

Article

A Transformer-Based Unified Framework for Multiple Target Track Association and Fusion under Incomplete Measurements

Likai Zeng ¹, Fan Liu ^{1,2,*} and Xin'an Wang ^{3,4}¹ Nanjing Research Institute of Electronics Technology, Nanjing 210039, China; bestzlk@163.com² School of Computer Science, Northwestern Polytechnical University, Xi'an 710072, China³ School of Software, Tsinghua University, Beijing 100084, China; wxa25@mails.tsinghua.edu.cn⁴ China Ship Development and Design Center, Shanghai 201100, China

* Correspondence: liufant800@mail.nwpu.edu.cn

Abstract

Multi-sensor multi-target track segment association and fusion play a critical role in maritime target surveillance systems; however, it is difficult to balance feature extraction efficiency, multi-task adaptability and kinematic state estimation accuracy under complex maneuvering scenarios. To tackle this problem, we propose an end-to-end Transformer-based multi-target track association and fusion framework (TMTAFF), which is an offline method. TMTAFF mainly comprises three core components: Track Completion Module (TCM), Track Association Module (TAM) and Track Fusion Module (TFM). The designed TCM is first used to complete the interrupted track segments, and the completed track segments are further fed into the TAM to accurately determine whether any two track segments originate from an identical target. Based on the outputs of the TAM, the TFM subsequently fuses the track segments of each target and outputs the final track fusion results. It is noted that TMTAFF can be simultaneously applied to both interrupted track segment association (ITSA) tasks and multi-source track segment association (MSTSA) tasks due to the adoption of the TCM. Experimental results based on a public dataset show that TMTAFF outperforms 11 state-of-the-art baselines, including traditional approaches and neural network-based models.

Keywords: track completion module (TCM); track association module (TAM); track fusion module (TFM); multi-target track association and fusion; transformer

Academic Editor: Douglas O'Shaughnessy

Received: 22 June 2026

Revised: 13 July 2026

Accepted: 18 July 2026

Published: 20 July 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

In the domain of battlefield situation awareness, target track association and track fusion serve as the fundamental prerequisites for acquiring accurate and integrated target tracks, and their performance exerts a profound impact on operational command and decision-making [1]. Nevertheless, the practical performance of these two core technologies remains far from satisfactory due to the influence of diverse complex factors, including incomplete data acquisition, heterogeneous multi-sensor observation systems, and complex electromagnetic interference in battlefield environments. Such unsatisfactory performance tends to yield adverse consequences, such as target misjudgment, information distortion, and decision-making delays in actual combat scenarios. Consequently, it is imperative to continuously improve the accuracy of track association and fusion algorithms

and enhance their performance and simplicity to meet the stringent requirements of modern battlefield situation awareness [2].

Generally, the generation of accurate and integrated target tracks based on multi-sensor detection data consists of two sequential procedures: track association and track fusion [3,4]. Track association refers to the process of determining which individual tracks observed by different sensors correspond to the same target, while track fusion is used to deeply fuse multiple low-precision or incomplete tracks belonging to an identical target to estimate the true target state, including critical parameters such as target position, velocity, and so on. The above two procedures are tightly coupled in practical implementation; however, most existing studies usually focus on developing models dedicated to solving only one of these two problems [5,6]. Such methodologies are unable to directly output the final fused target tracks from raw sensor measurements and thus do not constitute end-to-end frameworks. Consequently, it is difficult to meet the practical requirements in battlefield.

Track association typically involves two categories of scenarios [7]: interrupted track segment association (ITSA) for discontinuous tracks acquired by a single sensor, and multi-source track segment association (MSTSA) for track segments derived from heterogeneous sensors.

To tackle the ITSA problem, Yeom et al. [8] and Zhang et al. [9] proposed conventional methods based on a priori target motion hypothesis models. These methods construct an association framework for track segments by performing forward prediction or backward retrodiction between the termination of the previous track segment and the initiation of the new one. If the predicted or retrodicted trajectory meets predefined discrimination criteria, the two track segments are regarded as associated. In [10], an association method integrating the constant acceleration (CA) model and the coordinated turn (CT) model is proposed. By analyzing multiple model parameters, including acceleration and turn rate, this method accurately characterizes the motion features of the target and further employs fuzzy similarity metrics to evaluate the correlation between interrupted tracks. Raghu et al. tackle the target tracking interruption problem within a two-dimensional (2-D) assignment framework, where the constant velocity (CV) model is adopted to represent the continuous motion of the target, while the CT model is utilized during track interruptions [11], which optimizes the association performance by tuning the turn rate and applying the 2-D matching algorithm proposed in [12] to achieve a globally optimal solution. Nevertheless, the aforementioned traditional approaches based on a priori assumptions share a common limitation: when the actual motion state of the target deviates from the preset hypothetical model, the association accuracy and stability can hardly be ensured.

To solve the MSTSA problem, conventional methods aim to alleviate systematic biases and random measurement errors arising from heterogeneous sensing systems, relying on a variety of discriminative criteria, including statistical inference, structural feature similarity, fuzzy logic reasoning, and so on. In [13], an improved track-to-track association approach was developed, which introduces reference topology features to effectively suppress the adverse impacts of sensor systematic deviations. Additionally, the methodology reported in [13] converts the conventional track association problem into a non-rigid point registration framework, where structural similarity constraints are embedded to enhance the robustness and accuracy of association performance. Moreover, an innovative association algorithm is proposed in [14] based on a dynamically adjustable weight mechanism. By embedding target spatial position information into the weight function and iteratively updating the fuzzy weight set, this enhanced algorithm exhibits favorable association performance under dense target environments. However, such multi-source track association

schemes still face notable limitations in practical applications. On the one hand, inconsistent or time-varying sensor noise characteristics tend to degrade association stability and matching precision. On the other hand, most critical decision thresholds rely on manual empirical selection, which complicates the determination of a universally optimal threshold and restricts the generalization ability of these methods in complex scenarios.

