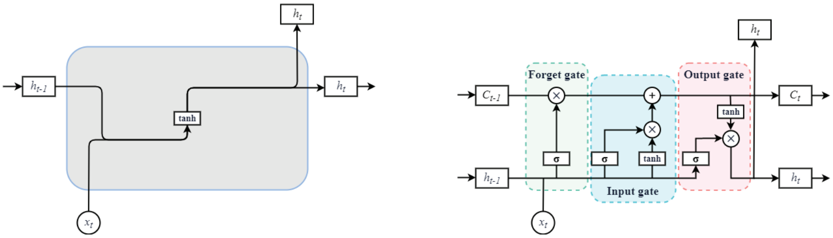


Figure 4. Seq2seq model architecture.

The RNN is a feature extractor of global information from the sequence and can be used to process the sequence data. In the RNN, neurons can accept information from other neurons and also their own information, to form a loop structure, as shown in Figure 5a. Here, the gradient disappeared due to the long input sequence.



(**a**) Structure of the RNN cell. (**b**) LSTM cell structure.

Figure 5. Structure of the recurrent neural network cell. The symbol x_t represents the input information of the RNN at the current time step t, h_{t-1} represents the hidden state of time step $t-1$, and h_t represents the output information of time step t.

To solve the problem of the vanishing gradient, a gating mechanism for forgetting the previously accumulated information is required. An LSTM is a type of RNN based on a gating mechanism. Compared with the traditional RNN, an LSTM introduces a gating mechanism to control the speed of information accumulation. Through the forgetting gate, input gate, and output gate, it forgets the previous information and simultaneously adds new information, which effectively solves the loss of learning information caused by a gradient explosion or disappearance. The unit structure is shown in Figure 5b. The notation σ represents the sigmoid activation function, h_{t-1} is the output of the previous LSTM unit, x_t indicates the state information of the input at the current time, and C_{t-1} is the internal state of the memory unit in the last moment. Each memory block has three gates to control the path of the information transmission.

a. Forget gate. h_{t-1}, C_{t-1}, and x_t are used as inputs to calculate the amount of information f_t (value is between 0 and 1) to be forgotten.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \cdot C_{t-1} \tag{1}$$

b. Input gate. The input information i_t and candidate status $\widetilde{C}_t$ can be obtained by the inputs h_{t-1} and x_t with a sigmoid function and tanh function, respectively. To calculate $i_t \cdot \widetilde{C}_t$, we need to update the information and forgotten information $C_{t-1} \cdot f_t$, and then obtain a new state C_t. The specific Equations are (2)–(4).

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

$$\widetilde{C}_t = tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{3}$$

$$C_t = i_t \cdot \widetilde{C}_t + C_{t-1} \cdot f_t \tag{4}$$

c. Output gate. h_{t-1} and x_t are input to the sigmoid function to obtain the output information O_t. The product of the output information and activated value of the current updated state is the information carried by the internal state at the current time h_t as the output information at time t.

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = O_t \cdot tanh(C_t) \tag{6}$$

Equations (1)–(6) introduce the operation of the LSTM unit in detail. The output function of the LSTM unit can be expressed as follows:

$$h_t = LSTMUnit(x_t, h_{t-1}, \theta) \tag{7}$$

where the *LSTMUnit*$(\cdot)$ function represents the operation rules forget, input, and output in Equations (1)–(6); and θ represents the parameters in the LSTM unit.

3.3.2. Attention Mechanism

The attention mechanism module is used in the decoder of the model, to improve the information resource allocation of the model. This can enable the model to dynamically adjust the weights of serial information and allow it to focus on important positions to achieve dynamic adjustment of the weights during the prediction process. The structure of the attention mechanism is shown in Figure 6.

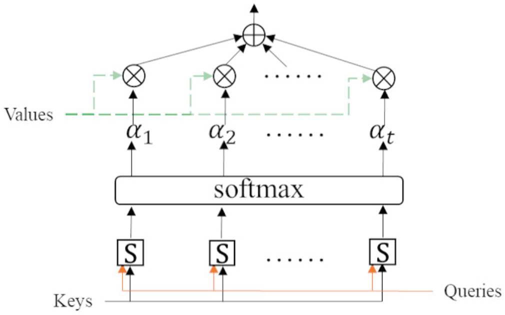


Figure 6. Attention mechanism structure.

Keys = *values* = $h_t(t \in \{1, \ldots, N\})$ are the outputs of all LSTM unit sequences in the encoder at all times, and *query* = h' is the output of the LSTM layer in the decoder. First, the correlation between h' and h_t is calculated using the attention-scoring function $s(\cdot)$. The calculation formula of h' is $h' = LSTMUnit(X_t, h_t, \theta^{'})$.

The models commonly used as $s(\cdot)$ are additive models, such as point product models or scaling point product models.In this study, a scaling point product model is selected as the score function. It can make better use of a matrix product in the process of matrix operation and can effectively solve the decrease in the *softmax* function gradient. D represents the dimension of the input vector.

$$s(h_t, h') = \frac{h_t^T h'}{\sqrt{D}} \tag{8}$$

The *softmax* function is used to map the output value of the score function between 0 and 1, and the attention distribution of h' with respect to h_t is obtained, α_t, which indicates the degrees of input vector at t. Finally, α_t is the weight, and h_t is the weighted sum of the corresponding positions.

$$\alpha_t = softmax(s(h_t, h')) = \frac{exp(s(h_t, h'))}{\sum_{j=1}^{N} exp(s(h_j, h'))} \tag{9}$$

$$att(H, h') = \sum_{t=1}^{N} \alpha_t h_t, H = [h_1, \ldots, h_N] \tag{10}$$

3.3.3. Feature Fusion Layer

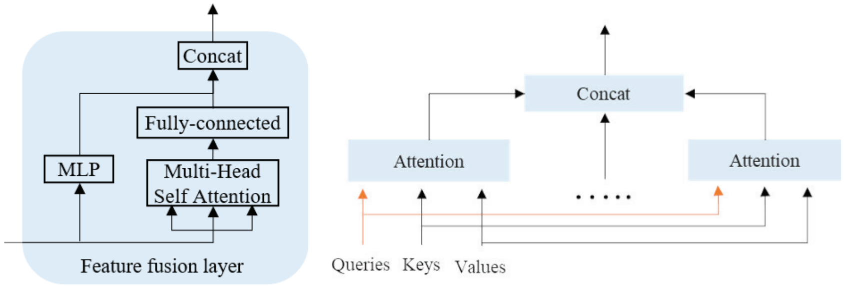
To model the potential relationships of the ship trajectory information between the historical state and current state, we propose a feature fusion layer. Its structure is shown in Figure 7a. This layer consists of two parts: a multi-layer perceptron (MLP) and a multi-head self-attention (MHSA) mechanism. The MLP is used for linear mapping of the input sequence information. The MHSA is used for calculating and selecting multiple information points from the input information in parallel (see Figure 7b). The original structure of the self-attention mechanism is shown in Figure 6, which is set as *Keys* = *Values* = *Query*. The MHSA can obtain the dependency information at the input stage, connect the input information, and extract the important features from the input data. Simultaneously, these features are spliced with the linear mapping information, which is extracted by the MLP from the input data. The calculation formulas are given as follows:

$$MH_{att} = \sigma\left(W_{MH}\left(concat\left(att((K, V), Q)_i\right)\right)\right), for\ i = 1, \ldots, d \tag{11}$$

$$M = \sigma(W_m \cdot X) \tag{12}$$

$$F = concat(MH_{att}, M) \tag{13}$$

where σ, W_{MH}, and W_m are the activation function in the full connection layer and weight parameters, respectively. The *concat*$(\cdot)$ function is used to connect multiple arrays without changing the existing array values. The subscript i of the attention function is the head number of the self-attention mechanism, where $K = V = Q = X$. Finally, X is the position and status information of the observed ship at time t, the previous k times, and the encounter ship.



(**a**) Structure of the feature fusion layer. (**b**) Multi-head attention mechanism structure.

Figure 7. Structure of the feature fusion layer and multi-head attention mechanism.

4. Numerical Experiments

4.1. Data Description

The AIS data used in this study were mainly collected from onboard AIS equipment in the Yangtze River delta region of China and the eastern coastal region of the United States. In this experiment, we collected a large amount of AIS data and selected six case studies of collision avoidance from the two regions. The research subject data from each region were collected on different dates. Table 1 provides detailed information about the collected data, and Figure 8 shows the specific location of each collision avoidance case.

In the experiment, the navigation trajectory sequence was sequentially sampled with a set sliding window length, where 60% and 40% of the samples were randomly divided into training and testing sets. The processing of experimental datasets for each situation was the same. The training set was used to train and determine the weights, deviation, and other parameters of the model. After training, the test set was used to evaluate the proposed model and other comparison models.

4.2. Setting of Experiments

4.2.1. Criterion of Model Evaluation

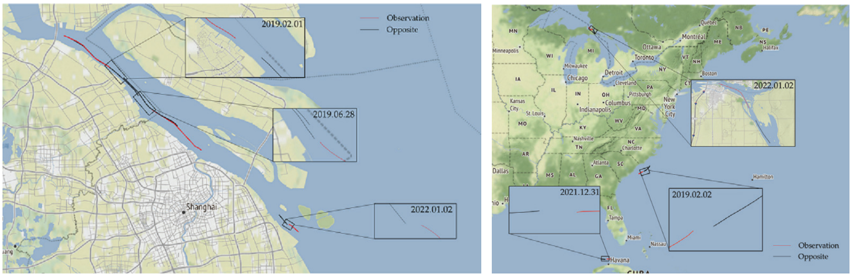
The mean square error (MSE) was used as the loss function to quantify the difference between the predicted and real values. After completing the model construction, we used the MAE and average displacement error based on the Haversine distance (HADE) to evaluate the model. These are calculated as follows:

$$MSE = \frac{1}{p}\sum_{l=i}^{p}(Y_i - \hat{y}_i)^2 \tag{14}$$

$$MAE = \frac{1}{p}\sum_{l=i}^{p}|Y_i - \hat{y}_i| \tag{15}$$

$$HADE = \frac{1}{p}\sum_{l=i}^{p} 2r \arcsin\left(\sqrt{\sin^2(\frac{\hat{lat}_i - lat_i}{2}) + \cos\hat{lat}_i \cos lat_i \sin^2(\frac{\hat{lon}_i - lon_i}{2})}\right) \tag{16}$$

where p is the total number of AIS data samples for training or testing, $\hat{y}_i$ is the estimated value of the ship trajectory longitude and latitude, Y_i is the measured value of the navigation longitude and latitude of the ship, r is the Earth's radius, $\hat{lat}_i$ and $\hat{lon}_i$ represent the predicted latitude and longitude, and lat_i and lon_i represent the true latitude and longitude, respectively. In the experiment, the ship status information containing the first 10 time steps of the current time t was used as input to the model, and the geographic location information of the 10 time steps after the current time t was predicted as the output.



(**a**) (**b**)

Figure 8. Two routes of the case ships; the red line represents the navigation trajectory of the observation ship; the black line represents the navigation trajectory of the opposite ship. (**a**) Navigation trajectory of collision avoidance for ships in Yangtze River delta. (**b**) Navigation trajectory of collision avoidance for ships in the eastern coastal area of the United States.

Table 1. Collision avoidance cases of experimental ship trajectories from different areas.

Water Area	MMSI	Number of AIS Points	Date	References
Yangtze River Delta	413450480	2124	1 February 2019	Figure 8a
	413425610	1529		
	414386000	912	28 June 2019	
	312958000	786		
	413556520	257	2 January 2022	
	413585000	207		
Eastern Coastal Area of the United States	367185680	1440	2 February 2019	Figure 8b
	304604000	823		
	372821000	1180	31 December 2021	
	311000375	669		
	316001635	1112	2 January 2022	
	316044371	1440		

4.2.2. Model Parameter Setting

The Adam adaptive learning rate optimization algorithm [42] was used to update the network parameters of the model structure. In the LSTM layer part of the sequence information extraction, the number of LSTM layers was between one to three. The number of hidden units in each layer of the LSTM searching for the optimal value was taken from [32, 320]. The serial batch size in the experiment was 512, the number of training simulations was 200, and the number of heads for the multi-head self-attention mechanism was 2. To prevent model overfitting, a dropout mechanism [43] and a regularization term were used in the training process. The optimization range and optimization interval granularity of each parameter and the parameter value ranges are shown in Table 2. After the comparison experiment with multiple sets of super parameter selection, using the encoder–decoder model based on the multi-module attention mechanism, the parameters of two different trajectory regions (Situation 1 and Situation 2) were chosen as shown in the last two columns of Table 2.

Table 2. Model training parameters.

Parameters	Optimization Range	Interval Granularity	Head-On Situation 1	Head-On Situation 2
Dropout rate	(0.1, 0.5)	0.1	-	-
Learning rate	(0.0001, 0.1)	0.0001	0.002	0.002
No. of LSTM layers	(1, 3)	1	3	3
No. of hidden cells	(32, 320)	32	128	128
No. of MHSA head	(2, 10)	1	2	2
Regularization parameter	(0.01, 1)	0.01	0.001	0.001

4.2.3. Introduction of Baseline Methods

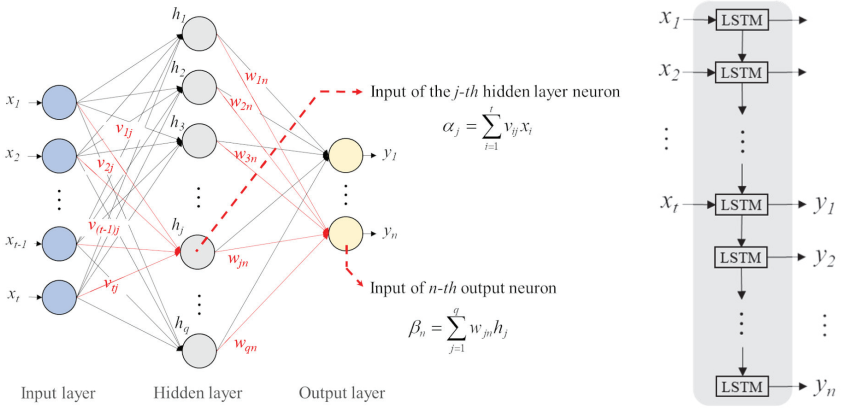
In the comparisons, we employed four baseline methods, as follows:

(1) The BPNN has the classic three layers: input layer, hidden layer, and output layer (see Figure 9a). In its network structure, the neurons are connected from the input layer to the output layer;

(2) LSTM is a classic sequence prediction model. The structure is shown in Figure 9b, and the unit structure of each LSTM cell can be found in Figure 5b;

(3) DANAE, Denoising automatic encoders (DAE) were proposed by Vincent et al. [44] and are used for prediction tasks, while DANAE is a deep denoising automatic encoder used for attitude estimation [45,46];

(4) EncDec-ATTN is a deep learning method used for ship trajectory prediction based on recurrent neural networks and was proposed by Capobianco et al. [28]. This method can learn spatiotemporal correlations from historical ship mobility data and predict future ship trajectories.



(**a**) BPNN (**b**) LSTM

Figure 9. The basic structure of the baseline method.

4.3. Prediction Analysis and Result Discussion

4.3.1. Analysis of Model Performance

MSE and MAE were used to evaluate the performance of the model, which was carried out using a Windows 10 system with a 2.90 GHz i5 central processor and 32 GB of memory. The model was coded using TensorFlow 2.4.0 in Python 3.8. The analyses of the prediction performance of our model and those of the other two models are shown in Table 3.

Table 3 shows that our model had the lowest MSE, MAE, and HADE. In both regions, the parameter number of the LSTM layer was 3, the number of hidden units was 128, the learning rate was 0.002, and the regularization parameter was 0.001. For Head-on Situation 1 from the Yangtze River, the MSE and MAE of the latitude predicted by our model were 2.8857×10^{-5} and 0.0042, respectively, and the MSE and MAE of the longitude were 2.5220×10^{-5} and 0.0041, respectively. For Situation 2 from the eastern coastal area, the MSE and MAE of the latitude predicted by our model were 3.8907×10^{-5} and 0.0044, respectively, and the MSE and MAE of the longitude were 3.0541×10^{-5} and 0.0042, respectively. Comparing with the experimental results of the other models, our model and EncDec-ATTN consistently outperformed the classic network models not using attention mechanisms. Moreover, the evaluation scores of our model decreased by 27.5% and 21.4%, respectively, compared with EncDec-ATTN, which indicated a greater advantage for trajectory prediction in this experiment. HADE reflects the displacement error of the predicted position of a ship from its true position in real scenarios, as shown in Figure 10. In terms of the prediction performance of the ship trajectories, our model had the lowest MSE and MAE values. Meanwhile, the HADE was also lower than that of the other prediction models.

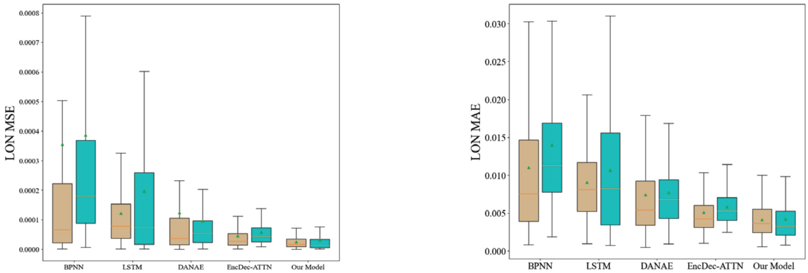
Table 3. Comparison with the performance index of classic models.

	Model	Position	MSE	MAE	HADE	Optimal Parameter
Head-on Situation 1 from Yangtze River delta	BPNN	LON LAT	3.1854×10^{-4} 3.2575×10^{-4}	0.0110 0.0113	1256.0985	3, (128, 64, 32), 0.01, 0.002
	LSTM	LON LAT	1.4240×10^{-4} 1.5876×10^{-4}	0.0093 0.0096	1072.9199	3, (128, 128, 128), 0.01, 0.002
	DANAE	LON LAT	1.2190×10^{-4} 1.5209×10^{-4}	0.0074 0.0085	890.1905	3, (128, 64, 10), 0.001, 0.002
	EncDec-ATTN	LON LAT	4.5361×10^{-5} 5.4565×10^{-5}	0.0051 0.0057	612.8078	2, (128, 128), 0.002, 0.001
	Our Model	LON LAT	2.5220×10^{-5} 2.8857×10^{-5}	0.0041 0.0042	480.3572	3, (128, 128, 128), 0.002, 0.001
Head-on Situation 2 from eastern coastal area of the United States	BPNN	LON LAT	3.8615×10^{-4} 3.6022×10^{-4}	0.0197 0.0190	1366.2123	3, (128, 64, 32), 0.01, 0.002
	LSTM	LON LAT	1.9693×10^{-4} 1.6729×10^{-4}	0.0107 0.0105	1045.1206	3, (128, 128, 128), 0.01, 0.002
	DANAE	LON LAT	9.1580×10^{-5} 9.7660×10^{-5}	0.0077 0.0082	670.2299	3, (128, 64, 10), 0.001, 0.002
	EncDec-ATTN	LON LAT	5.8020×10^{-5} 5.0178×10^{-5}	0.0058 0.0056	493.7096	2, (128, 128), 0.002, 0.001
	Our Model	LON LAT	3.0541×10^{-5} 3.8907×10^{-5}	0.0042 0.0044	422.7494	3, (128, 128, 128), 0.002, 0.001

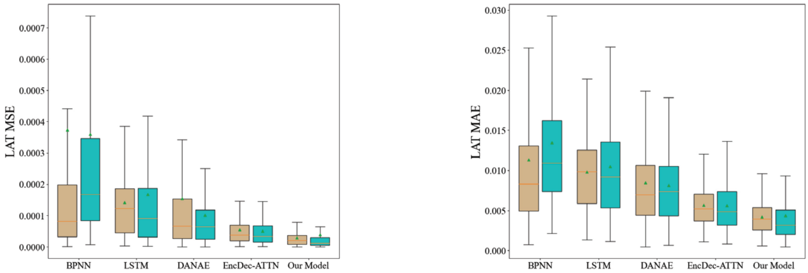
4.3.2. Discussion of the Prediction Results

To further evaluate the prediction ability, a comparison of the prediction results of the different prediction methods at different time steps is shown in Figures 11 and 12. The model prediction was evaluated according to the error between the predicted results and the actual longitude and latitude.

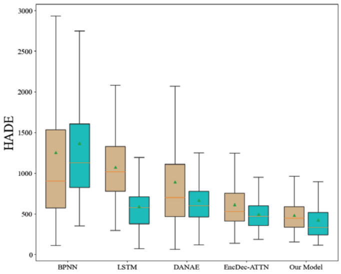
In Head-on Situation 1 from the Yangtze River, as shown in Figure 11a,c, where two ships meet and avoid each other, the first ship is sailing along the planned route, and the course is relatively stable. We observed that all models could predict accurately under relatively simple sailing conditions. As shown in Figure 11b, when the first ship began to change course, the quality of the trajectory prediction generated by our model was much lower than that of the other models.



(**a**) Longitude MSE of two situations. (**b**) Longitude MAE of two situations.



(**c**) Latitude MSE of two situations. (**d**) Latitude MAE of two situations.

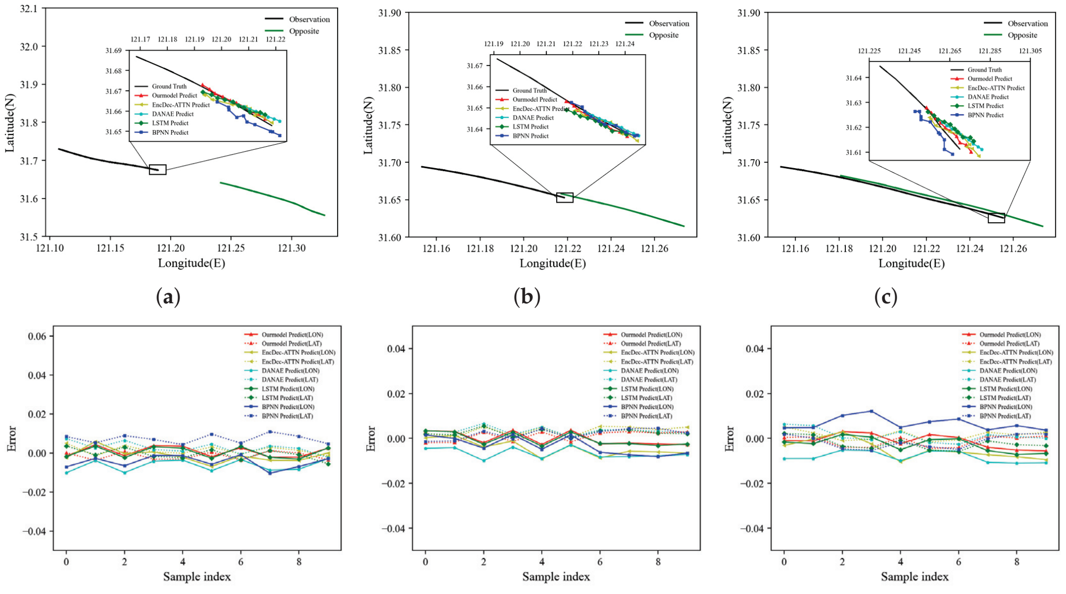


(**e**) HADE of two situations.

Figure 10. Results comparison of the MSE, MAE, and HADE. Brown and blue-green represent the evaluation indicators for Situations 1 and 2 from different areas, respectively. The green dots represent the average values.

In Head-on Situation 2 from the eastern coastal region, as shown in Figure 12a–d, during the initial sailing phase with the encounter ship, the first ship encountered the other ship and changed the sailing course. At this time, the prediction results of BPNN, LSTM, and DANAE have large errors. One benefit of the attention mechanism is that our model can predict ship trajectories with less variance from the actual ship trajectories.

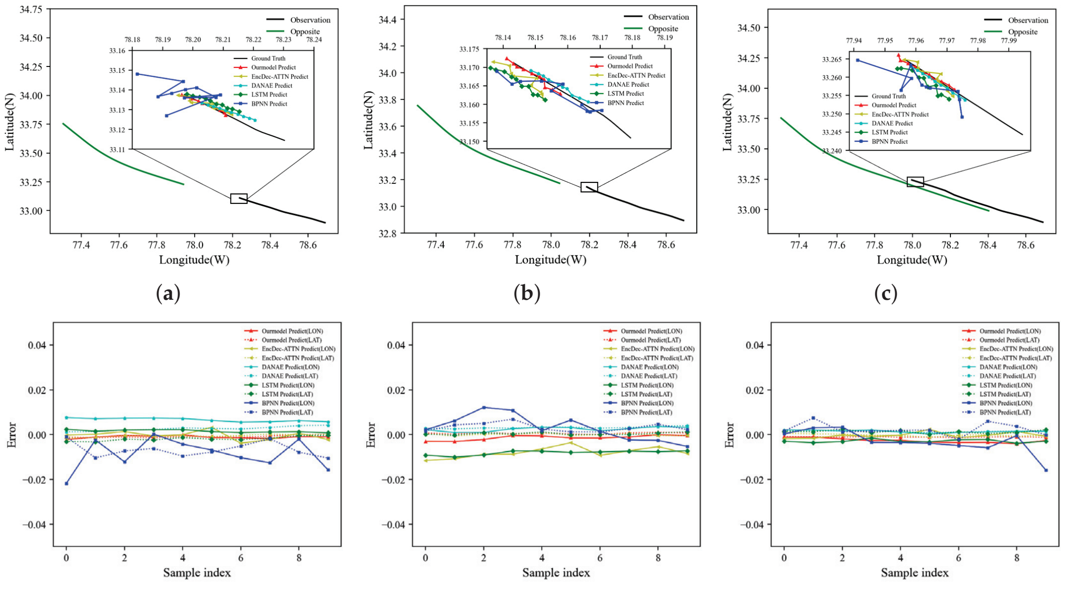
In previous studies [28,38–41], only a single attention mechanism was used to express the historical trajectory and current state information of ships, mainly focusing on hidden information within sequences. However, the correlation of information between the spatiotemporal sequence features is often overlooked. Unlike in previous studies, the proposed model introduces both global attention and multi-head self-attention mechanisms. The global attention mechanism is used to combine the trajectory history and current state information to obtain hidden information between sequences. The multi-head self-attention mechanism can capture the spatiotemporal correlations between the sequence-feature data and extract the fusion features, to generate a new feature representation. The two extracted parts of this information are correlated, to predict the ship trajectory at the next time point.



(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

Figure 11. The prediction results for Head-on Situation 1. (**a**–**c**) Visualization of ship navigation trajectory predicted at different time steps. (**d**–**f**) Error of predicted navigation trajectory at different time steps. The "ground truth" in the legend represents the true values of the experimental sample.



(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

Figure 12. The prediction results for Head-on Situation 2. (**a**–**c**) Visualization of the ship navigation trajectory predicted at different time steps. (**d**–**f**) Error of the predicted navigation trajectory at different time steps. The "ground truth" in the legend represents the true values of the experimental sample.

4.3.3. Analysis of the Attention Mechanism with the Weight Score

Our model includes two attention modules: The global attention mechanism recognizes the information interactions in the input sequence, and the multi-head self-attention mechanism extracts the effective feature information of the output target in the input sequence information. To explore the impact of the attention mechanism on model performance, we compared the performance of the attention mechanism module with and without the attention mechanism, as shown in Table 4.

Table 4. Comparison of the model performance with and without the attention mechanism.

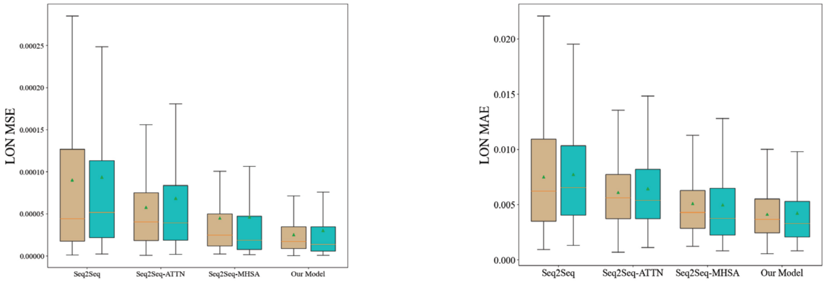
	Model	Position	MSE	MAE	HADE	Optimal Parameter
Head-on Situation 1 from Yangtze River delta	Seq2Seq	LON	9.0469×10^{-5}	0.0075	845.4042	3, (128, 128, 128), 0.01, 0.002
		LAT	9.2702×10^{-5}	0.0074		
	Seq2Seq-ATTN	LON	5.7978×10^{-5}	0.0061	717.5646	3, (128, 128, 128), 0.01, 0.002
		LAT	6.4726×10^{-5}	0.0064		
	Seq2Seq-MHSA	LON	4.5202×10^{-5}	0.0051	608.6507	3, (128, 128, 128), 0.01, 0.002
		LAT	5.2463×10^{-5}	0.0055		
	Our Model	LON	2.5220×10^{-5}	0.0041	480.3572	3, (128, 128, 128), 0.002, 0.001
		LAT	2.8857×10^{-5}	0.0042		
Head-on Situation 2 from eastern coastal area of the United States	Seq2Seq	LON	9.3664×10^{-5}	0.0077	758.1244	3, (128, 128, 128), 0.01, 0.002
		LAT	1.1874×10^{-4}	0.0078		
	Seq2Seq-ATTN	LON	9.0765×10^{-5}	0.0065	629.4445	3, (128, 128, 128), 0.01, 0.002
		LAT	7.0376×10^{-4}	0.0058		
	Seq2Seq-MHSA	LON	4.6450×10^{-5}	0.0050	512.9711	3, (128, 128, 128), 0.01, 0.002
		LAT	5.4739×10^{-5}	0.0057		
	Our Model	LON	3.0541×10^{-5}	0.0042	422.7494	3, (128, 128, 128), 0.002, 0.001
		LAT	3.8907×10^{-5}	0.0044		

Table 4 and Figure 13 show the comparison results of the models under different attention mechanisms. The Seq2Seq model showed a good prediction accuracy with the attention mechanism. Focusing on the prediction for Head-on Situation 1 from the Yangtze River, the prediction results of the model with a global attention mechanism showed decreases in MAE of 0.0014 and 0.0010 in the longitude and latitude predictions, and a decrease in HADE of 15.12%. The MAE of longitude and latitude predicted by the MHSA mechanism decreased by 0.0024 and 0.0019, respectively, and the HADE decreased by 28%. The results of comparing the MSE, MAE, and HADE showed that our model outperformed the other two models with the attention mechanism.

To explain the internal working of the neural network, we obtained the importance weight vector of the input sequence at each position in the prediction sequence and explored the influence of the attention mechanism on the proposed model. Inspired by Lee et al. [47] for the interpretability of the attention mechanism, we visualized the output of two self-attention heads in the prediction model, and the visualization of the output attention weight is shown in Figure 14.

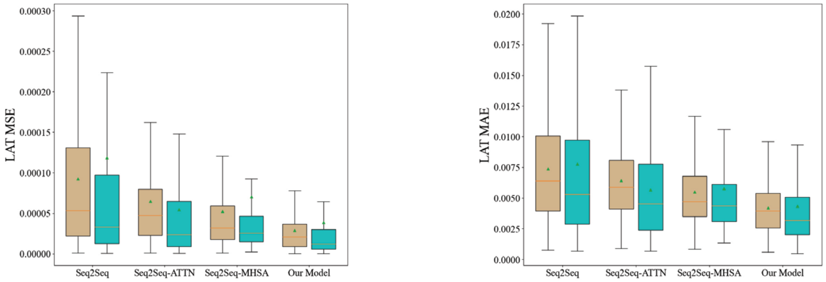
We visualized the weight values calculated by the attention mechanism of the Head-on Situation 2 in the model of different navigation state stages and explained the importance of the network to specific trajectory characteristics. The first column in Figure 14 shows the input, target, and prediction sequences of the different models; the second column shows the visualization of the global attention mechanism weight score of our model, and the last two columns show the visualization of the MHSA weight score of the feature fusion module in our model. In the thermodynamic diagram, the line represents the output sequence, and the list shows the weight distribution of the input sequence. Thus, it can be determined that positions in the history mode are considered more important when generating the predicted trajectory for the global attention mechanism (second column). Over time (from left to right), the model can influence the characteristics of the input sequence when generating the output sequence. At different stages of the predicted navigation status, the positions considered by the output series are different, as shown in Figure 14b,f, which pay more attention to the middle and tail segments of the input series, respectively.

Figure 14j shows a marginal difference in the weight values of the entire series. For the MHSA mechanism (the third and fourth columns), it can be observed from the figures that the different information focused on by the attention head was inconsistent, as shown in Figure 14k. The input X_1 sequence had a greater correlation with X_1, X_7, and X_8, while X_3 had a greater correlation with X_4 and X_9. The MHSA mechanism calculates the correlation representation between AIS information features and generates a better information feature code for the current input sequence by making full use of the position state information in the sequence. This allows the model to focus on the information of different positions in the input sequence, and it can also alleviate overfitting by integrating different attention heads, to improve the accuracy and robustness of the overall model.



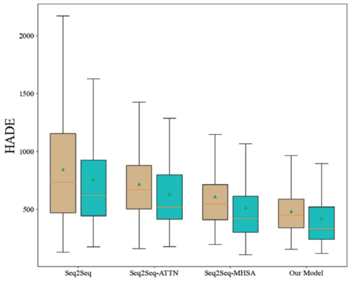
(**a**) Longitude MSE of the two situations.

(**b**) Longitude MAE of the two situations.



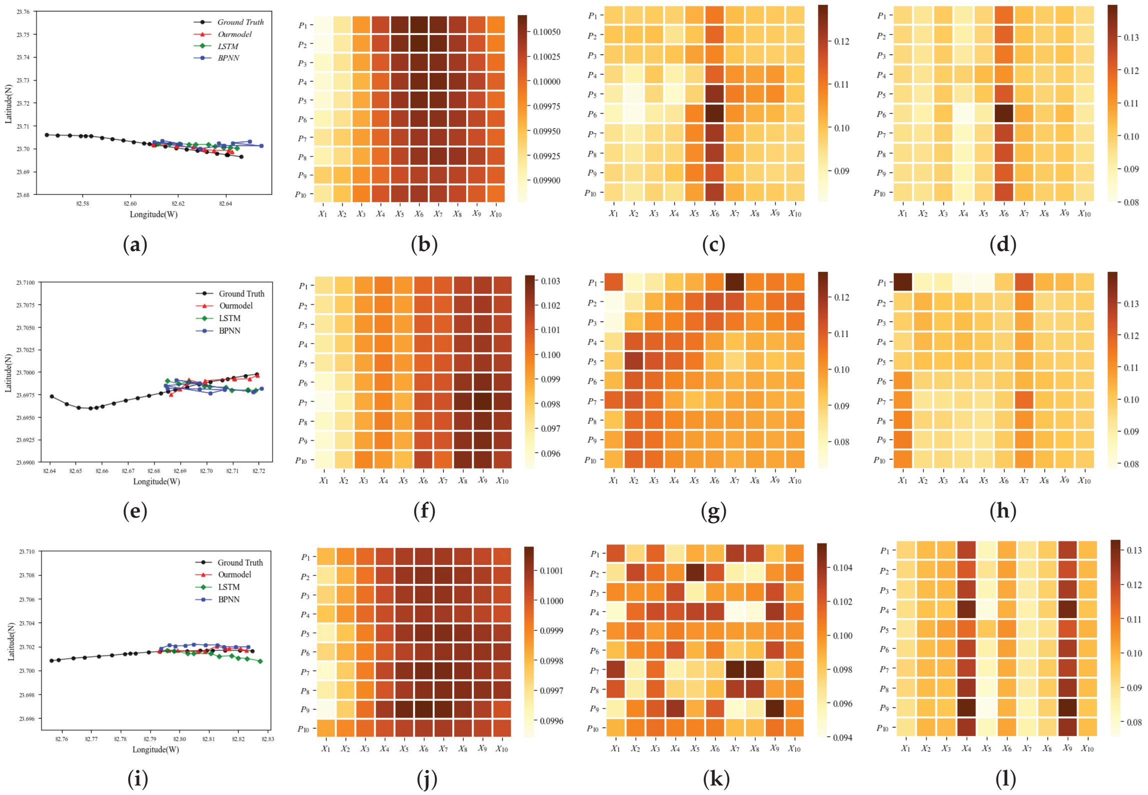
(**c**) Latitude MSE of the two situations.

(**d**) Latitude MAE of the two situations.



(**e**) HADE for the two situations.

Figure 13. Comparison results of MSE, MAE, and HADE. Brown and blue-green represent the evaluation indicators for Situations 1 and 2 from different areas, respectively. The green dots represent the average value.



(**a**) (**b**) (**c**) (**d**)

(**e**) (**f**) (**g**) (**h**)

(**i**) (**j**) (**k**) (**l**)

Figure 14. Weights of attention mechanism. (**a**,**e**,**i**) is the prediction results of the ship navigation trajectory at different conditions. (**b**,**f**,**j**) Weights of the global attention mechanism. (**c**,**g**,**k**) and (**d**,**h**,**l**) represent the weight values of the self-attention mechanism for the two heads, respectively.

4.3.4. Analysis of Model Validation

For both the Yangtze River delta and eastern coast, we selected two encounter scenarios to verify the effectiveness of the trajectory prediction. Test samples 1 and 2 for the Yangtze River delta contained encounter scenarios of MMSI (414386000, 312958000) on 28 June 2019 and MMSI (413556520, 413585000) on 2 January 2022. Test samples 1 and 2 for the eastern coast contained encounter scenarios of MMSI (372821000, 31100037) on 31 December 2021 and MMSI (316001635, 316044371) on 2 January 2022. The experimental results and comparative analysis are shown in Table 5.

From Table 5, it can be seen that our model achieved the best results for the various evaluation indicators, with the highest accuracy and a good predictive performance. In addition, statistical methods were used to analyze the results, as shown in Figure 15. In terms of the prediction performance of the ship trajectories, our model included an attention module that more effectively extracted important feature information from the trajectory sequences than BPNN, LSTM, Seq2Seq, and DANAE. Compared with the models containing a single attention mechanism, such as Seq2Seq-ATTN, Seq2Seq-MHSA, and EncDec-ATTN, our model effectively extracted correlations between the sequences and features, accounting for multiple attention structures. The experimental results also showed that our model had a good trajectory prediction performance under encounter situations.

Table 5. Comparison of the model performance for the test samples.

	Model	Position	Test Sample 1			Test Sample 2		
			MSE	MAE	HADE	MSE	MAE	HADE
Yangtze River delta	BPNN	LON LAT	5.5900×10^{-4} 5.8601×10^{-4}	0.0137 0.0142	537.5320	4.7147×10^{-4} 5.1134×10^{-4}	0.0217 0.0226	853.0951
	LSTM	LON LAT	1.6606×10^{-4} 1.7191×10^{-4}	0.0102 0.0108	399.8864	2.6896×10^{-4} 3.2279×10^{-4}	0.0116 0.0129	640.3072
	Seq2Seq	LON LAT	8.5295×10^{-5} 9.2641×10^{-5}	0.0075 0.0078	296.0660	9.9816×10^{-5} 1.0563×10^{-4}	0.0078 0.0080	485.1224
	Seq2Seq-ATTN	LON LAT	7.3838×10^{-5} 7.6468×10^{-5}	0.0069 0.0069	264.7203	8.0580×10^{-5} 8.4536×10^{-5}	0.0071 0.0071	414.3751
	Seq2Seq-MHSA	LON LAT	5.0014×10^{-5} 4.4967×10^{-5}	0.0056 0.0055	215.3634	6.2602×10^{-5} 6.7629×10^{-5}	0.0061 0.0065	350.034
	DANAE	LON LAT	1.2231×10^{-4} 2.1533×10^{-4}	0.0074 0.0083	301.2500	1.3935×10^{-4} 8.8652×10^{-5}	0.0083 0.0083	495.0518
	EncDec-ATTN	LON LAT	5.2074×10^{-5} 5.7575×10^{-5}	0.0055 0.0058	218.4405	5.6014×10^{-5} 7.9404×10^{-5}	0.0057 0.0068	334.0571
	Our Model	LON LAT	3.2121×10^{-5} 3.5942×10^{-5}	0.0044 0.0045	174.3734	2.9485×10^{-5} 3.5306×10^{-5}	0.0044 0.0048	250.2701
Eastern Coastal Area of the United States	BPNN	LON LAT	5.4331×10^{-4} 6.2152×10^{-4}	0.0233 0.0249	924.1445	5.2985×10^{-4} 5.7272×10^{-4}	0.0160 0.0168	1356.0606
	LSTM	LON LAT	1.9752×10^{-4} 2.3233×10^{-4}	0.0108 0.0111	767.8572	1.5630×10^{-4} 1.3083×10^{-4}	0.0100 0.0097	1090.1046
	Seq2Seq	LON LAT	5.8735×10^{-5} 9.7247×10^{-5}	0.0062 0.0076	586.9804	1.0423×10^{-4} 9.0954×10^{-5}	0.0078 0.0076	880.4276
	Seq2Seq-ATTN	LON LAT	5.0514×10^{-5} 7.4168×10^{-5}	0.0056 0.0068	520.9248	7.4921×10^{-5} 6.4257×10^{-5}	0.0066 0.0065	745.4695
	Seq2Seq-MHSA	LON LAT	4.6845×10^{-5} 5.7935×10^{-5}	0.0049 0.0059	383.6337	4.7334×10^{-5} 5.1199×10^{-5}	0.0053 0.0054	621.6424
	DANAE	LON LAT	9.3591×10^{-5} 1.2229×10^{-4}	0.0072 0.0084	629.2457	1.2319×10^{-4} 9.7121×10^{-5}	0.0081 0.0074	864.9335
	EncDec-ATTN	LON LAT	4.5829×10^{-5} 5.6622×10^{-5}	0.0056 0.0056	421.0406	5.0587×10^{-5} 4.9324×10^{-5}	0.0055 0.0057	632.9389
	Our Model	LON LAT	2.9595×10^{-5} 4.1747×10^{-5}	0.0043 0.0048	358.1493	3.2749×10^{-5} 3.5261×10^{-5}	0.0043 0.0042	487.4037

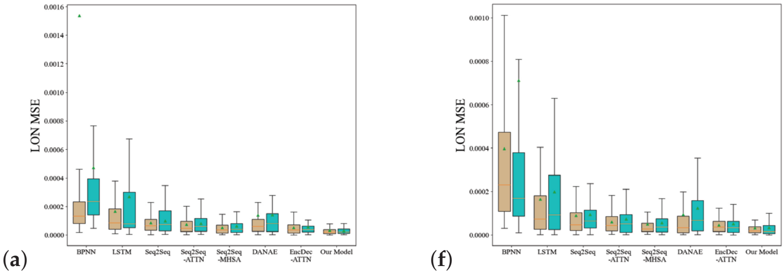
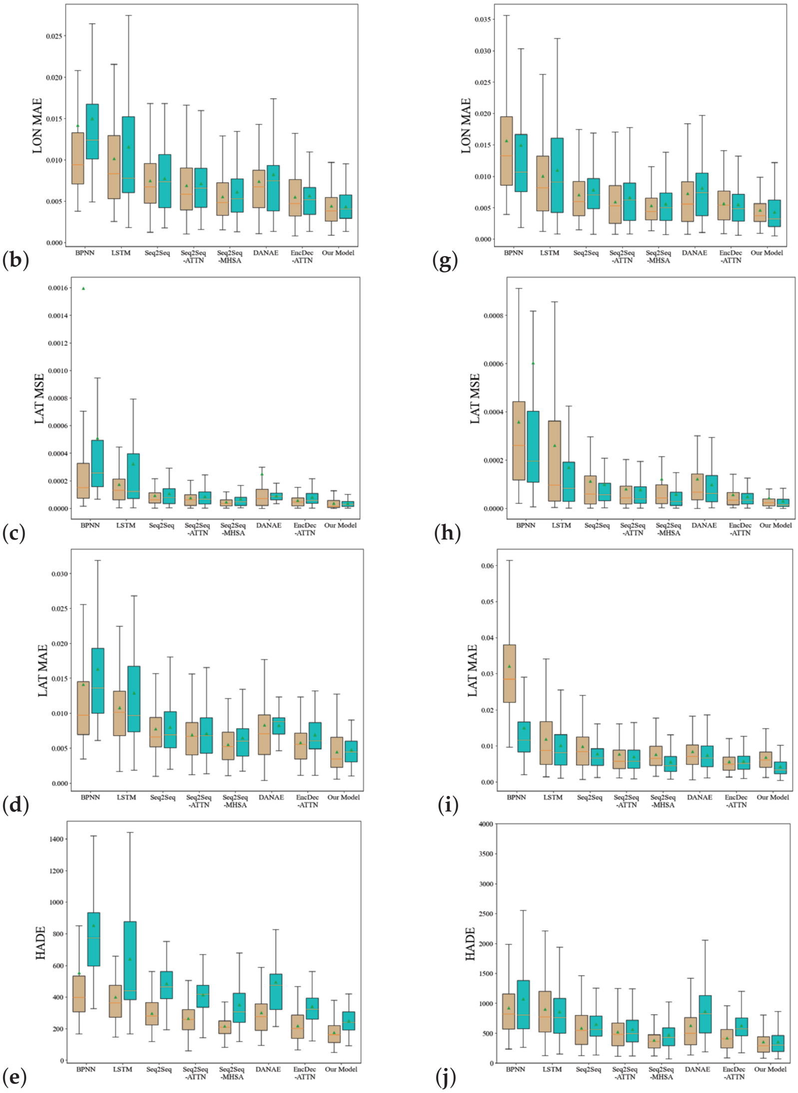


Figure 15. *Cont.*



Two test samples of the Yangtze River Two test samples of the eastern coastal area

Figure 15. Comparison analysis of the MSE, MAE, and HADE of the test samples from the different areas. Brown and blue-green boxplots represent the evaluation indicators for test samples 1 and 2, respectively. The green dots are the average value. Figure (**a**–**e**) and (**f**–**j**) represent the Yangtze River and eastern coastal area, respectively.

5. Conclusions

To predict the future trajectory of a ship in the case of encounter situations, a high-precision trajectory prediction model based on AIS navigation-history data was proposed. This method uses an LSTM neural-network model to encode and decode trajectory information from sequences. The framework of the proposed model uses the relative navigation state information of the encounter and observation ships as part of the input state information characteristics, to predict the observation ship trajectory. Compared with classical models, the proposed model has a stronger generalizability and better performance. Experiments showed that the attention-based model could effectively capture the potential information and correlations in a ship position sequence, so that the proposed model had a better prediction ability for the curve trajectory segment. This significantly improved on the performance of the existing models, which have strong advantages in trajectory prediction. It provides an effective safeguard for ship intelligent navigation systems, by providing real-time trajectory prediction and developing safe and efficient decision support.

In future work, we plan to consider more influencing factors around ships, in the case of multiple ship collisions, and provide further decision support for our model, for more cases of ship collision avoidance.

Author Contributions: L.Z.: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing-original draft, Visualization; Y.Z.: Conceptualization, Methodology, Validation, Formal analysis, Investigation, Resources, Data curation, Writing-original draft, Writing-review & editing, Visualization, Supervision, Project administrator, Funding Acquisition; T.L.: Conceptualization, Resources, Writing-review & editing, Project administrator, Funding Acquisition; C.L.P.C.: Conceptualization, Writing-review & editing, Supervision, Project administrator, Funding Acquisition.

Funding: This work was supported in part by the National Natural Science Foundation of China (grant nos. 52131101 and 51939001), the Liao Ning Revitalization Talents Program (grant no. XLYC1807046), and the Science and Technology Fund for Distinguished Young Scholars of Dalian (grant no. 2021RJ08).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data and model code of this study can be obtained from: https://github.com/zlingcheng2023/Ship-trajectory-prediction.git (accessed on 2 July 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mou, J.M.; van der Tak, C.; Ligteringen, H. Study on collision avoidance in busy waterways by using AIS data. *Ocean Eng.* **2010**, *37*, 483–490. [CrossRef]
2. Tu, E.; Zhang, G.; Rachmawati, L.; Rajabally, E.; Huang, G.B. Exploiting AIS Data for Intelligent Maritime Navigation: A Comprehensive Survey From Data to Methodology. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 1559–1582. [CrossRef]
3. Zhu, Y.; Zuo, Y.; Li, T. Modeling of Ship Fuel Consumption Based on Multisource and Heterogeneous Data: Case Study of Passenger Ship. *J. Mar. Sci. Eng.* **2021**, *9*, 273. [CrossRef]
4. Li, X.; Zuo, Y.; Jiang, J. Application of Regression Analysis Using Broad Learning System for Time-Series Forecast of Ship Fuel Consumption. *Sustainability* **2023**, *15*, 380. [CrossRef]
5. Cockcroft, A.; Lameijer, J. Part B—Steering and sailing rules. In *A Guide to the Collision Avoidance Rules*, 7th ed.; Cockcroft, A., Lameijer, J., Eds.; Butterworth-Heinemann: Oxford, UK, 2012; pp. 11–104.
6. Huang, Y.; van Gelder, P.; Wen, Y. Velocity obstacle algorithms for collision prevention at sea. *Ocean Eng.* **2018**, *151*, 308–321. [CrossRef]
7. Lehtola, V.; Montewka, J.; Goerlandt, F.; Guinness, R.; Lensu, M. Finding safe and efficient shipping routes in ice-covered waters: A framework and a model. *Cold Reg. Sci. Tech.* **2019**, *165*, 102795. [CrossRef]
8. Park, S.W.; Park, Y.S. Predicting Dangerous Traffic Intervals between Ships in Vessel Traffic Service Areas Using a Poisson Distribution. *J. Korean Soc. Mar. Environ. Saf.* **2016**, *22*, 402–409. [CrossRef]
9. Ristic, B.; La Scala, B.; Morelande, M.; Gordon, N. Statistical analysis of motion patterns in AIS Data: Anomaly detection and motion prediction. In Proceedings of the 2008 11th International Conference on Information Fusion, Cologne, Germany, 30 June–3 July 2008; pp. 1–7.
10. Rong Li, X.; Jilkov, V. Survey of maneuvering target tracking. Part I. Dynamic models. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1333–1364. [CrossRef]
11. Mazzarella, F.; Arguedas, V.F.; Vespe, M. Knowledge-based vessel position prediction using historical AIS data. In Proceedings of the 2015 Sensor Data Fusion: Trends, Solutions, Applications (SDF), Bonn, Germany, 6–8 October 2015; pp. 1–6.
12. Ma, X.; Liu, G.; He, B.; Zhang, K.; Zhang, X.; Zhao, X. Trajectory Prediction Algorithm Based on Variational Bayes. In Proceedings of the 2018 IEEE CSAA Guidance, Navigation and Control Conference (CGNCC), Xiamen, China, 10–12 August 2018; pp. 1–6.
13. Rong, H.; Teixeira, A.; Guedes Soares, C. Ship trajectory uncertainty prediction based on a Gaussian Process model. *Ocean Eng.* **2019**, *182*, 499–511. [CrossRef]

14. Barrios, C.; Motai, Y. Improving Estimation of Vehicle's Trajectory Using the Latest Global Positioning System With Kalman Filtering. *IEEE Trans. Instrum. Meas.* **2011**, *60*, 3747–3755. [CrossRef]
15. Perera, L.P.; Oliveira, P.; Guedes Soares, C. Maritime Traffic Monitoring Based on Vessel Detection, Tracking, State Estimation, and Trajectory Prediction. *IEEE Trans. Intell. Transp. Syst.* **2012**, *13*, 1188–1200. [CrossRef]
16. Hexeberg, S.; Flåten, A.L.; Eriksen, B.O.H.; Brekke, E.F. AIS-based vessel trajectory prediction. In Proceedings of the 2017 20th International Conference on Information Fusion (Fusion), Xi'an, China, 10–13 July 2017; pp. 1–8.
17. Guo, S.; Liu, C.; Guo, Z.; Feng, Y.; Hong, F.; Huang, H. Trajectory Prediction for Ocean Vessels Base on K-order Multivariate Markov Chain. In *Proceedings of the Wireless Algorithms, Systems, and Applications*; Chellappan, S., Cheng, W., Li, W., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 140–150.
18. Gao, D.W.; Zhu, Y.S.; Zhang, J.F.; He, Y.K.; Yan, K.; Yan, B.R. A novel MP-LSTM method for ship trajectory prediction based on AIS data. *Ocean Eng.* **2021**, *228*, 108956. [CrossRef]
19. Liu, J.; Shi, G.; Zhu, K. Online Multiple Outputs Least-Squares Support Vector Regression Model of Ship Trajectory Prediction Based on Automatic Information System Data and Selection Mechanism. *IEEE Access* **2020**, *8*, 154727–154745. [CrossRef]
20. Murray, B.; Perera, L.P. A Data-Driven Approach to Vessel Trajectory Prediction for Safe Autonomous Ship Operations. In Proceedings of the 2018 Thirteenth International Conference on Digital Information Management (ICDIM), Berlin, Germany, 24–26 September 2018; pp. 240–247.
21. Valsamis, A.; Tserpes, K.; Zissis, D.; Anagnostopoulos, D.; Varvarigou, T. Employing traditional machine learning algorithms for big data streams analysis: The case of object trajectory prediction. *J. Syst. Softw.* **2017**, *127*, 249–257. [CrossRef]
22. Zhang, Z.; Ni, G.; Xu, Y. Trajectory prediction based on AIS and BP neural network. In Proceedings of the 2020 IEEE 9th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), Chongqing, China, 11–13 December 2020; Volume 9, pp. 601–605.
23. Zhao, Y.; Cui, J.; Yao, G. Online Learning based GA-BP Neural Network to Predict Ship Trajectory. In Proceedings of the 2021 China Automation Congress (CAC), Beijing, China, 22–24 October 2021; pp. 3731–3735.
24. Tang, H.; Yin, Y.; Shen, H. A model for vessel trajectory prediction based on long short-term memory neural network. *J. Mar. Eng. Technol.* **2022**, *21*, 136–145. [CrossRef]
25. Qian, L.; Zheng, Y.; Li, L.; Ma, Y.; Zhou, C.; Zhang, D. A New Method of Inland Water Ship Trajectory Prediction Based on Long Short-Term Memory Network Optimized by Genetic Algorithm. *Appl. Sci.* **2022**, *12*, 4073. [CrossRef]
26. Ma, H.; Zuo, Y.; Li, T. Vessel Navigation Behavior Analysis and Multiple-Trajectory Prediction Model Based on AIS Data. *J. Adv. Transp.* **2022**, *2022*, 6622862. [CrossRef]
27. Zhang, X.; Fu, X.; Xiao, Z.; Xu, H.; Qin, Z. Vessel Trajectory Prediction in Maritime Transportation: Current Approaches and Beyond. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 19980–19998. [CrossRef]
28. Capobianco, S.; Millefiori, L.M.; Forti, N.; Braca, P.; Willett, P. Deep Learning Methods for Vessel Trajectory Prediction Based on Recurrent Neural Networks. *IEEE Trans. Aerosp. Electron. Syst.* **2021**, *57*, 4329–4346. [CrossRef]
29. Donandt, K.; Böttger, K.; Söffker, D. Short-term Inland Vessel Trajectory Prediction with Encoder-Decoder Models. In Proceedings of the 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), Macau, China, 8–12 October 2022; pp. 974–979.
30. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to Sequence Learning with Neural Networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2014; NIPS'14, pp. 3104–3112.
31. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All You Need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*; Curran Associates Inc.: Red Hook, NY, USA, 2017; NIPS'17, pp. 6000–6010.
32. Hinton, G.E.; Salakhutdinov, R.R. Reducing the Dimensionality of Data with Neural Networks. *Science* **2006**, *313*, 504–507. [CrossRef]
33. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
34. Chiu, C.C.; Sainath, T.N.; Wu, Y.; Prabhavalkar, R.; Nguyen, P.; Chen, Z.; Kannan, A.; Weiss, R.J.; Rao, K.; Gonina, E.; et al. State-of-the-Art Speech Recognition with Sequence-to-Sequence Models. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 4774–4778.
35. Forti, N.; Millefiori, L.M.; Braca, P.; Willett, P. Prediction oof Vessel Trajectories From AIS Data Via Sequence-To-Sequence Recurrent Neural Networks. In Proceedings of the 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 8936–8940.
36. Bahdanau, D.; Cho, K.; Bengio, Y. Neural Machine Translation by Jointly Learning to Align and Translate. In Proceedings of the 2015 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015.
37. Luong, T.; Pham, H.; Manning, C.D. Effective Approaches to Attention-based Neural Machine Translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*; Association for Computational Linguistics: Lisbon, Portugal, 2015; pp. 1412–1421.
38. Jiang, D.; Shi, G.; Li, N.; Ma, L.; Li, W.; Shi, J. TRFM-LS: Transformer-Based Deep Learning Method for Vessel Trajectory Prediction. *J. Mar. Sci. Eng.* **2023**, *11*, 880. [CrossRef]
39. Altan, D.; Marijan, D.; Kholodna, T. SafeWay: Improving the safety of autonomous waypoint detection in maritime using transformer and interpolation. *Marit. Transp. Res.* **2023**, *4*, 100086. [CrossRef]

40. Jiang, J.; Zuo, Y. Prediction of Ship Trajectory in Nearby Port Waters Based on Attention Mechanism Model. *Sustainability* **2023**, *15*, 7435. [CrossRef]
41. Capobianco, S.; Forti, N.; Millefiori, L.M.; Braca, P.; Willett, P. Recurrent Encoder—Decoder Networks for Vessel Trajectory Prediction With Uncertainty Estimation. *IEEE Trans. Aerosp. Electron. Syst.* **2023**, *59*, 2554–2565. [CrossRef]
42. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. In Proceedings of the 2015 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015.
43. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
44. Vincent, P.; Larochelle, H.; Bengio, Y.; Manzagol, P.A. Extracting and Composing Robust Features with Denoising Autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*; Association for Computing Machinery: New York, NY, USA, 2008; ICML'08, pp. 1096–1103.
45. Russo, P.; Ciaccio, F.D.; Troisi, S. DANAE: A denoising autoencoder for underwater attitude estimation. In Proceedings of the 2020 IMEKO TC-19 International Workshop on Metrology for the Sea, Naples, Italy, 5–7 October 2020; pp. 195–198.
46. Russo, P.; Di Ciaccio, F.; Troisi, S. DANAE++: A Smart Approach for Denoising Underwater Attitude Estimation. *Sensors* **2021**, *21*, 1526. [CrossRef]
47. Lee, J.; Shin, J.H.; Kim, J.S. Interactive Visualization and Manipulation of Attention-based Neural Machine Translation. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language: System Demonstrations; Association for Computational Linguistics: Copenhagen, Denmark, Copenhagen, Denmark, 9–11 September 2017; pp. 121–126.