Research on target track fusion using multi-source heterogeneous sensor data remains relatively limited. Existing studies predominantly focus on multi-target tracking, and a considerable portion of them relies on video data to track objects such as vehicles and pedestrians. For instance, Senel et al. [15] employed the unscented Kalman filter to fuse vehicle targets extracted from data collected by radar, LiDAR and cameras, thereby achieving multi-target tracking based on multiple sensors. Similarly, Sengupta et al. [16] put forward a tri-Kalman filter-based computational framework to integrate measurements from monocular cameras and millimeter-wave radars, enabling precise tracking of pedestrians in urban traffic scenarios. Guan et al. [17] developed a sequential fusion method grounded in the Extended Kalman Filter (EKF) algorithm to track orbital targets using data from multiple radar systems. These Kalman filter-based methods possess unique superiority in target track prediction and multi-sensor track fusion. Adopting a recursive predict-update framework, it extrapolates target motion states in advance to realize track prediction and only stores real-time state information instead of massive historical data, which satisfies the real-time computing requirements of sensing systems. By introducing covariance matrices to characterize estimation errors, the filter can optimally weight predicted states and noisy multi-source heterogeneous measurements, effectively restraining sensor clutter and Gaussian noise to smooth discontinuous raw tracks. For nonlinear maneuvering targets, the Extended Kalman Filter and Unscented Kalman Filter further overcome linear motion constraints and maintain stable prediction accuracy. Moreover, the modular structure of Kalman filtering allows easy integration with adaptive noise adjustment and multi-model algorithms, mitigating performance losses caused by fixed motion model assumptions, thereby delivering balanced precision and computational efficiency for complex target track fusion tasks. Nevertheless, the motion patterns of vehicles and pedestrians differ substantially from those of battlefield targets. In particular, orbital targets obey rigid and fixed motion laws, which is in sharp contrast to the highly random maneuvering characteristics of battlefield objects. Furthermore, conventional Kalman filtering-based fusion approaches rely heavily on predefined assumptions regarding target motion models. Hence, the aforementioned methods can hardly satisfy the practical application requirements in complex battlefield environments.

Although conventional methods have made achievements in track association and fusion, they suffer from inherent drawbacks, such as oversimplified model assumptions, noise sensitivity, manual parameter tuning, and limited performance [11,18]. In recent years, data-driven neural network methods have attracted significant attention in track processing [19,20], since they can automatically learn effective trajectory features from data and provide improved flexibility and accuracy. For example, Wang et al. utilized Graph Neural Network (GNN)-based text classification models for fire-door defect inspection, which shows strong potential and effectiveness for enhancing safety management practices in real-world construction environments [21,22].

To address the issue of ITSA, existing research has attempted to integrate deep learning architectures. For instance, the work presented in [23] adopts a hybrid structure consisting of LSTM [24] and CNN [25] to jointly model spatial and temporal characteristics contained in trajectory sequences. By measuring the feature-level similarity between different trajectories, the method is able to judge whether they correspond to the same target. In parallel, the approach proposed in [26] introduces graph neural networks to derive two levels of representations, namely node-level local feature embeddings of individual track

points and graph-level global embeddings of entire track structures, thereby supporting the association of discontinuous track segments. Although these studies have constructed dedicated network designs suitable for trajectory data and achieved target identification via similarity comparison under static conditions, they overlook the dynamic characteristics present in practical scenarios. In real environments, target tracks may emerge or vanish at arbitrary moments, requiring real-time computation of similarity and adaptive decision-making driven by the evolution of trajectory states [27–29].

As for the MSTSA problem, the method proposed in [2] employs a spatial-temporal hybrid feature extraction module to derive representations from both trajectory sequences and background scenes. The obtained features are further applied to estimate the homography matrix, thereby enhancing the discriminability of decision criteria and strengthening the reliability of association results. In addition, the approach introduced in [30] adopts an autoencoder structure to perform effective noise reduction on raw trajectory data. After that, fully connected layers are utilized to extract high-dimensional features that capture the inherent patterns of trajectories, while 3D convolutional layers are adopted to encode environmental information from the scene. By fusing trajectory-based and scene-based features, the method evaluates the probability that two trajectory segments belong to the same target. Nevertheless, similar to existing ITSA solutions, these methods [2,30] ignore the dynamic variations in real-world scenes. More specifically, they rely on complete trajectory sequences to perform association, rather than making online judgments based only on the information observed up to the current time step.

For the track fusion problem, Meinhardt et al. [31] adopted CNN to extract object features from visual images and employed Transformer to associate targets across consecutive frames, which enabled effective pedestrian tracking in video sequences. However, this approach, known as TrackFormer, relies solely on camera data and thus lacks adaptability to multi-sensor heterogeneous observation systems. Contrastively, Liu et al. [32] introduced the Transformer into radar-based multi-target tracking by utilizing raw radar plots, yet their core contribution focuses on data association at the plot level rather than track-to-track association. Furthermore, Xu et al. [33] proposed a multi-object tracking model built upon Swin Transformer Neck-YOLOX, which achieves robust detection and tracking of various vehicles from unmanned aerial vehicle videos. Although these data-driven methods have achieved remarkable performance improvements in visual or single-sensor tracking tasks, they still struggle to handle maritime target track fusion under multi-source heterogeneous sensor configurations. Qi et al. [19] utilized a long short-term memory (LSTM) network and an attention mechanism to extract spatiotemporal features from asynchronous target tracks, and useful features are then selected to construct the fused target tracks. Cui et al. [34] employed two individual Transformer-based models to separately estimate the position and velocity of the fused target tracks. While these methods achieve good performance, there remains room for performance improvement.

To address the aforementioned challenges, this paper proposes a Transformer-based **multi-target track association and fusion framework (TMTAFF)** under incomplete measurements across multiple sensors. The proposed TMTAFF consists of three tightly coupled modules: the track completion module, the track association module (TAM), and the track fusion module (TFM). Specifically, the TCM reconstructs interrupted tracks to mitigate the negative impacts caused by incomplete measurements on subsequent association and fusion. The TAM identifies track segments from heterogeneous sensors that correspond to the same target by exploiting cross-sensor detection information. The TFM then integrates multiple segments of the same target to accurately estimate target states, including position, speed, heading, and so on. Notably, beyond achieving promising performance in track association and track fusion, TMTAFF also exhibits strong adaptability and scalability since it can adapt to variable numbers of sensors and tolerate temporary

sensor failures, making it well-suited for reliable target track generation in complex battlefield environments.

The contributions of this work are four-fold:

First, the TMTAFF designed in this paper is constructed as a unified paradigm, which achieves a seamless integration of track association and track fusion within a unified computational pipeline. By taking raw target observations acquired from heterogeneous sensors as input, TMTAFF enables direct and reliable inference of state estimates for individual targets without intermediate post-processing. Conversely, most existing studies tend to concentrate on either track association or target state estimation, while limited efforts have been devoted to realizing joint track association and fusion simultaneously in multi-sensor observation scenarios.

Second, the proposed TAM is able to address both the ITSA problem and the MSTSA problem concurrently and demonstrates strong adaptability in adverse operating conditions, such as incomplete measurements and single sensor failure. In addition, the TAM is trained using the contrastive loss function, which effectively increases the accuracy of track association, making it more efficient than traditional methods.

Third, the designed TFM is particularly suitable for multi-sensor and multi-target scenes. Unlike conventional approaches, there is no need to pre-specify target motion models; in addition, it is not limited by the types and quantities of sensors, making it more applicable to target state estimation in complex battlefield environments.

Fourth, experimental results based on a public dataset demonstrate that the TMTAFF outperforms 11 baselines. In particular, TMTAFF exceeds six state-of-the-art track segment association methods and achieves 94.58% of mP_{CA} and 91.71% of mwP_{CA} for ITSA tasks, as well as 93.91% of mP_{CA} and 91.38% of mwP_{CA} for MSTSA tasks. Moreover, TMTAFF surpasses five cutting-edge track segment fusion methods and obtains $6.24 \times 10^{-4^\circ}$, $5.88 \times 10^{-4^\circ}$, 1.03 kn and 0.84° for mME_{lo} , mME_{la} , mME_{sp} and mME_{he} , respectively. Additionally, the effectiveness of the TCM in enhancing the performance of TMTAFF in track segment association and fusion is quantitatively demonstrated by ablation studies. Furthermore, the hyperparameters of the TMTAFF are fine-tuned to their optimal values.

The rest of this paper is organized as follows. Section 2 elaborates on the details of the proposed TMTAFF. In Section 3, the experimental results are evaluated. Subsequently, the limitations of the proposed TMTAFF are fully discussed in Section 4. Finally, the paper is concluded, and future work is discussed in Section 5.

2. Methods

In this section, we first illustrate the architecture and the workflow of the proposed TMTAFF, and then elaborate on the details of the designed TCM, TAM and TFM in order. Finally, the training of the model is explained in detail.

2.1. Overview of the TMTAFF

The architecture of TMTAFF is illustrated in Figure 1, which is predominantly made up of three core functional modules, namely, a track completion module (TCM), a track association module (TAM) and a track fusion module (TFM). Specifically, the input of TMTAFF is the target track segments acquired from multiple heterogeneous sensors. Prior to subsequent processing, the track segments undergo preprocessing procedures, such as time registration and spatial registration, to eliminate errors caused by sensor limitations. Subsequently, an input embedding module is employed to convert the preprocessed track segments into a matrix representation that is compatible with the Transformer-based TCM. After being processed by the TCM, each individual track segment is complemented within a predefined time duration, which ensures that all completed track segments adhere to

the same time duration and data length, thereby facilitating accurate comparison of similarity between any pair of track segments. Afterwards, the completed track segments are pairwise combined, with each track pair being individually input into the TAM for further processing. The TAM is responsible for determining whether the two input tracks in each pair originate from the same target. Based on the outputs of the TAM, an association matrix is constructed, by which it is feasible to represent which track segments in the dataset belong to the same target. Subsequently, for each target, its two track segments are fed into the TFM, which directly outputs the fusion result of these two track segments. If there are still track segments to be fused for the target, the aforementioned fusion result and one of the remaining track segments are fed back into the TFM. This iterative process continues until all track segments corresponding to the target have been processed, and the final output of the TFM is regarded as the final target track segment fusion results.