Journal of
*Marine Science
and Engineering*

Article

Multi-Feature Fusion-Guided Low-Visibility Image Enhancement for Maritime Surveillance

Wenbo Zhou [1,2,*], Bin Li [3] and Guoling Luo [2]

[1] School of Electronic and Information Engineering, South China University of Technology, Guangzhou 510641, China
[2] Zhuhai Metamemory Electronic Technology Co., Ltd., Zhuhai 519090, China; luo.guoling@zxelec.com
[3] School of Microelectronics, South China University of Technology, Guangzhou 510641, China; phlibin@scut.edu.cn
* Correspondence: zhou.wenbo@zxelec.com

Abstract: Low-visibility maritime image enhancement is essential for maritime surveillance in extreme weathers. However, traditional methods merely optimize contrast while ignoring image features and color recovery, which leads to subpar enhancement outcomes. The majority of learning-based methods attempt to improve low-visibility images by only using local features extracted from convolutional layers, which significantly improves performance but still falls short of fully resolving these issues. Furthermore, the computational complexity is always sacrificed for larger receptive fields and better enhancement in CNN-based methods. In this paper, we propose a multiple-feature fusion-guided low-visibility enhancement network (MFF-Net) for real-time maritime surveillance, which extracts global and local features simultaneously to guide the reconstruction of the low-visibility image. The quantitative and visual experiments on both standard and maritime-related datasets demonstrate that our MFF-Net provides superior enhancement with noise reduction and color restoration, and has a fast computational speed. Furthermore, the object detection experiment indicates practical benefits for maritime surveillance.

Keywords: multiple feature fusion; convolutional neural network; attention mechanism; low-visibility image enhancement; maritime surveillance

Citation: Zhou, W.; Li, B.; Luo, G. Multi-Feature Fusion-Guided Low-Visibility Image Enhancement for Maritime Surveillance. *J. Mar. Sci. Eng.* **2023**, *11*, 1625. https://doi.org/10.3390/jmse11081625

Academic Editor: Mihalis Golias

Received: 17 July 2023
Revised: 13 August 2023
Accepted: 15 August 2023
Published: 20 August 2023

1. Introduction

With the growth of the Internet of Things and artificial intelligence, the perception efficiency of maritime sensors has been employed for different tasks in ocean engineering, e.g., vessel trajectory prediction [1,2] and maritime surveillance [3]. In particular, visual sensors are widely used because of their unique intuitiveness and high timeliness [4]. However, imaging devices working in extremely low-visibility conditions, typically low-light and hazy, will generate images with severe distortion [5,6], which constantly suffer from low contrast, non-uniform noise, and details lost, as shown in Figure 1. Undoubtedly, the negative impact of low visibility will make it tricky to analyze critical information in the image, which brings difficulty in subsequent tasks [7]. For instance, it has been proven that low visibility will reduce the precision of object detection [8–10], image semantic segmentation [11,12], etc. Therefore, an effective and real-time method for low-visibility image enhancement is required in various domains, such as visual navigation [13], maritime management [14], etc.

Many academics have attempted to improve extremely low-visibility photos with both hardware- and software-enabled methods during the past several decades. The former attempts to increase the robustness of the visual sensors by applying extra artificial light sources, such as infrared and ultraviolet flashes [15], while the latter is more popular [16] due to the relatively low cost. Specifically, some traditional software-enabled methods have tried to employ some physical model and prior knowledge [15,17], which successfully enhanced the visibility but caused severe detail lost and failed to effectively overcome the noise interference. The convolutional neural network (CNN) has become increasingly popular in recent years for enhancement tasks [18]. The learnable convolutional kernel parameters enable CNNs to simultaneously eliminate noise interference [19]. However, the features extracted by convolutional layers are local, which works ineffectively for some non-uniform illumination patches, and the translation invariance of the CNN is incompatible to the non-linear relationship between the object and the background, which causes vague edges in enhanced images. Furthermore, to improve the receptive view of the convolutional kernel for better feature extraction, the computational complexity gradually increases with the deepening of the network structure [20].

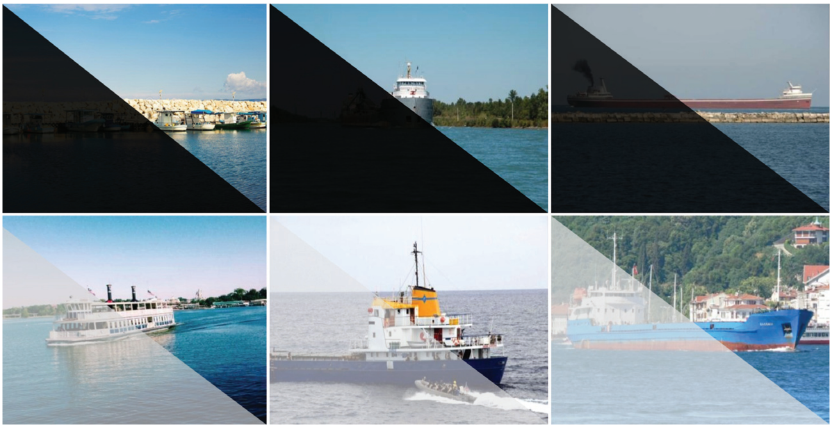


Figure 1. Examples of the comparison between maritime low-visibility images and clear images.

1.1. Motivation

For the convolutional layers, the critical mechanism is to learn a convolution kernel with fixed parameters and perform the same transformation process on the entire feature map. The size and stride settings of the convolution kernels only change the scope of action. The translation invariance is an important feature of the convolutional layer, which also makes it difficult for the CNN to extract the non-local features [21].

Meanwhile, the spatial attention mechanism is widely employed in computer vision tasks [22]. Unlike convolution, the receptive field of the spatial mechanism is larger and more diverse, which can extract the global features from the feature map and overcomes the limitation of the local features. However, compared with words in passages of text, the resolution of pixels is much higher, which requires more parameters to learn. In 2019, Huang et al. [23] proposed the criss-cross attention mechanism, which extracts the contextual information from full-image dependencies with competitive computational efficiency.

To let the comprehensive information guide the enhancement processing, we propose the multi-feature fusion-guided network. Specifically, inspired by [23], we employ the densely connected convolution layers and the cross attention module for local and global feature extraction and fuse them to form the general feature map, which helps the network enhance the low-visibility image with more detail preservation and better color recovery.

1.2. Contribution

In this paper, we present a multi-feature fusion-guided low-visibility image enhancement network (MFF-Net) for maritime surveillance advancement. It achieves a superior balance between the enhancement effect and computational time. The main contributions of our method are summarized as follows:

- We propose a multiple feature fusion-guided low-visibility image enhancement method for maritime surveillance advancement. It extracts the features of the image and reconstruct it with the supervision of a joint loss function to calculate both the Euclidean distance and angle difference between the output and the ground truth. The proposed network tackles two typical low-visibility problems, i.e., low-light and hazy, with the same framework.

- To overcome the limitation caused by the translation invariance of the CNN, we design a two-branched global and local features extraction block (GL-Block) comprising cross attention modules and densely residual convolutional layers. The output feature maps are then fused to guide the enhancement processing with more comprehensive information.
- Extensive experimental results show that our MFF-Net enhances both low-light and hazy maritime images with significant noise reduction and detail preservation, which outperforms other competitive methods. Furthermore, we evaluate the computational complexity of the MFF-Net. The results indicate an outperforming balance between the effect and the speed.

1.3. Organization

The rest of this paper is organized as follows: Section 2 reviews previous research on low-visibility image enhancement tasks. Section 3 introduces the proposed method and the detailed design of each component. Section 4 presents the experimental results compared with state-of-the-art methods on both enhancement performance and the running time cost. In addition, the ablation study investigates the necessity of the multi-feature fusion guidance for low-visibility image enhancement and the rationality about the weight settings of the joint loss function. The experiment on vessel detection demonstrates the practical benefits of our method. Section 5 summarizes the content of the paper and discusses future work.

2. Related Work

Low-light and haze are the most common low-visibility weathers in maritime surveillance. Many research studies have been proposed to over come these problems [24]. In this section, we briefly review the related works about low-light image enhancement and image dehazing, which can be generally classified into traditional and deep learning-based method.

2.1. Low-Light Image Enhancement Methods

Low-light image enhancement methods can generally be divided to mathematical model- and deep learning-based methods. Mathematical model-based methods include some famous theories such as histogram equalization (HE) [17], gamma correction (GC) [25], Retinex theory [26], and so on. HE firstly attempts to enhance the image with the most frequent intensity values uniformly. However, in practical applications, HE and its variants [27,28] are severely hampered by non-uniform noise. GC tries to increase the intensity of each pixel with an exponential function, which is also effective for contrast enhancement. However, it ignores adjacent pixels' correlation, resulting in artifacts and enlarged noise. Retinex theory is based on the retinal-imaging concept that decomposes images into illumination and reflection maps. It was first utilized in 1997 to lighten low-light images [26,29]. Many Retinex-based methods were proposed in subsequent years. For instance, Wang et al. [30] proposed a specially designed enhancement method for non-uniform illumination, and Guo et al. [31] proposed low-light image enhancement via illumination map estimation (LIME), which achieved competitive performance in low-light image enhancement. However, mathematical model-based methods generally use some specific functions to estimate the noise and illumination, which is non-uniform and difficult to express as a specific equation. Therefore, the results always suffer from severe color distortion. Noise interference is also a thorny problem for traditional mathematical model-based methods.

With the rapid advancement of computing devices, deep learning-based methods have produced outstanding results in low-light image enhancement. In 2018, Chen et al. proposed an end-to-end network trained using extremely low-light raw sensor data [32], which demonstrated the superior performance of the neural network in low-light image enhancement tasks. In the following years, a number of works were published on low-light enhancement [33,34]. For instance, KinD [35] proposed a CNN based on Retinex theory, which successfully correlated the mathematical model and neural network. Zero [36] formulated light enhancement as a task of image-specific curve estimation, which enhanced the low-light images with a lightweight neural network. Hap et al. [37] proposed a low-light image enhancement method, which decouples the model into two sequential stages to improve the scene visibility and suppress the rest degeneration factors separately. Guo et al. [38] designed a multi-scale deep stacking fusion enhancer to lighten the darkness in an intelligent transportation system. LLFlow [39] proposed a normalizing flow model to establish the relationship between the single low-light images to different normally exposed images. However, most deep learning methods suffer from several thorny problems like color distortion and detail lost, which are difficult to solve simultaneously by a lightweight CNN.

2.2. Image Dehazing Method

Image dehazing methods can be generally divided into prior- and deep-learning based methods. Prior-based dehazing methods exploit the statistical properties of clean images to estimate transmission maps, and then predict the haze-free image using the scattering model, which can be expressed as

$$H(x) = J(x) * t(x) + A * (1 - t(x)), \tag{1}$$

where $H(x)$ is the hazy image, $t(x)$ is the transmittance, and A is the atmospheric light intensity.

To acquire prior knowledge, early works attempted to concentrate on statistic analysis or observation of the haze-free images. Among them, He et al. [40] proposed the dark channel prior (DCP), which detects the haze distribution of hazy images by assuming that the lowest local intensity in the RGB channels are close to zero in a clear image. Zhu et al. [41] introduced the color attenuation prior, which supposes that in a linear model, the difference between the saturation and the pixel values are positively correlated with the depth of the scene. Although these methods have achieved certain dehazing effects, they are based on artificially constructed prior models, which cannot fully describe the real haze image. Therefore, these methods are highly restricted by the scene and have insufficient generalization ability.

The method based on deep learning also has a large application in dehazing. Cai et al. [42] proposed an end-to-end-based DehazeNet, which estimates the transmission map from a hazy image. Tang et al. [43] proposed a multi-scale network to exploit multi-scale information, which predicts the transmission by a coarse-scale net and a fine-scale one. Chen et al. [44] proposed a gated context aggregation network (GCANet), which employs a smooth dilated convolution to reduce the gridding artifacts led by the dilation technique. However, the image enhanced by GCANet still has unevenly distributed haze. However, these methods cannot recover the details of the image. Therefore, Qin et al. [45] further employed the application of the attention mechanism in dehazing work, which exploits a feature attention module that fuses the features with pixel and channel attention. Guo et al. [46] proposed a self-paced semi-curricular attention network to overcome the non-uniform distribution features of the hazy images.

3. Proposed Method

In practical applications, low-visibility weathers always bring challenges in traffic observation and navigational environment perception. An effective and efficient low-visibility enhancement method is beneficial for maritime surveillance. In this section, we introduce our method in detail. For a better understanding, Table 1 lists the main symbols adopted in this work.

Table 1. Summary of key notations.

Notation	Description
M	The feature map generated by neural networks
$M_{i,\_}$	The *i*-th row vectors of the feature map
$M_{\_,j}$	The *j*-th column vectors of the feature map
$M_{i,j}$	The vector at position $(i,\ j)$ of the feature map
λ	The weight of the loss function
$\mathcal{L}$	The loss function
P	The pixel of the output image
$\hat{P}$	The pixel of the ground truth image
h	The height of the image
w	The width of the image
c	The channel number of the image

3.1. Architecture

The overview of the network is presented in Figure 2. To reduce the computational complexity, we use 1 × 1 convolutional and max-pooling layers to downsample the low-visibility image. For feature extraction, we propose the GL-Block consisting of convolutional layers and cross attention modules. In the end, 1 × 1 convolutional and bilinear upsampling layers transform the output image to the corresponding fine scale.

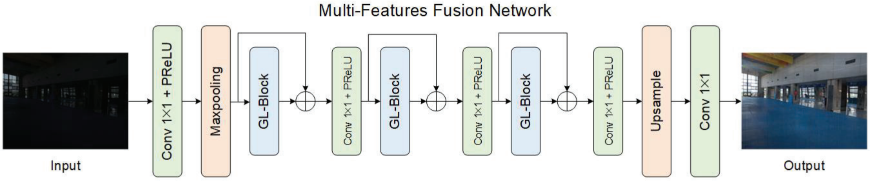


Figure 2. Flowchart of the proposed MFF-Net. Firstly, the low-visibility image is downsampled with a max-pooling layer. The multiple features are then extracted with three GL-Blocks to guide the enhancement process. The enhanced image is finally upsampled to the original scale.

3.2. GL-Block

We design a two-branched block to extract multiple features simultaneously. Firstly we employ cross attention modules [23] to extract global features, which collect global information in the horizontal and vertical directions to enhances the representative capability of each pixel, as shown in Figure 3. Specifically, 1 × 1 convolutional layers are used to obtain the query (Q), key (K), and value (V) matrix and generate the attention map M with an affinity operation. Unlike the common attention method, GL-Block achieves global spatial information interaction with two cross attention modules, which sufficiently reduces the computational complexity. The contextual information collected by the cross attention module can be expressed as

$$M_{i,j} = f(M^{'}_{i,j}, M^{'}_{i,\_}, M^{'}_{\_,j}), \tag{2}$$

where $M'_{i,j}$ represents each vector in the input feature maps, $M^{'}_{i,\_}$ and $M^{'}_{\_,j}$ represent the horizontal and vertical vectors, respectively, and f denotes the process of establishing the connection between each pixel. However, the cross attention mechanism will cause a black-line problem due to the extremely dark or bright pixels, as discussed in Section 4.6. Therefore, to balance the extreme non-uniformity, we optimize the cross attention module with two subsequent dilated convolutional layers. The kernel size is set to three, and the dilation steps are set to four and six, respectively.

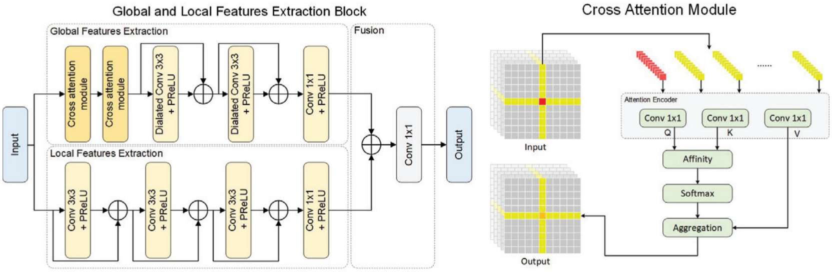


Figure 3. The detailed implementation of the GL-Block and cross attention module.

The other branch consists of several residual convolutional layers, designed to extract local features separately. In particular, inspired by [47], the convolutional layers are densely connected for better detail preservation. The kernel size and stride of convolutional layers are set to three and one, respectively. In the end, we merge the global and local feature maps and feed it into a 1 × 1 convolutional layer for feature fusion.

3.3. Loss Function

For the back-propagation process, we propose a joint loss function consisting of L1 loss $\mathcal{L}_1$, L2 loss $\mathcal{L}_2$, and color similarity loss $\mathcal{L}_{color}$ to supervise our network from both the Euclidean and angle difference. This can be defined as

$$\mathcal{L} = \lambda_1 \mathcal{L}_1 + \lambda_2 \mathcal{L}_2 + \lambda_3 \mathcal{L}_{color}, \tag{3}$$

where λ_1, λ_2, and λ_3 are the weights of each loss function.

L1 Loss. To ensure the quality of the generated images, we employ the widely used L1 loss function, which is based on the Euclidean distance between each pixel. It can be expressed as

$$\mathcal{L}_1 = \frac{1}{hwc} \sum_{i,k} \left\| P_i^k - \hat{P}_i^k \right\|, \tag{4}$$

where P_i^k and $\hat{P}_i^k$ are the pixels of the output images and ground truth, respectively. i and k represent the positions and channels, respectively. h, w, and c denote the height, width and the number of channels, respectively.

L2 Loss. Besides L1, the L2 loss function is also widely used in low-level computer vision tasks for the effective restriction on the output image, which can be expressed as

$$\mathcal{L}_2 = \frac{1}{hwc} \sum_{i,k} \| P_i^k - \hat{P}_i^k \|_2. \tag{5}$$

Color Similarity Loss. In RGB images, the Euclidean distance is a typical evaluation metrics to validate the similarity between two pixels, However, it ignores the angle difference between two RGB vectors, which also causes severe color differences between two pixels. To measure the deviation more comprehensively, we employ the cosine similarity between each vector as the color similarity loss to take the angle difference into consideration. The color loss function can be expressed as

$$\mathcal{L}_{color} = 1 - \frac{1}{hw} \cdot \sum_i \frac{P_i \cdot \hat{P}_i}{\max(\|P_i\|_2 \cdot \|\hat{P}_i\|_2, \epsilon)}, \tag{6}$$

where the cosine value of the angle between the RGB vectors P_i and $\hat{P}_i$ is calculated, which represents the angle differences of the pixel at the position i. ϵ = 0.001 is a hyper-parameter used to avoid zero becoming the denominator.

4. Experiments

In this section, we firstly describe the dataset and the implementation details used in the experiment. To comprehensively evaluate the performance of the MFF-Net, different aspects of the state-of-the-art methods and our model are compared including GT reference, noise reduction, color naturalness, and computational complexity. The ablation studies concerning the necessity of multiple feature fusion and the weight settings of the joint loss function are presented to demonstrate the rationality of the proposed method. Finally, to verify the practical benefits of the proposed method, we construct the vessel detection experiments on the enhanced images.

4.1. Dataset and Evaluation Indicators

Supervised learning requires the perfectly paired dataset to calculate the pixel difference between the output and ground truth. However, the current publicly available paired datasets (LOL [48], EnlightenGAN [49], I-HAZE [50], SMOKE [51], etc.) are not suitable for maritime low-visbility image enhancement[1], and the paired maritime low-visibility image dataset is difficult to obtain. We thus synthesize a large number of marine low-visibility images based on the Seaships dataset. Specifically, we select 1500 high-quality images from the Seaships dataset for training and 30 images for testing, as shown in Figure 4. It is noted that the characters in the image are the timestamps and locations of the camera, which is contained in the original dataset. In low-light image enhancement tasks, we also adopt traditional methods to synthesize low-light maritime images, which can be expressed as

$$L_{\text{maritime}}(x) = J(x) \times g(x), \tag{7}$$

where $L_{\text{maritime}}(x)$ is the low-light maritime image, $J(x)$ is the clear image, and $g(x)$ is the coefficient, which is a random number between 0.1 and 0.8. Meanwhile, we exploit Equation (1) to obtain synthetic training hazy data. We restrict $t(x)$ from 0.1 to 0.7, and set A from 0.2 to 0.8. For the test data, we synthesized three types of low-light images with different light levels, i.e., $g_1(x)$ = 0.2, $g_2(x)$ = 0.4, and $g_3(x)$ = 0.6 (termed Test-L). Similarly, we also synthesized three types of images using Equation (1) with different degrees of degradation, i.e., $t_1 = 0.4/A_1 = 0.9$, $t_2 = 0.2/A_2 = 0.8$, and $t_3 = 0.2/A_3 = 0.7$ (termed Test-H).

For the supervised neural network, the results closer to the ground truth represent a better performance. Therefore, for quantitative image quality assessment comparisons, we choose five reference evaluation indicators, i.e., peak signal-to-noise ratio (PSNR), structural similarity (SSIM) [52], feature similarity index measure (FSIM) [53], and visual saliency-induced index (VSI) [54] to evaluate the enhancement performance. It is noted that a higher PSNR, SSIM, FSIM, VSI represent better image quality and a closer proximity to the ground truth.

4.2. Implementation Details

We use Pytorch to build and train the MFF-Net. The network is trained for 300 epochs, and the ADAM optimizer is employed during training. The starting learning rate is set to 1×10^{-3} and is multiplied by 0.1 after every 100 epochs. In the loss function, to equally employ the Euclidean distance and the angle difference as the restraint, the weights of $\mathcal{L}_1$, $\mathcal{L}_2$, and $\mathcal{L}_{color}$ are set to 0.25, 0.25, and 0.5, respectively. For data augmentation, we randomly crop the 600 × 400 images to patches of size 128 × 128 for training and the original size for testing, and the running time costs are calculated on a laptop with an AMD Ryzen 7 5800H CPU accelerated by an NVIDIA GTX 3060 GPU. For a fair comparison, the parameters of all competing methods are from the open access checkpoints by the authors.

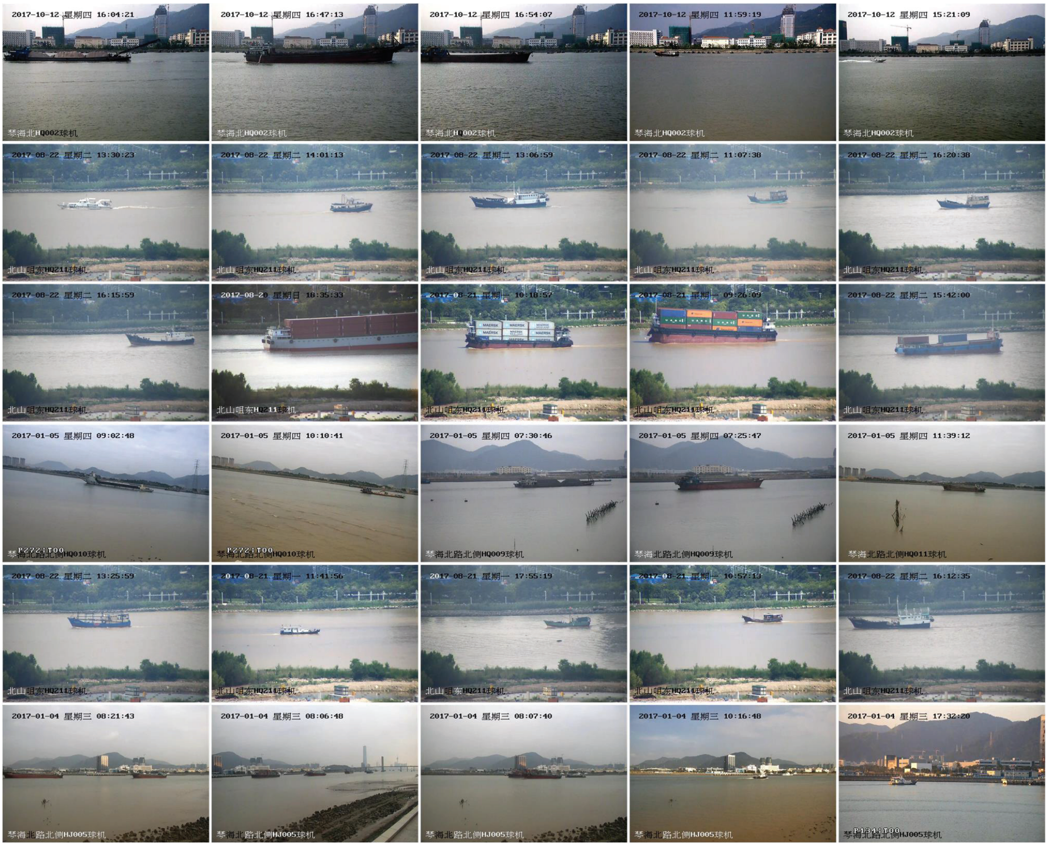


Figure 4. Thirty selected maritime images from the Seaships [55] dataset, which contains raw maritime surveillance data captured in different scenes.

4.3. Experiments on Maritime Low-Light Images

To verify the superior performance of our method, we select some competitive classical algorithms and state-of-the-art methods to compare: (a) traditional mathematical model methods, including HE [17], NPE [30], BCP [56], SRIE [57], and LIME [31]; (b) deep learning-based methods, including RetinexNet [48], LightenNet [58], MBLLEN [59], KinD [35], Zero [36], and StableLLVE [18]. The visual comparisons on Test-L are shown in Figure 5. In terms of mathematical model-based methods, the results of HE and BCP have obvious color distortion, some non-uniform artifact exists in the results enhanced by NPE, and LIME fails to lighten the low-light images effectively. In addition, for deep learning-based methods, RetinexNet suffers from severe color distortion, KinD only enhances the image with local features, which is incompatible with the illumination diversity between the non-adjacent patches, Zero sacrifices the enhancement effect for fast speed, making the results look a little dark, and StableLLVE fails to enlighten the extremely dark regions. Compared with these methods, our results look more natural with better recovery. As shown in Table 2, the quantitative experiment result indicates that our method achieves a competitive performance on the whole. Although not the best in terms of certain metrics, our proposed method has substantial advantages in running speed, as discussed in Section 4.5.

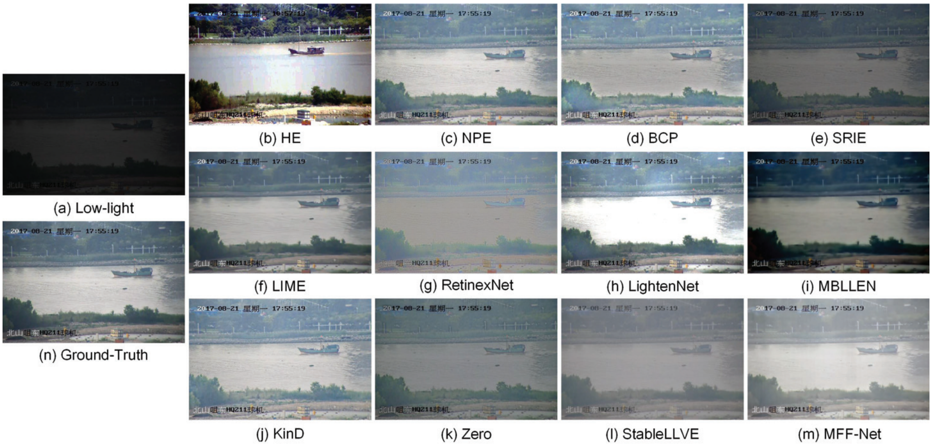


Figure 5. Visual comparison of the synthetic low-light images from the Seaships [55] dataset with other competitive methods: (**a**) low-light; (**b**) HE [17]; (**c**) NPE [30]; (**d**) BCP [56]; (**e**) SRIE [57]; (**f**) LIME [31]; (**g**) RetinexNet [48]; (**h**) LightenNet [58]; (**i**) MBLLEN [59]; (**j**) KinD [35]; (**k**) Zero [36]; (**l**) StableLLVE [18]; (**m**) MFF-Net; (**n**) ground truth.

Table 2. Quantitative comparison between our method and the state-of-the-art methods on the 90 maritime low-light images. The top three results are marked in red, blue, and green colors, respectively. The ↑ represents that the higher value means better result.

Methods	PSNR ↑	SSIM ↑	FSIM ↑	VSI ↑
HE [17]	18.011 ± 1.838	0.770 ± 0.073	0.896 ± 0.031	0.961 ± 0.013
NPE [30]	21.839 ± 2.936	0.946 ± 0.023	0.977 ± 0.008	0.992 ± 0.003
BCP [56]	16.577 ± 1.857	0.844 ± 0.051	0.929 ± 0.026	0.969 ± 0.016
SRIE [57]	18.044 ± 4.432	0.889 ± 0.074	0.946 ± 0.034	0.986 ± 0.008
LIME [31]	20.597 ± 3.719	0.945 ± 0.026	0.981 ± 0.008	0.993 ± 0.004
RetinexNet [48]	14.927 ± 1.927	0.691 ± 0.093	0.815 ± 0.049	0.943 ± 0.019
LightenNet [58]	11.985 ± 2.702	0.788 ± 0.072	0.916 ± 0.028	0.972 ± 0.008
MBLLEN [59]	13.721 ± 3.065	0.739 ± 0.064	0.950 ± 0.011	0.984 ± 0.004
KinD [35]	17.037 ± 1.413	0.906 ± 0.031	0.954 ± 0.015	0.984 ± 0.009
Zero [36]	17.074 ± 1.421	0.838 ± 0.036	0.891 ± 0.029	0.964 ± 0.017
StableLLVE [18]	14.853 ± 2.921	0.797 ± 0.044	0.898 ± 0.042	0.974 ± 0.011
MFF-Net	23.666 ± 4.222	0.941 ± 0.024	0.978 ± 0.009	0.994 ± 0.003

4.4. Experiments on Maritime Hazy Images

To demonstrate the dehazing ability of MFF-Net, we also select some classical traditional enhancement methods, including DCP [40], CAP [41], HL [60], F-LDCP [61], and GRM [62], and some state-of-the-art learning-based methods, including DehazeNet [42], MSCNN [63], AOD-Net [64], GCANet [44], HTDNet [65], and FFANet [45], for testing. As shown in Figure 6, the image dehazed by DCP suffers serious noise interference, especially on the water surface. Meanwhile, color distortion occurs in some areas. CAP and F-LDCP fail to dehaze the images thoroughly, resulting in the overall image being covered by a layer of haze. In contrast, HL dehazes the images better, but the enhanced image is too bright with serious noise interference. Furthermore, on the edge of the vessel and some of the water surface, the reflection phenomenon seriously influences the visual feeling, which brings a barrier to the lookout. On the whole, the learning-based methods achieve better performance than the traditional methods. However, DehazeNet, MSCNN, and AOD-Net still cannot dehaze the images thoroughly. Furthermore, although GCANet and FFA-Net can remove most of the haze, the images are still riddled with artifacts. The quantitative results are shown in Table 3. Compared with other methods, MFF-Net can successfully dehaze the images with a good balance between color restoration and detail preservation, benefiting from the strong learning ability of the CNN and the multi-feature fusion strategy. The expoeriments on the real captued images are shown in Figure 7. It can be seen that our method is effective for both real-captured low-light and hazy images.

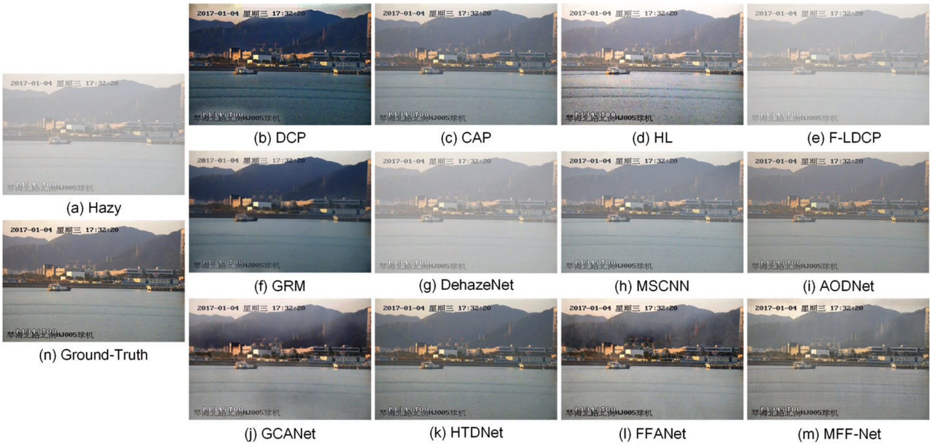

(a) Hazy (b) DCP (c) CAP (d) HL (e) F-LDCP (f) GRM (g) DehazeNet (h) MSCNN (i) AODNet (j) GCANet (k) HTDNet (l) FFANet (m) MFF-Net (n) Ground-Truth

Figure 6. Visual comparison on the synthetic hazy image from the Seaships [55] dataset with other competitive methods: (**a**) hazy; (**b**) DCP [40]; (**c**) CAP [41]; (**d**) HL [60]; (**e**) F-LDCP [61]; (**f**) GRM [62]; (**g**) DehazeNet [42]; (**h**) MSCNN [63]; (**i**) AODNet [64]; (**j**) GCANet [44]; (**k**) HTDNet [65]; (**l**) FFANet [45]; (**m**) MFF-Net; (**n**) ground truth.

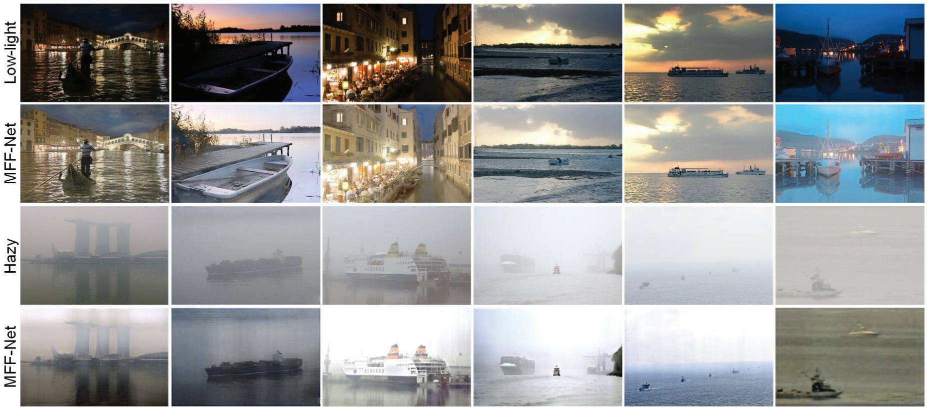


Figure 7. Visual performance on the physically captured low-visibility images. The first row contains the low-light images extracted from the TMDIED [66] dataset, and the third row contains the hazy images extracted from SMD [67] and online websites. The corresponding enhanced results of our MFF-Net are shown in the second and fourth rows.

4.5. Computational Complexity Analysis

In the practical maritime surveillance, the visual enhancement methods must take the running time into account. To evaluate the performance on computational complexity, we provide the running time cost on both low-light image enhancement and dehazing. For low-light enhancement, as shown in Table 4, MFF-Net is able to enhance the 400 × 600 images at over 20 FPS with the acceleration of an NVIDIA RTX 3060 GPU, which is faster than most other methods. LightenNet [58] and Zero [36] are more lightweight, but the enhancement effect is much worse than ours. For the dehazing method, as shown in Table 5, our MFF-Net also outperforms most of the previous methods. AOD-Net is faster, but the effectiveness of dehazing is worse than ours. In general, MFF-Net achieves a superior balance between the enhancement effect and the running time cost compared with the other methods, as depicted in Figure 8.

Table 3. Quantitative comparison between our method and the state-of-the-art methods on the 90 maritime hazy images. The top three results are marked in red, blue, and green colors, respectively. The ↑ represents that the higher value means better result.

Methods	PSNR ↑	SSIM ↑	FSIM ↑	VSI ↑
DCP [15]	15.687 ± 2.437	0.823 ± 0.059	0.946 ± 0.019	0.976 ± 0.011
CAP [41]	19.218 ± 4.078	0.848 ± 0.089	0.906 ± 0.057	0.978 ± 0.013
HL [60]	21.440 ± 2.353	0.917 ± 0.027	0.962 ± 0.019	0.980 ± 0.013
F-LDCP [61]	12.318 ± 1.454	0.645 ± 0.099	0.781 ± 0.077	0.948 ± 0.020
GRM [62]	20.271 ± 2.236	0.815 ± 0.068	0.888 ± 0.042	0.968 ± 0.013
DehazeNet [42]	13.092 ± 1.456	0.683 ± 0.101	0.808 ± 0.076	0.955 ± 0.019
MSCNN [63]	17.084 ± 2.132	0.843 ± 0.075	0.915 ± 0.048	0.980 ± 0.011
AODNet [64]	17.722 ± 2.661	0.761 ± 0.105	0.819 ± 0.074	0.958 ± 0.020
GCANet [44]	19.437 ± 3.066	0.878 ± 0.043	0.952 ± 0.014	0.981 ± 0.007
FFANet [45]	19.918 ± 4.805	0.858 ± 0.076	0.939 ± 0.031	0.984 ± 0.009
MFF-Net	22.192 ± 2.021	0.912 ± 0.028	0.955 ± 0.013	0.988 ± 0.006

Table 4. Average running time cost (unit: seconds) and parameter comparison on low-light images with different resolutions (400 × 600, 480 × 640, and 768 × 1024) of the different methods.

Methods	Platform	Parameters (K)	400 × 600	480 × 640	768 × 1024
HE [17]	Matlab (CPU)	-	0.1089	0.1344	0.3234
NPE [30]	Matlab (CPU)	-	4.6228	5.7649	14.705
BCP [56]	Matlab (CPU)	-	0.7711	0.9936	2.2191
SRIE [57]	Matlab (CPU)	-	5.1873	8.8056	20.323
LIME [31]	Matlab (CPU)	-	8.2369	11.429	46.887
RetinexNet [48]	Python (GPU)	8536.7	0.0714	0.0922	0.2070
LightenNet [58]	Matlab (GPU)	-	0.0570	0.0640	0.2093
MBLLEN [59]	Python (GPU)	450.2	0.0870	0.1045	0.2493
KinD [35]	Python (GPU)	8017.1	0.1040	0.1469	0.2225
Zero [36]	Python (GPU)	79.4	0.0165	0.0178	0.0315
StableLLVE [18]	Python (GPU)	4316.3	0.1045	0.1501	0.1953
MFF-Net	Python (GPU)	817.3	0.0457	0.0566	0.1657

Table 5. Average running time cost (unit: seconds) and the parameters comparison on hazy images with different resolutions (400 × 600, 480 × 640, and 768 × 1024) of the different methods.

Methods	Platform	Parameters (K)	400 × 600	480 × 640	768 × 1024
DCP [40]	Matlab (CPU)	-	0.8124	1.0282	2.8398
CAP [41]	Matlab (CPU)	-	1.1568	1.3286	2.6762
HL [60]	Matlab (CPU)	-	4.9637	5.1852	6.6912
F-LDCP [61]	Matlab (CPU)	-	1.2865	1.4799	3.5562
DehazeNet [42]	Matlab (GPU)	-	0.5243	0.6742	1.4918
MSCNN [63]	Matlab (GPU)	-	0.1965	0.2853	0.8973
AODNet [64]	Python (GPU)	9	0.0125	0.0141	0.0192
GCANet [44]	Python (GPU)	2758	0.0912	0.1055	0.1722
FFANet [45]	Python (GPU)	25,999	0.7236	0.7951	1.8978
MFF-Net	Python (GPU)	817.3	0.0457	0.0566	0.1657

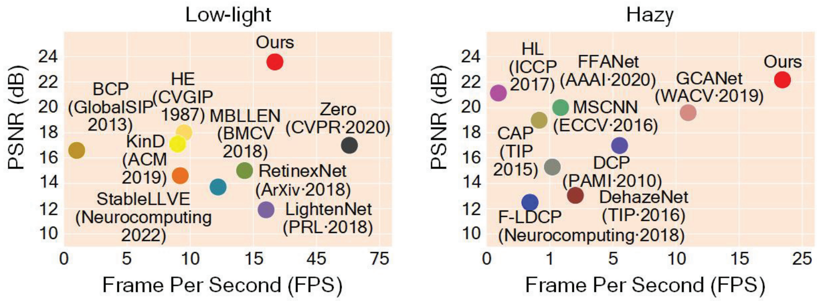


Figure 8. The trade-off between the visibility enhancement performance and the computational efficiency on several state-of-the-art low-light enhancement and dehazing methods. It is noted that the frame per second (FPS) metric is tested on a 600 × 400 resolution image.

4.6. Ablation Study

To validate the necessity of multiple feature fusion guidance, we first conduct the ablation experiment on the architecture with two incomplete versions: (a) with only the local feature guidance network (OLF-Net), which only employs the dense connected convolutional layers to extract the local features during the processing; and (b) with only the global feature guidance network (OGF-Net), which only uses the optimized cross attention module to extract global features. The visual results are shown in Figure 9, the enhanced image of OLF-Net suffers from obvious dark artifacts, which proves that the shallow convolutional layers cannot meet the learning capabilities required for extremely low-visibility image enhancement tasks. In addition, the noticeable black-line issue exists in the OGF-Net, due to the effect of exceptionally dark or bright pixels on correlated pixels in the cross attention module. The ablation study on the architecture indicates that multiple feature fusion guidance successfully improves the effectiveness of feature representation and alleviates the excessive influence of extremely bright or dark pixels in the cross attention mechanism, which is necessary in low-visibility enhancement tasks.


(a) Low light (b) OLF-Net (c) OGF-Net (d) MFF-Net

Figure 9. Visual comparison between the enhanced results of the MFF-Net with the incomplete versions on the standard low-light image enhancement dataset. It is noted that the results in the ablation study are output from the network trained and tested with the same implementation details as the MFF-Net.

We also investigate the effectiveness of the weight setting on different loss functions, including the Euclidean distance ($\mathcal{L}_1 + \mathcal{L}_2$) and angle difference ($\mathcal{L}_{color}$). For a fair experiment, the architecture of the network is set to MFF-Net. According to the comparison of the quantitative indicators shown in Table 6, the network shows the best performance on the proposed weight setting, which guarantee the enhancement quality comprehensively with a rational balance between the Euclidean distance and angle difference.

Table 6. Quantitative quality assessment comparison between different weight distributions of the loss functions on the testing data consisting of paired images extracted from the LOL [48] and EnlightenGAN [49] datasets. ↑ and ↓ represent that higher or lower values mean the better results, respectively.

Loss Functions	PSNR↑	SSIM↑	LPIPS↓
$0.25\mathcal{L}_1 + 0.25\mathcal{L}_2$	21.34 ± 4.4286	0.760 ± 0.1336	0.151 ± 0.1098
$0.25\mathcal{L}_1 + 0.25\mathcal{L}_2 + 0.25\mathcal{L}_{color}$	21.30 ± 4.6006	0.763 ± 0.1302	0.148 ± 0.1070
$0.125\mathcal{L}_1 + 0.125\mathcal{L}_2 + 0.25\mathcal{L}_{color}$	21.40 ± 3.6208	0.753 ± 0.1253	0.170 ± 0.1177
$0.25\mathcal{L}_1 + 0.25\mathcal{L}_2 + 0.5\mathcal{L}_{color}$	21.36 ± 4.2196	0.779 ± 0.1251	0.139 ± 0.1003

4.7. Improvement in Maritime Vessel Detection

To further demonstrate the practical benefits of our MFF-Net for maritime surveillance under low-visibility weathers, we apply YOLOv5 and YOLOX [68] to conduct maritime vessel detection experiments. The test images are randomly selected from the Test-L and Test-H. First, we select 1500 maritime-related images in the COCO dataset to train our detection networks. The evaluation tests are then constructed on the selected images. In low-visibility scenes, the vessel detection accuracy decreases heavily due to the low contrast and vague edge features, which can cause difficulties in maritime surveillance. In other words, the caption cannot make full use of the computer vision to assist the artificial lookout. After enhancement, the visual data can deliver clearer traffic scenes to the managers, and the detection accuracy is also significantly increased. The experimental results are illustrated in Figures 10 and 11. Compared with the state-of-the-art methods, the enhanced results of the MFF-Net perform better due to the application of multi-feature fusion. The quantitative comparison is shown in Table 7. It is noted that the input image will be first resized to 640 × 640 in YOLOX. However, most of the traditional method cannot enhance them within one second; thus, they cannot be applied in practical engineering. Therefore, we compare our method with the fastest representative traditional and deep learning-based methods in a quantitative experiment. The experimental results demonstrate that the MFF-Net has practical benefits in maritime surveillance, which is more beneficial for higher-level visual tasks under low-visibility weathers when assisting artificial observations, thereby improving maritime management.

Table 7. Quantitative experiments about the vessel detection accuracy improvement on YOLOX, which is tested on the Seaships [55] dataset. It is noted that mAP (clear), mAP (low-visibility), and mAP (enhancement) represent the mean average precision on clear, low-visbility, and enhanced images, respectively.

Weather	Method	mAP (Clear)	mAP (Low-Visibility)	mAP (Enhancement)
Low-light	HE [17]	40.97%	31.97%	33.26%
	Zero [36]	40.97%	31.97%	35.11%
	MFF-Net	40.97%	31.97%	36.42%
Hazy	DCP [15]	40.97%	21.66%	26.25%
	AOD-Net [64]	40.97%	21.66%	32.37%
	MFF-Net	40.97%	21.66%	34.81%

(a) Low-light (b) HE (c) NPE (d) BCP

(e) SRIE (f) RetinexNet (g) MBLLEN (h) KinD

(i) Zero (j) StableLLVE (k) MFF-Net (l) Ground Truth

Figure 10. Vessel detection experiment results on maritime low-light images between our method and other competitive methods: (**a**) low-light; (**b**) HE [17]; (**c**) NPE [30]; (**d**) BCP [56]; (**e**) SRIE [57]; (**f**) RetinexNet [48]; (**g**) MBLLEN [59]; (**h**) KinD [35]; (**i**) Zero [36]; (**j**) StableLLVE [18]; (**k**) MFF-Net; (**l**) ground truth. It can be seen that our MFF-Net significantly improves the accuracy of vessel detection under low-light environments, which demonstrates the benefits of our MFF-Net for practical ocean engineering.

(a) Hazy (b) DCP (c) CAP (d) HL

(e) F-LDCP (f) MSCNN (g) AODNet (h) GCANet

(i) DehazeNet (j) FFANet (k) MFF-Net (l) Ground Truth

Figure 11. Vessel detection experiment results on maritime hazy images between our method and other competitive methods: (**a**) hazy; (**b**) DCP [40]; (**c**) CAP [41]; (**d**) HL [60]; (**e**) F-LDCP [61]; (**f**) MSCNN [63]; (**g**) AODNet [64]; (**h**) GCANet [44]; (**i**) DehazeNet [42]; (**j**) FFANet [45]; (**k**) MFF-Net; (**l**) ground truth. It can be seen that our MFF-Net significantly improves the accuracy of vessel detection under hazy environments, which demonstrates the benefits of our MFF-Net for practical ocean engineering.

5. Conclusions

In this paper, we proposed an end-to-end multi-feature fusion-guided low-visbility enhancement method for maritime surveillance. Firstly, the maritime low-visibility images, i.e., low-light and hazy, are downsampled and then fed into the GL-Block comprising cross attention modules and dense residual convolutional layers. The GL-Block is designed to extract the global and local features to guide the enhancement processing simultaneously. After enhancement, the image is upsampled to a finer scale. For better constraint of the enhanced output, we introduced a joint loss function comprising L1 loss, L2 loss, and color similarity loss. In the experiments, we made massive comparisons on the visual performance, including quantitative image quality assessment, noise reduction, and color naturalness on both low-light enhancement and dehazing. Compared with other methods, the MFF-Net achieved a competitive quantitative and visual performance with effective noise reduction and superior color naturalness. Moreover, we evaluated the operating time cost and model size of the state-of-the-art methods, which indicates that MFF-Net can efficiently enhance extremely low-visibility images with lower computational complexity. In the ablation study, we conducted a series of experiments to investigate the necessity of multiple feature guidance and rational weight settings of the proposed loss function. Finally, the experiment of vessel detection indicate that our method is beneficial for practical maritime surveillance under low-visibility weathers.