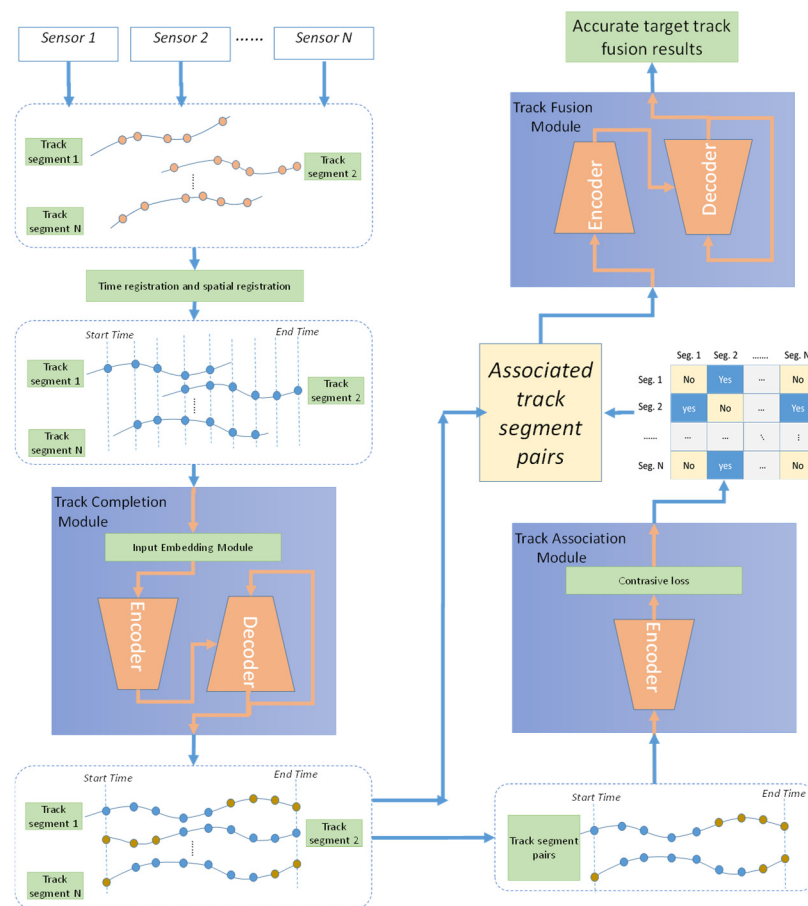


Figure 1. The workflow of the proposed TMTAFF.

2.2. Track Completion Module

Usually, the time duration and length of each track segment are different from each other, which makes it difficult to directly perform track association and track fusion operations. To solve this problem, the track completion module (TCM) is specifically designed to complete an interrupted track segment within a specified time range. Assuming that the track association and track fusion operations need to be performed within the time range of $T_{start} \sim T_{end}$ and the time range of a track segment after time registration and spatial registration is $T_i \sim T_j$ ($start < i < j < end$), the task of TCM is to complete the missing values in the time range of $T_{start} \sim T_i$ and $T_j \sim T_{end}$, and then splice the completed data segments with the original data segment together to generate the final completed track segment.

As shown in Figure 2, the proposed TCM consists of three components, i.e., an input embedding module, a Transformer encoder and a Transformer decoder [35]. The input embedding module is utilized to convert the track segment of a target into a unified vector representation. This vectorization process ensures that the track segment data is compatible with the subsequent Transformer-based processing, laying a solid foundation for accurate track completion and subsequent track association and track fusion procedures.

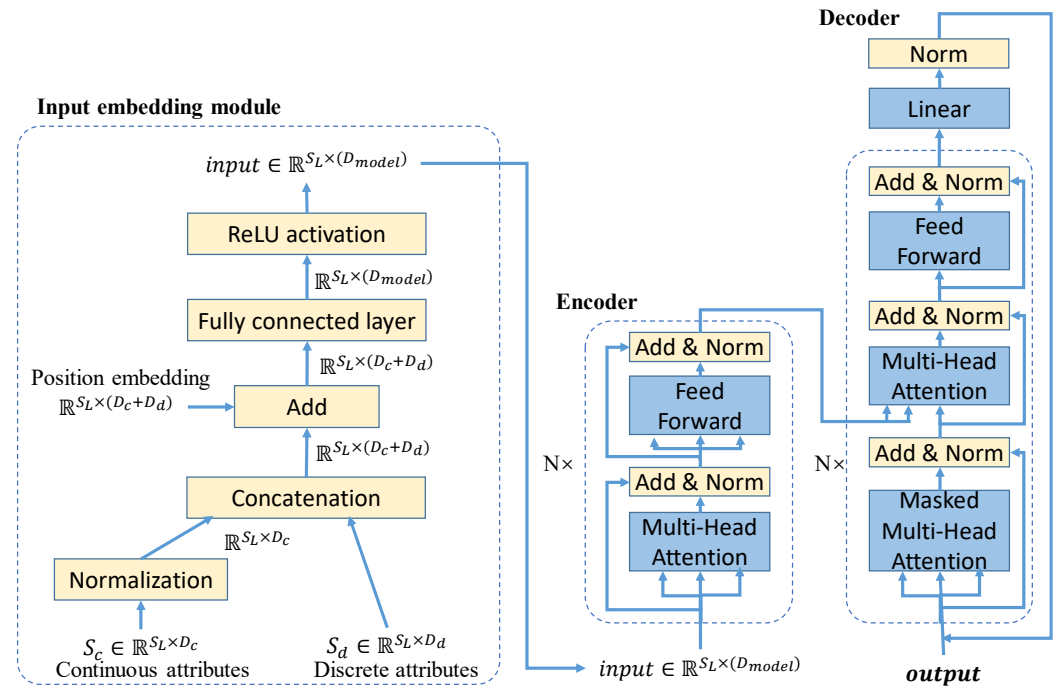


Figure 2. The architecture of the designed TCM.