In the future, we will test more methods for global feature extraction to demonstrate the advantages of multiple features for low-visibility image enhancement. Furthermore, high-definition videos cannot currently be enhanced in real time. We will thus optimize the architecture of the MFF-Net to achieve a better performance with lower computational complexity, which will enable the network to work on a diverse range of maritime edge devices.

Author Contributions: Conceptualization, W.Z. and B.L.; methodology, W.Z.; software, G.L.; validation, G.L.; formal analysis, B.L.; investigation, B.L.; resources, B.L.; data curation, W.Z.; writing—original draft preparation, W.Z.; writing—review and editing, W.Z.; visualization, W.Z.; supervision, B.L.; project administration, W.Z.; funding acquisition, B.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data available on request due to restrictions eg privacy or ethical.

Conflicts of Interest: The authors declare no conflict of interest.

Note

[1] LOL [48] and EnlightenGAN [49] are mainly captured in indoor or land environments. The haze in I-HAZE [50] and SMOKE [51] are artificially generated by smoke or steam, which is different from haze in the ocean.

References

1. Zhang, M.; Kujala, P.; Musharraf, M.; Zhang, J.; Hirdaris, S. A machine learning method for the prediction of ship motion trajectories in real operational conditions. *Ocean Eng.* **2023**, *283*, 114905. [CrossRef]
2. Zhang, M.; Taimuri, G.; Zhang, J.; Hirdaris, S. A deep learning method for the prediction of 6-DoF ship motions in real conditions. *Proc. Inst. Mech. Eng. Part J. Eng. Marit. Environ.* **2023**. [CrossRef]
3. Guo, Y.; Liu, R.W.; Qu, J.; Lu, Y.; Zhu, F.; Lv, Y. Asynchronous Trajectory Matching-Based Multimodal Maritime Data Fusion for Vessel Traffic Surveillance in Inland Waterways. *IEEE Trans. Intell. Transp. Syst.* 2023, *to be published*. [CrossRef]
4. Qu, J.; Liu, R.W.; Guo, Y.; Lu, Y.; Su, J.; Li, P. Improving maritime traffic surveillance in inland waterways using the robust fusion of AIS and visual data. *Ocean Eng.* **2023**, *275*, 114198. [CrossRef]
5. Yang, W.; Wang, S.; Fang, Y.; Wang, Y.; Liu, J. From fidelity to perceptual quality: A semi-supervised approach for low-light image enhancement. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 13–19 June 2020; pp. 3063–3072.
6. Liu, Y.L.; Lai, W.S.; Chen, Y.S.; Kao, Y.L.; Yang, M.H.; Chuang, Y.Y.; Huang, J.B. Single-image HDR reconstruction by learning to reverse the camera pipeline. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 14–19 June 2020; pp. 1651–1660.
7. Liu, D.; Wen, B.; Jiao, J.; Liu, X.; Wang, Z.; Huang, T.S. Connecting image denoising and high-level vision tasks via deep learning. *IEEE Trans. Image Process.* **2020**, *29*, 3695–3706. [CrossRef] [PubMed]

8. Zhou, Z.; Zhao, J.; Chen, X.; Chen, Y. A Ship Tracking and Speed Extraction Framework in Hazy Weather Based on Deep Learning. *J. Mar. Sci. Eng.* **2023**, *11*, 1353. [CrossRef]
9. Guo, Y.; Lu, Y.; Liu, R.W. Lightweight deep network-enabled real-time low-visibility enhancement for promoting vessel detection in maritime video surveillance. *J. Navig.* **2021**, *75*, 230–250. [CrossRef]
10. Ma, R.; Bao, K.; Yin, Y. Improved Ship Object Detection in Low-Illumination Environments Using RetinaMFANet. *J. Mar. Sci. Eng.* **2022**, *10*, 1996. [CrossRef]
11. Fan, Y.; Niu, L.; Liu, T. Multi-Branch Gated Fusion Network: A Method That Provides Higher-Quality Images for the USV Perception System in Maritime Hazy Condition. *J. Mar. Sci. Eng.* **2022**, *10*, 1839. [CrossRef]
12. Wang, H.; Chen, Y.; Cai, Y.; Chen, L.; Li, Y.; Sotelo, M.A.; Li, Z. SFNet-N: An Improved SFNet Algorithm for Semantic Segmentation of Low-Light Autonomous Driving Road Scenes. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 21405–21417. [CrossRef]
13. Liu, R.W.; Guo, Y.; Lu, Y.; Chui, K.T.; Gupta, B.B. Deep network-enabled haze visibility enhancement for visual IoT-driven intelligent transportation systems. *IEEE Trans. Industr. Inform.* **2022**, *19*, 1581–1591. [CrossRef]
14. Qu, J.; Gao, Y.; Lu, Y.; Xu, W.; Liu, R.W. Deep learning-driven surveillance quality enhancement for maritime management promotion under low-visibility weathers. *Ocean Coast. Manag.* **2023**, *235*, 106478. [CrossRef]
15. Krishnan, D.; Fergus, R. Dark flash photography. *ACM Trans. Graph.* **2009**, *28*, 96. [CrossRef]
16. Lu, Y.; Jung, S.W. Progressive Joint Low-Light Enhancement and Noise Removal for Raw Images. *IEEE Trans. Image Process.* **2022**, *31*, 2390–2404. [CrossRef]
17. Abdullah-Al-Wadud, M.; Kabir, M.H.; Dewan, M.A.A.; Chae, O. A dynamic histogram equalization for image contrast enhancement. *IEEE Trans. Consum. Electron.* **2007**, *53*, 593–600. [CrossRef]
18. Zhang, S.; Lam, E.Y. Learning to restore light fields under low-light imaging. *Neurocomputing* **2021**, *456*, 76–87. [CrossRef]
19. Liu, M.; Tang, L.; Zhong, S.; Luo, H.; Peng, J. Learning noise-decoupled affine models for extreme low-light image enhancement. *Neurocomputing* **2021**, *448*, 21–29. [CrossRef]
20. Qin, J.; Zhang, R. Lightweight Single Image Super-Resolution with Attentive Residual Refinement Network. *Neurocomputing* **2022**, *500*, 846–855. [CrossRef]
21. Hu, X.; Zhu, L.; Wang, T.; Fu, C.W.; Heng, P.A. Single-image real-time rain removal based on depth-guided non-local features. *IEEE Trans. Image Process.* **2021**, *30*, 1759–1770. [CrossRef]
22. Sun, Z.; Zhang, Y.; Bao, F.; Wang, P.; Yao, X.; Zhang, C. SADnet: Semi-supervised single image dehazing method based on an attention mechanism. *ACM Trans. Multimed. Comput. Commun. Appl.* **2022**, *18*, 1–23. [CrossRef]
23. Huang, Z.; Wang, X.; Huang, L.; Huang, C.; Wei, Y.; Liu, W. Ccnet: Criss-cross attention for semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 Oceober–2 November 2019; pp. 603–612.
24. Wang, C.; Fan, B.; Li, Y.; Xiao, J.; Min, L.; Zhang, J.; Chen, J.; Lin, Z.; Su, S.; Wu, R.; et al. Study on the Classification Perception and Visibility Enhancement of Ship Navigation Environments in Foggy Conditions. *J. Mar. Sci. Eng.* **2023**, *11*, 1298. [CrossRef]
25. Huang, S.C.; Cheng, F.C.; Chiu, Y.S. Efficient contrast enhancement using adaptive gamma correction with weighting distribution. *IEEE Trans. Image Process.* **2012**, *22*, 1032–1041. [CrossRef]
26. Jobson, D.J.; Rahman, Z.u.; Woodell, G.A. A multiscale retinex for bridging the gap between color images and the human observation of scenes. *IEEE Trans. Image. Process.* **1997**, *6*, 965–976. [CrossRef] [PubMed]
27. Lee, C.; Lee, C.; Kim, C.S. Contrast enhancement based on layered difference representation. In Proceedings of the 2012 19th IEEE International Conference on Image Processing, Orlando, FL, USA, 30 September–3 October 2012; IEEE: New York, NY, USA, 2012; pp. 965–968.
28. Lee, C.; Lee, C.; Kim, C.S. Contrast enhancement based on layered difference representation of 2D histograms. *IEEE Trans. Image Process.* **2013**, *22*, 5372–5384. [CrossRef] [PubMed]
29. Jobson, D.J.; Rahman, Z.U.; Woodell, G.A. Properties and performance of a center/surround retinex. *IEEE Trans. Image Process.* **1997**, *6*, 451–462. [CrossRef]
30. Wang, S.; Zheng, J.; Hu, H.M.; Li, B. Naturalness preserved enhancement algorithm for non-uniform illumination images. *IEEE Trans. Image Process.* **2013**, *22*, 3538–3548. [CrossRef]
31. Guo, X.; Li, Y.; Ling, H. LIME: Low-light image enhancement via illumination map estimation. *IEEE Trans. Image Process.* **2016**, *26*, 982–993. [CrossRef]
32. Chen, C.; Chen, Q.; Xu, J.; Koltun, V. Learning to see in the dark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 3291–3300.
33. Wang, K.; Gao, P.; Hoi, S.; Guo, Q.; Qian, Y. Extreme Low-Light Imaging with Multi-granulation Cooperative Networks. *arXiv* **2020**, arXiv:2005.08001.
34. Lu, Y.; Guo, Y.; Zhu, F.; Liu, R.W. Towards Low-Visibility Enhancement in Maritime Video Surveillance: An Efficient and Effective Multi-Deep Neural Network. In Proceedings of the 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), Indianapolis, IN, USA, 19–22 September 2021; IEEE: New York, NY, USA, 2021; pp. 2869–2874.
35. Zhang, Y.; Zhang, J.; Guo, X. Kindling the darkness: A practical low-light image enhancer. In Proceedings of the 27th ACM International Conference on Multimedia, Nice, France, 21–25 October 2019; pp. 1632–1640.

36. Guo, C.; Li, C.; Guo, J.; Loy, C.C.; Hou, J.; Kwong, S.; Cong, R. Zero-reference deep curve estimation for low-light image enhancement. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 14–19 June 2020; pp. 1780–1789.
37. Hao, S.; Han, X.; Guo, Y.; Wang, M. Decoupled Low-Light Image Enhancement. *ACM Trans. Multimed. Comput. Commun. Appl.* **2022**, *18*, 1–19. [CrossRef]
38. Guo, Y.; Lu, Y.; Qu, J.; Liu, R.W.; Ren, W. MDSFE: Multiscale Deep Stacking Fusion Enhancer Network for Visual Data Enhancement. *IEEE Trans. Instrum. Meas.* **2022**, *72*, 1–12. [CrossRef]
39. Wang, Y.; Wan, R.; Yang, W.; Li, H.; Chau, L.P.; Kot, A. Low-light image enhancement with normalizing flow. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 22 February–1 March 2022; pp. 2604–2612.
40. He, K.; Sun, J.; Tang, X. Single image haze removal using dark channel prior. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *33*, 2341–2353. [PubMed]
41. Zhu, Q.; Mai, J.; Shao, L. A fast single image haze removal algorithm using color attenuation prior. *IEEE Trans. Image Process.* **2015**, *24*, 3522–3533. [PubMed]
42. Cai, B.; Xu, X.; Jia, K.; Qing, C.; Tao, D. Dehazenet: An end-to-end system for single image haze removal. *IEEE Trans. Image Process.* **2016**, *25*, 5187–5198. [CrossRef] [PubMed]
43. Tang, G.; Zhao, L.; Jiang, R.; Zhang, X. Single image dehazing via lightweight multi-scale networks. In Proceedings of the 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 9–12 December 2019; IEEE: New York, NY, USA, 2019; pp. 5062–5069.
44. Chen, D.; He, M.; Fan, Q.; Liao, J.; Zhang, L.; Hou, D.; Yuan, L.; Hua, G. Gated context aggregation network for image dehazing and deraining. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 7–11 January 2019; IEEE: New York, NY, USA, 2019; pp. 1375–1383.
45. Qin, X.; Wang, Z.; Bai, Y.; Xie, X.; Jia, H. FFA-Net: Feature fusion attention network for single image dehazing. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 11908–11915.
46. Guo, Y.; Gao, Y.; Liu, W.; Lu, Y.; Qu, J.; He, S.; Ren, W. SCANet: Self-Paced Semi-Curricular Attention Network for Non-Homogeneous Image Dehazing. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 1884–1893.
47. Iandola, F.; Moskewicz, M.; Karayev, S.; Girshick, R.; Darrell, T.; Keutzer, K. Densenet: Implementing efficient convnet descriptor pyramids. *arXiv* **2014**, arXiv:1404.1869.
48. Wei, C.; Wang, W.; Yang, W.; Liu, J. Deep retinex decomposition for low-light enhancement. *arXiv* **2018**, arXiv:1808.04560.
49. Jiang, Y.; Gong, X.; Liu, D.; Cheng, Y.; Fang, C.; Shen, X.; Yang, J.; Zhou, P.; Wang, Z. Enlightengan: Deep light enhancement without paired supervision. *IEEE Trans. Image Process.* **2021**, *30*, 2340–2349. [CrossRef]
50. Ancuti, C.; Ancuti, C.O.; Timofte, R.; De Vleeschouwer, C. I-HAZE: A dehazing benchmark with real hazy and haze-free indoor images. In Proceedings of the Advanced Concepts for Intelligent Vision Systems: 19th International Conference, ACIVS 2018, Poitiers, France, 24–27 September 2018; Proceedings 19; Springer: Cham, Switzerland, 2018; pp. 620–631.
51. Jin, Y.; Yan, W.; Yang, W.; Tan, R.T. Structure representation network and uncertainty feedback learning for dense non-uniform fog removal. In Proceedings of the Asian Conference on Computer Vision, Macao, China, 4–8 December 2022; Springer: Cham, Switzerland, 2022; pp. 155–172.
52. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef]
53. Zhang, L.; Zhang, L.; Mou, X.; Zhang, D. FSIM: A feature similarity index for image quality assessment. *IEEE Trans. Image Process.* **2011**, *20*, 2378–2386. [CrossRef]
54. Zhang, L.; Shen, Y.; Li, H. VSI: A visual saliency-induced index for perceptual image quality assessment. *IEEE Trans. Image Process.* **2014**, *23*, 4270–4281. [CrossRef]
55. Shao, Z.; Wu, W.; Wang, Z.; Du, W.; Li, C. Seaships: A large-scale precisely annotated dataset for ship detection. *IEEE Trans. Multimed.* **2018**, *20*, 2593–2604. [CrossRef]
56. Fu, X.; Zeng, D.; Huang, Y.; Ding, X.; Zhang, X.P. A variational framework for single low light image enhancement using bright channel prior. In Proceedings of the IEEE Global Conference on Signal and Information Processing, Austin, TX, USA, 3–5 December 2013; IEEE: New York, NY, USA, 2013; pp. 1085–1088.
57. Fu, X.; Zeng, D.; Huang, Y.; Zhang, X.P.; Ding, X. A weighted variational model for simultaneous reflectance and illumination estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 2782–2790.
58. Li, C.; Guo, J.; Porikli, F.; Pang, Y. LightenNet: A convolutional neural network for weakly illuminated image enhancement. *Pattern Recognit. Lett.* **2018**, *104*, 15–22. [CrossRef]
59. Lv, F.; Lu, F.; Wu, J.; Lim, C. MBLLEN: Low-Light Image/Video Enhancement Using CNNs. In Proceedings of the BMVC, Newcastle, UK, 3–6 September 2018; p. 220.
60. Berman, D.; Treibitz, T.; Avidan, S. Air-light estimation using haze-lines. In Proceedings of the 2017 IEEE International Conference on Computational Photography (ICCP), Stanford, CA, USA, 12–14 May 2017; IEEE: New York, NY, USA, 2017; pp. 1–9.
61. Zhu, Y.; Tang, G.; Zhang, X.; Jiang, J.; Tian, Q. Haze removal method for natural restoration of images with sky. *Neurocomputing* **2018**, *275*, 499–510. [CrossRef]

62. Chen, C.; Do, M.N.; Wang, J. Robust image and video dehazing with visual artifact suppression via gradient residual minimization. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Cham, Switzerland, 2016; pp. 576–591.
63. Ren, W.; Liu, S.; Zhang, H.; Pan, J.; Cao, X.; Yang, M.H. Single image dehazing via multi-scale convolutional neural networks. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Cham, Switzerland, 2016; pp. 154–169.
64. Li, B.; Peng, X.; Wang, Z.; Xu, J.; Feng, D. Aod-net: All-in-one dehazing network. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4770–4778.
65. Guo, Y.; Lu, Y.; Liu, R.W.; Wang, L.; Zhu, F. Heterogeneous Twin Dehazing Network for Visibility Enhancement in Maritime Video Surveillance. In Proceedings of the 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), Indianapolis, IN, USA, 19–22 September 2021; IEEE: New York, NY, USA, 2021; pp. 2875–2880.
66. Vonikakis, V.; Bouzos, O.; Andreadis, I. Multi-exposure image fusion based on illumination estimation. In Proceedings of the Signal and Image Processing and Applications SIPA 2011, Crete, Greece, 22–24 June 2011; pp. 135–142.
67. Prasad, D.K.; Rajan, D.; Rachmawati, L.; Rajabally, E.; Quek, C. Video processing from electro-optical sensors for object detection and tracking in a maritime environment: A survey. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 1993–2016. [CrossRef]
68. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. Yolox: Exceeding yolo series in 2021. *arXiv* **2021**, arXiv:2107.08430.

Journal of
Marine Science and Engineering

Article

Research on Visual Perception for Coordinated Air–Sea through a Cooperative USV-UAV System

Chen Cheng [1,*], Dong Liu [2], Jin-Hui Du [1] and Yong-Zheng Li [1]

[1] School of Naval Architecture and Ocean Engineering, Jiangsu University of Science and Technology, Zhenjiang 212013, China; 212241801310@stu.just.edu.cn (J.-H.D.); justyzli@163.com (Y.-Z.L.)
[2] Yancheng Maritime Safety Administration of People's Republic of China, Yancheng 224008, China; ccccmaofei@163.com
* Correspondence: 202200000103@just.edu.cn

Abstract: The identification and classification of obstacles in navigable and non-navigable regions, as well as the measurement of distances, are crucial topics of investigation in the field of autonomous navigation for unmanned surface vehicles (USVs). Currently, USVs mostly rely on LiDAR and ultrasound technology for the purpose of detecting impediments that exist on water surfaces. However, it is worth noting that these approaches lack the capability to accurately discern the precise nature or classification of those obstacles. Nevertheless, the limited optical range of unmanned vessels hinders their ability to comprehensively perceive the entirety of the surrounding information. A cooperative USV-UAV system is proposed to ensure the visual perception ability of USVs. The multi-object recognition, semantic segmentation, and obstacle ranging through USV and unmanned aerial vehicle (UAV) perspectives are selected to validate the performance of a cooperative USV-UAV system. The you only look once-X (YOLOX) model, the proportional–integral–derivative-NET (PIDNet) model, and distance measurements based on a monocular camera are utilized to realize these problems. The results indicate that by integrating the viewpoints of USVs and UAVs, a collaborative USV-UAV system, employing the aforementioned methods, can successfully detect and classify different objects surrounding the USV. Additionally, it can differentiate between navigable and non-navigable regions for unmanned vessels through visual recognition, while accurately determining the distance between the USV and obstacles.

Keywords: visual perception; cooperative USV-UAV system; YOLOX; PIDNet; monocular camera vision

Citation: Cheng, C.; Liu, D.; Du, J.-H.; Li, Y.-Z. Research on Visual Perception for Coordinated Air–Sea through a Cooperative USV-UAV System. *J. Mar. Sci. Eng.* **2023**, *11*, 1978. https://doi.org/10.3390/jmse11101978

Academic Editor: Marco Cococcioni

Received: 18 September 2023
Revised: 8 October 2023
Accepted: 9 October 2023
Published: 12 October 2023

1. Introduction

As sensing technology, artificial intelligence algorithms, and intelligent control algorithms continue to advance, the development of intelligent spacecraft continues to advance. In recent years, the demand for unmanned ships has increased, and unmanned surface vehicles (USVs) have become research hotspots for numerous unmanned vehicles. USVs are small, intelligent ships that can navigate autonomously without the need for human operation and automatically complete specific water tasks [1].

The correct capture of surrounding environmental information is a crucial requirement for ensuring the safe and autonomous navigation of USVs in complex aquatic environments. USVs are able to effectively navigate through a dynamic environment while promptly avoiding obstacles and accurately identifying water surface target objects. USVs typically employ a range of sensors for various purposes, including radar navigation, millimeter-wave radars, LiDAR, sonar, and vision sensors. The initial sensors exhibit several limitations, including elevated costs, notable environmental ramifications, and a restricted capacity to perceive and gather comprehensive environmental data, hence impeding the acquisition of additional information [2,3]. Visual sensors have the potential to enhance the perception capabilities of USVs, allowing them to effectively observe a larger expanse of water and gather valuable information about the aquatic environment. This, in turn, enables USVs to gain a thorough understanding of water navigation situations. The advancement of image processing technology and deep learning-based detection and recognition technology has led to a notable enhancement in the perceptual accuracy of vision systems for unmanned ships. Visual sensors have found extensive applications in diverse mobile intelligent platforms, assuming a crucial function in the detection and recognition of water targets, monitoring aquatic environments, and mitigating potential collisions with unmanned vessels [4].

The limited installation height of vision sensors on USVs poses challenges in achieving a comprehensive perception of the surrounding environment, particularly in detecting obstructed areas ahead. Consequently, this creates a blind spot inside the field of vision. Simultaneously, the close proximity of visual sensors to the water surface introduces a susceptibility to environmental influences such as water ripples, reflections, and illumination, hence presenting challenges in image processing and target recognition. Unmanned Aerial Vehicles (UAVs) possess a notable advantage in terms of their elevated flying height, which significantly enhances their visual perception range. Consequently, the images captured by UAVs are subjected to lesser influence from the surrounding water environment [5]. The limited cargo capacity of UAVs necessitates a reliance on the battery module for flying power, resulting in a relatively short flight duration that restricts its ability to carry out prolonged, complex tasks. In addition, it should be noted that UAVs currently possess limited processing and computing capabilities, rendering them inadequate for executing intricate visual processing tasks within UAV systems.

Consequently, numerous researchers have directed their focus towards investigating the collaborative systems of UAVs and USVs, aiming to leverage their individual strengths in order to address the challenges posed by the limited endurance of UAVs and the restricted perception range of USVs. This paper aims to investigate the research pertaining to visual-based environment perception in collaborative UAV and USV systems. The findings of this study will offer fundamental technological assistance for the advancement and implementation of autonomous navigation, maritime supervision, and cruise control in unmanned maritime vessels.

The organization of this paper can be summarized as follows: Section 2 lists the recent related works. In Section 3, the several methods are introduced. Section 4 demonstrates the experimental setup and data process. The results are discussed in Section 5. Finally, the conclusion is summarized in Section 6.

2. Literature Review

The perception of autonomous ships can be categorized into two distinct aspects based on their perceived content: self-perception and the perception of the external environment. The accurate determination of USVs' own state can be achieved by utilizing GPS positioning sensors and IMU inertial units. The stability and precision of this determination are often dependent on the device's performance [6]. The process of perceiving the external environment primarily relies on the use of diverse sensory mechanisms. The heightened unpredictability of the environment is a significant obstacle in accurately recognizing the exterior surroundings of USVs, hence creating difficulties in ensuring their safe and autonomous navigation. The perception of the aquatic navigation environment can be categorized into two groups based on the various operating methods of sensors: active perception and passive perception [7].

Active perception refers to the act of transmitting signals to the external environment using sensing devices, and subsequently acquiring information about the surrounding environment by receiving the returned signal information. Examples of such sensing equipment are radar navigation and LiDAR sensors [8]. Carlos et al. (2009) incorporated radar technology into the ROAZ USV system in order to facilitate the identification of obstacles and the prevention of collisions [9]. Zhang et al. (2011) employed the Gaussian particle filtering technique to effectively analyze maritime radar data and successfully accomplish dynamic target tracking [10]. Han et al. (2019) presented a novel technique that combines radar technology with simultaneous location and map building for unmanned ship systems. This algorithm aims to overcome the issue of GPS signal loss in difficult surroundings, ultimately enabling accurate positioning in coastal areas [11]. Esposito et al. (2014) employed LiDAR technology for the purpose of automatically identifying docks, hence facilitating the autonomous docking of USVs [12].

Passive perception mostly pertains to the utilization of visual sensors for the acquisition of information concerning the surrounding navigation environment. The fundamental premise involves capturing visual data of the surrounding environment using visual sensors, followed by the interpretation of the environmental information based on the color and texture characteristics of the captured images [13,14]. Kristan et al. (2014) first utilized either monocular or stereo vision techniques to gather real-time data on the water surface environment. Subsequently, they used a water antenna recognition algorithm to ascertain the precise location of the water antenna. Next, they conducted a search for potential targets in close proximity to the water antenna in order to successfully detect water surface targets [15]. Wang et al. (2015) employed both monocular and binocular vision techniques to achieve the real-time and efficient identification of obstacles on the sea surface. Their approach enables the detection and localization of multiple objects across a distance range spanning from 30 to 100 m [16].

Following this, the integration of deep learning techniques was used in the domain of USV vision with the aim of enhancing the robustness and precision of algorithms pertaining to USV vision. Shi et al. (2019) made enhancements to the single-shot multi-box detector (SSD) algorithm in order to effectively identify and localize impediments and targets in the vicinity of unmanned vessels [17]. Song et al. (2019) put forth an algorithm for real-time obstacle identification. This approach utilized the Kalman filtering method to combine the SSD and Faster RCNN models. The objective of this algorithm was to detect obstacles on the sea surface for USVs [18]. Zhan et al. (2019) introduced a novel network segmentation algorithm that utilized self-learning techniques to identify and classify water and non-water surface regions in visual pictures captured by USVs. This approach aims to enable autonomous collision avoidance capabilities in USVs, hence ensuring the safety of their navigation [19].

The installation height of visual sensors on USVs presents a constraint that not only diminishes the sensing range of these ships, but also renders their vision vulnerable to the effects of water waves and reflections. Consequently, this low installation height poses challenges for the visual processing capabilities of USVs. In recent years, there has been significant advancement in UAV technology, leading to their widespread utilization across several domains. UAVs possess notable maneuverability capabilities and have an extensive perception range that is attributable to their elevated flight altitudes. Due to this rationale, numerous academics have endeavored to engage in collaborative research pertaining to USVS and UAVs in order to accomplish intricate aquatic assignments. Xu and Chen (2022) presented a comprehensive analysis of a multi-agent reinforcement learning (MARL) methodology, specifically designed for UAV clusters. The primary challenges were the assembly and formation maintenance in UAV cluster formation control [20]. Zhang et al. (2019) employed a distributed consistency technique in order to develop and simulate the control algorithm. In relation to the issue of cooperative path-following control between USVs and UAVs [21]. Li et al. (2022) introduced a novel conceptual framework for establishing a coherent and efficient connection between USVs and UAVs. This framework, referred to as the logical virtual ship–logical virtual aircraft guidance principle, aims to facilitate an effective association between these two types of unmanned vehicles [22].

3. Methodology

3.1. YOLOX

The predominant techniques employed for target recognition at present encompass SSD, CenterNet, and YOLO. The SSD algorithm is influenced by the anchor notion introduced in Faster R-CNN, wherein individual units establish previous boxes with varying scales or aspect ratios. The anticipated boundary boxes are derived from the past boxes, hence mitigating the challenges encountered during training. CenterNet, alternatively referred to as Objects as Points, has garnered significant attention from users, owing to its very straightforward and refined architecture, robust capability for handling diverse tasks, rapid inference speed, commendable accuracy, and the absence of the necessity for non-maximum suppression (NMS) post-processing. The YOLO algorithm addresses object detection by formulating it as a regression problem. Utilizing an independent end-to-end network, the task involves processing the input data that originate from the original image and generating the corresponding output, which comprises the positions and categories of objects. In comparison to these networks, YOLOX demonstrates superior performance in accomplishing the identical task, while concurrently preserving a highly competitive inference speed. The YOLOX object detection network comprises four components: the input terminal of the model, the Darket53 backbone network, the feature enhancement network neck, and the model prediction [23].

The YOLOX network Incorporates two data augmentation techniques, namely Mosaic and MixUp, at its input end. Additionally, it establishes a Focus structure. Mosaic data, firstly proposed in YOLO4, aim to improve the background of an image through the application of random scaling, cropping, and the arrangement of many photos [24]. MixUp is a supplementary augmentation method that is implemented in conjunction with Mosaic. This strategy significantly improves the quality of photos by merging two images together using a specific fusion coefficient, while minimizing the computational overhead [25]. The Focus structure is designed to extract four distinct feature layers from an image by selecting alternate pixels. These layers are subsequently combined to consolidate width and height information into channel information. This process results in a concatenation of feature layers, increasing the number of channels from three to twelve, thereby quadrupling the channel count.

The backbone serves as the primary architectural framework of YOLOX. Within YOLOX, the prominent feature extraction network employed is CSPDarknet53. CSPDarknet53 is composed of 72 convolutional layers, each possessing a dimension of 3 × 3 and a stride of 2. This configuration enables the network to effectively extract features and progressively down-sample the input data. The neck feature fusion structure employed in YOLOX is founded upon three fundamental elements: Feature Pyramid Networks (FPNs), Spatial Pyramid Pooling (SPP), and Path Aggregation Networks (PANs) [26]. The primary components of prediction encompass the decoupled head, anchor-free, label assignment, and loss calculation, which facilitate the execution of classification and regression tasks inside the model.

The primary filtering technique employed in YOLOX is SimOTA. Initially, the anchor boxes undergo a screening process to extract the position IoU matrix [27]. Subsequently, the loss function is computed for the chosen candidate detection boxes and ground truth. The cost function is then determined through a weighted summation of the resulting loss functions, as demonstrated below:

$$C_{ij} = L_{ij}^{clsloss} + \gamma \times L_{ij}^{regloss} \tag{1}$$

where C_{ij} denotes the total loss for a specific bounding box; $L_{ij}^{clsloss}$ is the classification loss, measuring the difference between the predicted and true class labels; and $L_{ij}^{regloss}$ is the regression loss, evaluating the disparity between the predicted and actual bounding box positions.

This paper employs the CBAM attention mechanism to enhance the conventional YOLOX network [28]. In contrast to the original network, the present network exhibits a heightened focus on pertinent characteristics of diminutive entities, hence minimizing the risk of detection oversight. Identifying little objects poses a greater challenge because of their low resolution and limited visual information, in contrast to larger objects. Consequently, the CBAM module is incorporated into the Dark3 module of the shallow network. Attention weights are then derived from both the spatial and channel dimensions. These weights are subsequently multiplied by the feature map ratio of 80 × 80, resulting in an enhanced feature response specifically for small objects. The purpose of the CBAM is accomplished by the utilization of two distinct attention modules: the channel attention module (CAM) for assessing the correlation among channels, and the spatial attention module (SAM) for evaluating the correlation among positions. The structure of the CBAM module can be viewed in Figure 1.

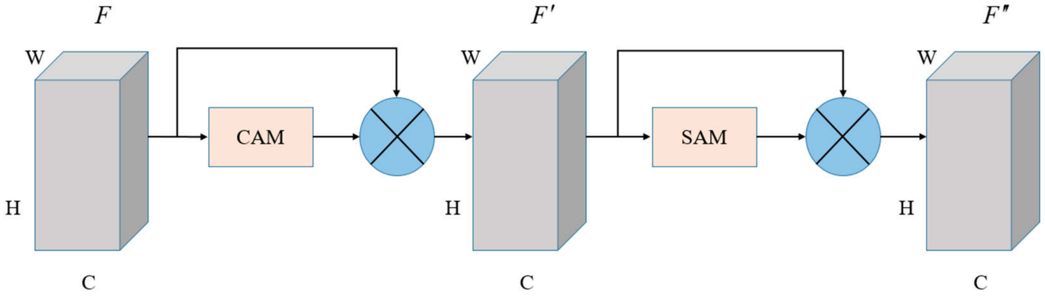


Figure 1. The structure of CAM module.

The convolutional attention module initially conducts spatial domain operations, specifically maximum pooling and average pooling, on the input feature map (F) with dimensions H × W × C. This process generates two channel information vectors of size 1 × 1 × C. These vectors are subsequently fed into a multi-layer perceptron (MLP) and individually summed. The application of the sigmoid activation function is the final step in obtaining the weight coefficient, M_c. This coefficient is then multiplied by the original feature map to derive the channel attention feature map, as shown in Equations (2) and (3).

$$M_c(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \tag{2}$$

$$F' = M_c(F) \otimes F \tag{3}$$

The IoU (Intersection over Union) loss function is employed in YOLOX, which is a widely utilized metric within the domain of object detection. The computation methodology for the IoU is as follows:

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|} \tag{4}$$

3.2. PIDNet

A proportional (P) controller, an integral (I) controller, and a derivative (D) controller comprise a PID controller, which can be viewed in Figure 2. The PI controller implementation might be expressed as:

$$c_{out}[n] = k_p e[n] + k_i \sum_{i=0}^{n} e[i] \tag{5}$$

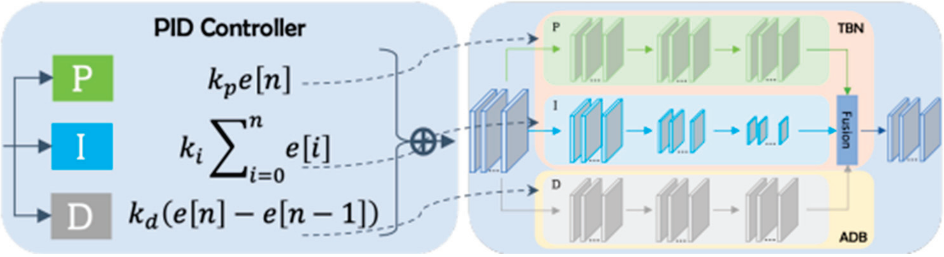


Figure 2. The structure of CBAM module.

The *P* controller concentrates on the present signal, whereas the *I* controller gathers all previous signals. Subsequently, the introduction of the *D* controller is implemented, wherein the *D* component assumes a negative value when the signal decreases, acting as a dampening mechanism to mitigate overshooting. In a similar manner, two-branch networks (TBNs) analyze the contextual and intricate information via the utilization of several convolutional layers, both with and without strides. In this particular one-dimensional example, it is worth noting that both the detailed and contextual branches comprise three layers, without the inclusion of batch normalization (BN) and rectified linear units (ReLUs). The output maps can be calculated as

$$O_D[i] = K^D_{i-3} I[i-3] + \ldots + K^D_i I[i] + \ldots + K^D_{i+3} I[i+3] \tag{6}$$

$$O_C[i] = K^C_{i-7} I[i-7] + \ldots + K^C_i I[i] + \ldots + K^C_{i+7} I[i+7] \tag{7}$$

where

$$K^D_i = k_{31}k_{22}k_{13} + k_{31}k_{23}k_{12} + k_{32}k_{21}k_{13} + k_{32}k_{22}k_{12} + k_{32}k_{23}k_{13} + k_{33}k_{21}k_{12} + k_{33}k_{22}k_{11} \tag{8}$$

$$K^C_i = k_{32}k_{22}k_{12} \tag{9}$$

and where k_{mn} refers to the *n-th* value of the kernel in layer *m*.

PIDNet is composed of three branches that have distinct roles: the proportional (*P*) branch is responsible for parsing and preserving detailed information in feature maps with high resolution; the integral (I) branch aggregates context information at both local and global levels to parse long-range dependencies; and the derivative (*D*) branch extracts high-frequency features to predict boundary regions. A semantic head is positioned at the output of the initial Pag module in order to generate an additional semantic loss, denoted as l_0, with the aim of enhancing the optimization process of the entire network. Instead of using dice loss, we employ weighted binary cross entropy loss, l_1, to address the issue of imbalanced boundary detection. This is because emphasizing the coarse border is desired in order to emphasize the boundary region and increase the characteristics for smaller items. The variables l_2 and l_3 are used to denote the cross-entropy (CE) loss in our study. Specifically, for l_3, we employ the boundary awareness CE loss (Towaki. 2019), which leverages the output of the boundary head to effectively coordinate the tasks of semantic segmentation and boundary detection. This approach enhances the functionality of the Bag module. Therefore, the final loss for PIDNet can be calculated as [29]:

$$Loss = \lambda_0 l_0 + \lambda_1 l_1 + \lambda_2 l_2 + \lambda_3 l_3 \tag{10}$$

3.3. Monocular Vision Scale–Distance by USVs

The objective here is to gather images of water surfaces, analyze them to determine the specific area where the target is situated within each image, and afterwards compute the greatest pixel ordinate value of that area, together with its related mean horizontal coordinate, utilizing the aggregated coordinates as the pixel coordinates for the points of observation. The objective is to determine the depth information of the observation point within the camera coordinate system by utilizing the camera's internal parameters and the geometric relationship of perspective projection. Next, the three-dimensional data of the observation point in the coordinate system of the USV attachment will be determined using rigid body transformation. Subsequently, the distance between the observation point and the USV will be calculated. The range-measuring model utilized in this paper is depicted in Figure 3. This model can be seen as a process that maps the items present in the three-dimensional (3D) scene onto two-dimensional (2D) images using a pinhole camera.

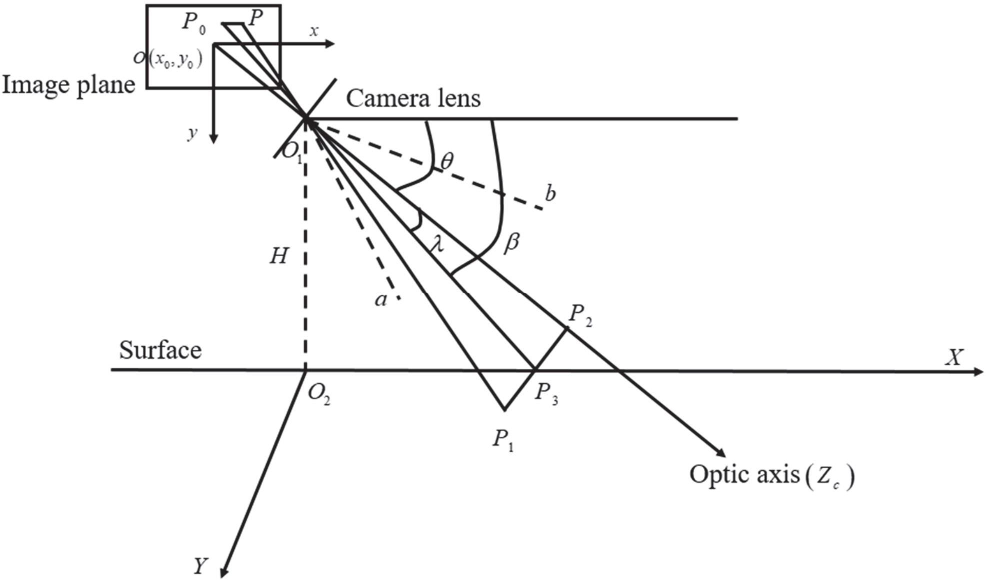


Figure 3. Ranging model.

In Figure 3, xoy denotes the image coordinate system while Z_c denotes the Z-axis and optical axis of the camera coordinate system; xo_2y represents the Z = 0 plane and water surface in the USV coordinate system; O_1 is the camera lens; the two dashed lines, a and b, represent visual field range; θ indicates the camera pitch angle. The observation point P_1 is positioned at the imaging point P on the image plane, the projection point P_2 on the optical axis, and the projection point P_3 on the X-axis; the point P_3 corresponds to the point P_0 when projected onto the picture plane.

The distance from the observation point to the center of the USV can be calculated as follows:

$$O_2P_1 = \sqrt{x_s^2 + y_s^2} \tag{11}$$

where x_s and y_s denote the coordinate values of $P_s(x_s, y_s, z_s)$ at the observation point in the USV fitted coordinate system, which can be estimated based on the link between the coordinates of the observation point in the pixel coordinate system and the coordinates in the USV fitted coordinate system:

$$z_c \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{R} & \mathbf{T} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_s \\ y_s \\ z_s \\ 1 \end{pmatrix} \tag{12}$$

where f_x, f_y, u_0, and v_0 indicate the intrinsic camera parameters, which can be calibrated based on Pei (2015) [30]. **R** and **T** denote the rotation matrix and translation matrix from the USV fitted coordinate system to the camera coordinate system, respectively. z_c is the depth coordinate value of the observation point in the camera coordinate system $P_c(x_c, y_c, z_c)$, which is the distance of O_1P_2 and can be calculated as

$$z_c = O_1P_2 = O_1P_3 \times \cos\lambda \tag{13}$$

$$O_1P_3 = \frac{H}{\sin(\theta + \lambda)} \tag{14}$$

where H denotes the height of camera; λ is the angle between the light at point P_0 and the camera's optical axis, which can be shown as follows:

$$\lambda = \arctan\frac{y - y_0}{f} \tag{15}$$

where y denotes the ordinate of point P in the image coordinate system;f is the focal length of the camera. In the pixel coordinate system, λ can be calculated as follows:

$$\lambda = \arctan\frac{(v - v_0) \times dy}{f_y \times dy} = \arctan\frac{v - v_0}{f_y} \tag{16}$$

where v denotes the ordinate of point P in the pixel coordinate system; v_0 is the vertical axis of the image center; dy is the unit pixel length in the y-direction; and f_y is the normalized focal length.

In order to mitigate the potential influence of measurement errors pertaining to camera height and pitch angle on the accuracy of ranging in situations characterized by uncertainty, this research study presents a calibration technique for the aforementioned camera parameters.

In the event that the sea surface is generally tranquil, it is feasible to approximate it as a flat plane. In Figure 4, $A(x_1, y_1, 0)$ is the intersection point between the plane $z = 0$ in the grid coordinate system and the camera optical axis (the Z-axis of the camera coordinate system), and the coordinate of point A in the camera coordinate system is $C_a(0, 0, z_1)$. $B(x_2, y_2, a)$ is the intersection point between the plane $z = 0$ in the grid coordinate system and the camera optical axis; the coordinate of point B in the camera coordinate system is $C_b(0, 0, z_2)$. The coordinate of the vertical projection point B' on the $z = 0$ plane of intersection B in the grid coordinate system is $(x_2, y_2, 0)$. The pitch angle of the camera is $\angle BAB'$, which can be shown as

$$\angle BAB' = a\cos\frac{AB \cdot AB'}{|AB||AB'|} \tag{17}$$

$$H = z_1 \times \sin\angle BAB' \tag{18}$$

Camera calibration is a process that yields the grid coordinate system, which lies on the calibration plate and is coplanar with the water surface. Additionally, it provides the rotation matrix ($\mathbf{R_1}$) and translation matrix ($\mathbf{T_1}$) of the camera coordinate system. The relationship between the coordinate $\boldsymbol{XX}$ in the grid coordinate system and the corresponding coordinate $\boldsymbol{XX_c}$ in the camera coordinate system can be expressed as follows:

$$XX_c = \mathbf{R}_1^* XX + \mathbf{T}_1 \tag{19}$$

Figure 4. Camera height and pitch angle model.

Substitute the coordinate variables of A and C_a into Equation (19):

$$\begin{pmatrix} 0 \\ 0 \\ z_1 \end{pmatrix} = \begin{pmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ y_1 \\ 0 \end{pmatrix} + \begin{pmatrix} T_1 \\ T_2 \\ T_3 \end{pmatrix} \tag{20}$$

Solve Equation (20) to obtain the coordinate values of A and C_a. Similarly, the coordinate values of B and C_b can also be calculated.

4. Experimental Setup

4.1. Data Processing

Pre-processing actions for images are deemed crucial in accordance with the stipulations of model application settings. The images within the experimental dataset were obtained from a total of 60 videos, each of which had a minimum duration of 60 s. During the cropping process, a proportional comparison is conducted on the target instance. If the resulting area of the target instance is equal to or more than 60% of the original instance area, the instance is retained. Otherwise, it is removed. The dataset has a total of 8588 pictures, with the training set and test set accounting for 80% and 20% of the dataset, respectively. The objective of this study is to examine the feasibility of the cooperative USV and UAV platform architecture. This is achieved by initially categorizing the dataset into five distinct groups, namely ship (representing various types of ships), USV, buoy, building, and people, with an equal proportion among each category.

In addition, 8588 images are also utilized to train and test the PIDNet model. The labelme software is employed for the purpose of annotating all photographs, with a primary focus on labelling distinct regions such as navigable water surfaces, non-navigable skies, and diverse barriers that are present on the water. In order to address intricate marine barriers, a multi-point framing approach is employed to accurately delineate the desired area. The detailed annotation method is shown in Figure 5.

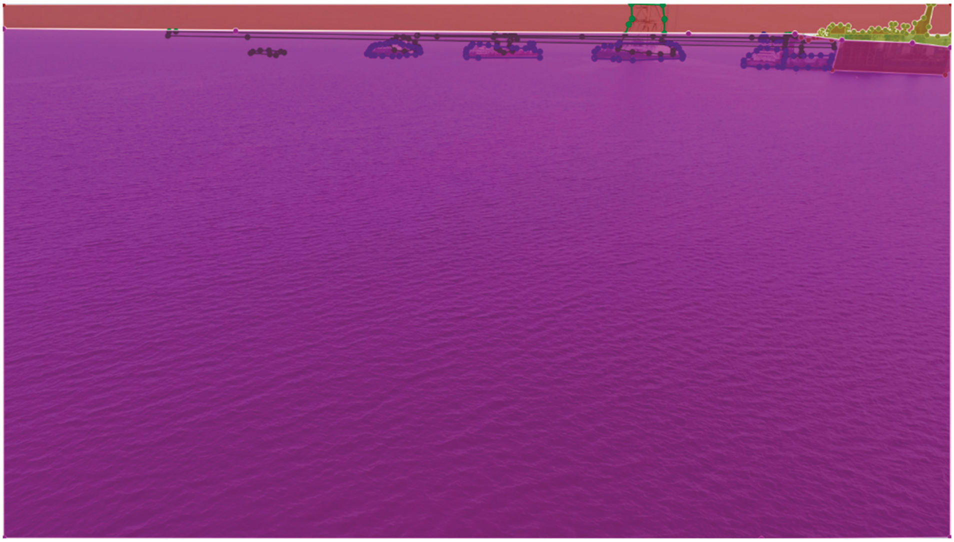


Figure 5. Annotation diagram.

4.2. Experimental Platform

Jiangsu University of Science and Technology invented and developed the cooperative USV-UAV platform, including an unmanned catamaran and a quadrotor, which can be viewed in Figure 6. Furthermore, this platform is equipped with a USV, with two lithium batteries housed within the hull. Additionally, a satellite communication module is located within the hull compartment, while four cameras are fitted atop the USV.

Figure 6. Cooperative USV-UAV platform.

The experimental training computer is configured with the Windows 10 operating system, with an NVIDIA GTX2080Ti graphics processing unit (GPU). The deep learning framework is Pytorch 1.5.0. Additional information can be observed in Table 1.

Table 1. Experimental computer environment.

Name	**Versions**
System Environment	Windows 10 64-bit
CPU	Intel(R) Core (TM) i9-9980XE
GPU	NVIDIA GTX 2080Ti
Python	3.6.0
Pytorch	1.5.0

4.3. Evaluation Criteria

Various occupations are evaluated using diverse metrics. This study introduces assessment criteria, including frames per second (FPS), precision (P), recall (R), and average precision (AP), which are developed by considering relevant demand variables. Frames per second (FPS) is a quantitative metric, utilized to gauge the rate at which images are processed within a given time frame of one second. Recall (R) is employed as a metric for evaluating the comprehensiveness of target detection. Conversely, precision (P) is utilized to ascertain the accuracy of recognition precision, which may be calculated as follows:

$$\begin{cases} Precision = \frac{N_{TP}}{N_{TP}+N_{FP}} \\ [l]\mathrm{Rec}all = \frac{N_{TP}}{N_{TP}+N_{FN}} \end{cases} \tag{21}$$

where N_{TP}, N_{FP}, N_{FN} indicate the number of successfully detected targets, the number of wrongly detected targets, and the correct number of targets missed by the model, respectively. The average precision (AP) is computed as follows:

$$Averageprecision = \int\limits_{0}^{1} P(R)dR \tag{22}$$

5. Results Analysis

5.1. Multi-Target Recognition

The field experiments were carried out on the Huanghai Sea in Yancheng city, China. The YOLOX model was selected to track the USV on calm water. Figure 7 demonstrates the test results. The model demonstrates a high level of effectiveness in accurately identifying the USV during its operation on undisturbed water surfaces. Additionally, the UAV is strategically positioned above the USV at this particular instance, which is considered the most favorable condition for optimal recognition. Moreover, even when the USV is not positioned in the center of the screen, the model is still capable of accurately detecting it and generating reliable detection outcomes.

This article focuses on the YOLO model and YOLOX is a fundamental model in the project, and is compared with other models (e.g., SSD, CenterNet, and other YOLO versions). According to the data presented in Table 2, it is evident that YOLOX exhibits superior speed performance compared to alternative models (baseline: FPS). The YOLOX model shows a recognition accuracy that surpasses the YOLO V4 model by 6.2 percent, and demonstrates superior performance compared to other existing models. The YOLOX model displays suboptimal recognition outcomes for small targets due to its tendency to prioritize the prediction of larger targets at higher levels while neglecting the accurate prediction of smaller targets. Despite the somewhat lower FPS achieved with the YOLOX model, it remains a highly promising approach. The YOLOX algorithm presents several benefits and holds potential for practical implementations due to its straightforward architecture, rapid processing capabilities, high precision, and efficient memory utilization.

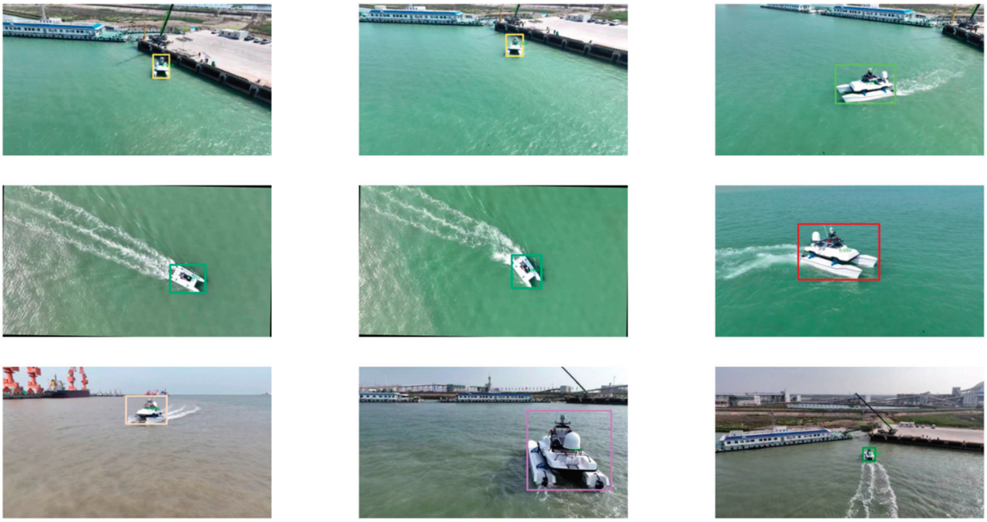


Figure 7. Results of the field test.

Table 2. Model comparison.

Model	FPS	AP_{50}(%)
SSD	52	79.9
CenterNet	57	81.5
YOLO V3	51	83.6
YOLO V4	46	84.1
YOLO V5	47	82.7
YOLOX	42	90.3

In this article, a UAV is employed to capture the testing photographs with the purpose of assessing the viability of the platform migration application. Figure 8 shows the multi-object recognition based on the UAV's perspective. From the standpoint of a UAV, it is possible to precisely monitor the movements of USVs from various vantage points. Additionally, the presence of other ships in motion near the unmanned vessel, as well as stationary obstructions like buoys, quayside barges, and moored ships, may be reliably detected and identified. The cooperative USV-UAV's perception system is highly successful in identifying even the smallest pixels in the distance of the image, including buoys and ships. This ensures the system's effectiveness in covering the entire water region. Figure 9 demonstrates the multi-object recognition from the USV's perspective. From this standpoint, the USV possesses the capability to effectively detect and classify diverse forms of impediments in its vicinity. Furthermore, it can precisely discern the movements of individuals situated beside the vessel. Nevertheless, the current perspective of the USV lacks sufficient breadth to adequately monitor diverse barrier conditions in every direction. Hence, the approach that relies on the cooperative USV-UAV system can effectively facilitate the comprehensive detection of expansive maritime regions by unmanned ships at sea. This approach holds resemblance to the concept of bird's-eye view (BEV) technology, employed in autonomous driving systems.

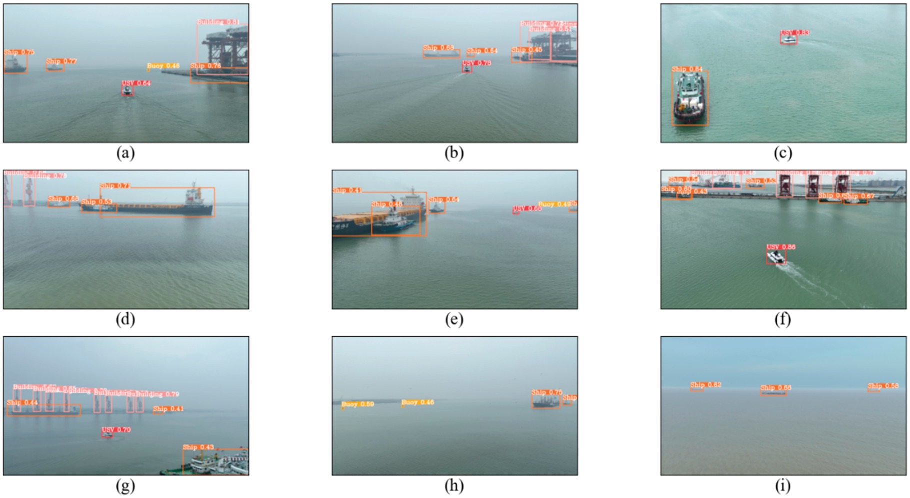

(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 8. Multi-object recognition based on UAV's perspective ((**a**–**i**) represent different angles of view from UAV).


(a) (b) (c) (d)

Figure 9. Multi-object recognition based on USV's perspective ((**a**–**d**) represent different angles of view from USV).

5.2. Semantic Segmentation

The reliable identification and classification of water surfaces play a crucial role in facilitating the autonomous movement of USVs. This is because any area that is not composed of water is highly likely to be an obstacle, hence presenting a possible hazard to the USV's navigation. The proposed PIDNet is based on the graph-based segmentation algorithm.

Figures 10 and 11 illustrate the process of semantic segmentation, specifically focusing on the identification and differentiation of water, surface obstacles, land, and sky. This segmentation is achieved through the utilization of both USV and UAV perspectives. The initial row showcases the unaltered input image, while the subsequent row presents the corresponding ground truth. The subsequent row exhibits the segmentation output that is predicted by PIDNet. The red region depicted in the diagram denotes the expanse of the water surface, which is designated as a navigable zone. The black region depicted in the illustration represents the celestial expanse known as the sky, which is deemed impassable for navigation purposes. Other colors are used to symbolize different barriers found on both water and land, namely locations that are not suitable for navigation. The results suggest that PIDNet demonstrates a high level of efficacy in discerning navigable and non-navigable regions on intricate water surfaces. This offers dependable technical assistance for the autonomous navigation of USVs, relying on visual inputs.

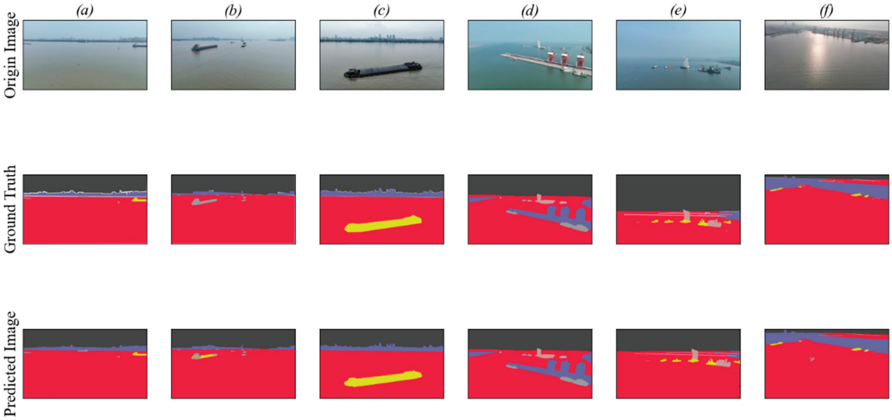


Figure 10. Semantic segmentation based on UAV perspective ((**a**–**f**) represent different angles of view from UAV).

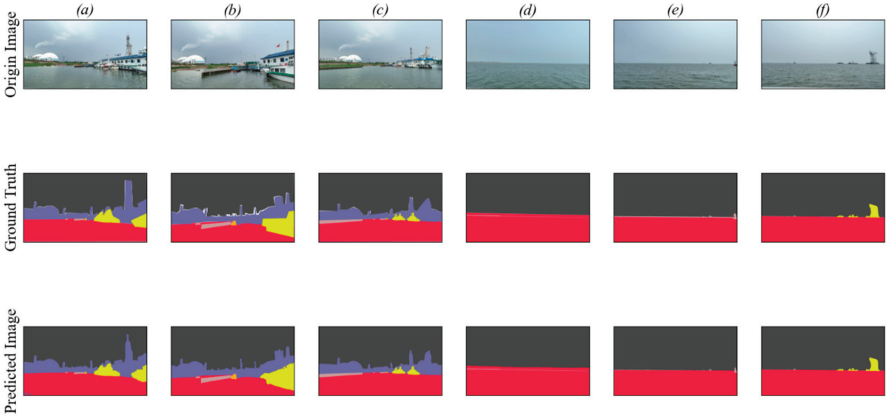


Figure 11. Semantic segmentation based on USV perspective ((**a**–**f**) represent different angles of view from USV).

In order to showcase the resilience of the PIDNet algorithm, we conducted a comparative analysis between our approach and other cutting-edge methodologies, specifically focusing on graph-based segmentation algorithms (U-Net, Refine-Net, and DeepLab). The performance evaluation of several models was conducted by quantitatively assessing their accuracy in semantic segmentation. This assessment involved the use of metrics such as mean intersection over union (MIoU), pixel accuracy (PA), and frames per second (FPS). It was observed that the accuracy of semantic segmentation was comparable to that of water surface segmentation. The results can be viewed in Table 3. The networks that are being compared in this study were subjected to retraining using our dataset, incorporating the most optimal hyperparameters. The results demonstrate that the PIDNet has the most accurate prediction ability, and the network's processing speed is exceptionally high as a result of its efficient architecture.

Table 3. Segmentation algorithms comparison.

Networks	Params (M)	MIOU (%)	PA (%)	FPS
U-Net	34.0	79.82	80.81	9
Refine-Net	55.1	81.63	84.26	15
DeepLab	44.3	87.22	89.13	30
PIDNet	29.5	91.08	94.32	40

Figure 12 shows the sea–skyline detection results based on the PIDNet. This part focuses on evaluating the effectiveness of PIDNet in detecting the sea–skyline in various water surface situations, including sunny, foggy, rainy, evening, and reflective conditions. It is evident that, despite the presence of reflection interference issues in the dataset, the alignment achieved using our technique roughly corresponds to the reference sea–skyline in both situations. The obtained outcome serves as evidence that our sea–skyline detection approach possesses the ability to effectively adapt to diverse environmental conditions.

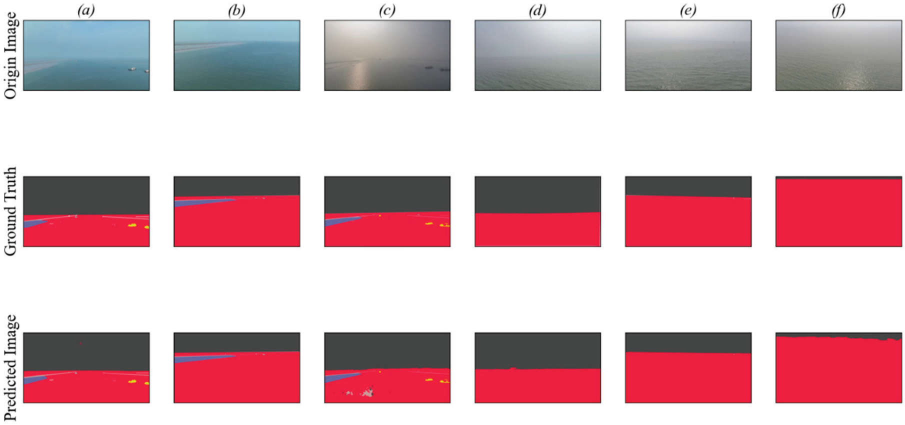


Figure 12. Sea–skyline detection algorithm based on PIDNet (**a**–**f**) represent different angles of view from UAV).

5.3. Stereo Distance Measurement

Based on the aforementioned distance measurement model, it is evident that the outcomes of distance measurement are subject to the influence of factors such as the camera height, pitch angle, and pixel coordinates of the observation site. The gathered single frame image is presented in Figure 13, extracting the area where the sea surface target (black box) is located. We determined the uppermost vertical coordinate value among the pixel values within the designated area, and computed the mean value of the related horizontal coordinates. The pixel coordinates obtained by combining the coordinates serve as the basis for calculating the distance between the observation point and the USV.

Figure 13. Target on water surface.

Through the manipulation of the camera height and pitch angle to collect photos of a consistent area, this study aims to examine the influence of camera height on the accuracy of range measurements. The findings of this investigation are presented in Table 4.

Table 4. The influence of different camera heights on ranging results.

Real Distance (m)	Camera Height (m)	Pitch Angle (Degree)	Test Distance (m)	Relative Error (%)
5.08	2.13	18	4.92	−3.1
	2.32	20	5.05	−0.5
	2.51	22	5.11	0.5
10.25	2.13	18	11.12	8.4
	2.32	20	10.94	6.7
	2.51	22	10.44	1.8
15.18	2.13	18	16.12	6.1
	2.32	20	15.88	4.6
	2.51	22	16.30	7.3
22.41	2.13	18	24.18	7.8
	2.32	20	25.02	11.6
	2.51	22	23.44	4.5
35.20	2.13	18	39.14	11.9
	2.32	20	38.52	9.4
	2.51	22	37.80	7.3

It is verified that the pitch angle has little impact on the ranging results [31]. Therefore, the camera height serves as the primary variable for evaluating the precision of range measurements. The distance values presented in Table 4 were obtained using radar technology. Subsequently, experiments were conducted to validate these measurements. The findings revealed that the relative error ((test distance − real distance)/real distance) diminishes as the camera height increases, but it increases with greater measurement distances.

6. Conclusions

A visual perception technology for coordinated air–sea via a cooperative USV-UAV system is proposed in this paper. The utilization of UAVs can serve as a means to address the limited visibility of unmanned maritime vessels. The primary purpose of this technology is to offer technological assistance in the realm of visual perception for USVs in complex sea regions. The research areas of USV visual perception encompass multi-object recognition, semantic segmentation, and obstacle recognition, which are regarded as highly significant. The main contributions of this paper can be summarized as follows:

1. The cooperative platform utilizes the YOLOX model to carry out a range of sea detection tasks, including ship recognition, various obstacle detection, and the identification of individuals. The findings of the YOLOX study demonstrate the versatility and effectiveness of the collaborative USV-UAV system, and provides improved detection accuracy and increased detection speed compared to other mainstream methods;
2. The PIDNet model is firstly used to handle the semantic segmentation of sea and air. Compared to other approaches, the results indicate that PIDNet has a significant degree of effectiveness in distinguishing between areas that can be navigated and those that cannot be navigated on complex water surfaces. This offers dependable technical assistance for the autonomous navigation of USVs, relying on visual inputs. The PIDNet model also has a strong ability to detect the sea–skyline in different environmental conditions;
3. The application of distance measurements based on monocular camera vision is used to range the distance between the USV and its targets. The results show that this method can effectively estimate the distance of obstacles. Nevertheless, the findings also suggest that, as the distance from the obstruction rises, the precision of the anticipated outcomes will correspondingly deteriorate. Hence, in instances where USVs exhibit high velocities, the utilization of visual ranging technology in isolation is inadequate for ensuring the safety of these USVs.

Author Contributions: C.C. finished the paper, designed the code and performed the experiments; D.L. provided the experimental fields and revised the paper; J.-H.D. carried out the experiments; Y.-Z.L. revised the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No data were used for the research described in the article.

Acknowledgments: We express our gratitude to the China Maritime Safety Administration for their provision of a sea test range.

Conflicts of Interest: The authors declare that they have no known competing financial interest or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Shao, G.; Ma, Y.; Malekian, R.; Yan, X.; Li, Z. A novel cooperative platform design for coupled USV-UAV systems. *IEEE Trans. Ind. Inf.* **2019**, *15*, 4913–4922. [CrossRef]
2. Anderson, K.; Gaston, K.J. Lightweight unmanned aerial vehicles will revolutionize spatial ecology. *Front. Ecol. Environ.* **2013**, *11*, 138–146. [CrossRef] [PubMed]
3. Woellner, R.; Wagner, T.C. Saving species, time, and money: Application of unmanned aerial vehicles (UAVs) for monitoring of an endangered alpine river specialist in a small nature reserve. *Biol. Conserv.* **2019**, *233*, 162–175. [CrossRef]
4. Campbell, S.; Naeem, W.; Irwin, G.W. A review on improving the autonomy of unmanned surface vehicles through intelligent collision avoidance maneuvers. *Ann. Rev. Control* **2012**, *36*, 267–283. [CrossRef]
5. Murphy, P.P.; Steimle, E.; Griffin, C.; Cullins, C.; Hall, M.; Pratt, K. Cooperative use of unmanned sea surface and micro aerial vehicles at Hurricane Wilma. *J. Field Robot.* **2008**, *25*, 164–180. [CrossRef]
6. Mostafa, M.Z.; Khater, H.A.; Rizk, M.R.; Bahasan, A.M. GPS/DVL/MEMS-INS smartphone sensors integrated method to enhance USV navigation system based on adaptive DSFCF. *IET Radar Sonar Navig.* **2018**, *13*, 1616–1627. [CrossRef]
7. Han, J.; Cho, Y.; Kim, J.; Kim, J.; Son, N.-S.; Kim, S.Y. Autonomous collision detection and avoidance for ARAGON USV: Development and field tests. *J. Field Robot.* **2020**, *37*, 987–1002. [CrossRef]
8. Ma, H.; Smart, E.; Ahmed, A.; Brown, D. Radar Image-Based Positioning for USV Under GPS Denial Environment. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 72–80. [CrossRef]

9. Almeida, C.; Franco, T.; Ferreira, H.; Martins, A.; Santos, R.; Almeida, J.M.; Carvalho, J.; Silva, E. Radar based collision detection developments on USV ROAZ II. In Proceedings of the OCEANS 2009—EUROPE, Bremen, Germany, 11–14 May 2009; pp. 1–6.
10. Zhang, J.Y.; Su, Y.M.; Liao, Y.L. Unmanned surface vehicle target tracking based on marine radar. In Proceedings of the 2011 International Conference on Computer Science and Service System (CSSS), Nanjing, China, 27–29 June 2011; pp. 1872–1875.
11. Han, J.; Cho, Y.; Kim, J. Coastal SLAM with marine radar for USV operation in GPS-restricted situations. *IEEE J. Ocean. Eng.* **2019**, *44*, 300–309. [CrossRef]
12. Esposito, J.M.; Graves, M. An algorithm to identify docking locations for autonomous surface vessels from 3-D Li DAR scans. In Proceedings of the IEEE International Conference on Technologies for Practical Robot Applications, Woburn, MA, USA, 14–15 April 2014; pp. 1–6.
13. Su, L.; Yin, Y.; Liu, Z. Small surface targets detection based on omnidirectional sea-sky-line extraction. In Proceedings of the 33rd Chinese Control Conference, Nanjing, China, 28–30 September 2014; pp. 4732–4736.
14. Tao, M.; Jie, M. A sea-sky line detection method based on line segment detector and Hough transform. In Proceedings of the 2nd IEEE International Conference on Computer and Communications (ICCC), Chengdu, China, 14–17 October 2016; pp. 700–703.
15. Kristan, M.; Perš, J.; Sulič, V.; Kovačič, S. A graphical model for rapid obstacle image-map estimation from unmanned surface vehicles. In Proceedings of the 12th Asian Conference on Computer Vision, Singapore, 1–5 November 2014; pp. 391–406.
16. Wang, H.; Mou, X.; Mou, W.; Yuan, S.; Ulun, S.; Yang, S.; Shin, B.-S. Vision based long range object detection and tracking for unmanned surface vehicle. In Proceedings of the 2015 IEEE 7th International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM), Siem Reap, Cambodia, 15–17 July 2015; pp. 101–105.
17. Shi, B.; Zhou, H. Marine Object Recognition Based on Deep Learning. In Proceedings of the International Conference on Computer, Network, Communication and Information Systems (CNCI 2019), Qingdao, China, 27–29 March 2019.
18. Song, X.; Jiang, P.; Zhu, H. Research on Unmanned Vessel Surface Object Detection Based on Fusion of SSD and Faster-RCNN. In Proceedings of the Chinese Automation Congress (CAC), Hangzhou, China, 22–24 November 2019; pp. 3784–3788.
19. Zhan, W.; Xiao, C.; Wen, Y.; Zhou, C.; Yuan, H.; Xiu, S.; Zhang, Y.; Zou, X.; Liu, X.; Li, Q. Autonomous visual perception for unmanned surface vehicle navigation in an unknown environment. *Sensors* **2019**, *19*, 2216. [CrossRef] [PubMed]
20. Xu, D.; Chen, G. Autonomous and cooperative control of UAV cluster with multi agent reinforcement learning. *Aeronaut. J.* **2020**, *126*, 932–951. [CrossRef]
21. Zhang, J.; Wang, W.; Zhang, Z.; Luo, K.; Liu, J. Cooperative Control of UAV Cluster Formation Based on Distributed Consensus. In Proceedings of the 2019 IEEE 15th International Conference on Control and Automation (ICCA), Edinburgh, UK, 16–19 July 2019; pp. 788–793.
22. Li, J.; Zhang, G.; Li, B. Robust adaptive neural cooperative control for the USV-UAV based on the LVS-LVA guidance principle. *J. Mar. Sci. Eng.* **2022**, *10*, 51. [CrossRef]
23. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. Yolox: Exceeding yolo series in 2021. *arXiv* **2021**, arXiv:2107.08430.
24. Bochkovskiy, A.; Wang, C.; Liao, H.-Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
25. Zhang, H.; Cisse, M.; Dauphin, Y.N.; Lopez-Paz, D. Mixup: Beyond empirical risk minimization. *arXiv* **2017**, arXiv:1710.09412.
26. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
27. Zheng, Z.; Wang, P.; Liu, W.; Lu, T.; Ye, R.; Ren, D. Distance-iou loss: Faster and better learning for bounding box regression. *Proc. AAAI Conf. Artif. Intell.* **2020**, *34*, 12993–13000. [CrossRef]
28. Woo, S.; Park, J.; Lee, J.K.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European conference on computer vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
29. Towaki, T.; David, A.; Varun, J.; Sanja, F. Gated-scnn: Gated shape cnns for semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 5229–5238.
30. Pei, X.Y. *Autonomous Navigation Technology of Unmanned Surface Vehicle*; Shanghai Maritime University: Shanghai, China, 2015.
31. Zhao, M.H.; Wang, J.H.; Zheng, X.; Zhang, S.J.; Zhang, C. Monocular vision based water-surface target distance measurement method for unmanned surface vehicles. *Transducer Microsyst. Technol.* **2021**, *40*, 47–54.

Journal of
*Marine Science
and Engineering*

Article

Ship Detection under Low-Visibility Weather Interference via an Ensemble Generative Adversarial Network

Xinqiang Chen [1], Chenxin Wei [1], Zhengang Xin [1], Jiansen Zhao [2,*] and Jiangfeng Xian [1]

[1] Institute of Logistics Science and Engineering, Shanghai Maritime University, Shanghai 201306, China; chenxinqiang@stu.shmtu.edu.cn (X.C.); 202130510052@stu.shmtu.edu.cn (C.W.); 202330510005@stu.shmtu.edu.cn (Z.X.); jfxian@shmtu.edu.cn (J.X.)
[2] Merchant Marine College, Shanghai Maritime University, Shanghai 201306, China
* Correspondence: jszhao@shmtu.edu.cn

Abstract: Maritime ship detection plays a crucial role in smart ships and intelligent transportation systems. However, adverse maritime weather conditions, such as rain streak and fog, can significantly impair the performance of visual systems for maritime traffic. These factors constrain the performance of traffic monitoring systems and ship-detection algorithms for autonomous ship navigation, affecting maritime safety. The paper proposes an approach to resolve the problem by visually removing rain streaks and fog from images, achieving an integrated framework for accurate ship detection. Firstly, the paper employs an attention generation network within an adversarial neural network to focus on the distorted regions of the degraded images. The paper also utilizes a contextual encoder to infer contextual information within the distorted regions, enhancing the credibility of image restoration. Secondly, a weighted bidirectional feature pyramid network (BiFPN) is introduced to achieve rapid multi-scale feature fusion, enhancing the accuracy of maritime ship detection. The proposed GYB framework was validated using the SeaShip dataset. The experimental results show that the proposed framework achieves an average accuracy of 96.3%, a recall of 95.35%, and a harmonic mean of 95.85% in detecting maritime traffic ships under rain-streak and foggy-weather conditions. Moreover, the framework outperforms state-of-the-art ship detection methods in such challenging weather scenarios.

Keywords: ship detection; adverse weather; image restoration; improved YOLOv5; intelligent maritime transportation

Citation: Chen, X.; Wei, C.; Xin, Z.; Zhao, J.; Xian, J. Ship Detection under Low-Visibility Weather Interference via an Ensemble Generative Adversarial Network. *J. Mar. Sci. Eng.* **2023**, *11*, 2065. https://doi.org/10.3390/jmse11112065

Academic Editor: Sergei Chernyi

Received: 6 October 2023
Revised: 24 October 2023
Accepted: 26 October 2023
Published: 29 October 2023

1. Introduction

With the rapid advancement of artificial intelligence and computer vision technologies, the traditional navigation methods of maritime ships are undergoing a process of transformation and elevation. Intelligent maritime traffic monitoring systems and automated ship navigation are gradually becoming tangible realities (Liu et al., Cheng et al., Volden et al. [1–3]). Therefore, it is imperative to accurately detect maritime traffic entities (such as ships, buoys, etc.) based on ship vision navigation or port surveillance videos in order to make precise navigational control decisions. This plays a pivotal role in enhancing the safety of automated ship navigation and maritime ship passage within waterways (Forti et al. [4]). To detect maritime navigating ships, various types of sensors, such as cameras and radar, are commonly employed in automated ship navigation. However, the unique maritime weather conditions and locations often expose these systems to adverse weather, such as fog and rain streaks, resulting in the deterioration of ship-monitoring video data (Bahnsen and Moeslund, and Li et al. [5,6]). The presence of rain streaks and fog in the atmosphere severely impacts the visibility of monitoring scenes. Low visibility is inefficient for accurately detecting maritime ships and increases the risk of maritime traffic accidents. Consequently, the development of effective image restoration techniques becomes crucial to achieve improved visual appearance or distinctive features. Providing clear maritime images to detection systems can significantly enhance the detection performance of maritime ships at sea (Fu et al., Lu et al. [7,8]).

Computer vision technology has become a crucial method for autonomous ship navigation (autonomous driving) and intelligent transportation applications. It can detect and recognize target objects in various scenarios with high precision, while also providing data support for intelligent control decision-making in the transportation field (Yu et al., Yao et al. [9,10]). Previous research has primarily focused on capturing high-quality maritime traffic video data. Wang et al. proposed a rapid and accurate ship detection algorithm based on YOLOv4, which incorporates K-means clustering, model structure refinement, and the Mixupfan method (Wang et al. [11]). Li et al. utilized a background filtering network for rapid filtering of background areas and employed a fine-grained ship classification network for the detection and classification of ship targets (Li et al. [12]). Fence et al. proposed a fast ship detection method based on multi-scale gradient features and a multi-branch support vector machine (Feng et al. [13]). Similar studies can be found in (Shao et al., Lv et al. and Chen et al. [14–16]). For some scenarios where video data cannot be directly obtained, previous research relied on limited exploration using radar data and multi-source data fusion. Chen et al. proposed a study similar to Radar-YOLONet that uses radar images for object detection (Chen et al. [17]). Wang et al. proposed a deep radar object-detection method called RODNet based on cross-fusion supervision of radar–camera data (Wang et al. [18]). Xu et al. used a multiple linear rescaling scheme to quantize the original satellite images into 8-b images, and proposed an adaptive weighting scheme to detect the loss between ships (Xu et al. [19]). Similar studies can be found in (Guo et al., Bai et al. [20,21]). The acquisition of multi-source data usually depends on special physical sensors, which are highly susceptible to the water environment and have high maintenance costs (Shang et al., Lin et al. [22,23]). With the development of deep learning, feature enhancement has been used to strengthen the perception of low-feature targets in low-visibility scenes. This addresses the issues of low accuracy and efficiency in traditional object-detection algorithms. Wang et al. constructed a new feature enhancement module (FEM) and utilized an attention mechanism to achieve real-time accurate detection of multiple targets in foggy conditions (Wang et al. [24]). M. Hassaballah et al. utilized an image enhancement scheme to achieve robust detection and tracking of vehicles (Hassaballah et al. [25]). While these methods can effectively detect target objects, they may not fully address the unique characteristics of maritime traffic environments, such as tides, water currents, and channel divisions. Therefore, they may not guarantee the safety of maritime traffic. It is important to consider these factors when developing and implementing object-detection algorithms for maritime traffic environments.

To address these problems, the paper presents an integrated framework for maritime ship detection under adverse weather conditions using computer vision techniques. This framework leverages adversarial neural networks to generate attention maps that focus on distorted regions within the images. These attention maps guide the contextual autoencoder in performing local feature inference, achieving a rational and effective restoration of distorted areas in low-visibility images. Moreover, the restored images are concurrently fed into the discriminative network to facilitate the evaluation of the restored regions in the generated images. This process serves as feedback to guide the generative network in achieving optimal results for the enhancement of low-visibility images. Next, a weighted bidirectional feature pyramid network (BiFPN) is introduced to achieve rapid multi-scale feature fusion, enhancing the accuracy of maritime ship detection (Tan et al. [26]). This involves iteratively applying top-down and bottom-up multi-scale feature fusion to enhance the accuracy of ship detection in repaired low-visibility images. Our proposed framework is evaluated on the synthetic SeaShip dataset, which includes challenges related to low-visibility conditions such as rain streaks and fog, as well as small-target detection. Experimental results show that the model framework we proposed exhibits effectiveness and superiority over existing algorithms. The main contributions of this work are summarized as follows:

- The paper has proposed a novel integrated framework for detecting and recognizing ships navigating in low-visibility maritime environments.
- The paper has proposed the use of a weighted BiFPN in the YOLOv5 detector, achieving top-down and bottom-up multi-scale feature fusion to improve the accuracy of ship detection in low-energy image restoration.
- The paper's proposed framework achieves an average accuracy of 96.3%, a recall of 95.35%, and a harmonic mean of 95.85% in detecting maritime traffic ships under rain streak and foggy weather conditions.

2. Materials and Methods

The proposed framework for ship detection in low-visibility maritime images in this paper consists of two main logical steps: image restoration for maritime traffic and ship detection in maritime traffic, as shown in Figure 1. Firstly, an attention map is generated in the recurrent network within the generative network to identify low-visibility areas in the image that are disturbed by rain streaks and foggy weather. Meanwhile, the context autoencoder within the generative network performs local inference and restoration on the rain streaks and fog areas, enabling them to generate more realistic local images. More specifically, firstly, an attention map is generated in the recurrent network within the generative network to identify low-visibility areas in the image that are disturbed by rain streaks and foggy weather. Meanwhile, the context autoencoder within the generative network performs local inference and restoration on the rain streaks and fog areas, enabling them to generate more realistic local images. More specifically, firstly, images of rainy and foggy weather are input into the model framework. The images, after passing through the generative attention map network (Residual Block and LSTM + Convs modules), generate an attention map for the rain streaks and fog (low-visibility) areas of the two-dimensional image. This enhances the perceptibility of the distorted areas and provides guidance for subsequent image restoration. Secondly, the generated attention map and the original image are passed into the generative contextual autoencoder (Convs + ReLu, Dilated Convs + ReLu, and Deconv + avgpool + ReLu modules). This allows for the extraction of surrounding structure and feature information from the distorted areas. By combining these extracted features, contextual information is inferred and restored, resulting in the generation of relatively intact images. Meanwhile, the restored images are input into the discriminator for image quality assessment. Finally, the restored images are input into the detection model. Since some areas may have lower restoration quality during the image restoration process, a multi-scale fusion method is used to achieve detection and recognition of low-resolution ships. The discriminative network evaluates whether the images generated by the generative network are realistic, and provides feedback to the generative network. Next, the detection model incorporates BiFPN into YOLOv5 for multi-scale feature fusion, further enhancing the accuracy of ship detection in the low-visibility image restoration regions.

2.1. Rain-Streak and Fog Imaging Modeling

Rain streaks and fog in images can affect the detection performance of ships in both human and computer vision. Therefore, removing rain streaks and fog, which means restoring blurry images to clean images, is an important problem in computer vision. To better remove rain streaks and fog, we first mathematically model the rain streaks and foggy scenes. The widely used rain streaks model scene (Quan et al., Luo et al. and Li et al. [27–29]) modeling formula is shown in Equation (1):

$$C = F_{img} - S \tag{1}$$

where F_{img} represents the rain streak image, C represents the background of clean water transportation and S represents the rain streaks. Therefore, we need to remove the rain streaks S.

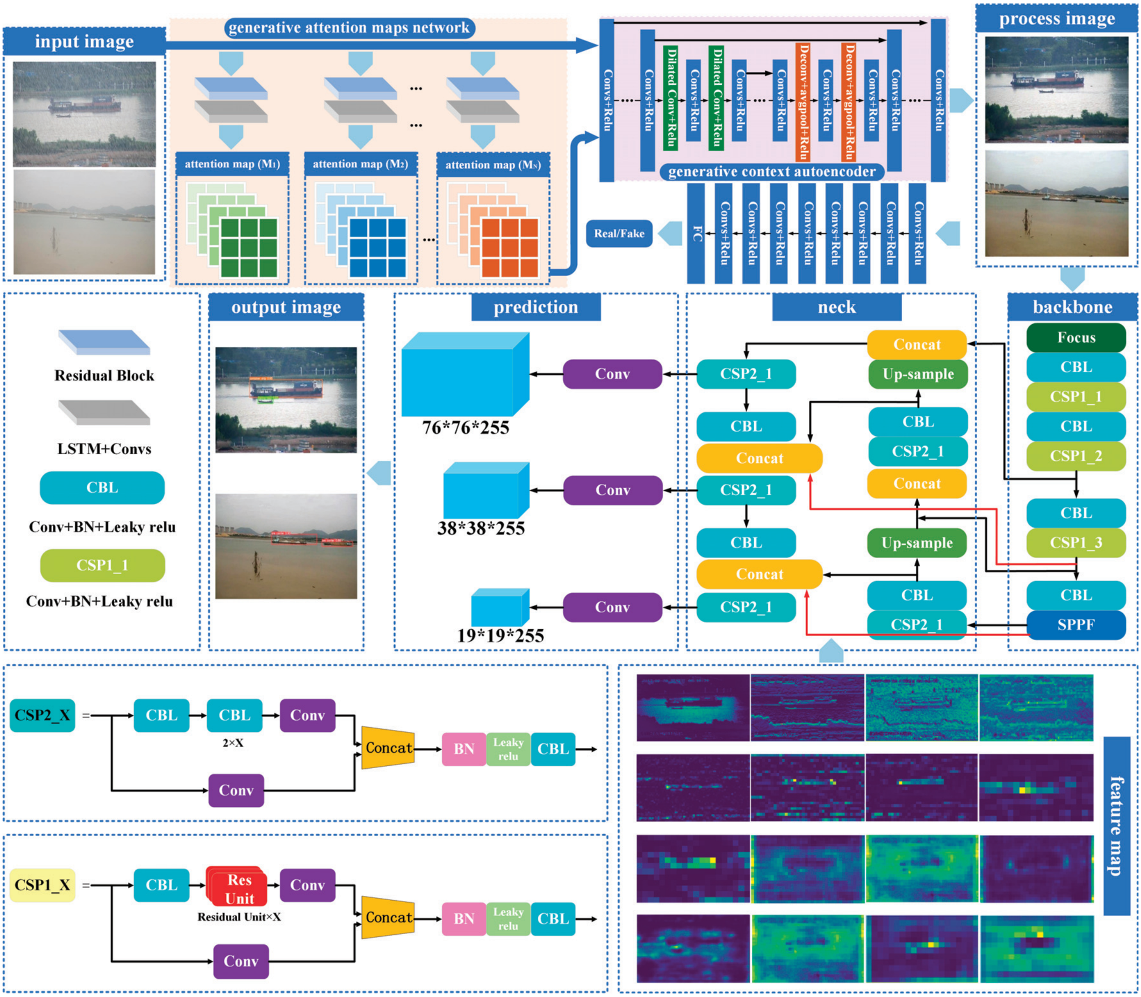


Figure 1. Schematic overview for ship detection framework for distorted image restoration under adverse weather.

For the simulation of foggy conditions, we have found that the most realistic methods predominantly utilize depth-based techniques to synthesize their own datasets. The widely used foggy model scene (Nayar and Narasimhan, Narasimhan and Nayar [30,31]) modeling formula is shown in Equation (2):

$$N(y) = \frac{H(y) - L}{e^{-\varphi d(y)}} + L \tag{2}$$

where $N(x)$ represents the clean image, $H(y)$ represents the fog image, and L represents the global atmospheric light. $e^{-\varphi d(y)}$ is the transmission map. φ is the attenuation coefficient and $d(y)$ refers to the image scene depth.

2.2. Generative Adversarial Network

Generative adversarial networks (GAN) (Goodfellow et al. [32]) have gained widespread application in the field of image restoration in recent years, and have yielded significant results (Qian et al. [33]). A GAN network consists of two components: a generator and a discriminator. The generator takes random noise as input and produces a feature vector representing the target as output. The discriminator is a classifier that takes a vector as input and outputs a judgment on whether that vector is real or fake. More specifically, the generator takes low-visibility ship images as input and generates the restored image after passing through the attention map and context autoencoder within the generator. Furthermore, the restored image is used as input to the discriminator to distinguish between the generated images and real images, thereby guiding the generator to produce more realistic images. Finally, the sigmoid function outputs 1 or 0, indicating real or fake for the restored image. To make the model more efficient, the generator and discriminator evolve in a minimax game, where they mutually constrain and encourage each other to achieve more realistic image outputs. The optimization objective function of the GAN model is shown in Equation (3):

$$\min_{G}\max_{D} V(D,G) = E_{T\sim R_{imgnoise}}[\log(D(T))] + E_{B\sim P_{imgdrop}}[\log(1-D(G(B)))] \tag{3}$$

where G represents the generator, D represents the discriminator, and B represents the low-visibility image input to the generator adversarial network. T represents the clear image sample corresponding to the low-visibility image of B. E represents the expected value. $D(T)$ is the output of the discriminator for the real clear image T, which is a probability value. $D(G(B))$ represents the output of the discriminator for the restored images generated by the generator $G(B)$.

2.3. Generative Attention Map Network

Rain streaks and fog in the atmosphere can significantly reduce the visibility of the maritime background in monitoring equipment, and cause image distortion. The distorted regions are perceptible to the human eye, but not explicitly delineated in computer-generated images. To address this problem, the paper proposes a method to identify the distorted regions by generating attention maps using adversarial neural networks. The generation of attention maps in this context is inspired by the principles of recurrent neural networks (RNNs) and residual networks. More specifically, as shown in Figure 1, the recurrent network consists of five layers of ResNet network layers and LSTM + Convs. The ResNet network is responsible for pre-extracting global features from the image (He et al. [34]). After passing through the LSTM + Convs module, both global and local features of the distorted image are fed back into the convolutional layers to generate the attention map.

The generated attention map is then concatenated with the original image and fed into the next identical module. In this process, the attention map from the previous layer guides the subsequent layer of the same network to focus more on the distorted regions. This iterative process is repeated in a loop. The attention map is essentially a two-dimensional array of the same size as the original image, where each element's value ranges from 0 to 1. The attention map is a non-binary mapping, signifying that attention gradually increases from non-raindrop regions to raindrop regions, with values varying even within the raindrop regions. This gradual increase in attention is meaningful because the areas around raindrops also need attention, and the transparency within the raindrop regions varies in reality. Therefore, a higher value in a specific region of the array indicates more attention from the attention map to that area, enabling focused restoration of the distorted regions in the image.

2.4. Generative Context Autoencoder

The framework obtains the distorted regions of the image through attention maps, but the original image information collected is missing (distorted) under the interference of rain streaks and fog. Therefore, the network uses a context autoencoder to help the generator produce a clear and complete image guided by the attention map, which is equivalent to restoring and repairing low-visibility images. Figure 1 shows the architecture of the context autoencoder. The network introduces a dilated convolution network to increase the size and perception ability of the receptive field, so as to better capture the global features of the input attention map and the context information of the distorted regions. At the same time, the generative network ensures the restoration of high-resolution in the distorted regions by introducing the Deconv + avgpool module. Since we need to extract image feature information from different network layers to infer more context information, we set up two loss functions in the context encoder: multi-scale loss and perceptual loss. The multi-scale loss can effectively extract image features to obtain context information on different scales and form outputs of different sizes to capture the details and structure information in the image. The use of multi-scale loss is effective in extracting image features to obtain context information on different scales and generating outputs of varying sizes to capture fine-grained details and structural information in the image. The objective function for the multi-scale loss is shown in Equation (4):

$$\varepsilon_M(\{F\},\{R\}) = \sum_{i=1}^{M} W_i \varepsilon_{MSE}(F_i, R_i) \tag{4}$$

where F_i represents the i-th output extracted from the context autoencoder, and R_i represents the i-th output ground-truth image information, which has the same scale as the F_i. ε_{MSE} represents the mean squared error between the output at different scales and the corresponding ground-truth image, and $\{W\}_{i=1}^{M}$ represents the weight magnitudes for different scales.

To generate more realistic images, the generative network needs to pay attention to the high-level structure and content of the image, rather than pixel-level noise. This approach prioritizes the perceptual features of the image over subtle pixel variations. Besides the pixel-based multi-scale loss in our image-based approach, the generative network also uses perceptual loss to ensure visual consistency and feature fidelity between the generated image and the target image. The objective function for the perceptual loss is shown in Equation (5):

$$\varepsilon_P(O,T) = \varepsilon_{MSE}(VGG(O), VGG(T)) \tag{5}$$

where O represents the output image of the generative network, which is the image after the restoration process. O is obtained by the generator G using the input image I and the attention map. $VGG(O)$ represents the image features extracted from O using a pre-trained VGG-16 network, while $VGG(T)$ represents the image features extracted from T. ε_{MSE} represents the mean-squared-error loss function, which calculates the difference between the features of the reconstructed image and the ground-truth image after the restoration process. The VGG-16 mentioned in this paper refers to a pretrained convolutional neural network (CNN) that is solely used for feature extraction from images. To summarize, the loss function for low-visibility image restoration is shown in Equation (6):

$$\varepsilon_G = \varepsilon_M(\{F\},\{R\}) + 10^{-2}\varepsilon_P(O,T) + \varepsilon_{GAN}(O) + \varepsilon_{A\sim M} \tag{6}$$

where $\varepsilon_{A\sim M}$ represents the loss value of the attention map with the distortion region mask and $\varepsilon_{GAN}(O) = 1 - D(G(O))$. To verify the authenticity of the repaired distorted images, the generative network utilizes attention maps to guide the discriminator's focus on the restored regions, evaluating the quality of the generated images based on both global- and local-image content. Additionally, the generative network employs fully connected layers to determine the authenticity of the restored low-visibility images.

2.5. Detection Method

Considering the navigation control decisions of maritime ships, the ability to detect other ships at sea in real time and accurately handle emergency situations (such as collision avoidance and locating missing vessels) is crucial. The YOLO model, as a one-stage detection model, has certain advantages in this regard. While two-stage object-detection models may offer superior accuracy, they do not stand out in terms of real-time performance. Furthermore, the equipment and monitoring devices on intelligent unmanned ships typically lack the computational capacity to support higher-precision target-detection algorithms. Moreover, this model not only detects the positions of ships but also classifies different types of ships. Therefore, choosing YOLOv5 for improvement can be effectively deployed in the ship's driving system to enhance the efficiency of maritime traffic management. YOLOv8 is the latest model-detection framework in the YOLO series. Although YOLOv8 has better accuracy and speed on GPU devices than YOLOv5, making it a better choice for real-time object detection, it is important to consider the device limitations of the ship perception system and the lack of GPU support. YOLOv5, with its smaller model parameters and ease of training, becomes a more suitable solution for such problems while maintaining a certain level of accuracy.

In this section, the detailed specifics of the YOLOv5 detector will be introduced. The network structure of YOLOv5 consists of three main components: backbone, neck and head. As shown in Figure 1, first, the restored low-visibility ship image is preprocessed (scaling the input ship image to a uniform size), and then sent to the backbone network, which transforms the original input image into multi-layer feature maps. The backbone network of YOLOv5 consists of CBL, CSP1_X and SPPF modules. The CBL module is composed of a convolutional layer, a batch normalization layer and an activation function. This module is mainly used to extract the local spatial information of the ship features and normalize the feature information. The CSP1_X convolutional module splits the input feature map into a backbone convolutional layer and a branch convolutional layer. The backbone convolutional layer uses 1*1 convolution to reduce the channel number and the parameter amount. The branch convolutional layer further extracts feature extraction on the feature map. This design effectively reduces the computational load by reducing the parameter count while enhancing the feature extraction capability. Additionally, this module adopts a residual approach, which further enhances the model's expressive power. The detection model employs spatial pyramid pooling (SPPF) to pool features from input feature maps of different scales, enabling the model to capture maritime ship objects at various scales. Then, the detection model introduces the BiFPN feature pyramid structure in the neck network to handle the ships of different scales and sizes that are distorted by image restoration.

In order to recognize different types of ships, the key is to collect image features of different ships (including color, shape, etc.) through supervised learning for further processing. Therefore, by inputting the annotated dataset into the model training, anchors of different sizes and aspect ratios are preset. The setting of these anchors can effectively divide the prediction box space into several subspaces, thereby reducing the difficulty of recognizing different types of ships. Through down-sampling by factors of 32, 16, and 8, different sizes of feature maps (20 × 20, 40 × 40, and 80 × 80) are produced. And these feature maps are input into the neck, where deep semantic features and low-level semantic features are fully fused. Each feature region is then input into the prediction head. Finally, the obtained feature maps are input into the head for feature regression and classification, fitting the best bounding boxes and positions for different types of ships.

2.6. Introduce the BiFPN Structure

Due to the attention adversarial network in the restoration process of low-visibility images, the distortion in the restoration of target ships in the images and the uneven distribution of small ships cause the image features of such target ships to be insignificant. Due to issues such as image distortion and occlusion, during the later stages of model training, the features extracted from the restored normal regions are significantly more prominent than those from the un-restored normal regions. The features of the restored normal area are found to be more significant than those of the un-restored normal area during the subsequent feature extraction, leading to a selective focus on the significant features. Therefore, to solve such problems, the YOLO series previously adopted the FPN (feature pyramid network) structure. As shown in Figure 2a, this network structure adopts a top-down approach to aggregate multi-scale features, allowing the high-level feature map to be transmitted to the low-level feature map (different square colors represent different feature maps, the arrow direction represents the direction of feature transmission.) and aggregating features on different scales. However, this information transmission can only be unidirectional, and cannot be reversed. More specifically, as the number of down-sampling or convolution operations increases, the receptive field of the high-level feature map gradually increases, and the overlap area between the receptive fields also continues to increase. At this time, the information represented by the pixel points is the information for a region, which has stronger semantic information and is more conducive to the classification of different ships. The low-level feature map can utilize more fine-grained feature information, ensuring that the network can capture more details. It has stronger positional information, which is more conducive to the positioning of ships. Then, the process of transmitting high-level features can lead to the loss and degradation of feature information. Therefore, a unidirectional FPN cannot effectively solve such problems.

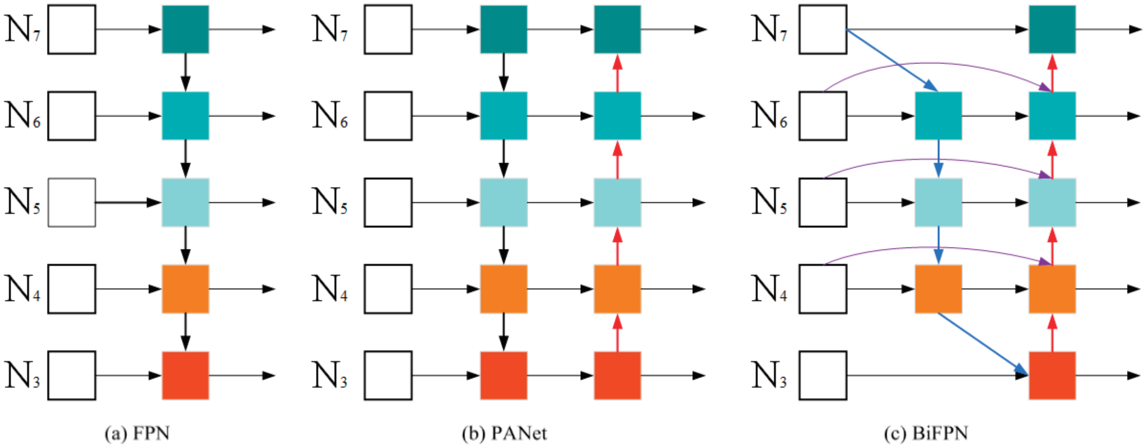


Figure 2. Comparative schematic diagram of different feature-fusion network structures: (**a**) FPN, (**b**) PANet, and (**c**) BiFPN.

To better solve the FPN problem, it is necessary to create a new path from bottom to top, transmitting the positional information to the predicted feature map as well, so that the predicted feature map simultaneously possesses higher semantic information and positional information (which is beneficial for object detection). As shown in Figure 2b, PANet proposes a bidirectional feature network from top to bottom and from bottom to top, and generates a new feature map from bottom to top. This is followed by adaptive feature pooling in the later stages. This network structure enhances the feature expression capability of the backbone network, allowing different target ships to choose different feature maps. This avoids the one-to-one matching between ship size and network depth. However, the ROI of this network structure can only rely on a single layer of features, leading to the problem of information loss from other feature layers. Then, the dual transmission paths can also lead to insufficient information transmission.

Therefore, in order to better solve the above problems, this study introduces the BiFPN weighted bidirectional (top-down + bottom-up)-feature pyramid network structure as a multi-scale feature-fusion method, and combines the idea of multi-level feature fusion. This is an effective bidirectional cross-scale weighted feature-fusion method, which enables the fusion and transfer of features from high-resolution ship images and low-resolution ship images. This further avoids the problems of erroneous ship detection and recognition caused by occlusions and image restoration distortions between ships of different sizes, and better balances the feature information on different scales under different circumstances. As shown in Figure 2c, the feature pyramid network structure is shown in the diagram, where the left part represents the input section consisting of feature maps from the backbone network. These feature maps have different levels and scales of information. Typically, lower-level feature maps have higher resolution but relatively less semantic information, while higher-level feature maps have lower resolution but contain more semantic information. The input feature maps are fused and propagated through various paths. The top-down path starts from higher-level feature maps and gradually increases the resolution of the feature maps through upsampling or interpolation operations to obtain higher-resolution feature maps. The bottom-up path starts from lower-level feature maps and gradually decreases the resolution of the feature maps through pooling or convolution operations to capture broader receptive fields and more detailed information. The lateral connections are used to fuse the feature maps from the top-down and bottom-up paths. By using 1×1 convolution operations, the channel dimensions of the feature maps from the bottom-up path are matched to be added to or concatenated with the feature maps from the top-down path. Multiple iterations are performed to enrich and diversify the levels of the feature pyramid. Finally, the fused and propagated feature map is output on the right side of the image. More specifically, the blue lines are the top-down pathways, which convey the semantic information of the high-level features; the red lines are the bottom-up pathways, which convey the location information of the low-level features; the purple lines are the newly added edges between the input nodes and the output nodes at the same level (N4, N5, N6), which fuse more image features without adding too much cost. Meanwhile, In the BiFPN network, nodes with only a single input edge are eliminated. This is because a node with just one input edge that does not perform feature fusion contributes minimally to the feature network that integrates different features. Therefore, removing such a node has a negligible impact on our network, while it simplifies the bidirectional network. This is applicable to the first node on the right of N7. If the original input node and the output node are at the same level, the network will add an extra edge between the original input node and the output node. This allows for the fusion of more features without significantly increasing costs, thereby improving the efficiency of ship detection and recognition.

3. Experimental Design

3.1. Data Description

To validate the proposed maritime ship detection framework for low-visibility scenarios at sea, it is essential to consider all influencing factors in the maritime environment to ensure the reliability and authenticity of the model validation. The SeaShip dataset is acquired by the monitoring cameras in a deployed coastline video-surveillance system. This dataset includes labels for various types of ships and high-precision bounding boxes, and covers all possible imaging variations, such as different scales, parts of the hull, lighting, viewpoints, backgrounds, and occlusions. This dataset comprises 7000 images, which are divided into two subsets: one simulates rain-streak conditions, and the other simulates foggy weather. Both datasets are split into training, testing, and validation sets, in a 2:1:1 ratio. The dataset encompasses six types of ships, namely ore carriers, bulk cargo carriers, container ships, general cargo ships, fishing boats, and passenger ships (Shao et al. [35]).

This dataset has collected maritime ship navigation data under forty-five different background conditions. Ship detection accuracy is often affected by background changes which pose challenges for separating foreground target ships from complex background environments. Note that the datasets also involved image distortion, occlusions, hull parts and small-target detection (i.e., small ship imaging size) interferences. Details for the datasets can be found in Table 1. The proposed method was implemented with PyTorch 1.7.1 framework and Python 3.7. The operating system is Ubuntu 20.04 OS, and the CPU is Intel(R) Xeon(R) Gold 6230R CPU @ 2.10 GHz. The GPU used for the experimental platform is Quadro RTX 5000.

Table 1. Information from Marine ship data.

Ship Category	Resolution	Image Distortion	Small Target Detection	Ship Obstruction	Hull Parts
Ore carrier	1920 × 1080	√	√	√	√
Bulk cargo carries	1920 × 1080	√	√	√	√
Container ship	1920 × 1080	√	/	√	√
General cargo ship	1920 × 1080	√	√	√	√
Fishing boat	1920 × 1080	√	√	√	√
Passenger ship	1920 × 1080	√	/	√	√

(Symbol √ indicates the situation that exists in the dataset.)

3.2. Evaluation Indicators

To validate the performance of the GYB framework proposed in this study, our approach involved five metrics: recall (R), F1-score, precision (P), and the average precision $AP_{0.5}$ and $AP_{0.5:0.95}$ to quantitatively evaluate the performance of the framework. the performance of the framework. Firstly, we need to introduce some common variables. TP (true positive) represents the positions and labels of the ships detected by different algorithms consistent with the ground truth. TN (true negative) represents that both the ground truth and detected ship labels are negative (which correctly predict the negative samples). FN (false negative) means that the different detection algorithms recognize the correct ship positions and labels as wrong (this sample is a positive sample). FP (false positives) means the different detection algorithms predict the wrong ship positions and labels as correct (this sample is a negative sample). Precision is the ratio of correctly predicted ship positions and labels among all predicted true ship labels within the range of [0, 1]. Recall represents the percentage of correctly predicted true ship labels among the total actual true ship labels, and lies within the range of [0, 1].

To better evaluate the performance of different algorithms, the F1-score, which is the harmonic mean, precision (P) and recall (R) are introduced. The F1-score reaches its optimum only when both precision and recall tend toward their maximum values. The average precision (AP) can be obtained by calculating the area under the precision–recall (P-R) curve, which is bounded by the horizontal and vertical axes. $AP_{0.5}$ and $AP_{0.5:0.95}$ are commonly utilized metrics, respectively signifying average precision values at an IOU (intersection-over-union) threshold of 50% and the mean values across IOU thresholds, ranging from 50% to 95%. In this study, the detection performance of the proposed method was assessed using $AP_{0.5}$ and $AP_{0.5:0.95}$. In accordance with Equations (7)–(10), the positions of the detected ships are closer to the ground truth when the values of P, R, F_1, $AP_{0.5}$ and $AP_{0.5:0.95}$ are larger. The frames-per-second (fps) is introduced as a performance criterion to assess the real-time performance of this framework, for which the calculation formula is shown in Equation (11):

$$P = \frac{TP}{TP + FP} \tag{7}$$

$$R = \frac{TP}{TP + FN} \tag{8}$$

$$F_1 = 2\frac{P \times R}{P + R} \tag{9}$$

$$AP = \int_0^1 P(r)dr \tag{10}$$

$$fps = \frac{1}{COT} \tag{11}$$

where COT represents the average time consumed per frame in the ship validation dataset.

4. Discussion and Result

4.1. Discussion

To illustrate the entire workflow, we provide a descriptive output of each step within the proposed framework in this paper. In the context of maritime surveillance video imaging, the interference caused by rain streaks and fog leads to reduced visibility in monitoring video data. Furthermore, the complexity and diversity of maritime ship navigation environments exacerbate this issue. Existing ship-detection algorithms tend to exhibit abnormal detection behavior in such scenarios, often resulting in missed detections, where they fail to accurately fit bounding boxes around navigating ships.

As shown in scenario one in Figure 3, under the interference of rain streaks and the complex water traffic environment, SSD and Faster_Rcnn failed to correctly distinguish the features of water obstacles and ships, resulting in false and missed detections of ships (purple dashed box). As shown in scenario two in Figure 3, the interference of rain streaks reduced the visibility, and the sailing ships were far from the monitoring equipment, resulting in YOLOv3 and SSD missing the detection of small-ship targets. Meanwhile, the YOLOv3 detector failed to generate the correct ore-carrier bounding box. The significant difference between the edge features of the general cargo ship and the edge features of the cargo on board caused Faster_Rcnn to generate multiple incorrect candidate regions (proposal regions), resulting in fishing boats and general cargo ships being generated (i.e., a single ship corresponds to multiple bounding boxes).