For the track point of a track segment at an arbitrary time step, it contains not only continuous attributes such as position, speed, heading and so on, but also discrete attributes, including target category, target type, and friend-or-foe attribute. Intuitively, discrete attributes can reflect the motion pattern of the target to a certain extent. For instance, the speed of a missile is usually higher than that of an aircraft; in addition, the turning speed of a ship is typically lower than that of an aircraft. Moreover, track segments with different discrete attributes are likely not to belong to the same target. For example, a friendly target is mostly not the same as a hostile target. Therefore, the utilization of discrete attributes during track completion is beneficial in two aspects: on the one hand, it helps improve the accuracy of the completed track points; on the other hand, it contributes to enhancing the correctness of the subsequent track association and track fusion processes.

For a given track segment S , we employ $S_c \in \mathbb{R}^{S_L \times D_c}$ to represent its continuous attributes, where S_L denotes the length of the track and D_c is the number of continuous attributes. Similarly, we utilize $S_d \in \mathbb{R}^{S_L \times D_d}$ to represent its discrete attributes, where S_L also denotes the length of the track and D_d is the sum of the number of unique discrete values of each discrete attribute. Since the count of attributes is usually limited, simply concatenating the S_c and S_d as the input of the Transformer encoder will restrict the mining of latent knowledge contained in the original data. Furthermore, intuitively, attribute values near the point to be predicted should be paid more attention. Therefore, prior to feeding track segment sequences into the Transformer encoder, an input embedding module is adopted to increase its dimension and embed its position information for the track segment. Specifically, as shown in Figure 2, we first normalize S_c by utilizing

the Z-Score method [36]. Then, the normalized continuous attributes are concatenated with the discrete attributes. Next, to provide each track point with positional cues across the complete track segment, the concatenated results are added with a position embedding vector $P_{pos} \in \mathbb{R}^{S_L \times (D_c + D_d)}$, which is computed as follows:

$$P_{pos,2i} = \sin\left(\frac{pos}{10000^{\frac{2i}{D}}}\right) \quad (1)$$

$$P_{pos,2i+1} = \cos\left(\frac{pos}{10000^{\frac{2i}{D}}}\right) \quad (2)$$

where pos represents the position of the target point in the track, and $2i$ and $2i + 1$ denote the dimensions. Afterwards, we employ a fully connected layer to increase the dimension to $\mathbb{R}^{S_L \times D_{model}}$ and then activate it using the ReLU function [37], so as to obtain the final input of the Transformer encoder, which is computed as follows:

$$Input = \text{ReLU}\left((\text{concatenated attributes} + P_{pos})W_{input}\right) \in \mathbb{R}^{S_L \times D_{model}} \quad (3)$$

where $W_{input} \in \mathbb{R}^{(S_c + S_d) \times D_{model}}$ is a trainable matrix and D_{model} is set much larger than the sum of S_c and S_d in order to strengthen the nonlinear representation and fitting capacity of the network.

As depicted in Figure 2, the TCM is composed of N Transformer encoder units and N Transformer decoder units. In the units, the most important technique is the multi-head attention mechanism [38], which enables the TCM to attend to the information from different representation subspaces at different positions concurrently and hence to capture various types of sequential dependencies, resulting in significantly boosting the expressive power and stability of the attention mechanism.

Let the $Input_{i,j} \in \mathbb{R}^{S_L \times D_{model}}$ denote the preprocessed input vector within $T_i \sim T_j$. For the multi-head attention mechanism, $Input_{i,j}$ is first transformed into multiple sets of matrices as below, and each set contains three matrices, namely query q , key k and value v .

$$q = Input_{i,j}W_q, \quad W_q \in \mathbb{R}^{D_{model} \times D_{model}} \quad (4)$$

$$k = Input_{i,j}W_k, \quad W_k \in \mathbb{R}^{D_{model} \times D_{model}} \quad (5)$$

$$v = Input_{i,j}W_v, \quad W_v \in \mathbb{R}^{D_{model} \times D_{model}} \quad (6)$$

where W_q , W_k , W_v are trainable parameters. Thereafter, the attention Att_{qkv} , which is utilized to determine how much information of v should be output at the current time step based on q and k , is computed as follows:

$$Att_{qkv} = \text{softmax}\left(\frac{qk^T}{\sqrt{D_k}}\right) \in \mathbb{R}^{S_L \times S_L} \quad (7)$$

where $D_k = D_{model}/h$ and h is the number of heads in the multi-head attention. It is notable that in the decoder, a mask matrix is added when computing the attention, as below:

$$Att_{qkv,masked} = \text{softmax}\left(\frac{qk^T}{\sqrt{D_k}} + M_{att}\right) \in \mathbb{R}^{S_L \times S_L} \quad (8)$$

where $M_{Att} \in \mathbb{R}^{S_L \times S_L}$ is an upper triangular mask matrix of which all entries in the upper triangular region are assigned to negative infinity. This guarantees that each current track point can only access the historical preceding information during the track completion procedure, and the future data are shielded so as to avoid information leakage in temporal prediction.