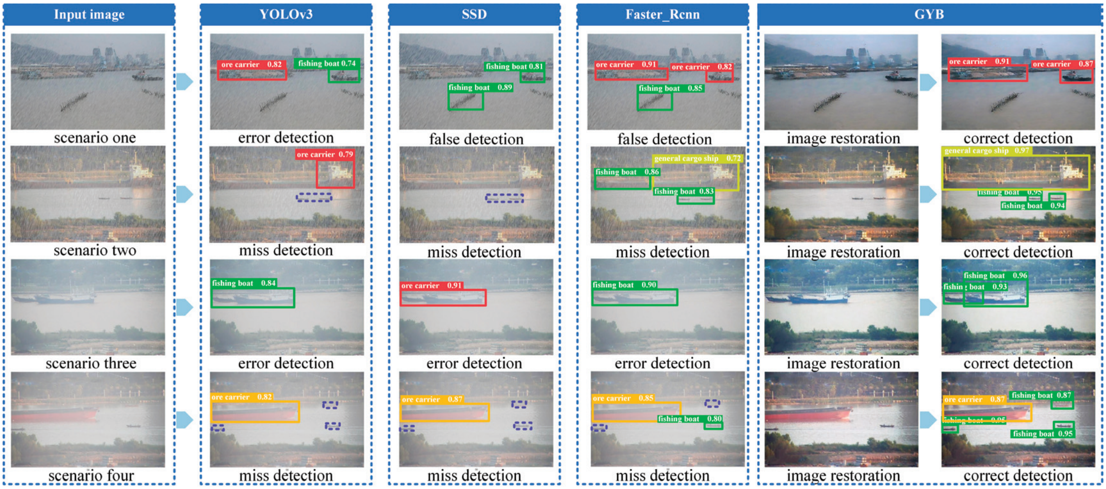


Figure 3. Comparative diagram of the effects of YOLOv3, SSD, Faster_Rcnn, and GYB on the detection of ships in typical water transportation under different weather disturbances.

As shown in scenario three in Figure 3, the visibility of the video image data collected under the interference of fog is significantly reduced, and the ships are occluded by each other; the features of small ships are similar to those of large ships, resulting in SSD, YOLOv3 and Faster_Rcnn being unable to distinguish fine-grained ship features, causing ship misdetection (i.e., multiple ships correspond to only one bounding box). As shown in scenario four in Figure 3, the small-target fishing boats in this scene are confused with the surrounding environmental features, and SSD, Faster_Rcnn and YOLOv3 cannot identify effective regions, resulting in some small ships being missed (purple dashed box).

By restoring the low-visibility images and obtaining a clear image as the input of the enhanced YOLOv5 detector, our proposed algorithm framework can effectively address the challenges of low-visibility detection in the aforementioned scenarios and achieve multi-scale fusion for water ship detection. Then, the restoration of the distorted area of the image becomes more important. As shown in Figure 4, the visualization of the low-visibility region attention-map learning process in the proposed framework is presented. This process focuses on the low-visibility regions by using the attention maps generated by the adversarial neural network, while understanding the structure and edge features around the low-visibility regions. And by guiding the context encoding encoder with the attention map, the global and local features and relations of the low-visibility regions are captured, generating high-quality image regions. In this way, we can ensure that the detector can extract more realistic and effective ship image features. In order to better quantify the effects of image dehazing and deraining, the PSNR (peak signal-to-noise Ratio) and SSIM (structural similarity) are introduced as performance criteria to assess the results after image restoration. PSNR (peak signal-to-noise ratio) is a reference value for evaluating image quality, while SSIM (structural similarity) is an indicator for measuring the similarity between the restored image and the real clear image. The evaluation indicators are summarized in Table 2. It can be observed that the PSNR metrics of the framework for image deraining and defogging are 30.32 and 32.68, respectively, and the SSIM metrics are 0.9289 and 0.9360, respectively. Overall, it has achieved a good quality of image restoration, providing an important foundation for subsequent ship detection and recognition.

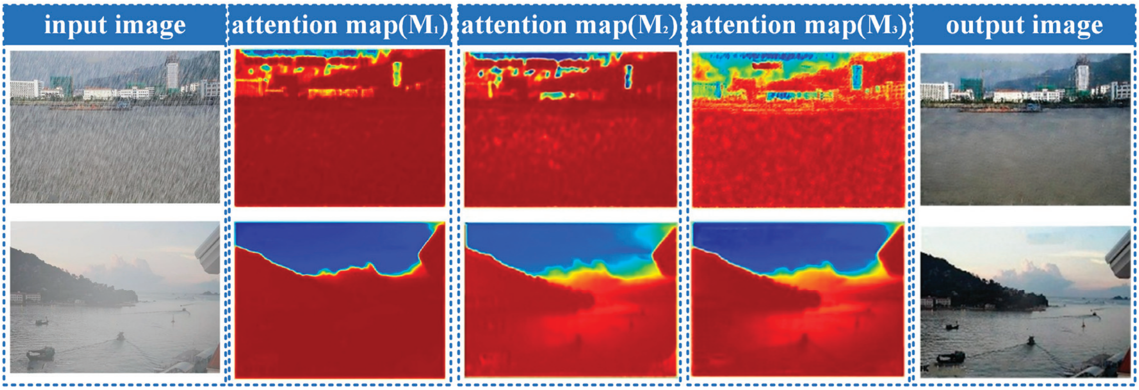


Figure 4. Schematic diagram of the process of image restoration by the context encoder guided by the attention map.

Table 2. Quantitative evaluation results of image deraining and dehazing.

Datasets	Evaluation Indicators	
	PSNR	SSIM
Rain streaks	30.32	0.9289
Fog	32.68	0.9360

4.2. Results

By calculating the data difference between the real position and the detected position of each type of ship in low-visibility maritime surveillance videos, we further quantified the experimental data regarding the framework's performance. The evaluation results are summarized in Table 3. For rain streak scene, the evaluation metrics (P,R,F1) of our proposed GYB framework model are 95.2%, 94.3%, 94.8%. $AP_{0.5}$ and $AP_{0.5:0.95}$ are 0.970 and 0.701, respectively, which are more than 10% higher than the traditional algorithms YOLOv3, SSD and Faster_Rcnn models. Similarly, for the fog scene, the evaluation indicators of the GYB model are more than 20% higher than traditional algorithms, while also meeting the real-time requirements. Simultaneously, we find that the framework proposed in this paper achieves a fps of 28.67 frames and 29.06 frames in the rain streak and fog scenes by calculating the fps of different models. Therefore, this framework ensures the accuracy of ship detection and classification while meeting the real-time requirements of ship systems. At the same time, in order to further verify the accuracy and reliability of the framework for recognizing different types of ships, we conducted a separate indicator evaluation for the detection and recognition of individual ships, and summarize the evaluation indicators in Tables 4 and 5. It can be seen that, whether in rainy or foggy weather, the P, R, and F1 of single-ship detection and recognition are all above 92%, especially due to the obstruction of ship structures of different sizes and limited visibility in rainy and foggy weather. In summary, the experimental results show that the proposed framework can effectively solve the problem of ship detection and recognition problems, even in low-visibility conditions.

Table 3. Performance statistics of ship detection for waterborne navigation in different weather conditions.

Data	**Model**	**Evaluation Indicators**					
		P	**R**	**F1**	$AP_{0.5}$	$AP_{0.5:0.95}$	**fps**
rain streaks	GYB	95.2%	94.3%	94.8%	0.970	0.701	28.67
	YOLOv3	71.7%	48.4%	57.8%	0.578	0.256	11.98
	SSD	83.5%	79.4%	81.4%	0.822	0.434	6.47
	Faster_Rcnn	60.6%	59.7%	61.0%	0.614	0.294	3.26
fog	GYB	97.4%	96.4%	96.9%	0.984	0.742	29.06
	YOLOv3	69.9%	58.9%	60.6%	0.867	0.268	12.68
	SSD	89.8%	84.9%	87.3%	0.867	0.408	7.03
	Faster_Rcnn	61.2%	63.5%	62.3%	0.655	0.317	3.78

Table 4. Performance statistics of detection and recognition of different types of ships under rain streak weather conditions.

Rain Streaks	**Evaluation Indicators**				
	P	**R**	**F1**	$AP_{0.5}$	$AP_{0.5:0.95}$
ore carrier	94.8%	92.6%	93.7%	0.971	0.660
passenger ship	94.4%	92.9%	93.6%	0.971	0.656
container ship	99.9%	100%	99.9%	0.995	0.783
bulk cargo carrier	92.8%	92.1%	92.4%	0.949	0.696
general cargo ship	95.9%	95.1%	95.5%	0.971	0.737
fishing boat	93.5%	93.0%	93.2%	0.963	0.662

Table 5. Performance statistics of detection and recognition of different types of ships under foggy weather conditions.

Fog	**Evaluation Indicators**				
	P	**R**	**F1**	$AP_{0.5}$	$AP_{0.5:0.95}$
ore carrier	98.4%	90.7%	94.4%	0.985	0.714
passenger ship	98.3%	96.5%	97.4%	0.966	0.715
container ship	96.7%	99.0%	97.8%	0.995	0.786
bulk cargo carrier	97.2%	97.7%	97.4%	0.99	0.766
general cargo ship	99.2%	97.7%	98.4%	0.992	0.771
fishing boat	94.7%	96.7%	95.7%	0.974	0.703

In order to gain a deeper understanding of the roles and importance of each part in the model framework, we conducted a series of ablation experiments. In these experiments, we sequentially added the image restoration module and the BiFPN module, observing their impact on the performance of the model framework. The image restoration method has been renamed as GY and compared with the GYB framework proposed in Table 6 for the aforementioned experiments. The evaluation metrics of the control experiment results are summarized in Table 6. We found that after adding image restoration and the BiFPN module, the detection and recognition metrics of different ships have improved to some extent. Under the conditions of rain streak weather, the YOLOv5 model with the added BiFPN module has seen improvements in its P, R, and F1 score by 9.1%, 7.5%, and 8.6%, respectively, compared to the original model. The GY model framework has seen improvements in its P, R, and F1 scores by 15.3%, 51.2%, and 38%, respectively, compared to the original model. Meanwhile, both $AP_{0.5}$ and $AP_{0.5:0.95}$ have improved to varying degrees. In combination with the model framework proposed in this paper, all evaluation metrics of the model have seen a significant improvement. This ensures that the accuracy of the model is improved without reducing the speed of model inference. Similarly, under the conditions of foggy weather, after adding different modules to the model all evaluation metrics of the model have seen a significant improvement.

Table 6. Comparison of experimental results of different modules.

Data	**Model**	**Evaluation Indicators**					
		P	**R**	**F1**	$AP_{0.5}$	$AP_{0.5:0.95}$	**fps**
Rain streaks	YOLOv5	65.0%	34.1%	44.7%	0.405	0.216	33.37
	YOLOv5 + BiFPN	74.1%	41.6%	53.3%	0.503	0.288	31.75
	GY	80.3%	85.3%	82.7%	0.887	0.516	30.45
	GYB	95.2%	94.3%	94.8%	0.970	0.701	28.67
Fog	YOLOv5	70.8%	55.5%	62.2%	0.622	0.368	35.37
	YOLOv5 + BiFPN	75.9%	49.1%	48.1%	0.580	0.351	30.85
	GY	83.6%	70.0%	76.2%	0.786	0.509	29.78
	GYB	97.4%	96.4%	96.9%	0.984	0.742	29.06

The experimental results show that the proposed framework can effectively solve the problem of ship detection and recognition problems, even in low-visibility conditions. It can be seen that, under the conditions of low-visibility small-target ship detection, large-area distortion of the image, and the challenge of area occlusion between ships, the new ship-detection framework (GYB) proposed in this paper has more robust performance than the traditional algorithms in the real complex low-visibility sea environment.

5. Conclusions

The detection of ships in maritime traffic navigation is of paramount practical significance for safeguarding navigation safety and facilitating intelligent control decision-making. In this paper, we propose an integrated framework for the detection and recognition of maritime ships under low-visibility conditions. This framework achieves accurate detection and recognition of ships in distorted maritime video data. The proposed framework utilizes adversarial neural networks to generate attention maps, which enable the identification of rain streaks and fog areas within the images. These attention maps guide the contextual autoencoder to selectively restore the low-visibility regions of the images, based on the surrounding information. Then, the restored images are input into the YOLOv5 detector, which incorporates multi-scale feature fusion, to achieve accurate detection and classification of navigating ships in the video images. We validated the performance of the proposed GYB framework using the SeaShip dataset. Experimental results showed that the proposed framework can achieve satisfactory performance for maritime ship detection under rainy and foggy weather conditions, with an average precision of 96.3%, an average recall of

95.4%, an average harmonic mean of 95.9% and an average fps of 28.87 frame. The average precision $AP_{0.5}$ and $AP_{0.5:0.95}$ are 0.977 and 0.722. The experimental results demonstrate that the proposed framework significantly improves the precision of ship detection and classification under adverse weather conditions.

The following directions can be expanded to further enhance the model applicability in the future. First, the SeaShip dataset exhibits a limited diversity in ship categories. It is advisable to augment the dataset with additional ship types, including small fishing boats, to enhance the model's robustness. Second, the density of maritime ships collected is not particularly high, and it is worthwhile for us to further investigate the validation of maritime ship detection in high-density scenarios. Last, but not least, we can also add the detection of ships with different rotation angles to further verify the performance of the model under different water transportation scenarios.

Author Contributions: Conceptualization, X.C. and J.Z.; methodology, X.C., C.W., Z.X. and J.Z.; writing—original draft preparation, X.C., C.W. and J.Z.; writing—review and editing, Z.X., C.W., J.Z. and J.X.; funding acquisition, X.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was jointly supported by National Natural Science Foundation of China (52331012, 52102397, 52071200, 72101157, 71942003).

Data Availability Statement: Readers interested in the work can email the corresponding author to request data sharing.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Liu, R.W.; Nie, J.; Garg, S.; Xiong, Z.; Zhang, Y.; Hossain, M.S. Data-Driven Trajectory Quality Improvement for Promoting Intelligent Vessel Traffic Services in 6G-Enabled Maritime IoT Systems. *IEEE Internet Things J.* **2021**, *8*, 5374–5385. [CrossRef]
2. Cheng, S.; Zhu, Y.; Wu, S. Deep learning based efficient ship detection from drone-captured images for maritime surveillance. *Ocean Eng.* **2023**, *285*, 115440. [CrossRef]
3. Volden, Ø.; Cabecinhas, D.; Pascoal, A.; Fossen, T.I. Development and experimental evaluation of visual-acoustic navigation for safe maneuvering of unmanned surface vehicles in harbor and waterway areas. *Ocean Eng.* **2023**, *280*, 114675. [CrossRef]
4. Forti, N.; d'Afflisio, E.; Braca, P.; Millefiori, L.M.; Carniel, S.; Willett, P. Next-Gen Intelligent Situational Awareness Systems for Maritime Surveillance and Autonomous Navigation [Point of View]. *Proc. IEEE* **2022**, *110*, 1532–1537. [CrossRef]
5. Bahnsen, C.H.; Moeslund, T.B. Rain Removal in Traffic Surveillance: Does it Matter? *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 2802–2819. [CrossRef]
6. Li, M.; Cao, X.; Zhao, Q.; Zhang, L.; Meng, D. Online Rain/Snow Removal from Surveillance Videos. *IEEE Trans. Image Process.* **2021**, *30*, 2029–2044. [CrossRef] [PubMed]
7. Fu, K.; Li, Y.; Sun, H.; Yang, X.; Xu, G.; Li, Y.; Sun, X. A Ship Rotation Detection Model in Remote Sensing Images Based on Feature Fusion Pyramid Network and Deep Reinforcement Learning. *Remote Sens.* **2018**, *10*, 1922. [CrossRef]
8. Lu, H.; Li, Y.; Lang, L.; Wu, L.; Xu, L.; Yan, S.; Fan, Q.; Zheng, W.; Jin, R.; Lv, R.; et al. An Improved Ship Detection Algorithm for an Airborne Passive Interferometric Microwave Sensor (PIMS) Based on Ship Wakes. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5302012. [CrossRef]
9. Yu, C.; Cai, J.; Chen, Q. Multi-resolution visual fiducial and assistant navigation system for unmanned aerial vehicle landing. *Aerosp. Sci. Technol.* **2017**, *67*, 249–256. [CrossRef]
10. Yao, P.; Sui, X.; Liu, Y.; Zhao, Z. Vision-based environment perception and autonomous obstacle avoidance for unmanned underwater vehicle. *Appl. Ocean Res.* **2023**, *134*, 103510. [CrossRef]
11. Wang, B.; Han, B.; Yang, L. Accurate Real-time Ship Target detection Using Yolov4. In Proceedings of the 2021 6th International Conference on Transportation Information and Safety (ICTIS), Wuhan, China, 22–24 October 2021; pp. 222–227.
12. Li, J.; Tian, J.; Gao, P.; Li, L. Ship Detection and Fine-Grained Recognition in Large-Format Remote Sensing Images Based on Convolutional Neural Network. In Proceedings of the IGARSS 2020–2020 IEEE International Geoscience and Remote Sensing Symposium, Waikoloa, HI, USA, 26 September–2 October 2020; pp. 2859–2862.
13. Feng, J.; Li, B.; Tian, L.; Dong, C. Rapid Ship Detection Method on Movable Platform Based on Discriminative Multi-Size Gradient Features and Multi-Branch Support Vector Machine. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 1357–1367. [CrossRef]
14. Shao, Z.; Wang, L.; Wang, Z.; Du, W.; Wu, W. Saliency-Aware Convolution Neural Network for Ship Detection in Surveillance Video. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *30*, 781–794. [CrossRef]
15. Lv, Y.; Li, M.; He, Y. An Effective Instance-Level Contrastive Training Strategy for Ship Detection in SAR Images. *IEEE Geosci. Remote Sens. Lett.* **2023**, *20*, 4007505. [CrossRef]

16. Chen, X.; Wang, Z.; Hua, Q.; Shang, W.L.; Luo, Q.; Yu, K. AI-Empowered Speed Extraction via Port-Like Videos for Vehicular Trajectory Analysis. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 4541–4552. [CrossRef]
17. Chen, X.; Guan, J.; Wang, Z.; Zhang, H.; Wang, G. Marine Targets Detection for Scanning Radar Images Based on Radar- YOLONet. In Proceedings of the 2021 CIE International Conference on Radar (Radar), Haikou, China, 15–19 December 2021; pp. 1256–1260.
18. Wang, Y.; Jiang, Z.; Li, Y.; Hwang, J.N.; Xing, G.; Liu, H. RODNet: A Real-Time Radar Object Detection Network Cross-Supervised by Camera-Radar Fused Object 3D Localization. *IEEE J. Sel. Top. Signal Process.* **2021**, *15*, 954–967. [CrossRef]
19. Xu, Q.; Li, Y.; Shi, Z. LMO-YOLO: A Ship Detection Model for Low-Resolution Optical Satellite Imagery. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 4117–4131. [CrossRef]
20. Guo, Y.; Liu, R.W.; Qu, J.; Lu, Y.; Zhu, F.; Lv, Y. Asynchronous Trajectory Matching-Based Multimodal Maritime Data Fusion for Vessel Traffic Surveillance in Inland Waterways. *IEEE Trans. Intell. Transp. Syst.* **2023**, 1–14. [CrossRef]
21. Bai, J.; Li, S.; Huang, L.; Chen, H. Robust Detection and Tracking Method for Moving Object Based on Radar and Camera Data Fusion. *IEEE Sens. J.* **2021**, *21*, 10761–10774. [CrossRef]
22. Shang, W.L.; Gao, Z.; Daina, N.; Zhang, H.; Long, Y.; Guo, Z.; Ochieng, W.Y. Benchmark Analysis for Robustness of Multi-Scale Urban Road Networks Under Global Disruptions. *IEEE Trans. Intell. Transp. Syst.* **2022**, 1–11. [CrossRef]
23. Lin, J.C.W.; Srivastava, G.; Zhang, Y.; Djenouri, Y.; Aloqaily, M. Privacy-Preserving Multiobjective Sanitization Model in 6G IoT Environments. *IEEE Internet Things J.* **2021**, *8*, 5340–5349. [CrossRef]
24. Wang, H.; Xu, Y.; He, Y.; Cai, Y.; Chen, L.; Li, Y.; Sotelo, M.A.; Li, Z. YOLOv5-Fog: A Multiobjective Visual Detection Algorithm for Fog Driving Scenes Based on Improved YOLOv5. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 2515612. [CrossRef]
25. Hassaballah, M.; Kenk, M.A.; Muhammad, K.; Minaee, S. Vehicle Detection and Tracking in Adverse Weather Using a Deep Learning Framework. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 4230–4242. [CrossRef]
26. Tan, M.; Pang, R.; Le, Q.V. EfficientDet: Scalable and Efficient Object Detection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 10778–10787.
27. Quan, R.; Yu, X.; Liang, Y.; Yang, Y. Removing Raindrops and Rain Streaks in One Go. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 9143–9152.
28. Luo, Y.; Xu, Y.; Ji, H. Removing Rain from a Single Image via Discriminative Sparse Coding. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 3397–3405.
29. Li, Y.; Tan, R.T.; Guo, X.; Lu, J.; Brown, M.S. Rain Streak Removal Using Layer Priors. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2736–2744.
30. Nayar, S.K.; Narasimhan, S.G. Vision in bad weather. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, 20–27 September 1999; Volume 822, pp. 820–827.
31. Narasimhan, S.G.; Nayar, S.K. Contrast restoration of weather degraded images. *IEEE Trans. Pattern Anal. Mach. Intell.* **2003**, *25*, 713–724. [CrossRef]
32. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. *Adv. Neural Inf. Process. Syst.* **2014**, 27.
33. Qian, R.; Tan, R.T.; Yang, W.; Su, J.; Liu, J. Attentive Generative Adversarial Network for Raindrop Removal from A Single Image. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2482–2491.
34. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
35. Shao, Z.; Wu, W.; Wang, Z.; Du, W.; Li, C. SeaShips: A Large-Scale Precisely Annotated Dataset for Ship Detection. *IEEE Trans. Multimed.* **2018**, *20*, 2593–2604. [CrossRef]

Journal of
*Marine Science
and Engineering*

Article

Maritime Transport Network in Korea: Spatial-Temporal Density and Path Planning

Jeong-Seok Lee, Tae-Hoon Kim and Yong-Gil Park *

Marine Bigdata & AI Center, Korea Institute of Ocean Science & Technology, Taejong-ro, Yeong-do,
Busan 49112, Republic of Korea; jslee90@kiost.ac.kr (J.-S.L.); thkim00@kiost.ac.kr (T.-H.K.)
* Correspondence: ygpark32@kiost.ac.kr

Abstract: The increase in maritime traffic and vessel size has strengthened the need for economical and safe maritime transportation networks. Currently, ship path planning is based on past experience and shortest route usage. However, the increasing complexity of the marine environment and the development of autonomous ships require automatic shortest path generation based on maritime traffic networks. This paper proposes an efficient shortest path planning method using Dijkstra's algorithm based on a maritime traffic network dataset created by extracting maritime traffic routes through a spatial-temporal density analysis of large-scale AIS data and Delaunay triangulation. Additionally, the depth information of all digital charts in Korea was set as a safety contour to support safe path planning. The proposed network-based shortest path planning method was compared with the path planning and sailing distance of a training ship, and compliance with maritime laws was verified. The results demonstrate the practicality and safety of the proposed method, which can enable the establishment of a safe and efficient maritime transportation network along with the development of autonomous ships.

Keywords: path planning; spatial-temporal density; maritime transportation; network analysis; Delaunay triangulation

Citation: Lee, J.-S.; Kim, T.-H.; Park, Y.-G. Maritime Transport Network in Korea: Spatial-Temporal Density and Path Planning. *J. Mar. Sci. Eng.* **2023**, *11*, 2364. https://doi.org/10.3390/jmse11122364

Academic Editor: Sergei Chernyi

Received: 28 November 2023
Revised: 9 December 2023
Accepted: 12 December 2023
Published: 14 December 2023

1. Introduction

Owing to the worldwide increase in maritime traffic and vessel size, the importance and connectivity of maritime transportation routes have become increasingly significant [1]. This implies that the marine transportation environment is becoming increasingly diverse, leading to an increasing demand for the analysis of maritime transportation routes [2]. Maritime transportation routes generally refer to areas where ships operate. Several such routes can be interconnected to form a maritime transportation network [3]. Maritime transportation networks provide various routes through the connection of ports for maritime logistics [4]. Lee and Cho stated that automatic identification system (AIS) data are essential for various methods of analyzing maritime transportation [5]. The International Maritime Organization (IMO) mandates the installation of AIS data on ships for safe navigation and environmental protection [6]. AIS data include information such as ship type, ship size, date and time, ship location, and ship specifications, and record ship tracks from the past [7]. By analyzing the AIS tracks of ships engaged in international navigation, optimal strategies for sailing from the origin to the destination can be planned based on the ship type and size [8]. The optimal route is economical and safe for ship operators, and most ships exhibit similar sailing patterns [9]. According to Akdağ et al. and Tan et al., the development of unmanned navigation technology leads to the active development of maritime autonomous surface ships (MASSs) and unmanned surface vehicles (USVs) [10,11]. Currently, ship operations are conducted through navigation planning and the actions of navigators and captains based on their experience and judgment. However, because of the decreasing number of seafarers, new navigation planning methods based on computer algorithms using AIS data, which have been accumulated over the years, have become essential, particularly owing to the increasing complexity of maritime traffic and the development of MASS technology [12–14].

In this study, a novel method for shortest path navigation planning is proposed through AIS data analysis to address the need for more efficient and safer navigation planning for ships. A spatial-temporal density analysis was performed using the AIS data accumulated over several years, and the navigational areas of the ships were represented as polygons. The AIS data underwent transformation from points to lines, and the analysis involved calculating the temporal values occupied within each grid, resulting in the spatial-temporal density analysis. Delaunay triangulation analysis was performed based on these polygons to divide maritime traffic areas into separate spatial units, using the traffic separation scheme (TSS) method to differentiate two-way traffic. The Delaunay triangulation results involved a process of filtering out areas, excluding those with high spatial-temporal density values. The resulting spatial units constituted a maritime traffic network dataset of about 800, which was supplemented with depth information from a digital chart to prevent grounding. The digital chart was based on the S-57 hydrographic data generated for navigation purposes. Dijkstra’s algorithm was used to analyze the navigation time and distance based on the proposed method to plan a safe and efficient route from the departure point to the destination. The proposed method was compared with the path planning of a training ship in terms of the navigation time, distance, and compliance with TSS regulations. The AIS data, along with the spatial-temporal density analysis and the maritime traffic network dataset generated using space partitioning algorithms, were analyzed to better represent the shortest path planning compared to the results of the reference training ship. The proposed maritime network dataset and depth information are expected to enhance the safety of ship navigation and can be used for the future navigation planning of MASSs and USVs.

2. Related Study

In this section, we classify the various analyses of maritime traffic into studies on maritime traffic route generation, maritime traffic network proposals, and ship path planning to represent the shortest distance. Additionally, we describe the main algorithms used in the analysis, including the method of spatial partitioning.

2.1. Studies on Maritime Traffic Routes

Lee et al. aimed to create a national maritime traffic map for South Korea’s coastal waters [5]. The study focused on cargo ships, tanker ships, passenger ships, and fishing vessels with lengths of 60 m or more, using AIS data for four weeks (28 days). A line density analysis was performed by connecting ship tracks and extracting the top 50% of the data using quantile partitioning to derive maritime traffic routes. Similarly, Kim et al. conducted a study on maritime traffic routes in the same area with four months (120 days) of AIS data and applied the same ship types and sizes [15]. They proposed a spatial-temporal density analysis method based on the occupied time of ship tracks within the analysis area. This method addressed the limitations of line density analysis and enabled the analysis of large amounts of maritime traffic data. This method utilized the approach provided by the European Marine Observation and Data Network (EMODnet) for ship density, which is publicly available [16]. Maritime traffic routes represent a way to depict high-density maritime traffic areas by using polygons to represent the areas, rendering them suitable for various applications.

2.2. Studies on Maritime Traffic Network

Pallotta et al. explained the effectiveness of using AIS data to infer spatial and temporal information for port and ocean platforms as the usage of AIS data increases [17]. They proposed a Traffic Route Extraction and Anomaly Detection (TREAD) method, which reviewed the utilization of this knowledge for the characteristics of maritime traffic, the extraction of traffic routes, and other related analyses based on the distribution analysis of maritime traffic. Fernandez et al. extracted maritime traffic networks using unsupervised methods on ship trajectory data, decomposing them into significant routes by detecting waypoints, and constructing maritime traffic networks by connecting ports and waypoints via graph-based connections [18]. Wang et al. extracted dense maritime traffic areas via the kernel density estimation method, and used image processing to detect their edges [19]. They proposed a polygon-based approach to represent areas, and generated centerlines connecting nodes and segments via Delaunay triangulation. Yan et al. transformed the rich positional information of ship trajectories into semantic objects called "ship trip semantic objects (STSO)", which represent the objects as a "stop-waypoint-stop" model [20]. They utilized graph theory to integrate the nodes and edges for maritime traffic network construction. Filipiak et al. utilized ship traffic volume derived from AIS data to construct maritime traffic networks [21]. Their method comprises three steps: maneuvering point detection, waypoint discovery, and edge construction, utilizing the k-d B-tree and Quadtree algorithms for spatial partitioning to extract waypoints. Graph-based maritime traffic networks were constructed using a genetic algorithm based on the extracted waypoints. The construction of a maritime traffic network that creates a path from a ship's origin to its destination is fundamental.

2.3. Studies on Path Planning

Shah et al. studied path planning for USVs operating in complex environments [22]. They utilized the A* algorithm at the visibility graph nodes, and employed quadtree spatial partitioning to efficiently calculate the nodes of the visibility graph. Various distance-based cost-to-go functions were proposed. Lee et al. also utilized quadtree to perform visibility graph-based path planning for USVs in Korean coastal waters [23]. The nodes generated through the quadtree were used to create a quadtree-based graph, and Dijkstra's algorithm was applied to extract the shortest path. Lee et al. used the depth data to detect the shortest maritime traffic routes for AIS-based ships [24]. Depth is the most important parameter for preventing maritime accidents when planning ship routes. In that study, a grid-based navigational area was extracted, and Dijkstra's, A*, and improved A* algorithms were used for the shortest path calculation. Prior to using the shortest path algorithm, they proposed a suitable algorithm for merchant ships that considered factors such as depth, restricted areas, and designated routes.

2.4. Research Improvement Plan

Merchant ships in international shipping generally use comfortable and safe routes to prevent accidents. Because navigation is impossible in certain areas, depending on the size of the ship, directly searching for the shortest route can reduce the usability of actual ship operations. In other words, the areas in which small vessels navigate are different from those in which large vessels navigate. This study presents the shortest route that can be applied to ships of a size capable of engaging in international navigation and suggests the following improvement measures.

In contrast to previous studies, this study utilized a large-scale AIS density analysis as basic data to represent maritime traffic areas as polygons. The high-density areas obtained from the density analysis provide evidence that ships have used them safely and economically. Therefore, we aim to produce maritime traffic routes based on the studies of Lee et al. and Kim et al. to represent the areas where ships primarily operate [5,15]. The polygons representing the navigation area of a ship can be used to create routes for ships by utilizing the Delaunay triangulation algorithm.

We differentiate our study by separating the routes, such that merchant ships can use them in two ways. Most maritime traffic networks represent single-line connections. However, major accidents are likely if ships operate in different directions along the same route. Therefore, when constructing maritime traffic networks, we assumed that ships could use these networks in a two-way manner.

Merchant ships engaged in international shipping must comply with mandatory navigation regulations. We built a network based on the premise of safe ship navigation by enforcing TSSs and using depth data.

3. Materials and Methods

The AIS data comprising spatial information are essentially a maritime traffic dataset including various attribute information [25]. Spatial information can be used as a basis for various analyses such as maritime traffic pattern recognition, maritime traffic prediction, and maritime traffic networks using geographic information systems (GIS) [26]. In this study, ArcGIS Pro 3.0.3 was used to generate maritime traffic routes, perform delay triangulation, build network datasets, and perform network analysis. To differentiate it from previous studies, AIS data were used as the basis for maritime traffic density analysis. Furthermore, a network dataset was developed based on ArcGIS to store the routes of various ships in the marine space, and a network analyst was used [27]. Additionally, the digital chart's shapefile files were combined with network analysis for practical ship use. Considering safety, adherence to navigation regulations, and accident prevention, we propose a shortest path maritime traffic network.

3.1. Study Overview

The entire Korean coastal area was selected to construct the maritime traffic network dataset, as shown in Figure 1. Previous research has focused on relatively small areas near ports or shallow waters for shortest path analysis. In contrast, this study aims to construct a national-scale maritime traffic network for ships to connect ports. Korea is located in Northeast Asia, and its geography depends strongly on the maritime industry, with 99.7% of its water volume used for shipping [28]. Therefore, the construction of a maritime traffic network is urgently required to handle the smooth flow of maritime traffic, and the importance of safe and efficient shortest path planning is increasing. This study emphasizes the need to construct a comprehensive maritime traffic network that covers all areas.

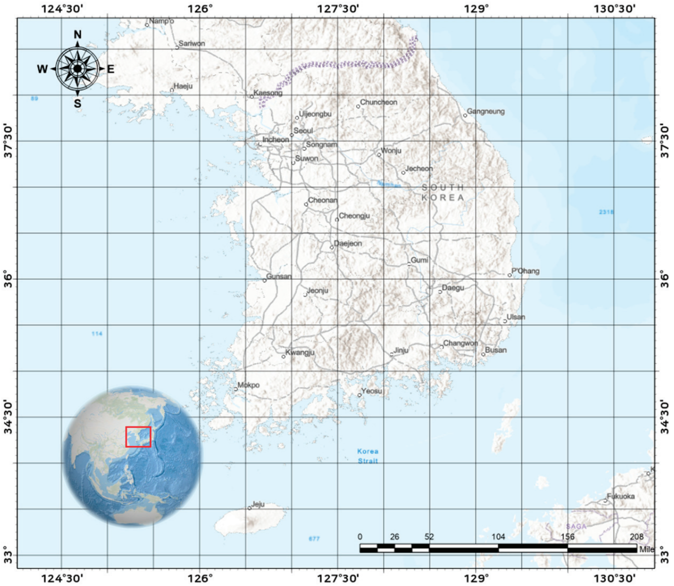


Figure 1. Detailed location and the analysis area.

The process for deriving the shortest path through a GIS-based network analysis is shown in Figure 2. Appropriate preprocessing is required because the initial AIS data are not directly usable. The AIS data, which are composed of point-based spatial information, are transformed into line-based information based on the ship's trajectories over time. Subsequently, a spatial-temporal density analysis is performed to identify the dense routes that are commonly used by ships. The resulting density can be represented as a polygon of the shipping area, and the Delaunay triangulation method is used to partition the area into triangles that are suitable for building a maritime traffic network. Subsequently, three equidistant points were inserted into each line segment to regulate the traffic flow of the ships. These points are subsequently connected in sequence to create a maritime traffic network, which is stored as a network dataset. This dataset, including safety contours and navigation regulation information from digital charts, enables the derivation of the shortest path through network analysis.

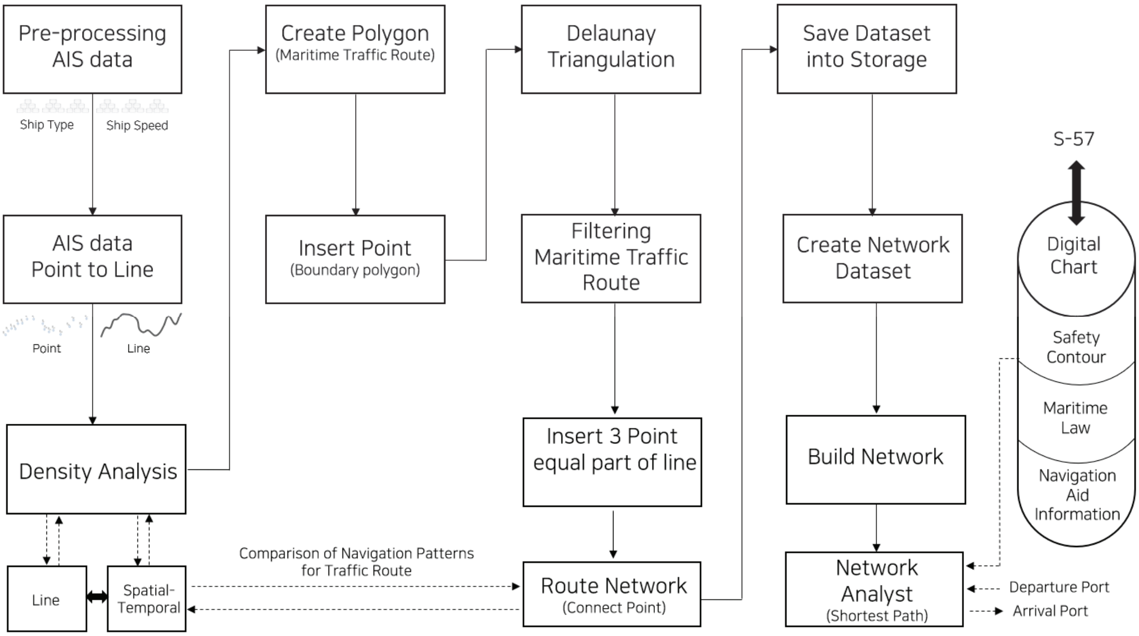
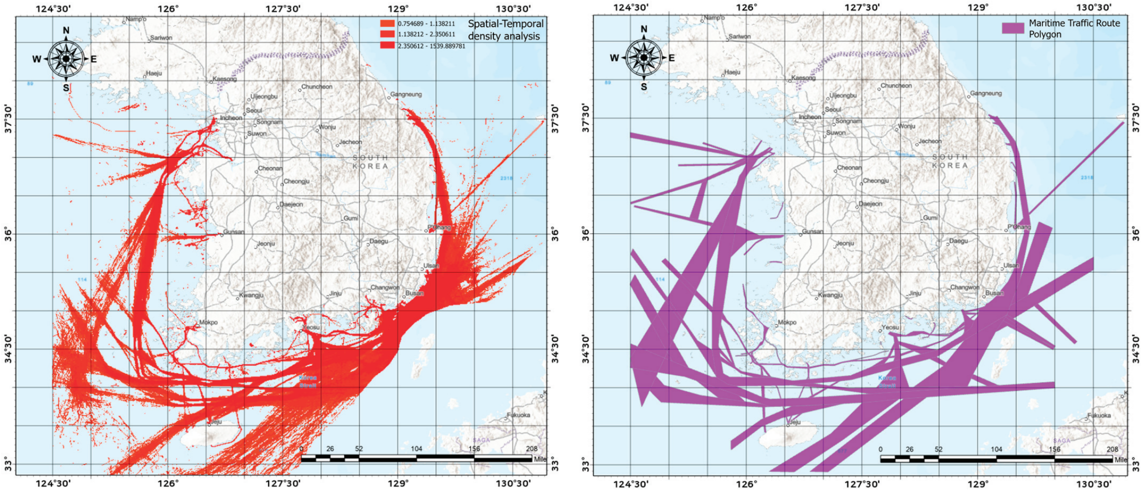


Figure 2. Flowchart of shortest path planning in maritime transport using network analysis.

3.2. Overview of AIS Data and Density Analysis

AIS data are recorded in real time, and the AIS is equipment installed to support the safe navigation of ships by providing information regarding their operations and specifications [29]. AIS data afford several advantages for maritime transportation analysis, including real-time data acquisition, high accuracy, and detailed ship information such as ship type, speed, and location, rendering them a valuable tool for improving navigation safety and optimizing shipping operations. In addition, AIS data can provide insight into maritime traffic patterns and trends, which can aid the development of effective maritime policies and regulations. The collected AIS data are divided into static and dynamic information. Static information includes the maritime mobile service identity (MMSI), name, type, IMO number, call sign, length, draft, and gross tons (GTs), whereas dynamic information includes the MMSI, date, latitude, longitude, speed over ground (SOG), course over ground (COG), and heading [30]. Preprocessing was performed by combining the static and dynamic information based on the MMSI data for analysis. In this study, the AIS data collected in Korea during the four seasons of 2018 were used for analysis, and data from March 1 to 7 (1 week), June 1 to 7 (1 week), September 1 to 7 (1 week), and December 1 to 7 (1 week) were used. According to Tsuji (1996), maritime traffic survey data should consider a minimum of 6–7 days of traffic volume, accounting for weekly fluctuations [31]. Therefore, this study performed a maritime traffic density analysis for 28 days, using 7 days of data for each season. Various types of ships use the sea. However, in this study, cargo ships, tanker ships, passenger ships, and towing ships were targeted to construct a network dataset for major shipping routes. Figure 3 shows the results of the spatial-temporal density analysis of AIS trajectories and traffic zones within the density area in polygon form. The spatial-temporal density analysis method used in Figure 3a follows the analyses used by Kim et al. and EMODnet, which are widely used scientific methods for ship density providing public services in Europe and Korea [15,16].



(**a**) (**b**)

Figure 3. Visualization of AIS data analysis: (**a**) results of spatial-temporal density analysis targeting AIS; and (**b**) polygon results of maritime traffic area including spatial-temporal density analysis results.

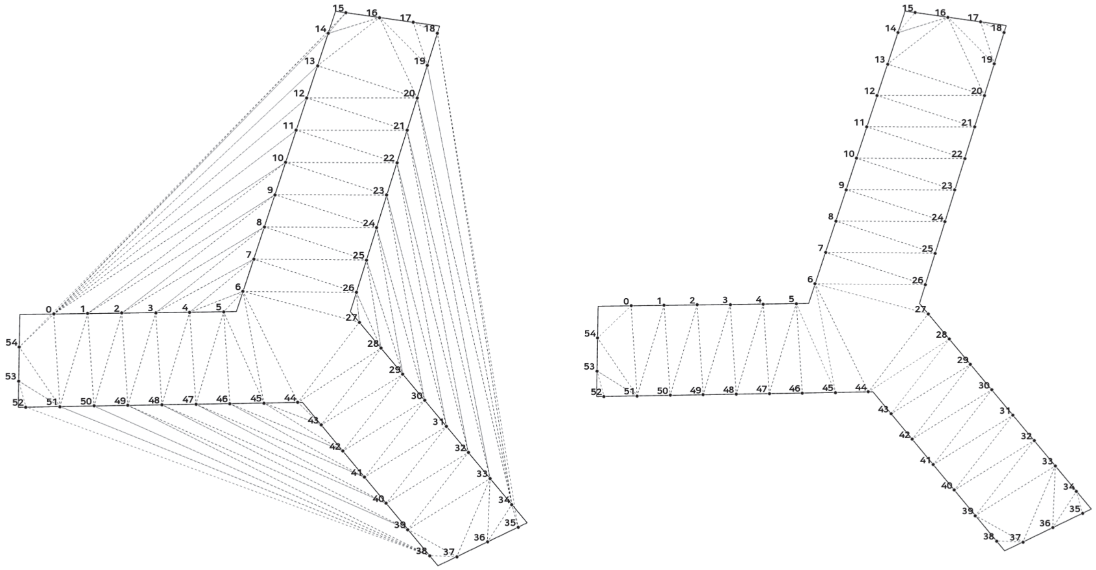
The density was calculated using Equation (1). When a line between two consecutive positions on a ship intersects two or more cells, the length of the line segment crossing these cells can be calculated. To calculate the time for which a ship occupied each cell, the length of the line segment was divided by the total length of the line and subsequently multiplied by the total time of the line. The density of each individual cell is subsequently determined by calculating the time value of each line segment and adding all the time values associated with the cell.

$$D_i = \sum_{j=1}^{n} \frac{S_j}{L_j} \times T_j \tag{1}$$

where D_i is the ship density (h) of cell i, L_j is the total length (km) of the line j, S_j is the partial length (km) of the line j that intersects with cell i, and T_j is the time (h) spent by each ship in each cell during the entire period (e.g., one month) and the number of lines related to the cell. Therefore, the density is the time (hours) each ship spends in a given cell over the entire period. The polygon representing the area where ships navigate in Figure 3b is included in the spatial-temporal density analysis result. This area can change and vary depending on changes in the maritime traffic volume and traffic flow trends according to the season. As the polygon is not a fixed element but can vary depending on the situation, it can be used in other analysis areas. In this study, the polygon, as shown in Figure 3b, was used to perform Delaunay triangulation.

3.3. Method for Performing Delaunay Triangulation Based on Density Analysis

Based on the density analysis, extracting the area where ships conventionally operate as polygons is feasible. This area is not only the flow area for the safe navigation of ships but also serves as a basis for further analysis. Delaunay triangulation constructs a polygon in the form of triangles based on points inserted at regular intervals on the edge of a maritime traffic area. Delaunay triangulation is a method for constructing and dividing triangles in a manner that excludes points other than those used to create the triangle. One of the most important features of Delaunay triangulation is that "the circumcircle of each triangle does not contain any points other than the three vertices of the triangle". This feature is useful for identifying the closest point, rendering it useful for data clustering, density analysis, and road network design [32]. Figure 4a shows the initial results of Delaunay triangulation on the generated polygon, and Figure 4b shows the filtered triangles generated outside the polygon area.

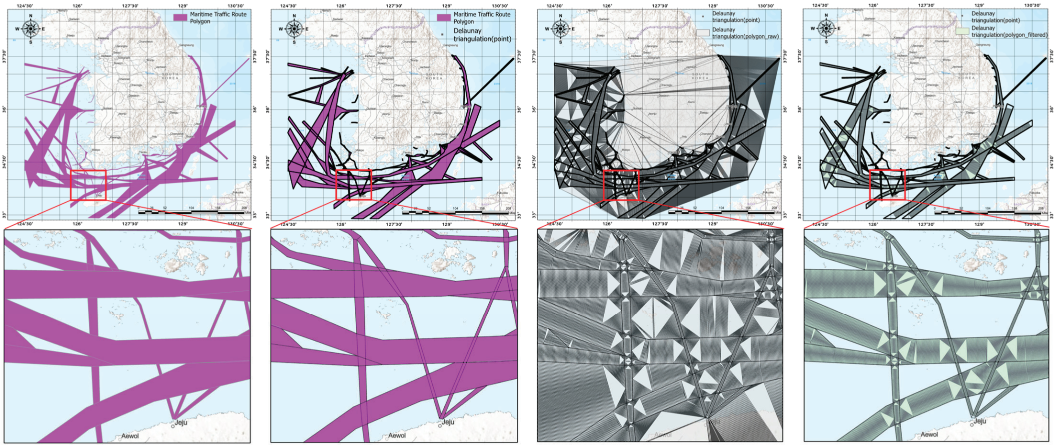


(a) Triangular network

(b) Filtered triangular network

Figure 4. Process of generating Delaunay triangulation: (**a**) Delaunay triangulation before filtering; and (**b**) Delaunay triangulation after filtering.

Maritime traffic in ocean spaces exhibits complex traffic flows depending on the types and sizes of ships. Shipping lanes for cargo ships have been established as a safe and efficient means of navigation, rendering the analysis of cargo ship density in Korean coastal waters important. The results of the density analysis and Delaunay triangulation for cargo ships in Korean coastal waters are shown in Figure 5.



(a) Polygon for Maritime traffic route (b) Insertion of points along the boundary line (c) Initial result of Delaunay triangulation (d) Filtered result of Delaunay triangulation

Figure 5. Results of the density-based Delaunay triangulation performed on the coastal waters of Korea: (**a**) spatial-temporal density-based polygon representing the maritime traffic flow, (**b**) insertion of points at regular intervals along the edges of the polygon, (**c**) initial results of Delaunay triangulation before filtering, and (**d**) results after filtering.

Figure 5a shows the initial state of the maritime traffic route and illustrates the results of the spatial-temporal density analysis extracted as polygons. In Figure 5b, point insertion along the boundary lines of the extracted polygons is performed at regular intervals. The points are inserted at a consistent interval of 1 km, equivalent to the grid size used in the density analysis. Figure 5c shows the results of Delaunay triangulation, encompassing areas beyond those extracted as maritime traffic routes. Subsequently, Figure 5d presents the filtering of the triangulation in land areas, retaining only the portions corresponding to maritime traffic routes. Maritime traffic flows differ depending on the types and sizes of ships. The analysis targeted the shipping routes for merchant ships that operate safely and efficiently, rendering them important elements of maritime traffic.

3.4. Network Analysis Method Using Digital Chart Data

The digital chart refers to all hydrographic information related to ship navigation, such as coastlines, contour lines, water depths, navigational aids, hazards, and shipping routes, produced according to the International Hydrographic Organization (IHO) standard specification S-57. It provides path planning, route monitoring, and navigation-related information for safe ship navigation [33]. The inclusion of digital chart information in network datasets is essential for safe ship navigation. In this study, we aim to create a new maritime traffic network dataset that includes depth and navigational information that can be utilized for actual ship operations. Figure 6 shows the detailed information on the S-57-based digital chart used for the maritime traffic network dataset.

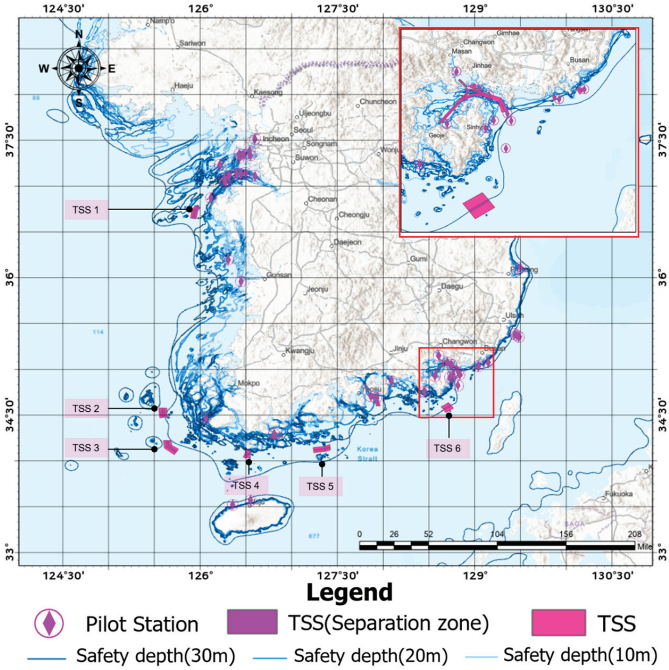


Figure 6. Detailed description of S-57-based digital charts for the maritime traffic network (pilot area, TSS, depth).

In general, a pilot boards a vessel to enter or leave a port. Similarly, for ships to leave the port, pilots board and safely unberth the vessel [34]. Thus, ships can navigate freely from the point of departure to their destination outside the port area, utilizing the most economical and safe routes. Therefore, the Pilot Station (P/S) can be used as information for departures and destinations. TSS is designed to prevent collisions between ships at sea and has rules for navigation according to the designated flow [35]. The navigation rules of the TSS are specified in Part B, Section I, Rule 10, Traffic Separation Schemes, as shown in Figure 7.

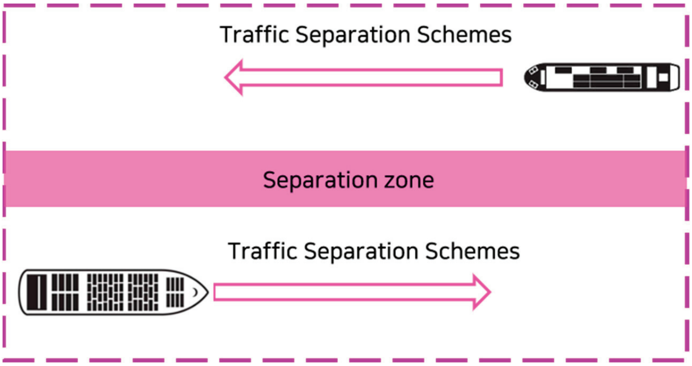


Figure 7. Detailed description of Traffic Separation Schemes (TSSs).

To ensure safe navigation for ships, TSSs are designated in wide areas of the sea beyond routes to ports of call. TSS was established to promote safe navigation in areas with heavy maritime traffic. The direction of the TSS should be considered when building a maritime transportation network to ensure the safe navigation of ships. Furthermore, the size of a ship significantly affects the factors considered in its navigation. Depth information is a critical factor that must be considered, particularly for larger ships. A larger ship requires a wider water area for maneuvering, rendering depth a sensitive factor. Therefore, depth data are crucial for preventing ship grounding accidents. In this study, depth information was applied to the network dataset at intervals of 10 m, 20 m, and 30 m. To secure a safe depth for ship navigation, depth information is set as a line obstacle in the shortest path detection algorithm.

3.5. Construction Method of Network Dataset Based on Delaunay Triangulation

Previous studies using Delaunay triangulation to construct maritime traffic networks connected these networks based on centerlines. A route based on centerlines is suitable for representing a single traffic flow but not for expressing various navigation patterns. The polygon inside the maritime traffic route includes not only ships sailing in a certain direction but also upstream, downstream, eastbound, westbound, and irregular navigation patterns. To induce regular maritime traffic flow, it is regulated to connect in the same direction as the TSS. Because TSS is a method for regulating bidirectional maritime traffic flow, constructing a bidirectional maritime traffic network dataset using Delaunay triangulation is necessary. Figure 8 illustrates the process of creating a bidirectional maritime traffic network that regulates maritime traffic in both directions.