Afterwards, the information in v that will be extracted and fed forward is calculated by using the obtained Att_{qkv} or $Att_{qkv,masked}$, as below:

$$Info = Att_{qkv}v \in \mathbb{R}^{S_L \times S_{model}} \quad (9)$$

$$Info_{masked} = Att_{qkv,masked}v \in \mathbb{R}^{S_L \times S_{model}} \quad (10)$$

In the multi-head attention mechanism, the aforementioned self-attention operation is computed independently for h times. Eventually, the output obtained from each self-attention operation is subsequently concatenated and fused to generate the final output of multi-head attention, which is illustrated as follows:

$$output = cat[Info_1, \dots, Info_h]W_{output} \in \mathbb{R}^{S_L \times S_{model}} \quad (11)$$

where $Info_i$ represents the information generated by the i_{th} self-attention operation, and $W_{output} \in \mathbb{R}^{S_{model} \times h \times S_{model}}$ is a trainable matrix.

The Feed Forward module shown in Figure 2 comprises two fully connected layers, which aim to conduct a non-linear transformation so as to enhance the expressive capabilities of the network. Concretely, the number of neurons contained in the two fully connected layers is $2 \times S_{model}$ and S_{model} , and these neurons are activated by using the GELU activation function [39]. After being processed by the Feed Forward module, the shape of the output remains the same as that of the input.

Notably, the information output by the decoder at each time step corresponds to the completed track data at the current moment. Specifically, we only extract the completed information segment of $T_j \sim T_{end}$, denoted as $Com_{j \sim end}$. In cases where the decoder fails to generate sufficient data to fill up the time range, the padding technique is employed to fill the missing values with zeros, ensuring consistent dimensions for each completed track segment. When completing the track within $T_{start} \sim T_i$, the preprocessed input track segment should be reversed before being fed into the encoder, by which the information of $T_i \sim T_{start}$ is completed sequentially. Then, we reverse the information again so as to produce the completed segment of $T_{start} \sim T_i$, denoted as $Com_{start \sim i}$. Eventually, we concatenate $Com_{start \sim i}$, the preprocessed track segment, i.e., the $Input \in \mathbb{R}^{S_L \times D_{model}}$, and $Com_{j \sim end}$ to form the complete track segment within $T_{start} \sim T_{end}$, as below:

$$Seg = cat[Com_{start \sim i}, Input, Com_{j \sim end}] \in \mathbb{R}^{S_{end-start} \times S_{model}} \quad (12)$$

Essentially, TCM is a self-supervised learning network. It takes the historical track segment preceding each real track point as the input sample and regards the corresponding current track point as the ground truth. The loss function of TCM is formulated as follows:

$$L_{TCM} = \frac{1}{S_L \times S_{model}} \sum_{i=1}^N \sum_{d=1}^{S_{model}} (y_{i,d} - \widehat{y}_{i,d})^2 \quad (13)$$

where S_L denotes the length of the input track segment, and S_{model} represents the feature dimension of inputs processed by the input embedding module, $y_{i,d}$ refers to the d -dimensional predicted value output by the decoder at the i_{th} time step, while $\widehat{y}_{i,d}$ stands for the ground-truth value of the d_{th} dimension at the i_{th} time step generated by the input embedding module.

2.3. Track Association Module

In this study, a track association module (TAM) based on the Siamese network technique is proposed in order to simultaneously solve both the ITSA problem and the MSTSA problem. Firstly, the completed track segments are grouped pairwise regardless of whether they come from the same target or the same sensor. As illustrated in Figure 3, let Seg_i and Seg_j represent the two completed track segments, respectively, in an arbitrary pair. Next, Seg_i and Seg_j are further processed, respectively, through two Transformer-

based encoders with the same network parameters, whose network architectures are consistent with the encoder in TCM, while the network parameters are not shared. Thus, in this section, the details of the encoder will not be elaborated again. Subsequently, a loss function is employed to determine whether Seg_i and Seg_j are originally from the same target. Adopting the Transformer model as the data processor of the Siamese network can better capture local temporal features compared with using a traditional convolutional neural network. This approach enables accurate discrimination of differences among complex trajectories characterized by close-range interleaving, similar trajectories and measurement noise. Leveraging the weight-sharing property of Siamese networks guarantees consistent criteria for feature extraction, thereby effectively reducing the probabilities of incorrect association and missed association in dense multi-target scenarios. Consequently, the accuracy and generalization robustness of trajectory association are substantially improved under complex electromagnetic environments.

For the Siamese network technique, the loss function is a critical factor that will significantly affect the track association performance of TAM. Two types of loss functions are adopted in this study, namely the contrastive loss function [40] and the cross-entropy loss function [41], in order to obtain better performance.

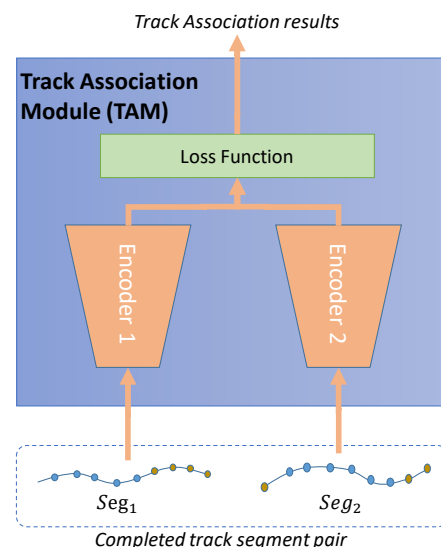


Figure 3. The architecture of the designed TAM based on Siamese network technique.

2.3.1. Utilization of Contrastive Loss Function

Contrastive loss function aims to reduce the feature distance between track segments belonging to the same target, while increasing the feature discrepancy among those from distinct targets. By constructing positive and negative sample pairs, this loss function facilitates the effective learning of discriminative data representations. It optimizes feature distribution by continuously minimizing the dissimilarity of analogous samples and maximizing the divergence between heterogeneous data instances.