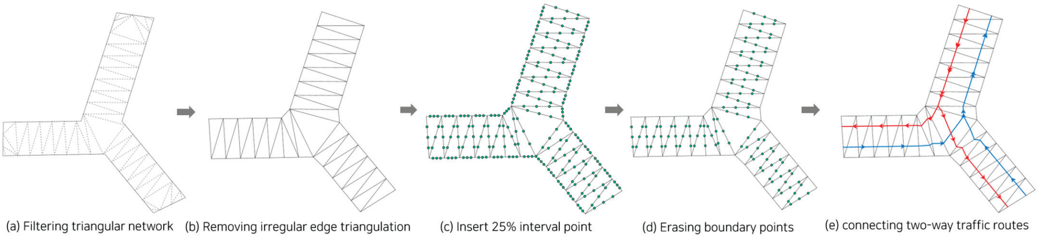


Figure 8. Construction process of two-way maritime traffic routes using Delaunay triangulation: (**a**) Initial results of Delaunay triangulation filtering, (**b**) Removal of irregular triangulation at the edges, (**c**) Insertion of points at 25% intervals in the triangulation results, (**d**) Removal of points along the boundary line, (**e**) Connection of two-way traffic routes.

Figure 8a shows the process of converting the initial delay triangulation, which is formed as a polygon, into line-based spatial information. Figure 8b shows the process of deleting the irregular edge triangulation, which enables forming an accurate maritime traffic route. Figure 8c inserts points at a distance of 25% for each line, resulting in three points being generated for each line, which represent the points that indicate the flow of maritime traffic. Figure 8d shows the process of deleting the points formed on the boundary by uniformly utilizing the difference in spatial information. Afterwards, Figure 8e forms three points only for the lines that exist inside, where the left point connects the south and west bounds and the right point connects the north and east bounds. The points formed in the middle are used as centerlines for the maritime traffic network route in this study. However, they can be created separately according to the separation provided by the TSS. Thus, maritime traffic flow is divided into two-way forms, preserving the regular flow of ships and focusing on accident prevention through the installation of separate buoys. The maritime traffic network dataset is constructed based on the created two-way route. The process of building by reorganizing network connections and attribute information is subsequently performed on the input network dataset. This implies rebuilding the included routes, such that they are available for use in the network, and the work speed can vary significantly depending on the data size during this process.

4. Results

4.1. Extraction Results of Network Based on Delaunay Triangulation

Various types of digital chart information have been considered when constructing maritime traffic networks. Maritime traffic includes areas with one-way navigation and areas where navigation is enforced to be two-way. Additionally, stations are available where pilots are required to board and disembark to enter and exit the ports. Lastly, safe water depths for the target vessels must be considered, as the permissible depths vary depending on the size of the ships. The ultimate goal of a maritime traffic network is to safely and economically transport large volumes of maritime cargo using ships, while considering various factors [36]. The maritime traffic network that can be established in South Korea's coastal waters is listed in Table 1. After constructing the maritime traffic network dataset, 485 datasets were generated based on six major trade ports located in the west (Incheon, Pyongtaek, Daesan, Boryeong, Gunsan, and Mokpo), with a total route length of 323,351 km and an average route length of 666.7 km.

Table 1. Results of maritime traffic network dataset.

Area	Number of Ports (Names)	Number of Route Networks	Route Total Distance (km)	Route Average Distance (km)
Western	6 (Incheon, Pyongtaek, Daesan, Boryeong, Gunsan, and Mokpo)	485	323,351	666.7
Southern	5 (Jeju, Wando, Yeosu, Busan Newport, and Busan port)	153	56,684	370.5
Eastern	3 (Ulsan, Pohang, and Donghae)	100	61,053	610.5
Outside	18	62	28,662	462.3
Total	32	800	469,750	527.5

In the south, five major trade ports (Jeju, Wando, Yeosu, Busan Newport, and Busan port) generated 153 datasets with a total route length of 56,684 km and an average route length of 370.5 km. The three major trade ports in the east (Ulsan, Pohang, and Donghae) generated 100 datasets with a total route length of 61,053 km and an average route length of 610.5 km. In addition to the network dataset of routes departing from major trade ports, data approaching trade ports from offshore were added as "outside". A total of 62 "outside" datasets were generated, with a total route length of 28,662 km and an average length of 462.3 km. Here, the term "outside" refers to routes approaching coastal areas from the open sea. Therefore, 32 starting points were selected in the coastal waters of South Korea, and 800 network datasets were constructed. Finally, density analysis was performed using the AIS data. The results of the maritime traffic network dataset based on Delaunay triangulation are shown in Figure 9.

The construction of a maritime traffic network must consider the actual flow of ship traffic, comply with regulations such as those designated in the TSS, and ensure a safe water depth to prevent grounding accidents. The established network dataset was subsequently used as the basis for performing shortest path analysis.

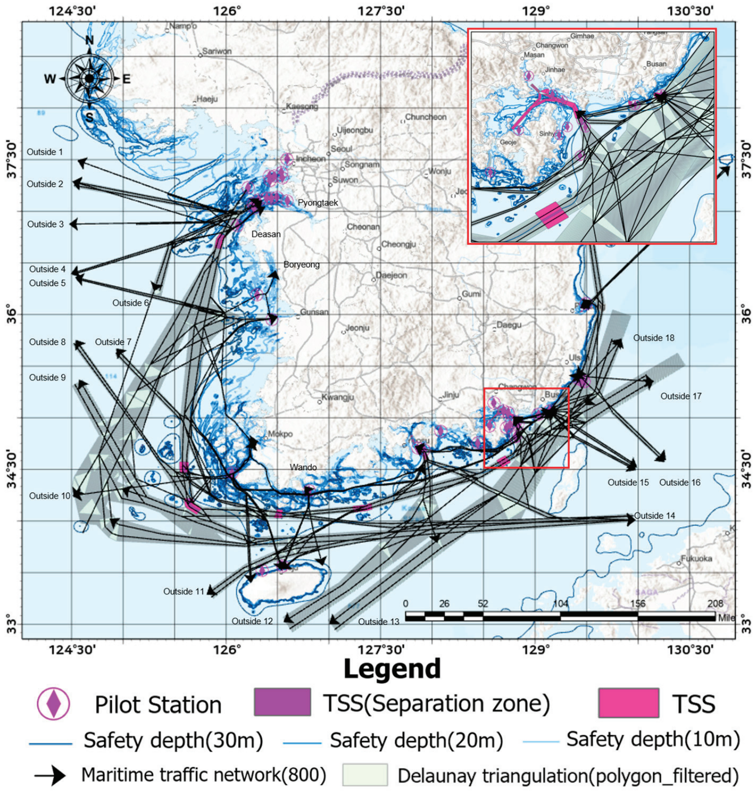


Figure 9. Result of the construction of the maritime traffic network dataset (32 starting points, 800 network datasets).

4.2. Shortest Path Network Analysis for Voyage Planning

For ships to navigate from the departure point to the destination, considering factors such as the economically shortest route, safe water depth, and compliance with designated routes is necessary. Dijkstra was the first to devise a method for generating the shortest route based on graph theory, and Silveira et al. and Wang et al. constructed the shortest route that ships can use using Dijkstra's algorithm [37–39]. In this study, Dijkstra's algorithm was used to calculate the shortest route, and the safety depth data were set as line obstacles in the network. This implies that by simultaneously inputting line obstacle values to maintain a safe water depth during ship navigation, the network dataset provides the shortest route among the available routes. Using Dijkstra's algorithm provided by ArcGIS Network Analyst, the safe area is set by considering points (marine facilities), lines (depth), and polygons (military training), and a safety check can be performed for the designated routes.

4.3. Comparison of Shortest Path Planning Algorithms

To compare the actual path planning of the training ship (T/S HANBADA) with the shortest path planning, the ship's specifications are listed in Table 2, and its photograph is presented in Figure 10.

Table 2. Specifications of T/S HANBADA.

Category	T/S HANBADA
Date built	8 December 2005
Length (overall)	117.20 (m)
Beam	17.80 (m)
Draught	8.15 (m)
Gross tonnage	6686.0 (tons)
Service speed	17.5 (knots)
Main engine	Diesel 8130 (HP)

Figure 10. Photograph of T/S HANBADA.

The training ship T/S HANBADA is currently in operation as a vessel for student onboard education. The ship's length is 117.20 m, the beam is 17.80 m, and the draught is 8.15 m, which is an important factor for designating the safety depth as the depth to which a vessel is immersed in water. Therefore, a 10 m depth boundary line was entered as a line obstacle in the maritime traffic network dataset. To compare the shortest path and compliance with designated routes provided by the network dataset, the path planning used in the actual operation of the training ship (Case A) from the P/S of Busan port to the P/S of Incheon port and the path planning (Case B) from the P/S of Incheon port to the P/S of Busan port were compared. Figure 11 shows the training ship's route, and Figure 12 shows the network dataset-based path planning connecting the P/S of the Busan and Incheon ports, two major trading ports in Korea.

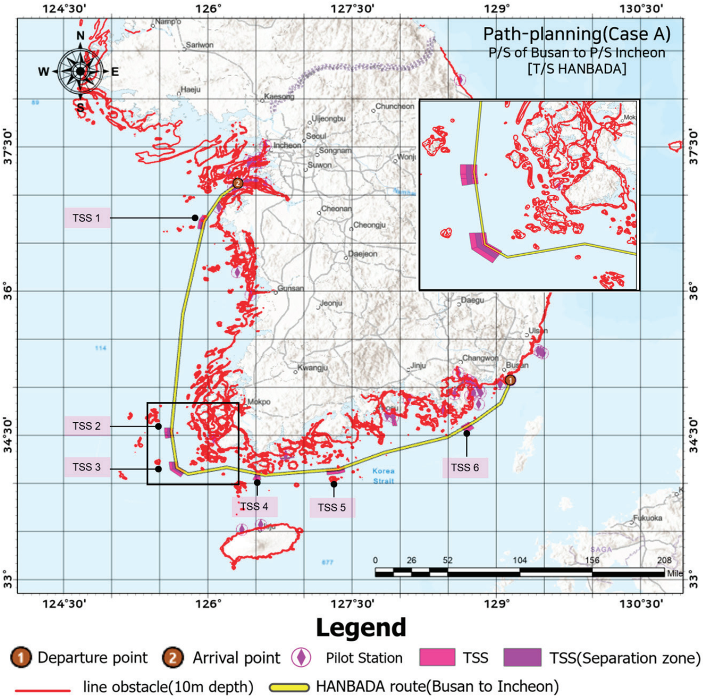


Figure 11. Path planning of the training ship (Case A) from Busan to Incheon.

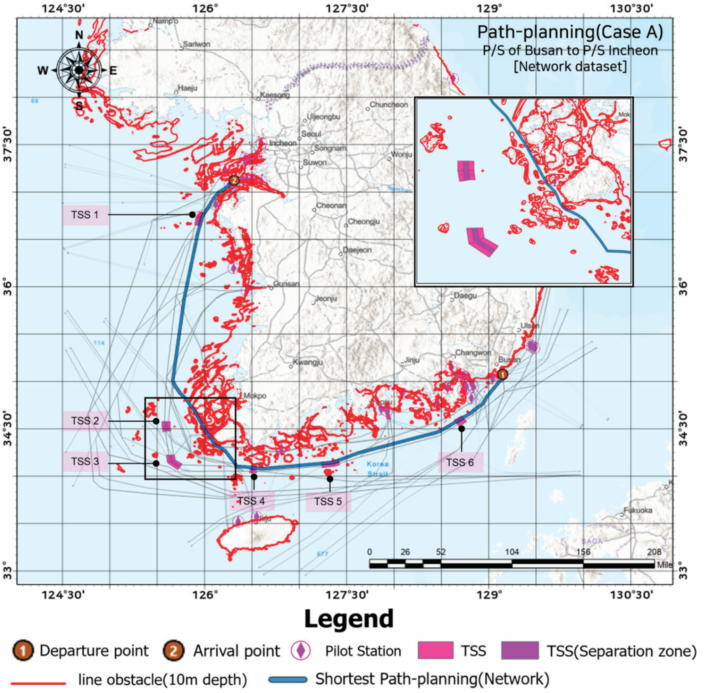


Figure 12. Path planning of the network dataset (Case A) from Busan to Incheon.

Table 3 presents a comparison of the path planning of the HANBADA ship and the shortest distance path generated by the network dataset in terms of distance, sailing time, and compliance with TSS regulations. The distance of HANBADA's path planning (Busan to Incheon) is 708.5 km, and the sailing time is analyzed to be 00 d 21 h 50 m when sailing at the service speed of 17.5 knots. Furthermore, the current ship operation plan is determined based on the captain's decision, and the ship may sail on routes other than the shortest distance. The analysis showed that the HANBADA ship passed through all six regulated TSSs in Korean coastal waters. Furthermore, the shortest distance path generated by the network dataset is analyzed to be 663.9 km, and it takes 00 d 20 h 28 m of time when sailing at 17.5 knots, which is the same speed as HANBADA. The shortest path of the network dataset navigated between the islands without passing through TSS 2 and TSS 3, but complied with all other TSS regulations and provided path planning. Therefore, the shortest distance path planning of the network dataset reduced the distance by 44.6 km relative to the HANBADA path planning, enabling arrival 01 h 22 m sooner.

Table 3. Comparison of path planning (Case A) between HANBADA and network dataset.

Category (Case A)	**Distance to Go (km)**	**Sailing Time (Distance/17.5 knot)**	**Comply with TSS 1**	**Comply with TSS 2**	**Comply with TSS 3**	**Comply with TSS 4**	**Comply with TSS 5**	**Comply with TSS 6**
HANBADA	708.5 km	00 d 21 h 50 m	Comply	Comply	Comply	Comply	Comply	Comply
Network dataset	663.9 km	00 d 20 h 28 m	Comply	No	No	Comply	Comply	Comply
Difference	−44.6 km	−00 d 01 h 22 m	-	-	-	-	-	-

Figure 13 shows the path planning of the training ship, whereas Figure 14 shows the path planning based on the network dataset, connecting the P/S of Incheon and Busan, and returning to the starting point.

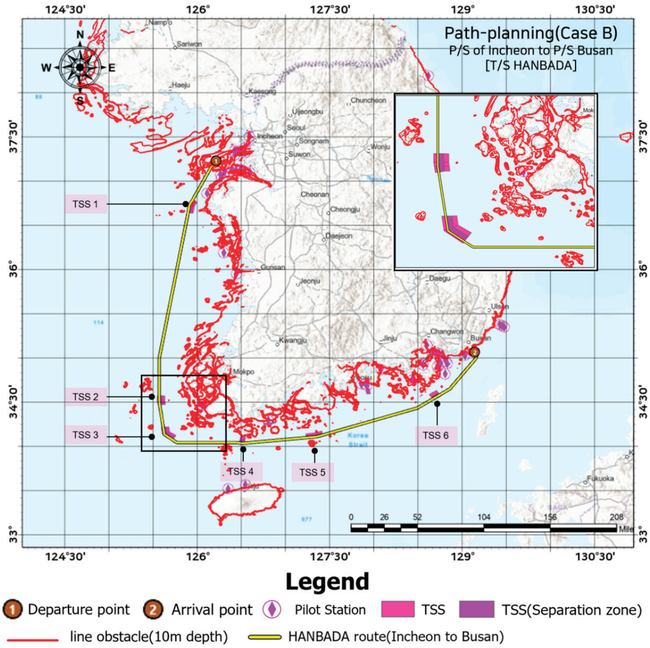


Figure 13. Path planning of the training ship (Case B) from Incheon to Busan.

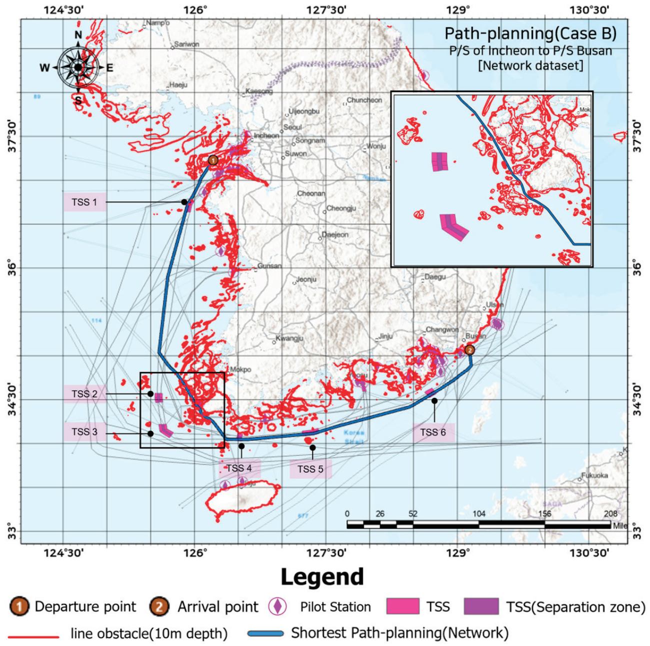


Figure 14. Path planning of the network dataset (Case B) from Incheon to Busan.

Table 4 compares the distance, sailing time, and compliance with TSS regulations of the HANBADA path planning and the shortest route provided by the network dataset for the return trip from Incheon P/S to Busan P/S. The HANBADA path planning covers a distance of 723.7 km and has a sailing time of 00 d 22 h 21 m when sailing at 17.5 knots, and complies with all regulations from TSS 1 to TSS 6. In contrast, the shortest route provided by the network dataset covers a distance of 680.7 km and has a sailing time of 00 d 21 h 02 m. Because the shortest route sails between the islands, it does not pass through TSS 2 and TSS 3. In summary, the results show that the network dataset can reduce the distance by 43 km and the sailing time by 01 h 19 m, while HANBADA's path planning complies with all TSS regulations.

Table 4. Comparison of path planning (Case B) between HANBADA and network dataset.

Category (Case A)	Distance to Go (km)	Sailing Time (Distance/17.5 knot)	Comply with TSS 1	Comply with TSS 2	Comply with TSS 3	Comply with TSS 4	Comply with TSS 5	Comply with TSS 6
HANBADA	723.7 km	00 d 22 h 21 m	Comply	Comply	Comply	Comply	Comply	Comply
Network dataset	680.7 km	00 d 21 h 02 m	Comply	No	No	Comply	Comply	Comply
Difference	−43.0 km	−00 d 01 h 19 m	-	-	-	-	-	-

5. Discussion

The increase in maritime traffic and vessel size has significantly changed the marine spatial planning (MSP) environment [40]. As the demand for various marine activities increases, the area covered by maritime traffic routes decreases. Simultaneously, the importance of maritime traffic networks for the smooth transportation and handling of cargo is increasing. The shrinking area of maritime routes and the increasing importance of networks signify the complexity of the marine environment. Furthermore, the future development of MASSs and USVs anticipates the operation of autonomous maritime transportation without human decision making. Fundamentally, compliance with predetermined routes is a priority for an unmanned MASS to operate from the point of origin to the destination. In this case, the predetermined routes are safe and can be used to determine the shortest distance. Therefore, analyzing the safe conditions and determining the shortest distance are necessary. The safe and shortest route mentioned here can be determined by analyzing the operating patterns of ships equipped with an AIS. If AIS data accumulated over several years can be used to identify areas where ships frequently operate, this route implies safe and economical results [41,42]. Therefore, maritime traffic routes, maritime traffic networks, and shortest path planning should be based on AIS data. In this study, we differentiated the following characteristics to construct a maritime traffic network dataset for analyzing shortest path planning.

Areas where ships frequently navigate and where AIS data have been accumulated over several years can be selected as safe and economically efficient routes. Accordingly, a maritime spatial-temporal density analysis was performed to extract the areas where ships operate the most.

A line that can induce ship flows is required to separate the flow of maritime traffic within a density-based polygon. The IMO regulates TSSs for densely trafficked areas, which separate traffic flows in a two-way direction. The Delaunay triangulation algorithm is used to achieve this within the polygon. Two-way traffic flow can facilitate preventing collision accidents because ships operate in a consistent direction, and this method can be effectively applied to future navigation plans for MASSs and USVs.

To construct the network dataset, the safety depth values based on a digital chart (S-57) were included, and the shortest path planning algorithm based on Dijkstra's algorithm was proposed. The results of shortest path planning were compared with the actual route used by the training ship. One distinctive feature of this study is that it compares the actual route used for navigation rather than comparing various algorithms for finding the shortest path. A limitation of this study is that, while considering factors such as water depth and designated routes utilized by ships, it did not account for various environmental factors in the sea, such as weather conditions, sea states, and currents. Additionally, the study did not encompass various aspects of maritime conditions, including navigation rules, encounters between vessels, and collision avoidance, which are crucial for ship operations.

6. Conclusions

This study contributes to the literature by providing an understanding of shortest path planning based on existing spatial-temporal density analysis results and network datasets. The main significance of the research findings is that the maritime traffic network dataset enables ships to be used in an easy, time-efficient, and safe manner for determining the shortest path. The comparison between the maritime network dataset and the shortest path planning for the training ship revealed the superiority of the AIS data-based network dataset. The comparisons for both Case 1 and Case 2 indicated a reduction in sailing time by over one hour and a distance decrease of more than 40 km. Maritime traffic networks are important; therefore, various countries are actively conducting analyses using different methods. Based on previous studies, this research proposes a novel approach to extract polygons using spatial-temporal density, separates traffic flows in a two-way direction, and suggests safe and economical shortest path planning using digital chart depth values. However, this study has a limitation in that it requires an accurate knowledge of maritime laws and navigation patterns. Although the traffic flow was separated in a two-way direction, the actual traffic environment at sea is considerably more complex and diverse. Developing a maritime traffic network requires combining various elements, such as ship encounters, safe speed, TSS, and overtaking. However, the current network dataset only considers TSS and safety depth, which is a limitation. Future research should aim to develop a maritime network dataset by combining these factors. Furthermore, there is a need to advance research by considering and incorporating a broader range of marine environmental factors, including weather conditions, sea states, and currents.

The analyzed area focused on Korea's coastal waters, which generate various trade ports and hundreds of routes; however, this area has the potential to be connected globally. Accumulated AIS data over several years are a valuable resource for inferring safe and economical routes. If a maritime traffic network dataset is constructed using the AIS data from all maritime areas, it will be useful to ship operators, cargo managers, shipowners, and government officials. This topic is of great interest when considering the potential use of MASSs or USVs as unmanned vessels in the future.

Author Contributions: Conceptualization, J.-S.L. and T.-H.K.; methodology, J.-S.L.; software, J.-S.L.; validation, Y.-G.P., T.-H.K., and J.-S.L.; formal analysis, J.-S.L.; investigation, J.-S.L.; resources, J.-S.L.; data curation, T.-H.K.; writing—original draft preparation, J.-S.L.; writing—review and editing, Y.-G.P.; visualization, T.-H.K.; supervision, Y.-G.P.; project administration, J.-S.L.; funding acquisition, T.-H.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Korea Institute of Marine Science & Technology Promotion(KIMST) funded by the Ministry of Oceans and Fisheries(20200495, "Development of satellite based system on monitoring and predicting ship distribution in the contiguous zone.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: This research was conducted with data provided by the Ministry of Ocean and Fisheries.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Parola, F.; Risitano, M.; Ferretti, M.; Panetti, E. The drivers of port competitiveness: A critical review. *Transp. Rev.* **2017**, *37*, 116–138. [CrossRef]
2. Oh, J.Y.; Kim, H.J. Spatiotemporal analysis of vessel trajectory data using network analysis. *J. Korean Soc. Mar. Environ. Saf.* **2020**, *26*, 759–766. [CrossRef]
3. Tovar, B.; Wall, A. The relationship between port-level maritime connectivity and efficiency. *J. Transp. Geogr.* **2022**, *98*, 103213. [CrossRef]
4. Tocchi, D.; Sys, C.; Papola, A.; Tinessa, F.; Simonelli, F.; Marzano, V. Hypergraph-based centrality metrics for maritime container service networks: A worldwide application. *J. Transp. Geogr.* **2022**, *98*, 103225. [CrossRef]
5. Lee, J.S.; Cho, I.S. Extracting the maritime traffic route in Korea based on probabilistic approach using automatic identification system big data. *Appl. Sci.* **2022**, *12*, 635. [CrossRef]
6. International Convention for the Safety of Life at Sea SOLAS. 1974. Available online: http://www.imo.org/About/Conventions/ListOfConventions/Pages/International-Convention-for-the-Safety-ofLife-at-Sea-(SOLAS),-1974.aspx (accessed on 27 November 2023).
7. Yan, Z.; Xiao, Y.; Cheng, L.; Chen, S.; Zhou, X.; Ruan, X.; Li, M.; He, R.; Ran, B. Analysis of global marine oil trade based on automatic identification system (AIS) data. *J. Transp. Geogr.* **2020**, *83*, 102637. [CrossRef]
8. Monios, J.; Ng, A.K.Y. Competing institutional logics and institutional erosion in environmental governance of maritime transport. *J. Transp. Geogr.* **2021**, *94*, 103114. [CrossRef]
9. Chen, J.; Lu, F.; Peng, G. A quantitative approach for delineating principal fairways of ship passages through a strait. *Ocean Eng.* **2015**, *103*, 188–197. [CrossRef]
10. Akdağ, M.; Solnør, P.; Johansen, T.A. Collaborative collision avoidance for Maritime Autonomous Surface Ships: A review. *Ocean Eng.* **2022**, *250*, 110920. [CrossRef]
11. Tan, G.; Zhuang, J.; Zou, J.; Wan, L. Adaptive adjustable fast marching square method based path planning for the swarm of heterogeneous unmanned surface vehicles (USVs). *Ocean Eng.* **2023**, *268*, 113432. [CrossRef]
12. Baumler, R.; Bhatia, B.S.; Kitada, M. Ship first: Seafarers' adjustment of records on work and rest hours. *Mar. Policy* **2021**, *130*, 104186. [CrossRef]
13. Toffoli, A.; Lefèvre, J.M.; Bitner-Gregersen, E.; Monbaliu, J. Towards the identification of warning criteria: Analysis of a ship accident database. *Appl. Ocean Res.* **2005**, *27*, 281–291. [CrossRef]
14. Jo, S.H.; D'agostini, E. Disrupting technologies in the shipping industry: How will MASS development affect the maritime workforce in Korea. *Mar. Policy* **2020**, *120*, 104139. [CrossRef]
15. Kim, Y.J.; Lee, J.S.; Pititto, A.; Falco, L.; Lee, M.S.; Yoon, K.K.; Cho, I.S. Maritime traffic evaluation using spatial-temporal density analysis based on big AIS data. *Appl. Sci.* **2022**, *12*, 11246. [CrossRef]

16. European Marine Observation and Data Network. EMODnet. In *EU Vessel Density Map Detailed Method_v1.5. EMODnet Human Activities*; European Parliament and of the Council: Roma, Italy, 2019.
17. Pallotta, G.; Vespe, M.; Bryan, K. Vessel pattern knowledge discovery from AIS data: A framework for anomaly detection and route prediction. *Entropy* **2013**, *15*, 2218–2245. [CrossRef]
18. Fernandez Arguedas, V.F.; Pallotta, G.; Vespe, M. Maritime traffic networks: From historical positioning data to unsupervised maritime traffic monitoring. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 722–732. [CrossRef]
19. Wang, G.; Meng, J.; Han, Y. Extraction of maritime road networks from large-scale AIS data. *IEEE Access* **2019**, *7*, 123035–123048. [CrossRef]
20. Yan, Z.; Xiao, Y.; Cheng, L.; He, R.; Ruan, X.; Zhou, X.; Li, M.; Bin, R. Exploring AIS data for intelligent maritime routes extraction. *Appl. Ocean Res.* **2020**, *101*, 102271. [CrossRef]
21. Filipiak, D.; Wecel, K.; Strózyna, M.; Michalak, M.; Abramowicz, W. Extracting Maritime Traffic Networks from AIS Data Using Evolutionary Algorithm. *Bus. Inf. Syst. Eng.* **2020**, *62*, 435–450. [CrossRef]
22. Shah, B.C.; Gupta, S.K. Long-distance path planning for unmanned surface vehicles in complex marine environment. *IEEE J. Ocean. Eng.* **2020**, *45*, 813–830. [CrossRef]
23. Lee, W.H.; Choi, G.H.; Kim, T.W. Visibility graph-based path-planning algorithm with quadtree representation. *Appl. Ocean Res.* **2021**, *117*, 102887. [CrossRef]
24. Lee, H.T.; Choi, H.M.; Lee, J.S.; Yang, H.; Cho, I.S. Generation of Ship's passage plan using data-driven shortest path algorithms. *IEEE Access* **2022**, *10*, 126217–126231. [CrossRef]
25. Lee, J.S.; Lee, M.S.; Cho, I.S. Changes in Maritime Traffic Patterns According to Installation of Floating LiDAR Using Spatial Analysis. *IEEE Access* **2023**, *11*, 74784–74795. [CrossRef]
26. Lee, J.S.; Son, W.J.; Lee, H.T.; Cho, I.S. Verification of novel maritime route extraction using kernel density estimation analysis with automatic identification system data. *J. Mar. Sci. Eng.* **2020**, *8*, 375. [CrossRef]
27. Hanaoka, K.; Nakaya, T.; Yano, K.; Inoue, S. Network-based spatial interpolation of commuting trajectories: Application of a university commuting management project in Kyoto, Japan. *J. Transp. Geogr.* **2014**, *34*, 274–281. [CrossRef]
28. Korea Shipowners' Association. KSA. Korean Seaborne Trade Volume. 2023. Available online: https://oneksa.kr:4431/eng/shipping/volume.php (accessed on 27 November 2023).
29. International Maritime Organization (IMO). *Guidelines for the Onboard Operational Use of Shipborne Automatic Identification Systems (AIS)*; IMO: London, UK, 2002.
30. Son, W.J.; Cho, I.S. Development of collision risk assessment model for bridge across waterways based on traffic probability distribution. *Ocean Eng.* **2022**, *266*, 112844. [CrossRef]
31. Tsuji, K. Methods of survey for marine traffic. *Jpn. Inst. Navig.* **1996**, *129*, 8–18.
32. Lee, D.T.; Schachter, B.J. Two algorithms for constructing a Delaunay triangulation. *Int. J. Comput. Inf. Sci.* **1980**, *9*, 219–242. [CrossRef]
33. International Hydrographic Organization (IHO). IHO Transfer Standard for Digital Hydrographic Data (ed 3.1.0), Special Publication No. 57. *IHO, Monaco, Principality of Monaco*. 2000. Available online: https://iho.int/uploads/user/pubs/standards/s-57/31Main.pdf (accessed on 11 December 2023).
34. Kang, E.J.; LEE, H.T.; Kim, D.G.; Yoon, K.K.; Cho, I.S. Grouping Pilots' Maneuvering Types According to Berthing Velocity Using Agglomerative Clustering Algorithm. *J. Mar. Sci. Eng.* **2022**, *10*, 1452. [CrossRef]
35. Liu, L.; Shibasaki, R.; Zhang, Y.; Kosuge, N.; Zhang, M.; Hu, Y. Data-driven framework for extracting global maritime shipping networks by machine learning. *Ocean Eng.* **2023**, *269*, 113494. [CrossRef]
36. Dijkstra, E.W. A note on two problems in connexion with graphs. *Numer. Math.* **1959**, *1*, 269–271. [CrossRef]
37. Silveira, P.; Teixeira, Â.P.; Guedes Soares, C.G. AIS based shipping routes using the Dijkstra algorithm. *TransNav. Int. J. Mar. Navig. Saf. Sea Transp.* **2019**, *13*, 565–571. [CrossRef]
38. Wang, H.; Mao, W.; Eriksson, L. A three-dimensional Dijkstra's algorithm for multi-objective ship voyage optimization. *Ocean Eng.* **2019**, *186*, 10631. [CrossRef]
39. Czermański, E.; Oniszczuk-Jastrząbek, A.; Zaucha, J.; Pawłowska, B.; Matczak, M.; Szydłowski, Ł. Preconditions of new container terminal location in the Maritime Spatial Planning framework. A case study for the Central Port Concept in Gdansk. *Mar. Policy* **2021**, *130*, 104585. [CrossRef]
40. Ivana, R.; Jelena, K.P.; Mladen, Z. Role of Marine Spatial Data Infrastructure and Marine Cadastre in a Sustainable World. *J. Mar. Sci. Eng.* **2022**, *10*, 1407.
41. Lee, W.H.; Cho, S.W. AIS trajectories simplification algorithm considering topographic information. *Sensors* **2022**, *22*, 7036. [CrossRef]
42. Park, K.S.; Seo, Y.J.; Kim, A.R. Seaport Network based on Change of Korean Liner Service Pattern. *Asian J. Shipp. Logist.* **2017**, *33*, 221–228. [CrossRef]

Journal of
Marine Science and Engineering

Article

Deep Learning Applications in Vessel Dead Reckoning to Deal with Missing Automatic Identification System Data

Atefe Sedaghat, Homayoon Arbabkhah, Masood Jafari Kang and Maryam Hamidi *

Department of Industrial and Systems Engineering, Lamar University, Beaumont, TX 77710, USA; asedaghat@lamar.edu (A.S.); harbabkhah@lamar.edu (H.A.); mjafarikang@lamar.edu (M.J.K.)
* Correspondence: mhamidi@lamar.edu

Abstract: This research introduces an online system for monitoring maritime traffic, aimed at tracking vessels in water routes and predicting their subsequent locations in real time. The proposed framework utilizes an Extract, Transform, and Load (ETL) pipeline to dynamically process AIS data by cleaning, compressing, and enhancing it with additional attributes such as online traffic volume, origin/destination, vessel trips, trip direction, and vessel routing. This processed data, enriched with valuable details, serves as an alternative to raw AIS data stored in a centralized database. For user interactions, a user interface is designed to query the database and provide real-time information on a map-based interface. To deal with false or missing AIS records, two methods, dead reckoning and machine learning techniques, are employed to anticipate the trajectory of the vessel in the next time steps. To evaluate each method, several metrics are used, including R squared, mean absolute error, mean offset, and mean offset from the centerline. The functionality of the proposed system is showcased through a case study conducted in the Gulf Intracoastal Waterway (GIWW). Three years of AIS data are collected and processed as a simulated API to transmit AIS records every five minutes. According to our results, the Seq2Seq model exhibits strong performance (0.99 R squared and an average offset of ~1400 ft). However, the second scenario, dead reckoning, proves comparable to the Seq2Seq model as it involves recalculating vessel headings by comparing each data point with the previous one.

Keywords: online traffic monitoring; ETL pipeline; AIS data; vessel trajectory prediction; dead reckoning; GIWW

Citation: Sedaghat, A.; Arbabkhah, H.; Jafari Kang, M.; Hamidi, M. Deep Learning Applications in Vessel Dead Reckoning to Deal with Missing Automatic Identification System Data. *J. Mar. Sci. Eng.* **2024**, *12*, 152. https://doi.org/10.3390/jmse12010152

Academic Editors: Xinqiang Chen, Dongfang Ma and Ryan Wen Liu

Received: 16 December 2023
Revised: 4 January 2024
Accepted: 5 January 2024
Published: 12 January 2024

1. Introduction

The availability of maritime data, collected through an extensive network of terrestrial and satellite Automatic Identification System (AIS) receivers, has created unprecedented opportunities for transformative analyses and the extraction of valuable insights in maritime traffic monitoring. This abundance of information enables various crucial applications, including vessel trajectory prediction, anomaly detection, threat assessment, and tracking and classification of maritime activities [1].

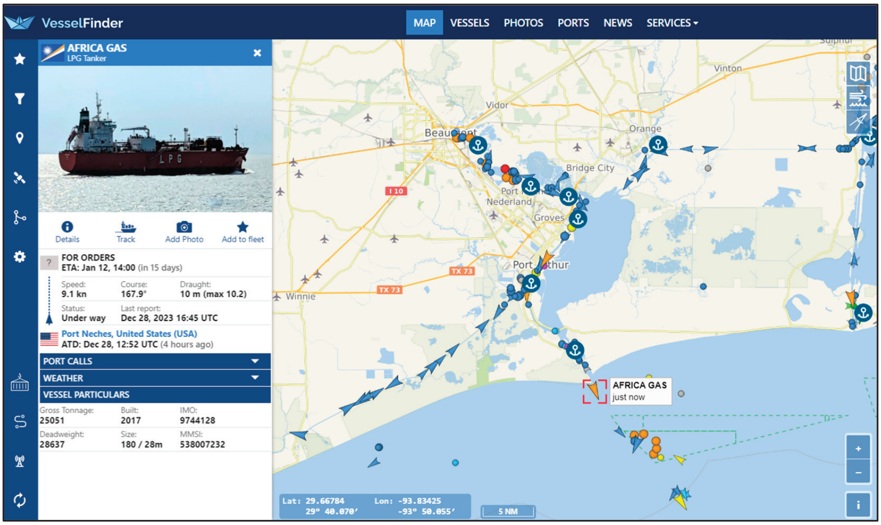
At the core of this data-driven revolution is AIS technology, which plays a central role in maritime operations for real-time tracking and monitoring of vessels. Utilizing Very High Frequency (VHF) signals, AIS facilitates the exchange of encoded information containing various attributes of a ship at regular intervals. These attributes include key details such as the ship's position coordinates, speed over ground, course over ground, Maritime Mobile Service Identities (MMSI), and more. AIS data are categorized into static and dynamic information, with static details encompassing essential ship-related information and dynamic data continuously transmitted and varying based on the vessel's motion [2,3].

The management challenges posed by the high volume and velocity of AIS data underscore the necessity for compression and efficient data processing. The frequent transmission of AIS signals generates substantial data, posing challenges for real-time analysis, decision-making procedures, and the development of intelligent services and applications [4]. To tackle this issue, compression techniques are applied to reduce storage and computing costs associated with processing AIS data. These techniques aim to decrease the overall data volume by retaining essential information while eliminating redundant or negligible data points [5].

In the practical landscape of vessel monitoring, existing systems such as VesselFinder [6], Marine Cadastre [7], and AccessAIS [8], as outlined in Table 1, demonstrate capabilities in vessel tracking. In Figure 1, the dashboard of the VesselFinder platform is shown. Such systems let users find vessel locations and some static information like a vessel's name, picture (if any), speed, destination (if provided), as well as estimated time of arrival (ETA). However, the current systems exhibit deficiencies in real-time analysis, data compression, and traffic analysis. To address these limitations, a novel system has been crafted. The proposed system not only monitors vessel movements but also conducts real-time data analysis, presenting the results on an interactive map. Additionally, the system is equipped to analyze historical data, enhancing its overall functionality.

Table 1. Comparison of current marine systems and our proposed system.

		Features					
		Real-Time Map	**Historical Analysis**	**Real-Time Analysis**	**Traffic Analysis**	**Data Volume Reduction**	**Dealing with Missing Records**
Current systems	VesselFinder	✓					
	Marine Cadastre	✓					
	AccessAIS	✓					
Proposed system	ETL pipeline	✓	✓	✓	✓	✓	✓



AFRICA GAS
LPG Tanker

FOR ORDERS
ETA: Jan 12, 14:00 (in 15 days)

Speed: 9.1 kn
Course: 167.9°
Draught: 10 m (max 10.2)
Status: Under way
Last report: Dec 28, 2023 16:45 UTC

Port Neches, United States (USA)
ATD: Dec 28, 12:52 UTC (4 hours ago)

VESSEL PARTICULARS
Gross Tonnage: 25051
Built: 2017
IMO: 9744128
Deadweight: 28637
Size: 180 / 28m
MMSI: 538007232

Lat: 29.66784 29° 40.070′
Lon: -93.83425 -93° 50.055′

Figure 1. Dashboard of VesselFinder.

In narrow waterways and channels, where operational intricacies are heightened, the significance of missing AIS points becomes even more pronounced. The scheduling of vessels in such areas often relies heavily on manual intervention and observation, lacking a systematic monitoring approach. Aside from manual scheduling, current Collision Avoidance Systems rely on observers to ensure safe navigation. The 1972 Convention (COLREGs) was a significant effort, introducing rules and guidelines governing vessel conduct, lights, sounds, and exemptions. The integration of advanced technologies like radar, AIS, and automatic radar plotting aid (ARPA) could provide maritime operators with real-time data, predictive tools, and automated alerts. However, these systems heavily depend on the accuracy and consistency of data inputs. AIS provides the major source of data, and any technical issues, data errors, or data absence can weaken the system's effectiveness. Apart from vessels that are not required to have an AIS transporter, there is a notable occurrence of AIS transponders intentionally or accidentally turning off. This adds a layer of complexity, as the absence of real-time vessel information can pose significant challenges for ensuring safe navigation. Our study introduces an intelligent framework designed to track vessel traffic effectively and predict the next location of vessel movements accurately in real time. The algorithm employs an Extract, Transform, and Load (ETL) pipeline to dynamically clean, compress, and process AIS data. Furthermore, it enriches raw AIS data with valuable information such as online traffic volume, origin/destination details, vessel trips, trip directions, and vessel routing. By storing the processed data in a database, this advanced system could be a replacement for the current of collecting and storing raw AIS data. To facilitate user interactions and access to the system's intelligent services, we have developed a user interface that allows end users to query the database and retrieve real-time information displayed on an interactive map. This intuitive interface empowers users to make informed decisions and gain valuable insights.

In the second phase of our study, we use processed data to predict the next location of vessel movements employing two distinct approaches: classical dead reckoning and machine learning methods. These approaches are evaluated and compared based on prediction errors, enabling an assessment of their performance to determine the most accurate prediction method. Combining data processing, prediction algorithms, and a user-friendly interface, our framework provides a comprehensive solution for online traffic monitoring and trajectory prediction. By leveraging its sequential-to-sequential architecture, the model can learn patterns from historical AIS data, predict vessel trajectories, and fill in the gaps caused by missing points, including instances where AIS transponders are intentionally or accidentally turned off. This not only enhances the accuracy of predictions but also introduces a level of automation to the monitoring and scheduling processes, compensating for the limitations of manual observation. The model's ability to predict trajectories even in the presence of missing data due to AIS transponder outages contributes to more robust and reliable maritime operations in narrow waterways. The algorithm's functionality is tested using the Gulf Intracoastal Waterway (GIWW). Three years of data are collected and fed to the ETL pipeline using a simulated API that sends AIS messages every 5 min. Results indicate that the proposed algorithm processes millions of data rapidly and predicts the vessel trajectory with 99% accuracy in terms of R squared. To reinforce our model, we also define other evaluation metrics as the mean offset from actual points as well as the mean offset from the channel centerline. On average, the best model predicts vessel points in a buffer of 1500 ft around actual points.

The paper is organized as follows: Processing based on historical AIS data (offline mode) and both historical and current AIS data (online mode) are discussed in Section 2. The methodology of the paper including the ETL process and prediction methods are presented in Section 3. In Section 4, the result using a dashboard is visualized and the prediction errors are investigated; in Section 5, the discussion and future direction are mentioned, and finally, the conclusions are presented in Section 6.

2. Literature Review

We examine research papers that have employed historical AIS data for vessel tracking and trajectory analysis, particularly focusing on online monitoring and intelligent frameworks. These frameworks dynamically compare real-time AIS data with historical data to analyze vessel movements and identify trajectories. In the final review, we delve into the application of machine learning, deep learning, and dead reckoning methods in predicting vessel trajectories.

2.1. Historical Analysis

Li et al. proposed a multi-step algorithm that integrates Dynamic Time Warping (DTW), Principal Component Analysis (PCA), and an improved center clustering approach for trajectory clustering. The goal is to identify customary routes and detect abnormal trajectories [9]. Zhang et al. utilized data-driven algorithms, including density-based spatial clustering of applications with noise (DBSCAN) and Ant Colony Optimization (ACO), to infer vessel routes from AIS data [10]. Ren et al. introduced a network based on a multi-clustering algorithm combining k-means, DBSCAN, and Affinity Propagation (AP) clustering methods to generate high-dimensional trajectories and measure their similarity [11]. Eljabu et al. emphasized the significance of automatic methods for extracting traffic routes from AIS data, demonstrating the potential of density-based clustering algorithms [12]. Kang et al. analyzed AIS data from the Houston Ship Channel to explore vessel congestion patterns, factors contributing to congestion, and speed variations [13]. Kabir et al. developed a framework and algorithms for capturing significant directional changes in vessel trajectories for maritime traffic management [14]. Zohoori et al. presented a vectorized algorithm for analyzing waterway traffic characteristics, reducing processing time compared to loop-based methods [15]. Wu et al. proposed an AIS-based method to identify hot spots in waterways experiencing frequent vessel conflicts and examined time-of-day impacts on conflict frequency [16]. Additionally, Wu investigated vessel travel behavior in hotspots using AIS data, focusing on speed distributions and flow speeds for different vessel types [17]. Zohoori et al. developed an algorithm to model and quantify delays caused by beam restrictions in narrow waterways, providing insights for vessel scheduling and expansion projects [18].

2.2. Real-Time Analysis

Evmides et al. introduced an intelligent framework for vessel traffic monitoring that integrates data analytics, machine learning, and visualization techniques [19]. Chi et al. proposed a framework to monitor vessel efficiency in real time using AIS data, leading to cost savings and environmental benefits [20]. Zhang and Li presented a methodology involving online data cleaning, compression, partition, and clustering of AIS data to identify traffic patterns and anomalies [2]. Kontopoulos et al. offered a method to detect intentional AIS switch-off in real time for improved safety [21]. Gao and Shai introduced a ship spatiotemporal key feature point extraction algorithm for AIS trajectory data, beneficial for ship traffic flow analysis [22]. Sedaghat et al. proposed a smart framework to dynamically separate and compress AIS data without compromising data quality, enabling the study of the online traffic flow of vessels [23].

2.3. Trajectories Prediction

We delve into trajectory prediction methods, categorizing them into two main groups: machine learning algorithms and deep learning algorithms. Machine learning algorithms utilize historical trajectory data and models like regression, decision trees, and support vector machines to predict future object movements based on past trajectories. In contrast, deep learning algorithms, including recurrent neural networks (RNNs) and sequence-to-sequence models, excel in capturing temporal dependencies and complex patterns, exhibiting promising results in trajectory prediction.

2.3.1. Machine Learning Methods

Fuentes extensively discussed various machine learning prediction techniques [24], encompassing regression models such as the Linear Regression Model (LRM) [25], the autoregressive model (AR) [26], Support Vector Regression (SVR), Gaussian Process Regression

(GPR), neural networks like artificial neural networks (ANN), as well as the Kalman Filter (KF) and Random Forest (RF). These models typically require ship velocity, acceleration, heading, and position data for training. The Linear Regression Model (LRM) is often used for time series prediction due to its real-time forecasting capability, although it may face challenges in predicting long-term linear ship trajectories and susceptibility to overfitting. However, the Kalman Filter (KF) excels in estimating the state of moving targets and making predictions. The Random Forest (RF) algorithm, a versatile method incorporating decision trees, finds applications in predicting arrival ports and sailing times of ships.

2.3.2. Deep Learning Methods

Deep learning methods prove highly effective in handling complex and dynamic trajectory data, showcasing robust learning and adaptability. Notably, they demonstrate outstanding performance in predicting ship trajectories based on AIS data [27]. The Long Short-Term Memory (LSTM) model addresses the short-term memory issue of RNNs by incorporating dedicated gate controls for both short and long-term memories in ship trajectory prediction. Integrated models based on LSTM, such as the multiple vessels prediction model [28], vessel location prediction [29], the Trajectory-based Similarity Search Prediction model (TSSPL) [28], the Context-Aware LSTM (C-LSTM) model [30], and the federated deep learning-based method (Conv LSTM) [31], handle complex trajectory problems. The authors of [1,32] develop a model for predicting vessel trajectories using AIS data, employing neural sequence-to-sequence models with an LSTM encoder-decoder architecture. Their experiments on real AIS data demonstrate the superiority of these models over traditional methods. Abada et al. explore the synergy between deep learning and big data, showcasing the prowess of artificial neural networks in deciphering complex patterns within extensive datasets. It highlights applications in predictive analytics, image analysis, and language processing [33].

3. Methodology

We consider an Extract, Transform, Load (ETL) pipeline to deal with the extraction of the stream of AIS data, process the data, and load the result into a database as depicted in Figure 2. Not only does this procedure help to build customized and useful information to be used instead of raw AIS data but it also helps in making online predictions of vessel movements; it is a big help for the port authorities to know the estimated location of the vessels in a real-time manner when facing a disconnection or a vessel intentionally switch off its location. To do the prediction, some traditional methods and deep learning methods are implemented, and the error of each method has been evaluated. In the following, every component of the ETL pipeline is explained in detail.

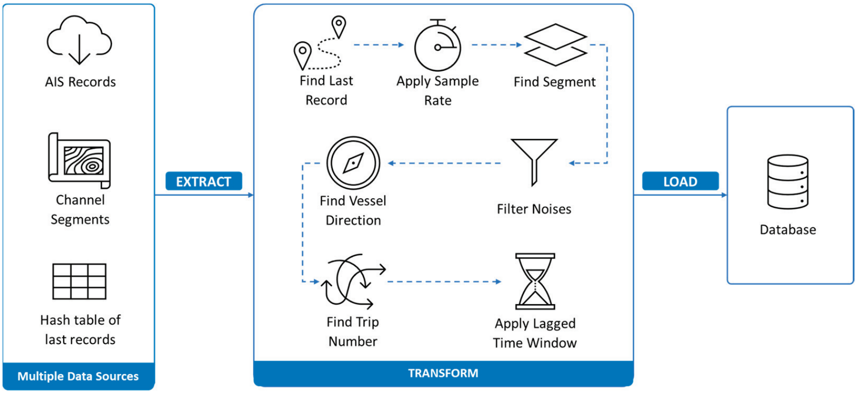


Figure 2. Proposed system (ETL pipeline) diagram.

3.1. Extracting Data

The first step of the ETL pipeline starts by extracting data from an external source of AIS data. Practically, the stream of real-time data would be provided through APIs. To testify to the proposed method's functionality in real-world cases, we collected historical data for North America and simulated an API that generates AIS messages in a given time interval. The retrieved data are then filtered based on a specific boundary called the Area of Interest (AoI) to extract only the relevant portions of interest. After filtering, the code performs a data-cleaning process by removing any incomplete or null values. This ensures that the data are accurate and suitable for further analysis or processing. The final result of the code is a cleaned dataset containing the essential and valid data, ready to be used for the Transformation step.

3.2. Transforming Data

The second component of the ETL procedure is the "Transform". This step considers the output we have gathered as clean raw AIS data from the Extract part as an input for further processing. The other input is a geographical information system (GIS) layer of all waterways located in the AoI. Thanks to the QGIS toolbox, we employ the "split line to maximum length" function to split the GIS layer into smaller, equally sized segments as depicted in Figure 3. Then a series of functions have been applied to the input data to transform the raw AIS data. The following sections talk about these functions in detail.

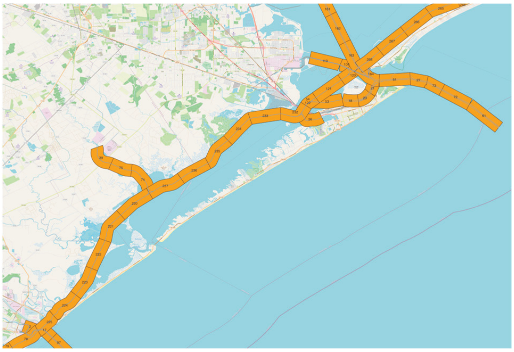


Figure 3. Generated segments along waterways in the AoI.

Step 1: Finding the last record

The algorithm is developed based on a hash table, which stores the latest record of each vessel's data. As the algorithm receives a stream of data; it compares each vessel's record with the last record already stored in the hash table to compute features such as the time and space distance between a pair of consecutive points. The calculated features will be used in the next steps. If there is no record in the hash table, the algorithm stores the new data as the last record. However, if the vessel's information is already stored, the algorithm updates the existing data with the newly received data. This allows the algorithm to progressively calculate and update the calculated values in real time, ensuring the latest information is stored and processed accurately.

Step 2: Applying the sample rate

The other inputs of the transform algorithm are the minimum and maximum acceptable time difference between two consecutive records collected for a vessel. Assuming Δt is a time difference between each vessel's AIS record and the previous record, the algorithm keeps records only when Δt is between the minimum and maximum time difference. It drops redundant information and breaks the sequence of the AIS records if there is a long pause. Our assumption is to set the minimum acceptable time difference at 5 min and the maximum acceptable time difference at 120 min (2 h).

Step 3: Finding segments

This step takes the intersection of each AIS record and the pre-defined segments of the AoI to assign the segment's ids to the records. this helps us summarize the data for each segment and calculate some traffic features like traffic density. Additionally, it drops any records located outside of the segments.

Step 4: Filtering noises

Trajectory trackers sometimes may generate wrong records that appear as noises in a sequence of locations. To capture and drop such records, we add this step to our transformer. This step uses Δt and Δl—that is, the distance between each record and the previous record—to calculate the average speed. If the average speed is not in a rational range of the vessel's speed, from 0 to 30 knots, it drops the record.

Step 5: Determining the vessel's direction

To define the vessel's direction, we first define the unit vector of the segments' centerline. As Figure 4 shows, we keep the first and last point of each segment centerline and create a segment vector $\vec{a}$. Next, we consider vector b as the distance difference between the new record and the previous record for each vessel. Finally, we utilize the inner product, Equation (1), to calculate the angle between these two arrows:

$$a{\cdot}b = |a|\ |b| \cos(\theta) \tag{1}$$

where
a: segment centerline vector
b: vessel's movement vector

Figure 4. Segment and vessel vectors.

These two vectors (a, b) make an angle, θ. Based on Equation (1), we have the following:

$$\begin{cases} 0 < \theta \leq 90 & \text{if} \quad a{\cdot}b > 0 \\ 270 < \theta \leq 360 & \text{if} \quad a{\cdot}b > 0 \\ 90 < \theta \leq 270 & \text{if} \quad a{\cdot}b < 0 \end{cases} \tag{2}$$

Therefore, if the inner product of two vectors has a positive value, it means that the cos θ is a positive value, which means we can interpret the two vectors as having the same direction, and we consider it as the "inbound" direction, while if cos θ is a negative value, the direction of the vessel is in the opposite direction of the segment vector, so the vessel's direction is "outbound". The example, shown in Figure 3, is an inbound trip because the inner product of (a, b) is a positive value and the θ between them is simply between 0 and 90 degrees. This method requires all segment unit vectors to be organized in the head-to-tail position. We use the QGIS toolbox to order segments and find each segment unit vector. We also assume the vessel's direction is zero when its speed is below 2 knots. Therefore, we define three directions based on the vessel's speed, the inner product of the vessel's movement vector, and the segment vector as listed in Table 2.

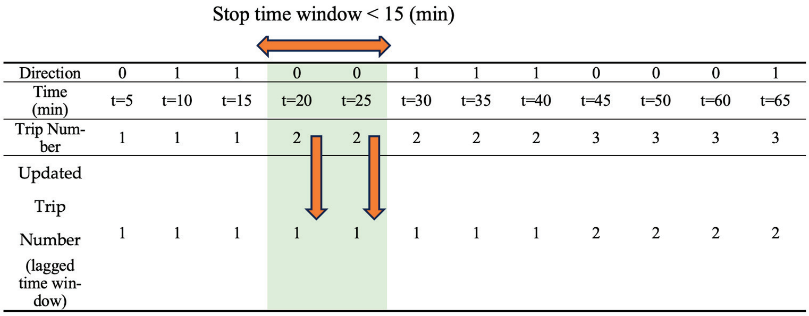
Table 2. Direction Values.

Vessel Status	Inbound	Stop	Outbound
Direction value	1	0	−1

Step 6: Determining the Trip Number

The proposed algorithm aims to determine trip numbers, considering every stop-to-stop interval as a separate trip. To determine the trip number, the code first checks if the new record's direction is "stopped" and differs from the previous record's direction. If this condition is met, it implies the start of a new trip. The code increments the trip number by one, assigning it as the previous record's trip number plus one.

In instances where vessels initiate a trip and make stops during the journey, the algorithm tends to designate each stop as a new trip number, even if the vessel halts briefly and does not signify the initiation of a new trip. To address this, time-lagged windows have been implemented. For trips in which vessels stop during the journey, if the stop time is below a predefined threshold, the algorithm disregards the stop status, considering the trip number as the previous trip number and not initializing it as a new trip. Consequently, the algorithm updates the last vessel's recorded trip number with a delay, ensuring that the stop status of the vessel is deemed negligible. For more clarification, in Figure 5, we have a record from an MMSI every five minutes, and the algorithm considers each stop to stop as a single trip. Therefore, before using a time-lagged window, the trip number for this specific vessel can increase to three, while in the movement of the vessel, it stops for about 5 min. If we predefine our threshold at 15 min, based on the time-lagged window, we must ignore the stop status and consider it to be the previous trip number. In this case, we have two trip numbers, since at the third stop, the vessel does not stay less than 15 min. Therefore, we initiate a new trip number.



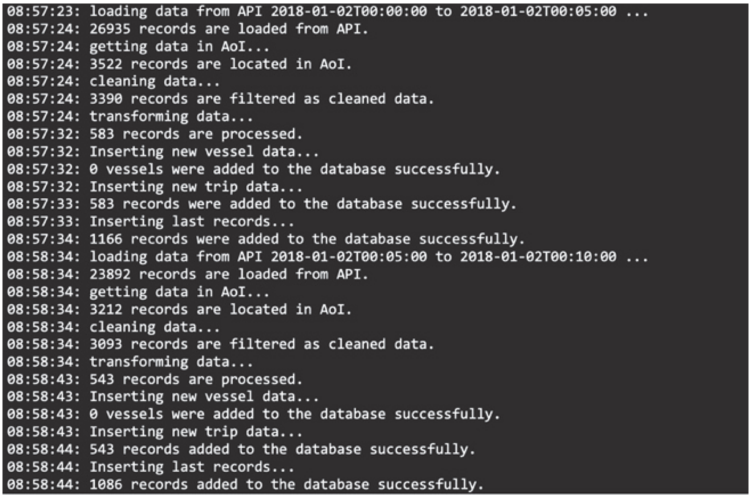
Stop time window < 15 (min)

Direction	0	1	1	0	0	1	1	1	0	0	0	1
Time (min)	t=5	t=10	t=15	t=20	t=25	t=30	t=35	t=40	t=45	t=50	t=60	t=65
Trip Number	1	1	1	2	2	2	2	2	3	3	3	3
Updated Trip Number (lagged time window)	1	1	1	1	1	1	1	1	2	2	2	2

Figure 5. Updated Trip number using a time-lagged window.

3.3. Load the Processed Data

The final stage of the ETL pipeline involves the loading of processed data from the preceding section into a database. We have developed a comprehensive database to store the processed data obtained from the ETL pipeline output. This database incorporates a combination of static and dynamic tables to effectively organize the data. The static tables encompass vessel profile, segments, vessel status, and vessel type. On the other hand, the dynamic tables consist of trips and the last records, which are used for efficient tracking and analysis. As mentioned above, we developed a simulated API to replicate real-world operations of receiving a stream of AIS records. The API uses actual AIS data from 2018 to 2022 but releases them in 5 min time intervals. Then, we connect our pipeline to this API and set it running for all data. Figure 6 shows the log message when running the pipeline. As shown in log messages, the pipeline is highly efficient so that it can process 26,000 records in 10 s and compress them into 583 records.



```
08:57:23: loading data from API 2018-01-02T00:00:00 to 2018-01-02T00:05:00 ...
08:57:24: 26935 records are loaded from API.
08:57:24: getting data in AoI...
08:57:24: 3522 records are located in AoI.
08:57:24: cleaning data...
08:57:24: 3390 records are filtered as cleaned data.
08:57:24: transforming data...
08:57:32: 583 records are processed.
08:57:32: Inserting new vessel data...
08:57:32: 0 vessels were added to the database successfully.
08:57:32: Inserting new trip data...
08:57:33: 583 records were added to the database successfully.
08:57:33: Inserting last records...
08:57:34: 1166 records were added to the database successfully.
08:58:34: loading data from API 2018-01-02T00:05:00 to 2018-01-02T00:10:00 ...
08:58:34: 23892 records are loaded from API.
08:58:34: getting data in AoI...
08:58:34: 3212 records are located in AoI.
08:58:34: cleaning data...
08:58:34: 3093 records are filtered as cleaned data.
08:58:34: transforming data...
08:58:43: 543 records are processed.
08:58:43: Inserting new vessel data...
08:58:43: 0 vessels were added to the database successfully.
08:58:43: Inserting new trip data...
08:58:44: 543 records added to the database successfully.
08:58:44: Inserting last records...
08:58:44: 1086 records added to the database successfully.
```

Figure 6. ETL pipeline log messages for a couple of hours.

3.4. Trajectory Prediction

This section discusses two distinct approaches used for predicting vessels' trajectories. The first method is a traditional approach. In contrast, the second approach involves utilizing a sequence-to-sequence recurrent neural network model (RNN).

3.4.1. Dead Reckoning

In the realm of vessel navigation, a traditional prediction method known as dead reckoning is employed to estimate the next position in a series of trajectory data using current and previous records. Dead reckoning allows us to estimate the next location of the vessel based on its past movements, even when real-time data, such as AIS data, are unavailable or disrupted. There are two scenarios in which dead reckoning comes into play. In Scenario I, we utilize the speed and course over ground obtained from the AIS record as an input to predict its next location. On the other hand, in Scenario II, we rely on the pre-record location of the vessel to find its next location. By extrapolating the vessel's historical data, we can project its next position, assuming it maintains a consistent speed. To facilitate this prediction process, we undertake the following steps:

Step 1. Preparing the inputs

To streamline the process, we retrieve relevant data from the database. In the trip table, we introduce two additional columns that showcase the pre-record and next-record locations of the vessels.

Step 2. Dead reckoning method

Assuming our aim is to obtain predictions for each specific time interval, we apply Equation (3) to calculate the distance traveled by the vessel within the defined time interval, which is assumed to be 5 min (0.083 h). Regarding [34], in Equation (4) the angular distance is calculated by dividing the linear distance by the earth's radius. In

Equation (5), the latitude of the next point is calculated. To find the longitude of the next point, first we should calculate the projected latitude difference in Equation (6). The ratio n in Equation (7) is introduced to account for the fact that, as the vessel moves along a rhumb line, the meridional scale (change in latitude) is not equal to the zonal scale (change in longitude) due to the convergence of meridians toward the poles. Therefore, the term n is used to adjust for the variation in the size of a degree of the longitude with the latitude. We have the longitude difference and the longitude of the next point in Equations (8) and (9), respectively.

$$\mathrm{d} = 1.15\ \mathrm{s}\cdot\Delta_t \tag{3}$$

$$\delta = \mathrm{d}/\mathrm{R} \tag{4}$$

$$\varphi_2 = \varphi_1 + \delta\cdot\cos(\theta) \tag{5}$$

$$\Delta\psi = \ln(tan(\pi/4 + \varphi_2/2)/tan(\pi/4 + \varphi_1/2) \tag{6}$$

$$\mathrm{n} = \Delta\varphi/\Delta\psi \tag{7}$$

$$\Delta\lambda = \delta\cdot sin(\theta)/\mathrm{n} \tag{8}$$

$$\lambda_2 = \lambda_1 + \Delta\lambda \tag{9}$$

where
d: distance in mile
s: vessel's current speed in knots
Δ_t: given time interval (5 min = 0.083 h)
R: earth's radius in miles (3958.80 miles)
θ: vessel's course over ground
φ_1: latitude of current point
φ_2: latitude of next point
$\Delta\varphi$: latitude difference
$\Delta\psi$: projected latitude difference
n: adjustment ratio for the variation in the size of a degree of longitude with latitude
$\Delta\lambda$: longitude difference
λ_1: longitude of current point
λ_2: longitude of next point
φ_0: latitude of previous point
λ_0: longitude of previous point

In scenario II, instead of using the course over the ground, as recorded in the AIS data, we calculate it by comparing the current record and the previous one. Figure 7 shows the dead reckoning method for finding the next location of the vessel using the pre-record and the new record.

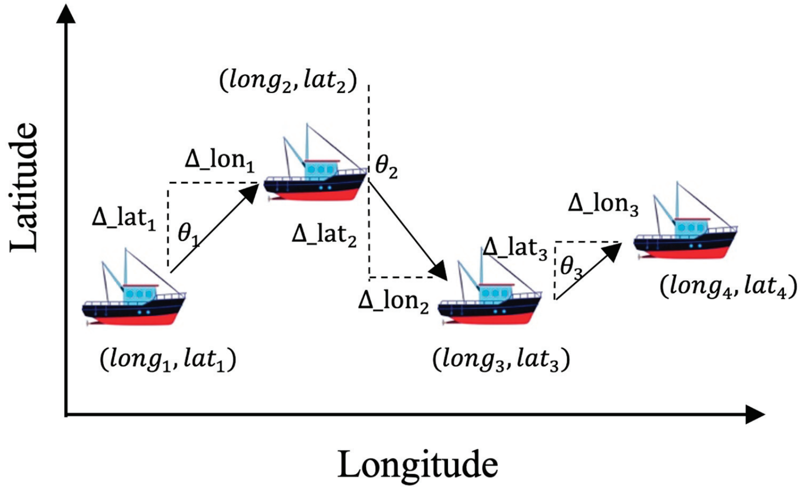


Figure 7. Dead reckoning.

3.4.2. RNN Method

The sequence-to-sequence recurrent neural network (RNN) method is a more advanced approach for predicting the vessel's next timestamp location. This technique leverages the power of deep learning to model sequential data effectively. The RNN architecture is designed to handle sequential input data, such as a vessel's position and timestamp over time. It is capable of capturing temporal dependencies and learning complex patterns from the historical trajectory of the vessel. However, standard RNNs have limitations in capturing long-term dependencies, making them less suitable for tasks requiring long-term memory [1]. Long Short-Term Memory (LSTM) is a powerful variant of RNNs that addresses the challenges of capturing long-term dependencies in sequential data, making it well-suited for tasks that require modeling complex sequential patterns, such as sequence-to-sequence learning and natural language processing.

3.4.3. LSTM Structure

LSTM is a type of recurrent neural network (RNN) architecture designed to address the vanishing gradient problem and capture long-term dependencies in sequential data. It is particularly effective in tasks involving sequences, such as time series prediction and natural language processing. The components include the following:

- Input Gate: Determines which information from the current input should be stored in the cell state;
- Forget Gate: Controls what information should be discarded from the cell state;
- Cell State updates: Maintains the long-term memory information;
- Output Gate: Determines the next hidden state based on the cell state.

3.4.4. Sequence-to-Sequence Model

A sequence-to-sequence (Seq2Seq) model (Figure 8) is designed for tasks where the input and output are both sequences of varying lengths. Common applications include machine translation and text summarization. The components include the following:

- Encoder: Processes the input sequence and encodes it into a fixed-size context vector;
- Decoder: Generates the output sequence based on the context vector produced by the encoder.

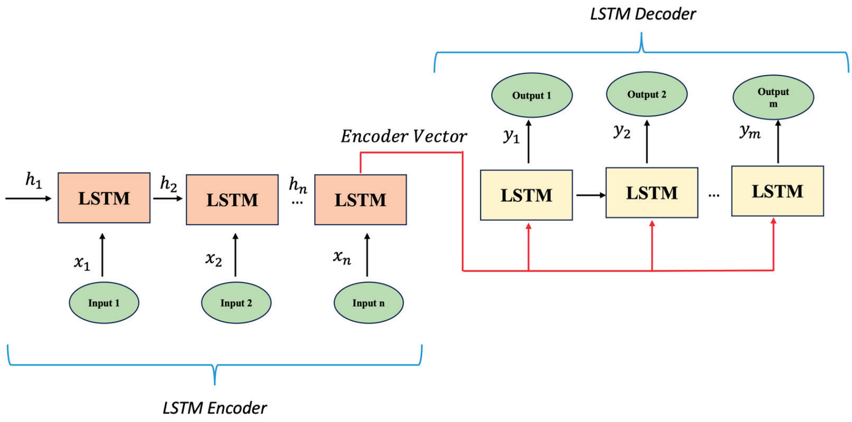


Figure 8. The Seq2Seq model structure.

In our case, we use different lengths to predict the points, so we use a Seq2Seq structure, which is structured as below:

- Step 1: Preparing the inputs;

  First, we call processed AIS data from the database and store it as a dataset.

- Step 2: Feature selection and normalization.

  Vessel latitude and longitude coordinates are essential for capturing the vessel's movement patterns. Therefore, we use these two features as the input for the Seq2Seq model. Additionally, as shown in Table 3, we add the vessel's speed over the ground, the course over the ground, and the heading independent variables. Then we perform data normalization on the selected columns to bring them to a similar scale. This is crucial for ensuring that the LSTM model can effectively learn from the data and avoid numerical instabilities during training.

Table 3. X and Y features.

Features	Speed over Ground	Course over Ground	Vessel Heading	Longitude	Latitude
X features	✓	✓	✓	✓	✓
Y features				✓	✓

- Step 3: Converting data into tensors.

  A sliding window method is applied to convert data tables into a three-dimensional matrix tensor. Figure 9 illustrates how the sequences, and their corresponding target values, have been selected. The output of this step is x and y tensors, which are the input sequence (here this is four sequences) and target sequence (here this is two sequences), respectively.

- Step 4: Training the Seq2Seq model.

  We split the data into 80% training and 20% validation sets. This operation does not require data shuffling and sorts data based on the record date and time to choose the first 80 percent of the data as the training dataset. As depicted in Figure 10, three years of data from January 2018 to April 2020 is selected and based on the chronological split, 80% of the data falls between Jan 2018 and Aug 2019.

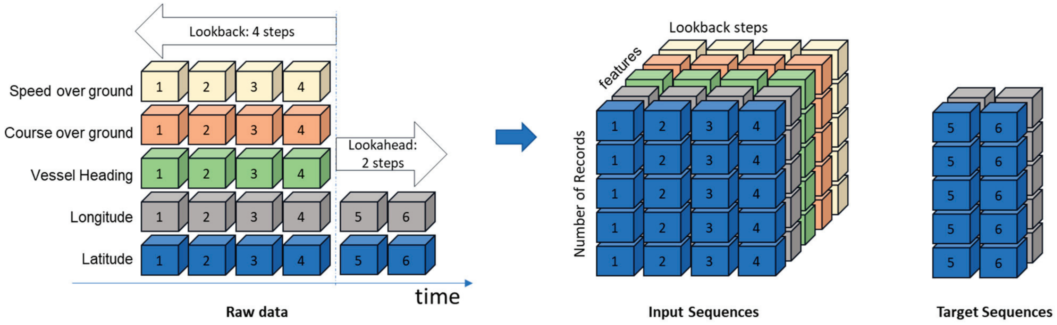


Figure 9. An example of input and target sequences with different lengths feed into the Seq2Seq model.

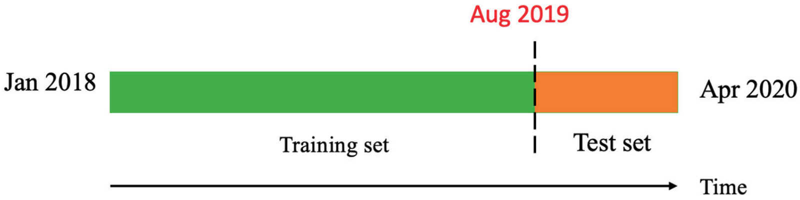


Figure 10. Chronological split.

- Step 5: Seq2Seq model architecture.

In the context discussed in Section 3.4.4, where input and target sequences vary in length, the model architecture, as illustrated in Figure 8, has been tailored accordingly. To accommodate diverse configurations for sensitivity analysis, the generalized format is presented in Figure 11. Using TensorFlow, we create a sequence model with 50 epochs and 64 hidden layers. The goal is to predict the vessel's trajectory based on historical data using this model. The initial step involves selecting the input to the model with a specified lookback value of 5, where 5 denotes the number of features in the X dataset. This input is then fed into the LSTM layer, producing an output with a hidden layer of 64 units and a repeated vector, also known as the encoder vector. This fixed vector is generated 'm' times and serves as the input for the subsequent LSTM layer, utilizing the value (lookahead, 64). Consequently, the final output of the model is represented as a lookahead value of 2, where 2 signifies the number of features in the Y dataset. This architectural design enables the model to effectively handle varying lookback and lookahead values during the sensitivity analysis.

- Step 6: Model evaluation.

In order to evaluate the model performance, the following metrics are defined:

- R2

R-squared measures the proportion of the variance in the dependent variable (the actual next coordinates) that is explained by the independent variable (the predicted next coordinates). It ranges from 0 to 1, where 1 indicates a perfect fit, meaning that the model's predictions perfectly match the actual data.

$$R^2 = 1 - \frac{\sum_{i=1}^{n} \left(y_i - \widetilde{y_i}\right)^2}{\sum_{i=1}^{n} (y_i - \overline{y_i})^2} \tag{10}$$

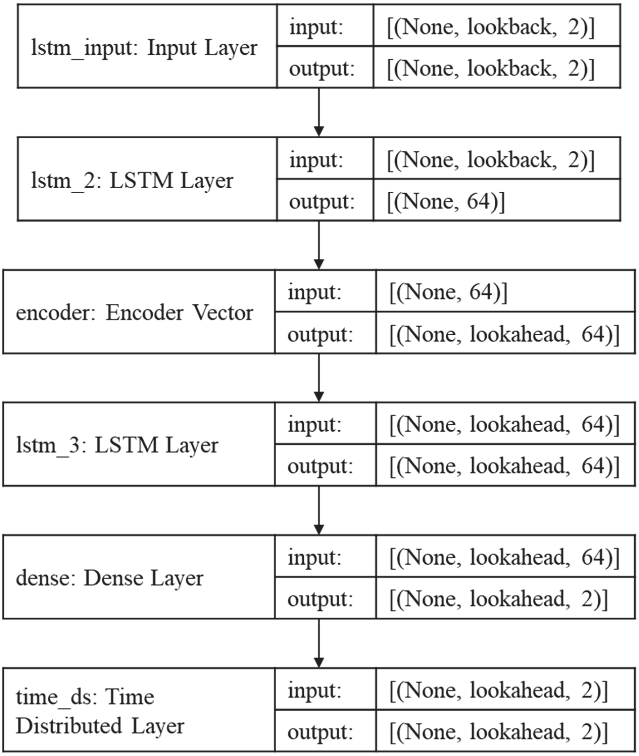


Figure 11. Seq2Seq model architecture.

- Mean Square Error (*MSE*)

MSE calculates the average of the squared differences between the predicted and actual values. It penalizes large errors more heavily. A lower *MSE* indicates that the model's predictions are closer to the actual values.

$$MSE = \frac{1}{n\sum_{i=1}^{n}\left(y_i - \widetilde{y_i}\right)^2} \tag{11}$$

- Mean Absolute Error (*MAE*)

MAE calculates the average of the absolute differences between the predicted and actual values. It provides a measure of the average magnitude of errors.

$$MAE = \frac{1}{n}\left|\left(y_i - \widetilde{y_i}\right)\right| \tag{12}$$

- Mean Absolute Percentage Error (*MAPE*)

Aside from the conventional metrics, we calculate the distance between the predicted and actual coordination in miles and feet and take the average of the distance of all the predicted points.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{\left(y_i - \widetilde{y_i}\right)}{y_i}\right| \times 100 \tag{13}$$

- Mean offset from Centerline (*MFC*)

We determine the mean offset from the channel centerline by measuring the distance between the predicted points and the centerline of the channel and then calculating the average of these distances. This helps us to have a better estimation of our model if it is predicting the point in the land rather than the waterway. The red circles in Figure 12 are predicted points and the dashed line is the centerline of the channel.

$$MFC = \frac{1}{n}\left|\vec{r}\right| \cdot \sin(\theta) \tag{14}$$

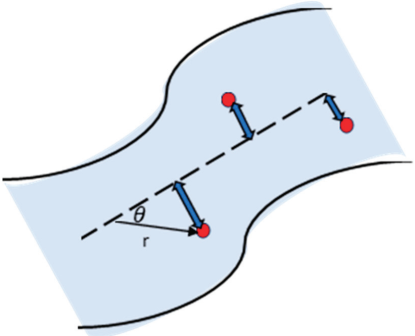
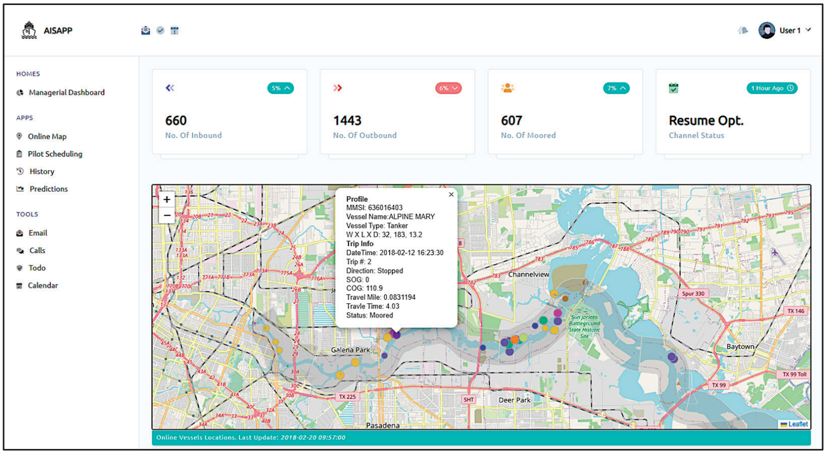


Figure 12. Mean offset from the centerline.

4. Application

4.1. User Interface Dashboard

The proposed algorithm has been deployed for implementation in the Texas coastal lines region, specifically targeting the GIWW (Gulf Intracoastal Waterway). As part of the implementation, we simulate a continuous stream of AIS data from 2018. These data are then processed using an ETL pipeline and stored in a database. To provide an intuitive user interface, we have designed a dashboard that allows end users to retrieve the data they need. The dashboard initially displays real-time vessel traffic, as shown in Figure 13. In addition to real-time data, the dashboard provides access to historical information. Users can query various metrics such as traffic flow, dwell time, OD matrix, trip generation, trip attraction, and individual vessel trips. These queries can be filtered based on the vessel type and specific date ranges.



660
No. Of Inbound

1443
No. Of Outbound

607
No. Of Moored

Resume Opt.
Channel Status

Profile
MMSI: 636016403
Vessel Name:ALPINE MARY
Vessel Type: Tanker
W X L X D: 32, 183, 13.2
Trip Info
DateTime: 2018-02-12 16:23:30
Trip #: 2
Direction: Stopped
SOG: 0
COG: 110.9
Travel Mile: 0.0831194
Travle Time: 4.03
Status: Moored

Online Vessels Locations. Last Update: *2018-02-20 09:57:00*

Figure 13. Online traffic vessel monitoring at a section at GIWW region.

4.2. Data Processing Efficiency

The proposed ETL pipeline is highly efficient, processing data at a rate of 0.000001 s per record. For example, if we want to process ten million records, it will take just ten seconds using our algorithm. The speed of this method represents a notable advancement compared to traditional data processing. It allows for real-time monitoring of vessel traffic efficiently. Additionally, the ETL has been tested on a simulated environment and can work simultaneously with the current AIS collection systems. As a result, it does not require a high-end computer to process raw AIS data because it processes a chunk of the most recent data at each iteration.

4.3. Prediction Evaluation

As discussed, in the previous section, the dead reckoning method and LSTM algorithms have been implemented to predict the next location of the vessel's movement. To visualize the prediction of both methods, we use the folium library in Python. The real location and the predicted location of a single vessel using the dead reckoning II method are depicted in red and green colors, respectively, in Figure 14.

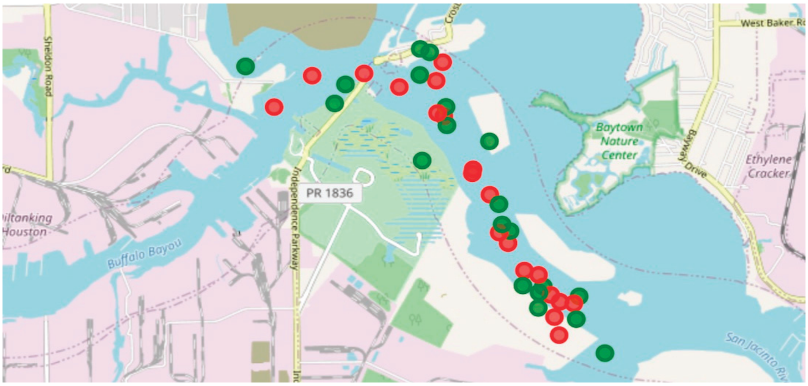


Figure 14. Real vs. predicted locations.

To evaluate the method in Scenarios I and II, the mean offset and other model evaluation metrics have been investigated, and the result is shown in Table 4. While the traditional method may be simple to implement, it may not capture complex patterns or account for irregularities in the vessel's movement. Additionally, the first two approaches can only predict one location ahead of time, and they are not able to predict more than one step in the future. They also may not perform well in situations where the vessel's trajectory is subject to sudden changes or nonlinear behavior.

Table 4. Model evaluation results.

Metrics	Dead Reckoning I	Dead Reckoning II	Seq2Seq
r square	0.9996	0.9999	0.9999
mean absolute error	0.0111	0.0032	0.0028
mean square error	0.0149	0.0064	0.0046
mean absolute percent error (%)	2.3	0.68	0.59
mean offset (mile)	1.03	0.32	0.28
mean offset (ft)	5433	1674	1454
mean offset from centerline(mile)	---	---	0.12

4.4. Sensitivity Analysis

To gain deeper insights into the impact of the input sequence length on the predictions, various values were experimented with for both lookback and lookahead settings. Through a comprehensive analysis of these combinations, it was determined that the optimal configuration for lookback and lookahead is (5, 3) as shown in Table 5. This signifies that the model exhibits strong predictive performance when provided with five historical records to forecast the subsequent three records. The sensitivity analysis for different combinations of lookback and lookahead values is shown in Figure 15, which denotes that the combination (5, 3) shows better results in terms of the mean distance. Additionally, Figure 16 provides a visual representation of both accuracy and loss per epoch. The graphs on the right illustrate the loss per epoch, showcasing closely aligned training and validation curves, indicating that the model is not overfitting. On the left, the accuracy per epoch plots reveals consistently high values, approaching one, indicative of a positive trend. Notably, the a1 plot exhibits particularly promising results, suggesting that selecting a lookahead of three yields enhanced the accuracy. While other lookahead values display slightly decreased accuracy compared to lookahead three, the overall trend remains favorable. This indicates that opting for lookahead values of five, seven, and ten still maintains a satisfactory level of accuracy without a significant decline.

Table 5. Sensitivity analysis with different combinations of lookback and lookahead.

Mean Offset (mile)		**Lookahead Window**		
		3	**5**	**7**
Lookback window	5	4.03	4.72	5.25
	10	5.74	6.7	5.94
	15	6.27	7.02	7.86

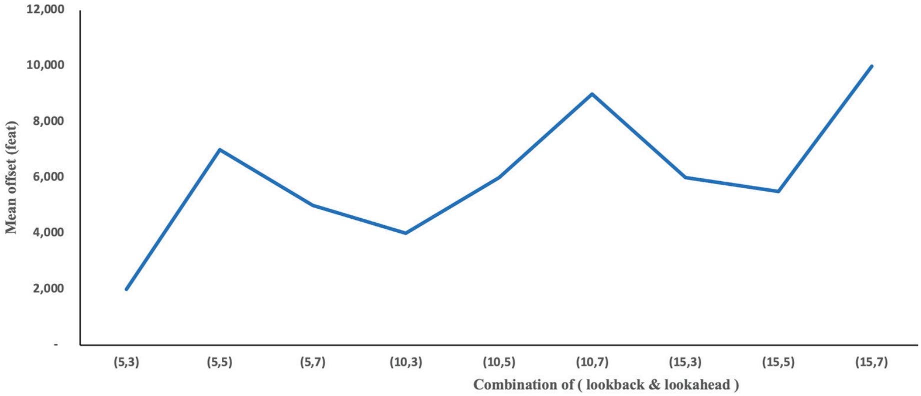
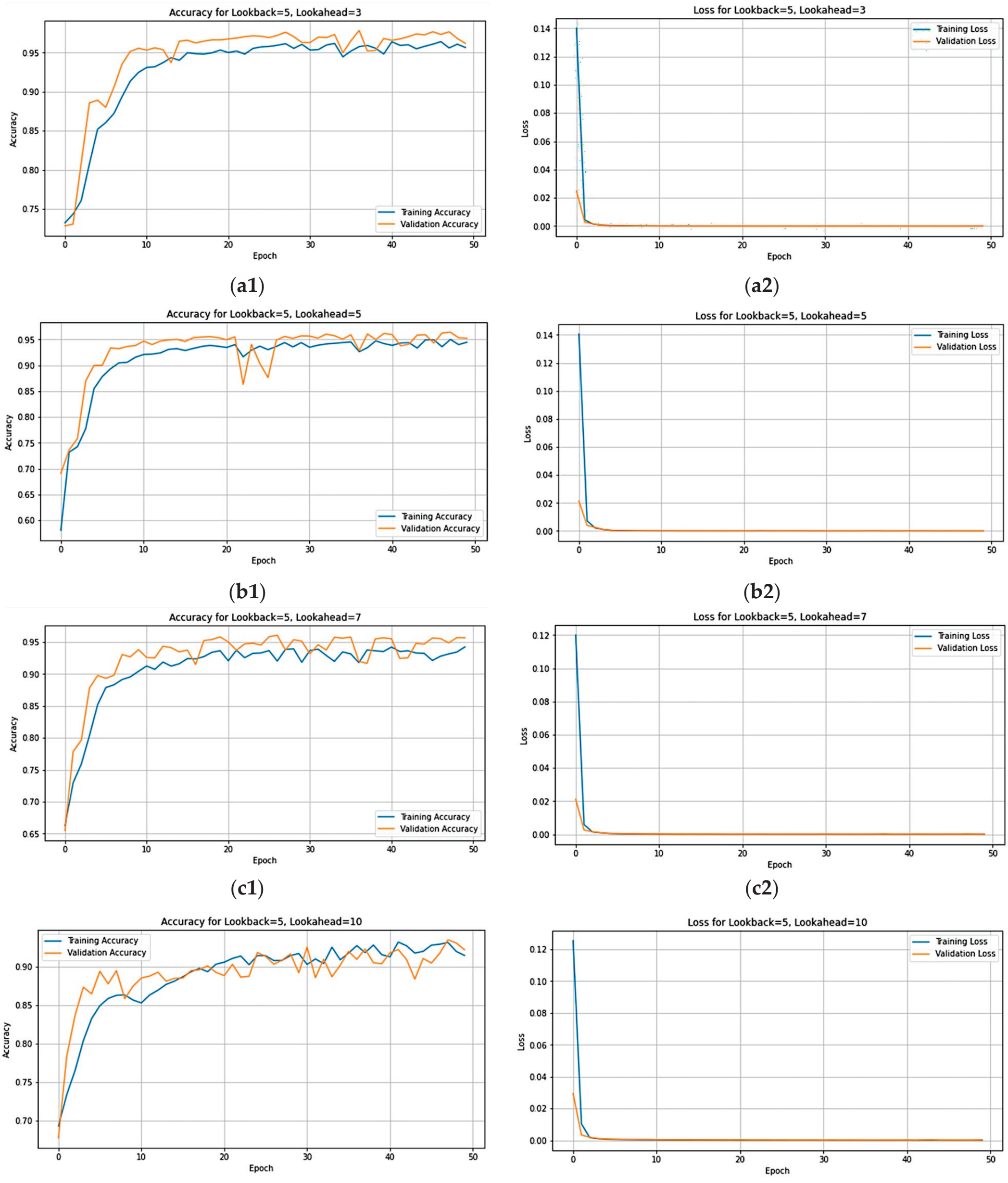


Figure 15. Sensitivity analysis on different lookback and lookahead values.

Note that it was impossible for us to capture weather data and merge them with AIS data based on time and location. Thus, we plotted our model's error (mile offset between the predicted and actual point) versus the month of prediction to show the weather's impact on our model's performance. As shown in Figure 17, there is no pattern in different months.



(**a1**) (**a2**)

(**b1**) (**b2**)

(**c1**) (**c2**)

(**d1**) (**d2**)

Figure 16. Accuracy (**a1**–**d1**) and loss (**a2**–**d2**) plots for a lookback value of 5 with different lookahead values {3, 5, 7, 10}. The subfigures (**a1**–**d1**) depict the accuracy plots, while (**a2**–**d2**) represent the corresponding loss plots under varying lookahead conditions.

Obviously, different vessel types show different maneuvers in channels and narrow waterways. Especially in channels that have width or depth limitations and put restrictions on vessels based on their type and dimensions, as shown in Figure 18, tug tows have the lowest error since their operations are hardly impacted by such restrictions. The average offset values of tankers and cargos are also low, which is a good sign, because, in practice, we care more about tankers and cargo's locations in channels.

In Figure 19, the traffic density of the Galveston port with id segments of 18, 23, and 73 and the Houston ship channel with id segments of 43, 66, and 94 is illustrated.

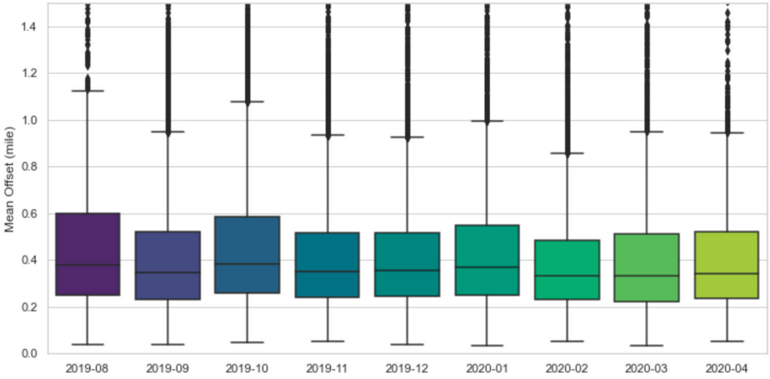


Figure 17. The boxplot of test months vs. mean offset.

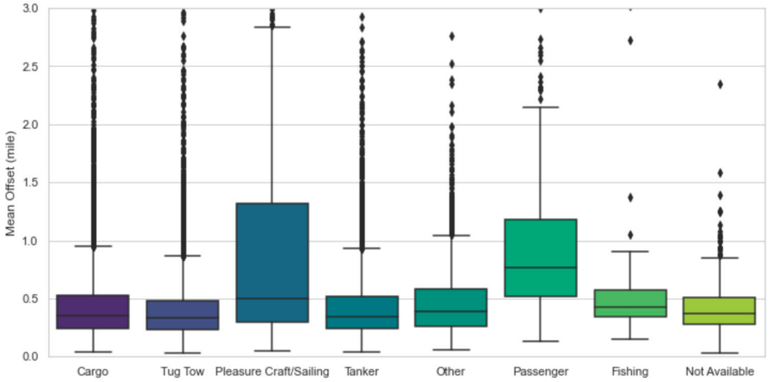


Figure 18. The boxplot of vessel type vs. mean offset.

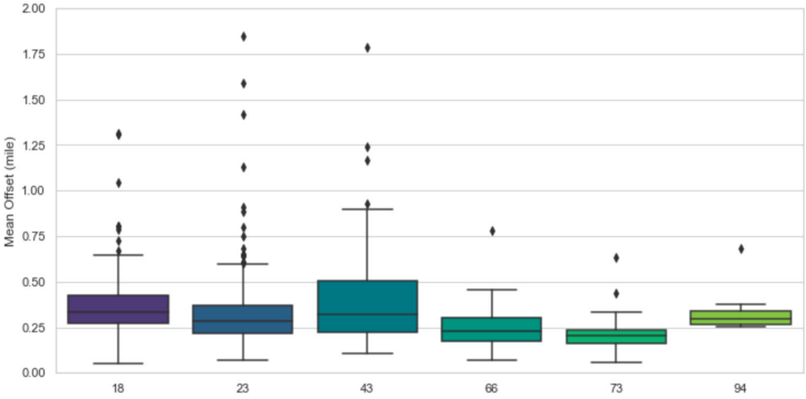


Figure 19. The boxplot of the segments at the Galveston port and the Houston ship channel area vs. mean offset.

5. Discussion

Looking forward, the research suggests two important future directions. Firstly, an examination of the system's feasibility and its applicability to LSTM networks is essential. Evaluating the scalability and adaptability of the proposed framework under different operational conditions, such as high-traffic scenarios and diverse waterway characteristics, will provide valuable insights. Additionally, exploring the integration of advanced machine learning techniques beyond LSTM could enhance predictive capabilities. Secondly, the research lays the groundwork for autonomous vessel systems. The system's ability to handle false or missing AIS records through dead reckoning and machine learning techniques sets the stage for developing intelligent, self-adjusting vessels. This evolution aligns with industry trends toward autonomy, particularly in navigating narrow waterways and channels, offering a potential solution to the challenges of pilot training and scheduling. Exploring these future directions will not only contribute to academic discourse but also offer practical insights for the ongoing digital transformation of maritime operations.

6. Conclusions

In the realm of maritime tracking systems, current platforms like Marine Traffic, VesselFinder, and AccessAIS serve as vital tools for monitoring vessel movements, providing real-time insights into vessel types, names, and directions. However, these systems face notable limitations, including the absence of detailed historical data and the cumbersome process of downloading raw information, hindering users seeking comprehensive insights. Additionally, these platforms lack features such as traffic information, mooring locations, and advanced predictive capabilities for estimated time of arrival (ETA). Our proposed model addresses these gaps by introducing innovative features and addressing existing drawbacks. Through real-time transformation, the model processes and analyzes data in smaller, more manageable chunks, optimizing time efficiency. Using a 5 min sample rate reduces unnecessary data volume and structures data for storage efficiency by converting repetitive columns into dimension tables. The model also introduces new features, including trip number, direction, travel miles, travel time, segments, and origin/destination, enhancing the depth of analysis. Furthermore, the model addresses issues of noise and missing records, ensuring a more reliable and comprehensive maritime tracking solution.

Our research introduces an inclusive and effective framework for the processing of maritime data, tracking vessel traffic, and predicting their next locations. Through the implementation of an Extract, Transform, and Load (ETL) pipeline, we have successfully processed raw AIS data, augmenting it with additional attributes like vessel direction and trip number. The utilization of an inner product context for defining vessel direction and a time-lagged window for trip estimation has proven highly effective, enabling the accurate processing of millions of data entries within seconds. The newly designed system, which eliminates the necessity for raw AIS data, exhibits the capacity to handle extensive information, rendering it a valuable tool for maritime applications. Processed data are stored in a database, and our user interface offers real-time visualizations of vessel traffic, providing port authorities with effective monitoring capabilities. In the prediction phase, we explored two distinct approaches: the conventional dead reckoning method and a deep learning technique using a decoder-encoder model. Our findings revealed that the second scenario of the dead reckoning method, considering the angle between the pre-record and the new record, resulted in lower prediction errors compared to the course of the ground-based approach. Furthermore, the Seq2Seq model demonstrated promising outcomes in predicting the trajectories of vessels based on historical data. Our algorithm is versatile and applicable to diverse maritime scenarios, offering valuable insights and facilitating improved decision-making processes. For this study, we applied the framework to the Gulf Intracoastal Waterway (GIWW) in the Texas region, simulating AIS APIs. The results underscore the system's efficiency in processing large amounts of data and achieving precise vessel location predictions.

Author Contributions: Conceptualization, A.S. and M.J.K.; methodology, A.S. and M.J.K.; software, M.J.K. and A.S.; validation, H.A.; formal analysis, M.J.K. and H.A.; investigation, A.S.; resources, M.J.K.; data curation, M.J.K. and H.A.; writing—original draft preparation, A.S.; writing—review and editing, M.J.K. and M.H.; visualization, A.S. and M.J.K.; supervision, M.H.; project administration, M.H.; funding acquisition, M.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw AIS data are extracted from https://marinecadastre.gov/ais/ (accessed on 1 January 2024) and the dashboard for visualization and results are available at https://aisapp.lamarhamidi.com/ (accessed on 15 December 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Forti, N.; Millefiori, L.M.; Braca, P.; Willett, P. Prediction Oof Vessel Trajectories From AIS Data Via Sequence-To-Sequence Recurrent Neural Networks. In Proceedings of the ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 8936–8940.
2. Zhang, Y.; Li, W. Dynamic Maritime Traffic Pattern Recognition with Online Cleaning, Compression, Partition, and Clustering of AIS Data. *Sensors* **2022**, *22*, 6307. [CrossRef] [PubMed]
3. Brooks, P.; Harati-Mokhtari, A.; Wall, A.; Wang, J. Automatic Identification System (AIS): Data Reliability and Human Error Implications. *J. Navig.* **2007**, *60*, 373–389. [CrossRef]
4. Herodotou, H.; Aslam, S.; Holm, H.; Theodossiou, S. Big Maritime Data Management. In *Maritime Informatics*; Lind, M., Michaelides, M., Ward, R., Watson, R.T., Eds.; Springer International Publishing: Cham, Switzerland, 2021; pp. 313–334. ISBN 978-3-030-50892-0.
5. Zhao, L.; Shi, G. A Method for Simplifying Ship Trajectory Based on Improved Douglas–Peucker Algorithm. *Ocean Eng.* **2018**, *166*, 37–46. [CrossRef]
6. Available online: https://www.vesselfinder.com/ (accessed on 1 August 2022).
7. Available online: https://noaa.maps.arcgis.com/apps/mapviewer/index.html?webmap=1d2ed214322c4130a034c691a9c462b0 (accessed on 1 April 2022).
8. Available online: https://marinecadastre.gov/accessais/ (accessed on 8 August 2018).
9. Li, H.; Liu, J.; Liu, R.W.; Xiong, N.; Wu, K.; Kim, T. A Dimensionality Reduction-Based Multi-Step Clustering Method for Robust Vessel Trajectory Analysis. *Sensors* **2017**, *17*, 1792. [CrossRef]
10. Zhang, S.; Shi, G.; Liu, Z.; Zhao, Z.; Wu, Z. Data-Driven Based Automatic Maritime Routing from Massive AIS Trajectories in the Face of Disparity. *Ocean Eng.* **2018**, *155*, 240–250. [CrossRef]
11. Ren, F.; Han, Y.; Wang, S.; Jiang, H. A Novel High-Dimensional Trajectories Construction Network Based on Multi-Clustering Algorithm. *EURASIP J Wirel. Commun. Netw.* **2022**, *2022*, 18. [CrossRef]
12. Eljabu, L.; Etemad, M.; Matwin, S. Spatial Clustering Method of Historical AIS Data for Maritime Traffic Routes Extraction. In Proceedings of the 2022 IEEE International Conference on Big Data (Big Data), Osaka, Japan, 17–20 December 2022; pp. 893–902.
13. Kang, M.J.; Zohoori, S.; Hamidi, M.; Wu, X. Study of Narrow Waterways Congestion Based on Automatic Identification System (AIS) Data: A Case Study of Houston Ship Channel. *J. Ocean Eng. Sci.* **2022**, *7*, 578–595. [CrossRef]
14. Kabir, M.; Kang, M.J.; Wu, X.; Hamidi, M. Study on U-Turn Behavior of Vessels in Narrow Waterways Based on AIS Data. *Ocean Eng.* **2022**, *246*, 110608. [CrossRef]
15. Zohoori, S.; Jafari Kang, M.; Hamidi, M.; Craig, B. A Vectorized Algorithm for Waterway Traffic Analysis Using AIS Data. *J. Ocean Technol.* **2022**, *16*. Available online: https://www.academia.edu/69696254/A_Vectorized_Algorithm_for_Waterway_Traffic_Analysis_using_AIS_Data?ri_id=1137263 (accessed on 15 December 2023).
16. Wu, X.; Mehta, A.L.; Zaloom, V.A.; Craig, B.N. Analysis of Waterway Transportation in Southeast Texas Waterway Based on AIS Data. *Ocean Eng.* **2016**, *121*, 196–209. [CrossRef]
17. Wu, X.; Rahman, A.; Zaloom, V.A. Study of Travel Behavior of Vessels in Narrow Waterways Using AIS Data—A Case Study in Sabine-Neches Waterways. *Ocean Eng.* **2018**, *147*, 399–413. [CrossRef]
18. Sepideh, Z.; Uttara, R.; Maryam, H.; Xing, W. Quantifying Wide-Body Vessel Navigation Delay in Narrow Waterways: A Case Study at the Houston Ship Channel. *J. Waterw. Port Coast Ocean Eng.* **2022**, *148*, 04022010. [CrossRef]
19. Evmides, N.; Odysseos, L.; Michaelides, M.P.; Herodotou, H. An Intelligent Framework for Vessel Traffic Monitoring Using AIS Data. In Proceedings of the 2022 23rd IEEE International Conference on Mobile Data Management (MDM), Paphos, Cyprus, 6–9 June 2022; pp. 413–418.

20. Chi, H.; Pedrielli, G.; Kister, T.; Ng, S.H.; Bressan, S. An AIS-Based Framework for Real Time Monitoring of Vessels Efficiency. In Proceedings of the 2015 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Singapore, 6–9 December 2015; pp. 1218–1222.
21. Kontopoulos, I.; Chatzikokolakis, K.; Zissis, D.; Tserpes, K.; Spiliopoulos, G. Real-Time Maritime Anomaly Detection: Detecting Intentional AIS Switch-Off. *Int. J. Big Data Intell.* **2020**, *7*, 85–96. [CrossRef]
22. Gao, M.; Shi, G.-Y. Ship Spatiotemporal Key Feature Point Online Extraction Based on AIS Multi-Sensor Data Using an Improved Sliding Window Algorithm. *Sensors* **2019**, *19*, 2706. [CrossRef]
23. Sedaghat, A.; Kang, M.J.; Hamidi, M. A Heuristic ETL Process to Dynamically Separate and Compress AIS Data. In Proceedings of the 2023 Systems and Information Engineering Design Symposium (SIEDS), Charlottesville, VA, USA, 27–28 April 2023; pp. 159–164.
24. Fuentes, G. Generating Bunkering Statistics from AIS Data: A Machine Learning Approach. *Transp. Res. E Logist. Transp. Rev.* **2021**, *155*, 102495. [CrossRef]
25. Neri, P. Time-Domain Simulator for Short-Term Ship Manoeuvring Prediction: Development and Applications. *Ships Offshore Struct.* **2019**, *14*, 249–264. [CrossRef]
26. Qiang, H.; Jin, S.; Feng, X.; Xue, D.; Zhang, L. Model Predictive Control of a Shipborne Hydraulic Parallel Stabilized Platform Based on Ship Motion Prediction. *IEEE Access* **2020**, *8*, 181880–181892. [CrossRef]
27. Li, H.; Jiao, H.; Yang, Z. AIS Data-Driven Ship Trajectory Prediction Modelling and Analysis Based on Machine Learning and Deep Learning Methods. *Transp. Res. E Logist. Transp. Rev.* **2023**, *175*, 103152. [CrossRef]
28. Ma, H.; Zuo, Y.; Li, T. Vessel Navigation Behavior Analysis and Multiple-Trajectory Prediction Model Based on AIS Data. *J. Adv. Transp.* **2022**, *2022*, 6622862. [CrossRef]
29. Alesheikh, A.A.; Alizadeh, D.; Sharif, M. Vessel Trajectory Prediction Using Historical Automatic Identification System Data. *J. Navig.* **2021**, *74*, 156–174. [CrossRef]
30. Mehri, S.; Alesheikh, A.A.; Basiri, A. A Contextual Hybrid Model for Vessel Movement Prediction. *IEEE Access* **2021**, *9*, 45600–45613. [CrossRef]
31. Hammedi, W.; Brik, B.; Senouci, S.M. Toward Optimal MEC-Based Collision Avoidance System for Cooperative Inland Vessels: A Federated Deep Learning Approach. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 2525–2537. [CrossRef]
32. Zhao, L.; Zuo, Y.; Li, T.; Chen, C.L.P. Application of an Encoder-Decoder Model with Attention Mechanism for Trajectory Prediction Based on AIS Data: Case Studies from the Yangtze River of China and the Eastern Coast of the U.S. *J. Mar. Sci. Eng.* **2023**, *11*, 1530. [CrossRef]
33. Abada, R.; Abubakar, A.M.; Bilal, M.T. An Overview on Deep Leaning Application of Big Data. *Mesopotamian J. Big Data* **2022**, 31–35. [CrossRef]
34. Available online: https://www.movable-type.co.uk/scripts/latlong.html (accessed on 1 March 2023).

Journal of
*Marine Science
and Engineering*

Article

Automatic Identification System-Based Prediction of Tanker and Cargo Estimated Time of Arrival in Narrow Waterways

Homayoon Arbabkhah, Atefe Sedaghat, Masood Jafari Kang and Maryam Hamidi *

Department of Industrial and Systems Engineering, Lamar University, Beaumont, TX 77710, USA; harbabkhah@lamar.edu (H.A.); asedaghat@lamar.edu (A.S.); mjafarikang@lamar.edu (M.J.K.)
* Correspondence: mhamidi@lamar.edu

Abstract: In maritime logistics, accurately predicting the Estimated Time of Arrival (ETA) of vessels is pivotal for optimizing port operations and the global supply chain. This study proposes a machine learning method for predicting ETA, drawing on historical Automatic Identification System (AIS) data spanning 2018 to 2020. The proposed framework includes a preprocessing module for extracting, transforming, and applying feature engineering to raw AIS data, alongside a modeling module that employs an XGBoost model to accurately estimate vessel travel times. The framework's efficacy was validated using AIS data from the Port of Houston, and the results indicate that the model can estimate travel times with a Mean Absolute Percentage Error (MAPE) of just 5%. Moreover, the model retains consistent accuracy in a simplified form, pointing towards the potential for reduced complexity and increased generalizability in maritime ETA predictions.

Keywords: travel time; ETA prediction; AIS data; XGBoost

Citation: Arbabkhah, H.; Sedaghat, A.; Jafari Kang, M.; Hamidi, M. Automatic Identification System-Based Prediction of Tanker and Cargo Estimated Time of Arrival in Narrow Waterways. *J. Mar. Sci. Eng.* **2024**, *12*, 215. https://doi.org/10.3390/jmse12020215

Academic Editors: Ryan Wen Liu, Dongfang Ma and Xinqiang Chen

Received: 16 December 2023
Revised: 15 January 2024
Accepted: 22 January 2024
Published: 25 January 2024

1. Introduction

Shipping plays a pivotal role in the intricate web of global trade, serving as the lifeblood of international commerce. It facilitates the movement of goods across vast oceans as well as connecting markets and economies in a seamless exchange of products and resources. With over 80% of the world's trade volume being carried by ships, the maritime industry is a cornerstone of the global economy [1]. Consequently, the efficiency of the shipping system, particularly in ports, yields a profound influence on global trade and supply chains. Port congestion, which is attributed to 93.6% of delays, primarily stems from congestion issues, and it underscores the critical need for effective port operational planning [2]. For ports to function smoothly, ships must adhere to arrival schedules. Research indicates that shipping delays significantly impact port operations. Consequently, the estimated time of arrival (ETA) assumes a pivotal role in port operational planning and management. ETA represents the anticipated date and time of a shipment's arrival at a specified destination. An uncertain ETA hampers the ability of ports to formulate efficient logistics plans, emphasizing the crucial role of accurate arrival time predictions.