In the TAM, a track segment pair (Seg_i, Seg_j) is defined as a positive sample if Seg_i and Seg_j originate from the same target, irrespective of their sensor sources. Otherwise, it is labeled as a negative sample. Taking Figure 4 as an example, there are two sensors denoted as S_1 and S_2 , respectively, by which two targets, denoted as T^A and T^B , are detected. Owing to certain factors such as sensor malfunctions, a total of 8 interrupted track segments are ultimately obtained, i.e., $T_{S1,1}^A$, $T_{S1,2}^A$, $T_{S2,1}^A$, $T_{S2,2}^A$, $T_{S1,1}^B$, $T_{S1,2}^B$, $T_{S2,1}^B$, $T_{S2,2}^B$. Herein, the notation $T_{S_m,i}^X$ represents the i_{th} track segment of target T^X detected by sensor S_m . Obviously, any pair consisting of two track segments selected from $T_{S1,1}^A$, $T_{S1,2}^A$, $T_{S2,1}^A$ and $T_{S2,2}^A$ is a positive pair. Similarly, all pairs formed by arbitrary two track

segments among $T_{S1,1}^B$, $T_{S1,2}^B$, $T_{S2,1}^B$ and $T_{S2,2}^B$ are also positive pairs. In contrast, any pair whose track segments come from the first group and the second group, respectively, is marked as a negative pair. Figure 5 illustrates whether each of the aforementioned track pairs is positive or negative.

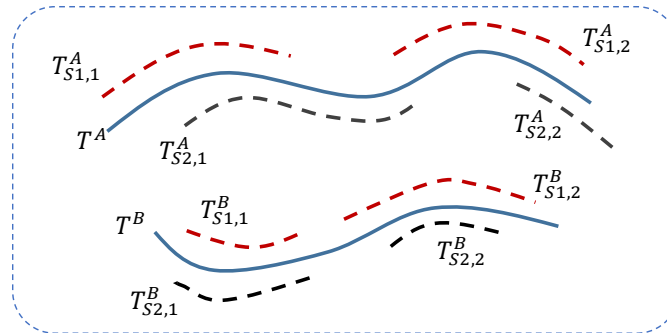


Figure 4. Track segments of multiple targets detected by multi-sensors.

	$T_{S1,1}^A$	$T_{S1,2}^A$	$T_{S2,1}^A$	$T_{S2,2}^A$	$T_{S1,1}^B$	$T_{S1,2}^B$	$T_{S2,1}^B$	$T_{S2,2}^B$
$T_{S1,1}^A$	--	P	P	P	N	N	N	N
$T_{S1,2}^A$	P	--	P	P	N	N	N	N
$T_{S2,1}^A$	P	P	--	P	N	N	N	N
$T_{S2,2}^A$	P	P	P	--	N	N	N	N
$T_{S1,1}^B$	N	N	N	N	--	P	P	P
$T_{S1,2}^B$	N	N	N	N	P	--	P	P
$T_{S2,1}^B$	N	N	N	N	P	P	--	P
$T_{S2,2}^B$	N	N	N	N	P	P	P	--

Figure 5. The illustration of whether each track pair is positive or negative. “P” and “N” represent positive and negative, respectively.

For a training batch with N track segment pairs, the label of pair (Seg_i, Seg_j) is denoted as $y_{i,j}$, as below:

$$y_{i,j} = \begin{cases} 1, & \text{for positive pairs} \\ 0, & \text{for negative pairs} \end{cases} \quad (14)$$

Let $V_i = \text{Encoder}(Seg_i)$ and $V_j = \text{Encoder}(Seg_j)$ be the output of the two transformer encoders corresponding to Seg_i and Seg_j , respectively. Since the architecture of the encoder in TAM is the same as that in TCM, $V_i \in \mathbb{R}^{S_L \times D_{model}}$ and $V_j \in \mathbb{R}^{S_L \times D_{model}}$. In this study, Euclidean distance $\|V_i - V_j\|_2$ is adopted to quantify the similarity between Seg_i and Seg_j , and a smaller distance signifies that the two segments are more likely to belong to the same target. Therefore, the total loss l_{FAM} of a batch is calculated as follows:

$$l_{FAM} = \frac{1}{2N} \sum_{i=1}^N [y_{i,j} \cdot \|V_i - V_j\|_2 + (1 - y_{i,j}) \cdot \max(0, m - \|V_i - V_j\|_2)] \quad (15)$$

where m is a margin parameter to ensure effective separation of negative pairs.

2.3.2. Utilization of Cross-Entropy Loss Function

In this study, the track association task is formulated as a binary classification problem, which aims to determine whether two track segments originate from the same target or distinct targets. As a standard loss function for binary classification tasks, cross-entropy loss quantifies the discrepancy between predicted classification outputs and ground-truth

labels of samples [41]. A loss smaller than a predefined threshold signifies the two track segments belong to the same target. Network optimization implemented with cross-entropy loss presents prominent merits, including stable training gradients, rapid convergence and strong feature discriminability. This optimization strategy can also adaptively fit the nonlinear distribution characteristics of tracks in complex operational scenarios.