In maritime terms, a "narrow waterway" designates a constricted waterway characterized by limited breadth and depth, posing navigational challenges. Contrasted with the ample maneuvering space in open seas or larger water bodies, these confined areas demand precise navigation. Skillful handling and often the aid of local pilots are essential for navigating these constrained spaces [3]. Challenges include a restricted turning radius where big vessels have to sail extra miles to make a U-turn [4], possible strong currents or tides, and close quarters with other vessels or the boundaries of the waterway itself [5]. Deep-draft vessel navigation in main waterways is managed by channel pilots, tasked with ensuring safe, orderly bidirectional traffic. The process begins as arriving deep-draft vessels request pilotage. A pilot is assigned to a vessel when two key criteria are met: the availability of the intended dock and the accessibility of the channel. Channel unavailability can arise from various factors, such as fog or the transit of large vessels. Since halting in the main channel's centerline is prohibited, vessels await at sea buoys or designated anchorages until these conditions are fulfilled [3].

The Estimated Time of Arrival (ETA) in narrow waterways is crucial and significantly impacts maritime operations. When a vessel completes its ocean journey and enters shallow waterways, terminal operations need to be meticulously managed. This involves guiding vessels through the waterway to their destination terminal for loading/unloading. Predicting the ETA to the destination greatly affects terminal efficiency. The ETA is vital for various operational decisions, such as scheduling at the terminal, assigning pilots to vessels, controlling traffic at waterway–highway intersections such as bascule bridges, and decision-making regarding dredging and temporary channel closures to manage maritime traffic in channels and narrow waterways. This study focused on the critical role of travel time estimation in narrow waterways for its significant implications for maritime transportation.

While the estimation of travel time on highways and roadways has been extensively explored in the literature, there is a noticeable gap in research concerning the Estimated Time of Arrival (ETA) for shipping systems. Traditional traffic estimation methods typically rely on aggregated data such as traffic flow, average speed, and congestion distribution. Additionally, statistical modeling of travel time has been applied in city planning. More recently, machine learning techniques have gained prominence in studying vehicle travel times within urban routes. These methods encompass established approaches like random forest and decision trees, as well as sophisticated deep learning architectures. In contemporary city planning, diverse deep learning structures, including Recurrent Neural Networks (RNN) with Long-Short Term Memory (LSTM) [6] and even Graph Neural Networks (GNN) [7], are employed for travel time estimation. However, the majority of studies focus on travel time estimation within city routes. This research uniquely employs a machine learning approach to estimating the time of arrival for ship movement in waterways, extending from a specific origin to their destined locations.

To tackle the challenge of estimating travel time, this study introduces a machine learning framework that leverages AIS data for predicting arrival times between two points along sea routes. This framework is designed to not only process AIS data but also incorporate additional spatial information about vessel trajectories, such as path weight and segment features, as supplementary data inputs. The proposed framework involves the initial preprocessing of AIS data, followed by inputting the prepared data into an XGBoost (v 2.0.1) model and determining the optimal parameters for the model. Subsequently, the trained model undergoes testing using historical AIS data to assess its performance. The experimentation in this study is conducted on historical AIS data of Houston Ports in the United States. The results demonstrate the efficacy of the model, evaluated through five different metrics, namely, mean absolute error (MAE), root mean square error (RMSE), R squared, mean absolute percentage error (MAPE), and root mean square logarithmic error (RMSLE).

The subsequent sections of this paper are structured as follows: Section 2 provides a review of related work in the field of estimating time of arrival. Section 3 outlines the AIS data, defines the problem, and extracts important information from the dataset. Section 4 details the proposed model framework and its constituent modules. Finally, Section 5 presents our conclusions and outlines potential future research.

2. Related Works

Considerable research has been conducted on the precise and timely prediction of Estimated Time of Arrival (ETA) to enhance decision-making across diverse application domains. ETA prediction plays a vital role in air traffic control, impacting arrival sequencing, scheduling, methods for assigning airport gates, and flight arrival time [8–13]. In the realm of road transportation, studies have been undertaken to forecast vehicle management [14], as well as the ETAs of buses [15,16], emergency ambulance services [17,18], and cargo. The current literature lacks studies on estimating vessel arrival times in ports using historical tracking data. Vessel ETA from AIS messages are often unreliable due to manual input. Despite the requirement for a 72 h advance notice, accurate predictions are challenging, as authorities must verify notifications, the lead time is sometimes too long, and there is a need for insight into approaching vessel volumes for optimal port operations. Therefore, investigating the theoretical foundations and practical applications of machine learning in a business context, research has delved into the performance of various algorithms, including Random Forests, Neural Network architectures, and Linear Regression, on different datasets related to maritime transportation [19]. These machine learning algorithms are applied to provide a qualitative estimation of vessel ETA, aiming to alleviate the consequences of inconsistent arrivals at ports [20].

2.1. Path Finding/Other Methods

In recent years, the Dijkstra algorithm [21], which is often used to find the shortest path in problems, and its derivative, the A* algorithm [22], have been used in studies to calculate routes considering weather conditions. Alessandrini et al. [23] discussed a novel data-driven method for estimating vessel arrival times in port areas, leveraging the abundant data available from ship reporting systems like (AIS) and Long-Range Identification and Tracking (LRIT). The approach utilizes historical maritime traffic data from these systems, focusing on a specific area of interest. It employs an optimized data-driven path-finding algorithm to process these data. Chen et al. employed maritime image sequences for predicting the trajectories of ships [24]. Park et al. [2] presented an ETA prediction system based on a path-finding algorithm. With increasing container volumes and vessel sizes, efficient port operations are crucial. The proposed methodology utilizes AIS data-driven techniques, including data mining and reinforcement learning, to identify possible vessel trajectories. Additionally, the Markov Chain property and Bayesian Sampling are introduced to estimate the vessel's speed over ground (SOG). Wu et al. [25] introduced an AIS-data-based model for the precise estimation and distribution of vessels' travel time and trip numbers in narrow channels, crucial for efficient traffic control. The model involves identifying a vessel's destination dock, arrival/departure times, and estimating travel time between specific points. Additionally, the model addresses the separation of a vessel's trips, contributing to a comprehensive understanding of its journey. Applied to the Houston Ship Channel, the model reveals that travel times are lognormally distributed and influenced by vessel characteristics. Interestingly, trip numbers and travel times exhibit a correlation, providing valuable insights for channel management.

Wu et al. [26,27] examined the transit patterns of tankers and cargos through the Sabine–Neches Waterway (SNWW). Unlike highway traffic, vessel travel time at the SNWW and the entrance of Galveston Ship Channel showed independence from traffic density [28]. Kang et al. [29] studied 15 legs in the Singapore Strait, observing a correlation between vessel travel time and traffic density. The disparity in findings may be attributed to differing vessel volumes, with the Singapore Strait experiencing higher traffic than SNWW and Galveston Ship Channel.

2.2. Machine Learning

Artificial intelligence is about building systems capable of understanding and solving real-world problems by acquiring knowledge from experience. Machine learning is a subfield of artificial intelligence that refers to the ability of extracting insights from data. Machine learning models include deep learning, which is an advanced class of machine learning, inspired by the human brain function and based on Artificial Neural Networks (ANN) and representation learning. Deep learning has networks with the capability to learn complicated concepts even with unstructured data.

2.2.1. Road Application

Several studies have used machine learning (ML) approaches to predict travel times based on GPS traces from vehicles [30,31] or the so-called live Automatic Vehicle Locations (AVL) data [32,33]. Larsen et al. [34] employed an NN to predict the travel times of buses using open real-time data derived from the Sao Paulo City bus fleet location, real-time traffic data, and traffic forecast from Google Maps. Alam et al. [35] used a Recurrent NN (RNN) architecture to predict the ETA irregularities by exploring live AVL data from buses, provided by the Toronto Transit Commission, along with schedules retrieved from GTFS and weather data. Chondrodima et al. [36] addressed the challenge of predicting public transport ETA using General Transit Feed Specification (GTFS) data. The proposed approach employs a novel combination of Particle Swarm Optimization (PSO) and Radial Basis Function (RBF) neural networks, incorporating a modified PSO-NSFM algorithm for training. A unique pre-processing pipeline, CR-GTFS, is introduced for cleansing and reconstructing GTFS data.

2.2.2. Waterway Application

ANNs have been applied to solve some issues in shipping, for instance, container flow forecasting [37], container dwell time [38], navigational behavior prediction [39], and detecting navigable area for autonomous navigation [40]. To minimize the unpredictability of ship arrivals, recent studies have turned to data mining methods for arrival prediction. In [41], for instance, the author utilized a Neural Network (NN) model to forecast the time intervals between ship arrivals. Subsequently, the predicted interval times are integrated into a model that optimizes the allocation of human resources, leading to successful outcomes that offer valuable support to planners. Pani et al. [42,43] utilized both k-means and Ward's method to cluster daily records from the Cagliari International Container Terminal. This clustering aims to categorize arrival delays into three levels. Subsequently, Classification and Regression Trees (CART), Random Forest (RF), and Naive Bayes (NB) were employed to estimate the delay level. Notably, RF demonstrated superior predictive performance, boasting a relative absolute error of 29% when compared to CART and NB. In a different approach. Pallotta et al. [44] introduced the unsupervised method Traffic Route Extraction and Anomaly Detection (TREAD) to learn a statistical model from AIS data for maritime traffic at the Cagliari International Container Terminal. Pani et al. [20] employed Logistic Regression (LR), CART, and RF to estimate arrival ship deviations at both the Cagliari International Container Terminal and the PSA-Antwerp terminal. Additionally, Parolas et al. [45] applied Support Vector Machines (SVM) and NN to predict ETA for container ships arriving at the Port of Rotterdam. The results showed that both SVM and NN outperformed ETA predictions based on ship agent estimations, with SVM models surpassing NN in Mean Absolute Error. Collectively, these studies hold significance and provide valuable references for the application of data mining in predicting ship arrival times at specific ports. Noman et al. [46] investigated the use of Gradient Boosting Decision Trees (GBDT), Multi-Layer Perceptron Neural Networks (MLP), and Gated Recurrent Unit Neural Networks (GRU) for predicting vessel ETA in inland waterways. It used historical AIS data for training and compared the accuracy of these methods. The GRU algorithm outperformed the others. Yu et al. [47] focused on ship arrival prediction and its impact on the daily operations of Gangji (Yining) Container Terminal (GYCT) in China. Utilizing data mining methods such as Back-Propagation network (BP), CART, and RF, the study aims to enhance the accuracy of predicting ship arrival delays or advances. The results indicate that RF outperforms BP and CART, with ETA month and ship length identified as crucial factors influencing arrivals at GYCT.

3. Problem Definition

AIS Data

The Automatic Identification System (AIS), first introduced in 1990, is used in maritime traffic to record the historical trajectory of vessels. Its main objectives are to enhance maritime safety, improve situational awareness, and facilitate efficient maritime operations. In 2004, the International Maritime Organization (IMO) required all ships exceeding 300 gross tonnages to record and broadcast AIS data [48].

The frequency of AIS message transmission from vessels varies from 2 to 30 s, depending on their speed. The range at which these messages can be received is influenced by various factors such as signal propagation conditions, sea state, and the height and strength of the transmitting and receiving antennas. Reception ranges can vary from 20 nautical miles to up to 350 nautical miles under optimal conditions. Typically, an AIS receiver network is expected to achieve an average reception radius of around 40 nautical miles.

AIS operates by acquiring position and movement data from the vessel's GPS system or an internal sensor within the AIS unit. These data, along with other programmable information from the AIS unit (such as Maritime Mobile Service Identity (MMSI) number, vessel name, destination, and cargo type), are periodically transmitted. The system not only sends out information but also receives data from other vessels' AIS systems.

Each AIS message contains both static and dynamic information. Static information includes vessel attributes, while dynamic information covers the spatial-temporal data of the vessel [3]. The MarineCadastre website provides access to AIS data [49]. Table 1 displays the static and dynamic information contained in AIS data.

Table 1. Static and dynamic information in AIS data.

Static Information	Dynamic Information
MMSI Number	Ship's Position with Accuracy indication
IMO Number	Position timestamp (in UTC)
Name and Call Sign	Course Over Ground (COG)
Length and Beam	
Type of Ship	
Location of Position	

The first challenge in ETA analysis is to transform raw AIS messages into useful data required in ETA prediction. As shown in Table 1, AIS data do not include trip information such as trip number, trip origin, destination, start time, and end time. To get such information, we apply a trip separation algorithm to raw AIS data. The algorithm works based on comparing each vessel's AIS message and the previous one. It uses a hash table to capture the last AIS record of each vessel. Then, it calculates the time and length difference between the current and the last records. The calculated time difference and spatial distance help us to filter redundancies and noises. As the next step, it assumes a vessel's direction is "Stopped" if its speed is less than two knots (~1.151 miles per hour). Defining one stop to the next one as a vessel trip, the algorithm assigns trip numbers to the processed AIS data. Finally, the processed data are stored in a local database. This paper does not discuss the algorithm but rather uses its output to predict ETA for vessels. For an in-depth exploration of the methodology, we recommend reviewing our previous works [50,51].

In this study, we applied our trip separation method to extract trip data from raw AIS data. To estimate the time of arrival for vessels, this information has to be transformed into the form of a complete trip for each vessel. A complete trip dataset includes longitude and latitude coordinates from the origin to the destination, the time taken for each trip, and a sequence of segments representing the vessel's path. The methodology section provides a comprehensive explanation of the algorithm.

4. Methodology

The model proposed in this study consists of three modules. In the first module, AIS data undergo preprocessing, and new features are incorporated into the dataset. The second module utilizes the preprocessed data to train an XGBoost (v 2.0.1) model, incorporating hyperparameter optimization and defining a validation strategy. The final module is dedicated to applying the trained model to a test dataset and comparing the results with the actual travel time. The three modules are illustrated in Figure 1.

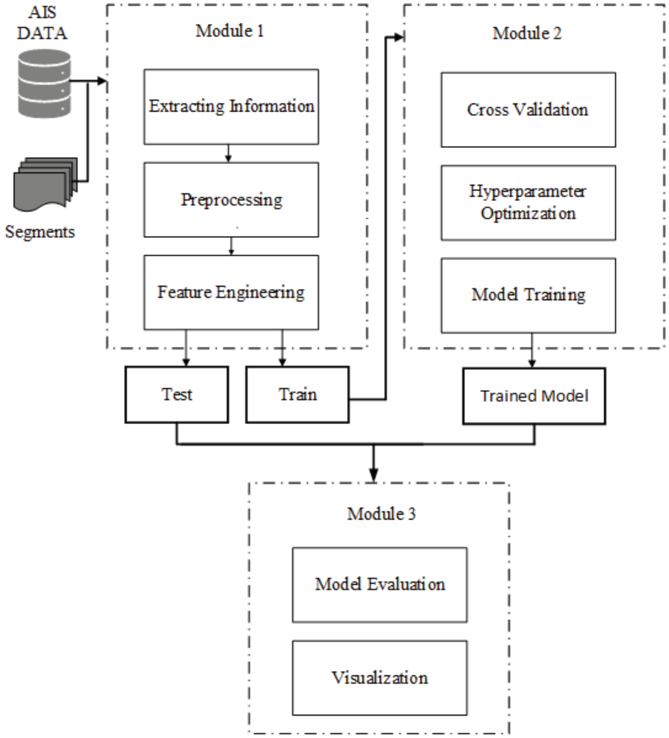


Figure 1. Overview of the proposed framework.

Subsequent sections provide a more detailed explanation of each module.

4.1. Module 1: Preprocessing

The model module cannot directly use raw AIS data; hence, preprocessing is essential before feeding the data into the model. This preprocessing module encompasses various steps, such as addressing missing values, feature engineering, and implementing Principal Component Analysis for dimensionality reduction. The subsequent sections elaborate on each of these steps in greater detail.

4.1.1. Segmentation of Area of Interest

The vessel's journey from its origin to its destination involves traversing a predefined route. To facilitate the tracking of the vessel's path, a network of segments is established using a Geographical Information System (GIS) layer, which maps all waterways within the designated Area of Interest (AoI). For this study, the AoI is the Gulf Intracoastal Waterway (GIWW). To segment the waterway, a function in QGIS (v 3.32.3) named "split line to maximum length" is utilized, dividing the channel's centerline into segments of 2 miles each. As the vessel moves through these segments, the IDs of the segments it passes are captured and stored as features in the database. A schematic representation of this network of segments is depicted in Figure 2.

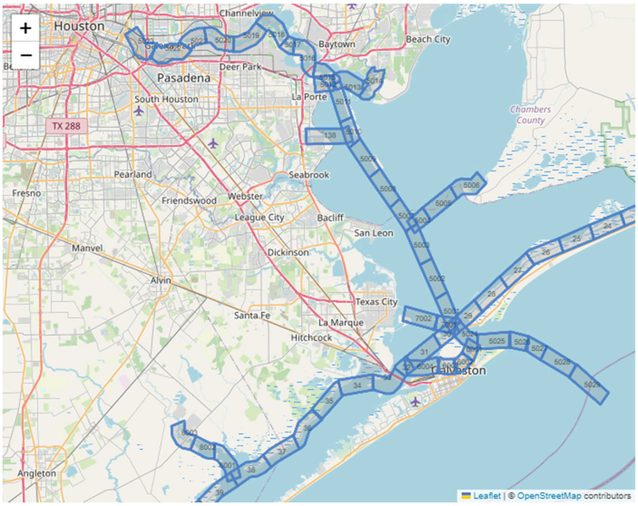


Figure 2. Schematic representation of segment network.

4.1.2. Extracting Features from Raw AIS Data

AIS data encompass details such as a vessel's location, speed, date, and time. In this section, the primary features are initially extracted from the raw data utilizing the Sedaghat [50,51] algorithm. Their approach enables the retrieval of information related to a vessel's route, velocity, direction, and trip number.

In the next step, the extracted information is processed to compute a complete trip for each vessel. A complete trip normally starts from the ocean and ends at a terminal and vice versa. A trip is deemed complete when there is a change in the vessel's trip number. Consequently, the data processed in the previous step are organized by trip number for each vessel, considering the latitude and longitude of the origin and destination. Subsequently, the travel time for the entire trip is computed. A visual representation of a complete trip is depicted in Figure 3.

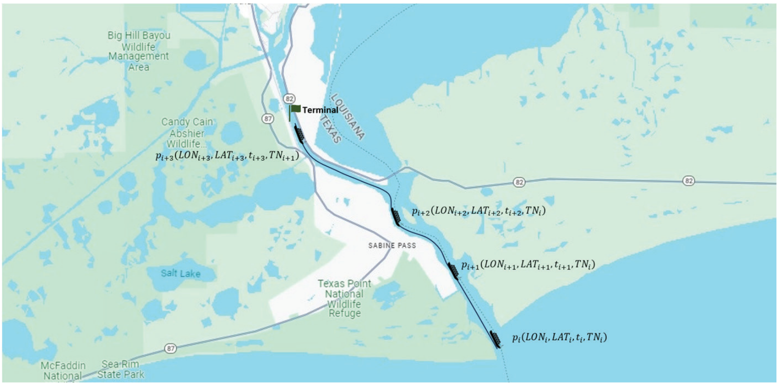


Figure 3. Visual representation of a complete trip.

As illustrated in Figure 3, when there is a change in the vessel's trip number, it signifies the completion of the previous trip. Consequently, the AIS information of the origin and destination, along with the corresponding travel time, can be computed. For example, in Figure 3, a complete trip can be calculated using Equation (1)

$$\mathrm{OD_{trip}} = \mathrm{P_{i+2}}(\mathrm{LON_{i+2}}, \mathrm{LAT_{i+2}}, \mathrm{t_{i+2}}, \mathrm{TN_{i+1}}) - \mathrm{P_i}(\mathrm{LON_i}, \mathrm{LAT_i}, \mathrm{t_i}, \mathrm{TN_i}) \tag{1}$$

where $\mathrm{LON_i}, \mathrm{LAT_i}, \mathrm{t_i},$ and $\mathrm{TN_i}$ represent the longitude, latitude, time, and trip number at location i, respectively.

4.1.3. Principal Component Analysis

Principal Component Analysis (PCA) is a powerful statistical technique used to reduce the dimensionality of data while preserving as much information as possible. This is achieved by identifying a set of orthogonal (uncorrelated) directions called principal components (PCs) that capture the greatest variance in the data. By projecting the data onto these PCs, we can obtain a lower-dimensional representation that is often sufficient for many tasks, including visualization, data analysis, and machine learning [52]. Principal components can be calculated by using singular value decomposition (SVD) of the dataset matrix. The SVD of the matrix X can be expressed as Equation (2)

$$\mathrm{X} = \mathrm{U\Sigma V^T} \tag{2}$$

where:

- U is an N × N orthogonal matrix containing the left singular vectors.
- Σ is an N × D diagonal matrix containing the singular values on the diagonal.
- V is a D × D orthogonal matrix containing the right singular vectors.

To compute the principal components (PCs) of the X dataset, the singular value decomposition of the dataset must be calculated initially. Subsequently, by selecting the top K right singular vectors $(\mathrm{V_K})$ and their corresponding singular values (Σ_K), where K represents the desired dimensionality of the reduced data, the projected X in the new dimension can be calculated using Equation (3):

$$\mathrm{X_{new}} = \mathrm{XV_K}\Sigma_\mathrm{K}^{(-1/2)} \tag{3}$$

where $\Sigma_\mathrm{K}^{(-1/2)}$ is a diagonal matrix containing the reciprocal square root of the top K singular values.

PCA method also can be used as a method to reduce noise in the dataset by mapping data to a new space without reducing the dimensionality of the dataset. In this study PCA algorithm is applied to the latitude and longitude of the origin and destination feature to map the data to a new space with lower noise.

4.1.4. Feature Engineering

In machine learning, feature engineering involves extracting and manipulating data to transform them into a format suitable for training and improving the performance of machine learning models. This process allows the model to better understand the underlying patterns and relationships within the data, leading to more accurate and generalizable predictions. In this study, season, hour, minutes, and day of the week of trip beginning are extracted from raw AIS data and added to the dataset to improve model accuracy. Also, two new features are introduced including vessel segments path and path weight. These new features are explained in more detail in the following sections.

Vessel Segments Path

In the AIS data provided for ETA prediction, the trajectory of vessels that pass through different segments is introduced. In this study, all segments for every vessel are incorporated into the dataset as a one-hot-encoded vector appended to the original dataset. While this approach introduces additional sparsity to the dataset, it enriches the dataset with more information about vessel paths from origin to destination, thereby aiding the model in achieving more accurate travel time estimations. The encoded representation of each segment is defined by Equation (4):

$$\mathrm{e_{segment}} = \begin{cases} 1 \text{ if vessel path through segment} \\ 0 \quad \text{otherwise} \end{cases} \tag{4}$$

Path Weight

The trajectory each vessel follows plays a crucial role in determining its arrival time, with vessels navigating through busier routes expected to experience longer travel durations. Consequently, it is reasonable to infer that each path taken during shipment holds distinct weights that significantly influence arrival times. In this research, we propose a new feature termed “path weight”, which serves as an indicator of the congestion level along a vessel’s route. Given that our dataset represents vessel paths through discrete segments, we define the segment weight based on the frequency of vessel passages through these segments. As the number of vessels traversing a segment increases, the corresponding segment weight is proportionally amplified. The mathematical representation of segment weight is denoted as Equation (5):

$$\mathrm{w_{segment}} = \frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}} \mathrm{e_{segment}} \tag{5}$$

where N is the total number of samples, and $\mathrm{e_{segment}}$ represents one hot encode of the segment, as defined by Equation (4).

Path weight is the total sum of segments if the vessel path is through the segment and is defined by Equation (6):

$$\mathrm{W_{path,i}} = \sum_{\mathrm{j}=0}^{\mathrm{j}=\mathrm{L}} \mathrm{W_{segments}} * \mathrm{I_{i,j}}' \tag{6}$$

where L is the total number of segments, and $\mathrm{W_{segments}}, \mathrm{I_{I,j}}$ is defined by Equation (7).

$$\begin{gathered} \mathrm{W_{segments}} = [\mathrm{w_0}, \mathrm{w_1}, \mathrm{w_2}, \ldots..\mathrm{w_L}] \\ \mathrm{I_{i,j}} = [\mathrm{e_0}, \mathrm{e_1}, \mathrm{e_2}, \ldots..\mathrm{e_L}] \end{gathered} \tag{7}$$

The concept of path weight essentially represents the significance of a vessel’s trajectory during each trip. As the path weight increases, it signifies a longer travel time for the vessel, indicating that the trajectory traverses through busier segments.

4.2. Module 2: Modeling

Once the raw data have been thoroughly preprocessed, they are fed into the modeling module for further analysis. This module encompasses the training of the model and the identification of optimal hyperparameters, accomplished through a predefined validation strategy. The subsequent paragraphs delve into a more comprehensive explanation of each of these steps.

In this research, the XGBoost (v 2.0.1) algorithm serves as the modeling module for Estimated Time of Arrival (ETA) prediction. XGBoost, a prominent member of the ensemble learning family, is chosen for its potency and widespread application in machine learning. XGBoost combines the strengths of both bagging and boosting techniques, creating a robust and highly accurate model. The algorithm works by iteratively training weak learners, typically decision trees, and boosting their performance by focusing on the mistakes made in previous iterations. It employs a unique regularization term in its objective function, which helps prevent overfitting and enhances generalization [53].

Based on gradient boosting method, XGBoost uses the k additive function to predict the output as expressed by Equation (8).

$$\hat{y}_i = \sum_{k=1}^{K} f_k(X_i), \quad f_k \in F \tag{8}$$

where f_k is an independent Classification and Regression Tree (CART) at each k step that maps input variable X_i to output variable y_i. And F is space of all possible CARTs. The XGBoost algorithm tries to minimize the regularized objective function which is defined in Equation (9):

$$\begin{gathered} \mathrm{Obj} = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \\ \Omega = \gamma T + \tfrac{1}{2}\lambda \|w\|^2 \end{gathered} \tag{9}$$

where T is number of leaves in the tree, w is the score in corresponding leaves, and γ, λ are regularization coefficients. The regularized objective function comprises two parts. Training loss function l and regularization term Ω. The training loss l indicates the difference between the predicted $(\hat{y}_i)$ and actual (y_i) value. The regularization term shows the complexity of models, which helps the model to avoid overfitting to the dataset.

XGBoost incorporates two key techniques: shrinkage and column subsampling. Shrinkage reduces the impact of each tree by scaling down the weights added in each boosting step, which helps in mitigating overfitting. On the other hand, column subsampling enhances the training speed by selecting a random subset of input features for the construction of each tree.

XGBoost exhibits high sensitivity to its hyperparameters, with an increase in the tree size potentially leading to overfitting issues. Consequently, identifying appropriate hyperparameters is crucial for achieving a well-generalized model. This study attained optimal hyperparameters for the model by assessing its performance on a validation dataset. Employing the stratified cross-validation method, the model's hyperparameters were determined based on its performance in the validation dataset. The dataset is divided into 5 folds, ensuring consistency in the frequency of the day-of-the-week feature across all partitions. Consequently, the model's hyperparameters are estimated to minimize a predefined metric (Root Mean Square Logarithmic Error or RMSLE) on the validation dataset.

Metrics

In this study, five distinct metrics were employed to assess the model's performance. These metrics encompass R^2, Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Root Mean Square Logarithmic Error (RMSLE). The mathematical definitions of each of these metrics are provided in Equation (10).

$$\begin{gathered} R^2 = 1 - \frac{\sum_{i=1}^{N}(y(i)-\hat{y}(i))^2}{\sum_{i=1}^{N}(y(i)-\bar{y}(i))^2} \\ \mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{N}\|y(i)-\hat{y}(i)\|^2}{N}} \\ \mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y(i)-\hat{y}(i)| \\ \mathrm{MAPE} = \frac{100}{N}\sum_{i=1}^{N}\left|\frac{y(i)-\hat{y}(i)}{y(i)}\right| \\ \mathrm{RMSLE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\log\frac{1+\hat{y}(i)}{1+y(i)}\right)^2} \end{gathered} \tag{10}$$

where $y(i)$ is the actual target value, and $\hat{y}(i)$ is the predicted target value for all evaluation metrics. Note that RMSLE is used as a metric to estimate the best hyperparameters of the model.

4.3. Module 3: Apply Model

In this module, the trained model with optimized hyperparameters is applied to the test dataset, and its performance is evaluated using the metrics defined in the Metric section. Additionally, this module includes visualizations to illustrate the efficiency of the model.

5. Experimental Study

The proposed methodology to estimate the time of arrival of vessel was evaluated by considering port of Houston in the United States. Port of Houston was ranked the second busiest seaport in the United States by total tonnage in 2013. The Houston Ship Channel (HSC), spanning 52 miles, is home to approximately 200 private and public industrial terminals. Annually, the HSC facilitates the transportation of over 247 million tons of cargo through the passage of more than 8200 vessels and 215,000 barges [2]. This vital waterway is integral to the support of Texas' energy and petrochemical sectors. The detailed use of the AIS data is summarized in Table 2.

Table 2. Summarized AIS data.

Variable	Description
Data	AIS data of 4330 Cargo and Tankers
Historical Period	January 2018 to April 2020
Features	MMSI, Vessel type, Date, Latitude, Longitude

In this study, data from 2018 were used for model training, and data from 2019 and 2020 were used in testing the model performance. These raw data were first fed to the preprocessing module to extract proper features for model training.

5.1. Preprocessing Module

Within this module, feature extraction, identification of missing values, feature engineering, and the division of data into training and testing sets were carried out. Following the approach suggested by Sedaghat et al. [50,51], the initial features were extracted from the raw data in the first step. Table 3 displays the list of features extracted from the raw data.

Table 3. Extracted initial features from raw AIS data.

Features		
MMSI	Vessel Type	Trip Number
Direction	Date	Location
Segment id		

To apply AIS data to the ETA problem, the initial features need to be transformed into the structure of a complete trip, encompassing a specific origin, destination, and corresponding travel time.

The resulting dataset, formatted with origin and destination information, is subsequently input into the dimension reduction and feature engineering section. This process is undertaken to compute the path weight and the path of vessel segments.

The path weight distribution of the training and test data is shown in Figure 4. As Figure 4 shows, the majority of shipment trips passed through lower-traffic routes. However, there are cases where vessels follow busier routes, leading to increased travel time and, consequently, elevating the significance of the corresponding segments in the estimation of travel time.

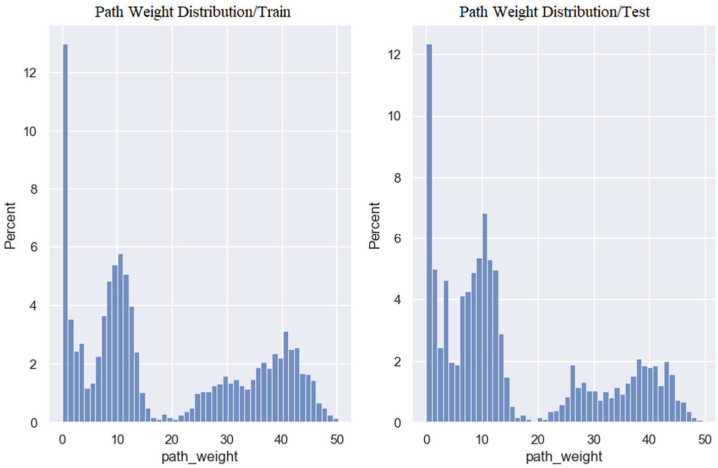


Figure 4. Path weight distribution.

5.2. Modeling Module

5.2.1. Hyperparameter Optimization

As previously mentioned, XGBoost's performance heavily relies on its hyperparameters. Therefore, selecting the right hyperparameters is crucial for enhancing the model's generalization abilities. In this research, the model's optimal hyperparameters were determined based on its validation dataset performance. To identify a suitable validation dataset, a stratified cross-validation approach was employed, dividing the training dataset into five distinct segments. Additionally, to ensure a balanced distribution of trips across all segments, the day of the week was used as the stratification feature. Furthermore, the Root Mean Square Logarithmic Error (RMSLE) was adopted as the metric for optimal parameter identification. Table 4 displays the optimal hyperparameters and their ranges. Our experiments achieved an overall validation score of RMSLE = 0.07.

Table 4. Optimal hyperparameters and their corresponding range.

Hyperparameter	Range	Increment Method	Optimal Value
λ	$(1 \times 10^{-3}, 10)$	Loguniform	0.009
α	$(1 \times 10^{-3},10)$	Loguniform	1.03
Colsample by tree	(0.3, 1.0)	0.1	0.7
Sub sample	(0.4, 1.0)	0.1	0.4
Learning rate	(0.008, 0.02)	0.001	0.014
Max depth	(10, 80)	10	40
Min child weight	(1, 300)	uniform	5

5.2.2. Model Training

This section is divided into two distinct parts. The first part involves applying a tuned model to the training dataset with all predefined features, followed by an evaluation of its performance using the defined metrics in the Metrics section. In the second part, the effect of reducing the size of the feature is investigated, and the model is reassessed on metrics.

The performance of the tuned model is illustrated in Figure 5. Figure 5 demonstrates that the difference between the training and validation data is sufficiently minimal, indicating that the model is not overfitted to the training data across all folds.

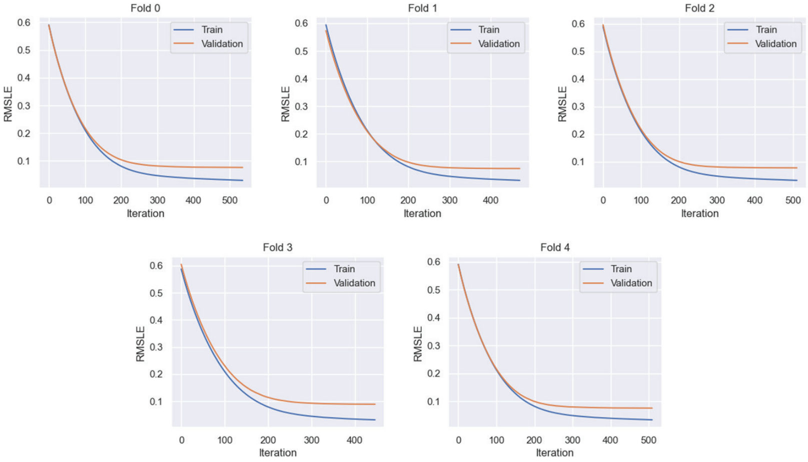


Figure 5. Model performance across folds.

Figure 6 illustrates the Cumulative Density Function of feature importance. The visualization in Figure 6 reveals that the number of features essential for model accuracy is considerably fewer than the total number of features. Our analysis indicates that merely 28% of features, encompassing all the segments traversed by ships, account for 99% of the model's feature importance. Critical among these are path weight, the latitude and longitude of both origin and destination and the busiest segments, which significantly influence the model's performance. Notably, our findings suggest that the date and time of the trip do not significantly impact the accuracy of the model.

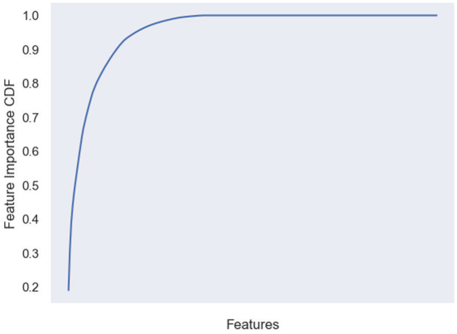


Figure 6. Feature importance cumulative density function.

Table 5 presents a comparison of the performance metrics for both the tuned model and the simplified model. This comparison highlights that removing less significant features has a negligible impact on the overall performance of the model. However, it significantly simplifies the model by reducing the sparsity of the dataset.

Table 5. Tuned full and simplified model accuracies.

Metric	Tuned Full Model	Reduced Model
R^2	0.99	0.98
MAE [min]	6.35	6.81
MSE [min]	95.01	105.21
MAPE	0.05	0.05

5.3. Apply Model

This module assesses the performance of the trained model using a test dataset, which comprises historical AIS data from trips taken during 2019 and 2020. The corresponding performance evaluation is depicted in Figure 7. To construct Figure 7, the travel times of vessels are segmented into 30 min intervals. Within each interval, the average travel time is compared with the distribution of the model's estimated travel times. As indicated by Figure 7, the model's estimated time distribution aligns closely with the average travel time for each interval. Notably, the model exhibits greater accuracy for shorter travel durations. However, for longer trips (longer than 180 min), the figure shows that the travel time estimates are more dispersed around the average, indicating reduced accuracy for these longer journeys.

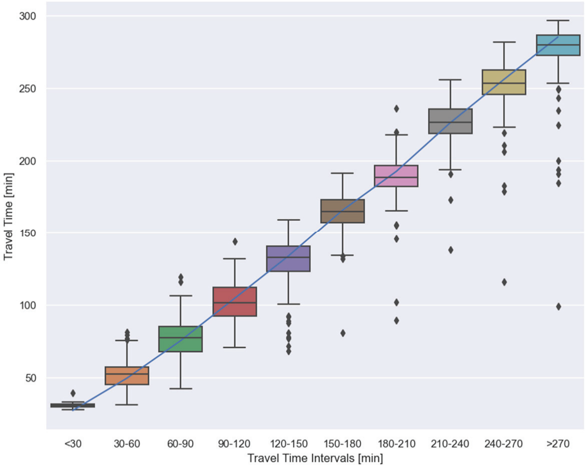


Figure 7. Model accuracy on the test dataset.

Table 6 displays the performance of the model on the test dataset. It demonstrates that both the trained model and the trained simplified model maintain consistent performance across all metrics on the test dataset, comparable to their performance on the training data.

Table 6. Model accuracy on test data.

Metric	Tuned Full Model	Reduced Model
R^2	0.98	0.98
MAE [min]	6.49	6.41
MSE [min]	136.01	127.61
MAPE	0.05	0.05

6. Conclusions

This research presents a comprehensive study on estimated time of arrival (ETA) for vessels in channels and narrow waterways. Using historical Automatic Identification System (AIS) data from 2018 to 2020, this study introduces a machine learning framework to transform raw data, perform feature engineering and preprocessing, and, finally, predict vessel arrival time in channels. The proposed XGBoost model shows high performance across all metrics. The model can predict travel time with only 5% mean absolute error and with 98% R^2. Moreover, the experimental results show that the model can maintain consistent accuracy even using a simplified structure. The less complex model not only preserves accuracy but also offers computational efficiency by addressing the sparsity in the dataset. The model also shows varying accuracy across different trip durations. While it shows higher precision for shorter trips, its predictions for longer trips (over 180 min) display a wider dispersion around the average values. It should be noted that our study focuses on ETA prediction for vessels from anchorage areas in the ocean to their destination terminals and vice versa. Therefore, a small portion of trips are longer than 180 min. The other area for future research would be model production in real-time and integrating our model results with port and channel operations.

Author Contributions: Conceptualization, M.J.K.; Methodology, H.A.; Software, A.S. and M.J.K.; Validation, A.S.; Formal analysis, A.S. and M.J.K.; Data curation, H.A.; Writing—original draft, H.A.; Writing—review & editing, A.S., M.J.K. and M.H.; Visualization, H.A.; Supervision, M.H.; Project administration, M.H.; Funding acquisition, M.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw AIS data are extracted from https://marinecadastre.gov/ais/ (accessed on 1 January 2024).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. *Review of Maritime Transport*; United Nations Publications: New York, NY, USA, 2022; ISBN 978-92-1-113073-7.
2. Park, K.; Sim, S.; Bae, H. Vessel Estimated Time of Arrival Prediction System Based on a Path-Finding Algorithm. *Marit. Transp. Res.* **2021**, *2*, 100012. [CrossRef]
3. Kang, M.J.; Hamidi, M. *Quantifying and Predicting Waterway Traffic Conditions: A Case Study of Houston Ship Channel*; Lamar University: Beaumont, TX, USA, 2021.
4. Kabir, M.; Kang, M.J.; Wu, X.; Hamidi, M. Study on U-Turn Behavior of Vessels in Narrow Waterways Based on AIS Data. *Ocean Eng.* **2022**, *246*, 110608. [CrossRef]
5. Cho, Y.; Park, J.; Kim, J.; Kim, J. Autonomous Ship Collision Avoidance in Restricted Waterways Considering Maritime Navigation Rules. *IEEE J. Ocean. Eng.* **2023**, *48*, 1009–1018. [CrossRef]
6. Agafonov, A.; Yumaganov, A. Bus Arrival Time Prediction with LSTM Neural Network. In *Proceedings of the Advances in Neural Networks—ISNN 2019*; Lu, H., Tang, H., Wang, Z., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 11–18.
7. Derrow-Pinion, A.; She, J.; Wong, D.; Lange, O.; Hester, T.; Perez, L.; Nunkesser, M.; Lee, S.; Guo, X.; Wiltshire, B.; et al. ETA Prediction with Graph Neural Networks in Google Maps. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management, Gold Coast, QLD, Australia, 1–5 November 2021; Association for Computing Machinery: New York, NY, USA, 2021; pp. 3767–3776.
8. Carr, G.C.; Erzberger, H.; Neuman, F. Fast-Time Study of Airline-Influenced Arrival Sequencing and Scheduling. *J. Guid. Control Dyn.* **2000**, *23*, 526–531. [CrossRef]
9. Roy, K.; Levy, B.; Tomlin, C. Target Tracking and Estimated Time of Arrival (ETA) Prediction for Arrival Aircraft. In Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit; Guidance, Navigation, and Control and Co-located Conferences, Keystone, CO, USA, 21–24 August 2006; American Institute of Aeronautics and Astronautics: Reston, VA, USA, 2012.
10. Lim, A.; Rodrigues, B.; Zhu, Y. Airport Gate Scheduling with Time Windows. *Artif. Intell. Rev.* **2005**, *24*, 5–31. [CrossRef]
11. Narciso, M.E.; Piera, M.A. Robust Gate Assignment Procedures from an Airport Management Perspective. *Omega* **2015**, *50*, 82–95. [CrossRef]
12. Yang, Z.; Wang, Y.; Li, J.; Liu, L.; Ma, J.; Zhong, Y. Airport Arrival Flow Prediction considering Meteorological Factors Based on Deep-Learning Methods. *Complexity* **2020**, *2020*, 6309272. [CrossRef]
13. Ayhan, S.; Costas, P.; Samet, H. Predicting Estimated Time of Arrival for Commercial Flights. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, London, UK, 19–23 August 2018; Association for Computing Machinery: New York, NY, USA; pp. 33–42.
14. Karbassi, A.; Barth, M. Vehicle Route Prediction and Time of Arrival Estimation Techniques for Improved Transportation System Management. In Proceedings of the IEEE IV2003 Intelligent Vehicles Symposium. Proceedings (Cat. No.03TH8683), Columbus, OH, USA, 9–11 June 2003; pp. 511–516.
15. Mazloumi, E.; Rose, G.; Currie, G.; Sarvi, M. An Integrated Framework to Predict Bus Travel Time and Its Variability Using Traffic Flow Data. *J. Intell. Transp. Syst.* **2011**, *15*, 75–90. [CrossRef]
16. Achar, A.; Bharathi, D.; Kumar, B.A.; Vanajakshi, L. Bus Arrival Time Prediction: A Spatial Kalman Filter Approach. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 1298–1307. [CrossRef]

17. Propp, D.A.; Rosenberg, C.A. A Comparison of Prehospital Estimated Time of Arrival and Actual Time of Arrival to an Emergency Department. *Am. J. Emerg. Med.* **1991**, *9*, 301–303. [CrossRef]
18. Fleischman, R.J.; Lundquist, M.; Jui, J.; Newgard, C.D.; Warden, C. Predicting Ambulance Time of Arrival to the Emergency Department Using Global Positioning System and Google Maps. *Prehospital Emerg. Care* **2013**, *17*, 458–465. [CrossRef]
19. Shmueli, G.; Bruce, P.C.; Patel, N.R. *Data Mining for Business Analytics: Concepts, Techniques, and Applications with XLMiner*; Wiley: Hoboken, NJ, USA, 2016; ISBN 9781118729137.
20. Pani, C.; Vanelslander, T.; Fancello, G.; Cannas, M. Prediction of Late/Early Arrivals in Container Terminals—A Qualitative Approach. *Eur. J. Transp. Infrastruct. Res.* **2015**, *15*, 536–550. [CrossRef]
21. Hart, P.E.; Nilsson, N.J.; Raphael, B. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Trans. Syst. Sci. Cybern.* **1968**, *4*, 100–107. [CrossRef]
22. Shin, Y.W.; Abebe, M.; Noh, Y.; Lee, S.; Lee, I.; Kim, D.; Bae, J.; Kim, K.C. Near-Optimal Weather Routing by Using Improved A* Algorithm. *Appl. Sci.* **2020**, *10*, 6010. [CrossRef]
23. Alessandrini, A.; Mazzarella, F.; Vespe, M. Estimated Time of Arrival Using Historical Vessel Tracking Data. *IEEE Trans. Intell. Transp. Syst.* **2018**, *20*, 7–15. [CrossRef]
24. Chen, X.; Wang, M.; Ling, J.; Wu, H.; Wu, B.; Li, C. Ship Imaging Trajectory Extraction Via an Aggregated you only Look once (YOLO) Model. *Eng. Appl. Artif. Intell.* **2024**, *130*, 107742. [CrossRef]
25. Wu, X.; Roy, U.; Hamidi, M.; Craig, B.N. Estimate Travel Time of Ships in Narrow Channel Based on AIS Data. *Ocean Eng.* **2020**, *202*, 106790. [CrossRef]
26. Wu, X.; Mehta, A.L.; Zaloom, V.A.; Craig, B.N. Analysis of Waterway Transportation in Southeast Texas Waterway Based on AIS Data. *Ocean Eng.* **2016**, *121*, 196–209. [CrossRef]
27. Wu, X.; Rahman, A.; Zaloom, V.A. Study of Travel Behavior of Vessels in Narrow Waterways Using AIS Data—A CASE study in Sabine-Neches Waterways. *Ocean Eng.* **2018**, *147*, 399–413. [CrossRef]
28. Roy, U.; Wu, X. Ais-data based vessel traffic's characteristics and travel behaviour analysis: A case study at houston ship channel. *J. Ocean Technol.* **2019**, *14*, 58–74.
29. Kang, L.; Meng, Q.; Liu, Q. Fundamental Diagram of Ship Traffic in the Singapore Strait. *Ocean Eng.* **2018**, *147*, 340–354. [CrossRef]
30. Ghanim, M.S.; Shaaban, K.; Miqdad, M. An Artificial Intelligence Approach to Estimate Travel Time along Public Transportation Bus Lines. In Proceedings of the International Conference on Civil Infrastructure and Construction, Doha, Qatar, 2–5 February 2023.
31. Liu, H.; Xu, H.; Yan, Y.; Cai, Z.; Sun, T.; Li, W. Bus Arrival Time Prediction Based on LSTM and Spatial-Temporal Feature Vector. *IEEE Access* **2020**, *8*, 11917–11929. [CrossRef]
32. Petersen, N.C.; Rodrigues, F.; Pereira, F.C. Multi-Output Bus Travel Time Prediction with Convolutional LSTM Neural Network. *Expert Syst. Appl.* **2019**, *120*, 426–435. [CrossRef]
33. Ranjitkar, P.; Tey, L.-S.; Chakravorty, E.; Hurley, K.L. Bus Arrival Time Modeling Based on Auckland Data. *Transp. Res. Rec. J. Transp. Res. Board* **2019**, *2673*, 1–9. [CrossRef]
34. Larsen, G.H.; Yoshioka, L.R.; Marte, C.L. Bus Travel Times Prediction Based on Real-Time Traffic Data Forecast Using Artificial Neural Networks. In Proceedings of the 2020 International Conference on Electrical, Communication, and Computer Engineering (ICECCE), Istanbul, Turkey, 12–13 June 2020; pp. 1–6.
35. Alam, O.; Kush, A.; Emami, A.; Pouladzadeh, P. Predicting Irregularities in Arrival Times for Transit Buses with Recurrent Neural Networks Using GPS Coordinates and Weather Data. *J. Ambient. Intell. Humaniz. Comput.* **2020**, *12*, 7813–7826. [CrossRef]
36. Chondrodima, E.; Georgiou, H.; Pelekis, N.; Theodoridis, Y. Particle Swarm Optimization and RBF Neural Networks for Public Transport Arrival Time Prediction Using GTFS Data. *Int. J. Inf. Manag. Data Insights* **2022**, *2*, 100086. [CrossRef]
37. Milenković, M.; Milosavljevic, N.; Bojović, N.; Val, S. Container Flow Forecasting through Neural Networks Based on Metaheuristics. *Oper. Res.* **2021**, *21*, 965–997. [CrossRef]
38. Kourounioti, I.; Polydoropoulou, A.; Tsiklidis, C. Development of Models Predicting Dwell Time of Import Containers in Port Container Terminals—An Artificial Neural Networks Application. *Transp. Res. Procedia* **2016**, *14*, 243–252. [CrossRef]
39. Gao, M.; Shi, G.; Li, S. Online Prediction of Ship Behavior with Automatic Identification System Sensor Data Using Bidirectional Long Short-Term Memory Recurrent Neural Network. *Sensors* **2018**, *18*, 4211. [CrossRef]
40. Kim, J.; Lee, C.; Chung, D.; Kim, J. Navigable Area Detection and Perception-Guided Model Predictive Control for Autonomous Navigation in Narrow Waterways. *IEEE Robot. Autom. Lett.* **2023**, *8*, 5456–5463. [CrossRef]
41. Fancello, G.; Pani, C.; Pisano, M.; Serra, P.; Zuddas, P.; Fadda, P. Prediction of Arrival Times and Human Resources Allocation for Container Terminal. *Marit. Econ. Logist.* **2011**, *13*, 142–173. [CrossRef]
42. Pani, C.; Cannas, M.; Fadda, P.; Fancello, G.; Frigau, L.; Mola, F. Delay Prediction in Container Terminals: A Comparison of Machine Learning Methods. In Proceedings of the WCTR (World Conference on Transport Research), Rio de Janeiro, Brazil, 15–18 July 2013.
43. Pani, C.; Fadda, P.; Fancello, G.; Frigau, L.; Mola, F. A data mining approach to forecast late arrivals in a transhipment container terminal. *Transport* **2014**, *29*, 175–184. [CrossRef]
44. Pallotta, G.; Vespe, M.; Bryan, K. Vessel Pattern Knowledge Discovery from AIS Data: A Framework for Anomaly Detection and Route Prediction. *Entropy* **2013**, *15*, 2218–2245. [CrossRef]

45. Parolas, I.; Tavasszy, L.; Kourounioti, I.; van Duin, R.; Cities, K. Prediction of Vessels' Estimated Time of Arrival (ETA) Using Machine Learning–a Port of Rotterdam Case Study. In Proceedings of the 96th Annual Meeting of the Transportation Research, Washington, DC, USA, 8–12 January 2017; pp. 8–12.
46. Noman, A.A.; Heuermann, A.; Wiesner, S.A.; Thoben, K.-D. Towards Data-Driven GRU based ETA Prediction Approach for Vessels on both Inland Natural and Artificial Waterways. In Proceedings of the 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), Indianapolis, IN, USA, 19–22 September 2021; pp. 2286–2291.
47. Yu, J.; Tang, G.; Song, X.; Yu, X.; Qi, Y.; Li, D.; Zhang, Y. Ship Arrival Prediction and Its Value on Daily Container Terminal Operation. *Ocean Eng.* **2018**, *157*, 73–86. [CrossRef]
48. Notteboom, T.E. The Time Factor in Liner Shipping Services. *Marit. Econ. Logist.* **2006**, *8*, 19–39. [CrossRef]
49. Available online: https://marinecadastre.gov/ (accessed on 1 January 2024).
50. Sedaghat, A.; Kang, M.J.; Hamidi, M. A Heuristic ETL Process to Dynamically Separate and Compress AIS Data. In Proceedings of the 2023 Systems and Information Engineering Design Symposium (SIEDS), Charlottesville, VA, USA, 27–28 April 2023; pp. 159–164.
51. Sedaghat, A.; Arbabkhah, H.; Kang, M.J.; Hamidi, M. Deep Learning Applications in Vessel Dead Reckoning to Deal with Missing Automatic Identification System Data. *J. Mar. Sci. Eng.* **2024**, *12*, 152. [CrossRef]
52. Wold, S.; Esbensen, K.; Geladi, P. Principal component analysis. *Chemom. Intell. Lab. Syst.* **1987**, *2*, 37–52. [CrossRef]
53. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; Association for Computing Machinery: New York, NY, USA; pp. 785–794.

MDPI
St. Alban-Anlage 66
4052 Basel
Switzerland
www.mdpi.com

Journal of Marine Science and Engineering Editorial Office
E-mail: jmse@mdpi.com
www.mdpi.com/journal/jmse

MDPI

Academic Open
Access Publishing

mdpi.com

ISBN 978-3-7258-0656-0