When employing the cross-entropy loss function, given an input sample, i.e., two track segments denoted, respectively, as Seg_i and Seg_j , the two segments are first encoded via two Transformer encoders with shared weights. Subsequently, the feature difference between Seg_i and Seg_j is calculated. The derived feature differences are then mapped to a category label through a fully connected layer. Finally, the obtained label is further converted into probability values using the sigmoid function. The detailed calculation procedure is presented as follows:

$$\hat{y}_i = \text{sigmoid}(\text{reshape}(|\text{Trans}(Seg_i) - \text{Trans}(Seg_j)|)W) \in \mathbb{R}^{1 \times 1} \quad (16)$$

where $\text{Trans}(\cdot)$ denotes the computational output of the Transformer model, $|\cdot|$ represents the absolute value operation, the function $\text{reshape}(\cdot)$ is adopted to transform the dimension of the obtained difference values into $\mathbb{R}^{1 \times (S_L \times S_{model})}$, $W \in \mathbb{R}^{(S_L \times S_{model}) \times 1}$ is a trainable weight matrix used to derive the classification result of the corresponding input sample, and the $\text{sigmoid}()$ function further maps the classification result to the interval (0,1).

For a batch of input samples, the overall cross-entropy loss is calculated via the formula presented below:

$$L_{FAM} = -\frac{1}{N} \sum_{i=1}^N [y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)] \quad (17)$$

where N denotes the total number of input samples within a single batch. y_i stands for the ground-truth label of the i_{th} input sample, and $\hat{y}_i$ refers to the corresponding predicted label. Specifically, label 1 indicates that the two trajectory segments originate from the same target, while label 0 means that they belong to different targets.

2.4. Track Fusion Module

Benefiting from the outputs of TAM, we can easily determine whether any two track segments belong to the same target. On this basis, pairwise fusion is then iteratively conducted for all track segments derived from the same target in order to construct the final fused track of each target. As a powerful architecture, the Transformer has been extensively applied in autonomous driving [35,42], pedestrian monitoring [43] and UAV video-based aerial target tracking [33], achieving remarkable performance in practical tasks. Accordingly, this study adopts the Transformer to design a Track Fusion Module (TFM) for track segment fusion.

As illustrated in Figure 6, the proposed TFM consists of two Transformer-based encoders and one Transformer-based decoder. The two encoders separately encode two track segments, after which feature fusion is performed. The decoder subsequently generates the fused track results based on the fused features. Concretely, an association matrix is first constructed to denote whether any two track segments originate from the same target, as shown in Figure 7. Taking the track segment Seg_1 as an example, we select another track segment Seg_2 that belongs to the identical target. Both Seg_1 and Seg_2 are then encoded by the two Transformer encoders. These two encoders adopt the same network architecture as the encoder in the TCM, while their parameters are not shared. Prior to being input into the Transformer decoder, the outputs of the two encoders are fused. Specifically, the outputs of the two Transformer encoders are first concatenated along the temporal dimension, followed by flattening across both temporal and feature dimensions.

Two fully connected layers are then applied to conduct a nonlinear transformation, enabling sufficient feature interaction and fusion between the two track segments. The processed features are finally reshaped to the original dimension for subsequent decoding. The decoder outputs the fused track results at each time step sequentially. After this, if there remain unprocessed track segments for the current target, the above fusion procedure is repeatedly deployed on the existing fused result and the remaining segments until all track segments of the target are fully fused. The final output of the encoders is regarded as the ultimate fused track of the target. It is notable that the rows corresponding to the processed track segments are removed from the association matrix before starting the track fusion process for the next target. The above track fusion process is shown in Algorithm 1 in detail.

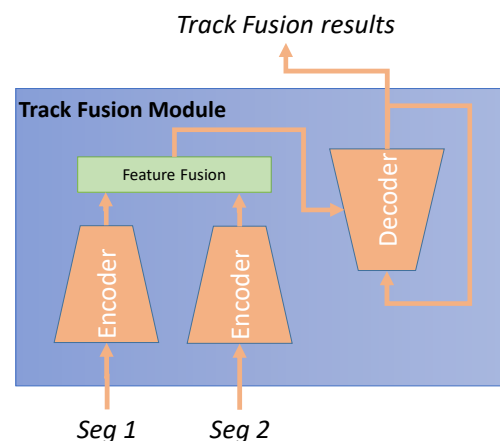


Figure 6. The architecture of the proposed TFM.

	Seg. 1	Seg. 2		Seg. N
Seg. 1	No	Yes	...	No
Seg. 2	yes	No	...	Yes
.....	...	...	⋮	⋮
Seg. N	No	yes	...	No

Figure 7. The association matrix outputted by TAM. “Yes” means the two track segments originate from the same target, and “no” indicates otherwise.

It should be noted that during the training phase, only paired track segments from the same target are fed into the proposed TFM, and iterative processing of all track segments of a target is not required. In contrast, the TFM is applied iteratively to multiple track segments of each target in the inference phase to obtain the final fused track.

In this study, the longitude, latitude, speed and heading of the targets are selected as the original features. Generally, the value ranges of these features are quite different. Therefore, direct normalization of the raw features will make it impossible to recover the actual physical parameters of the target from the outputs of the Transformer decoder. However, simply summing the original loss terms across different targets will inevitably result in unbalanced optimization if the normalization is not employed. Specifically, features with larger numerical magnitudes dominate the overall loss gradient and hinder the

learning of features with smaller ranges. Such numerical scale mismatch greatly deteriorates the joint optimization performance and further leads to partial convergence failure in multi-dimensional prediction tasks.

Algorithm 1: The Track fusion Process

Definition:

- Let AM represent the association matrix that has m rows and n columns.
- Let $TFMencoder()$ stand for the process of the Transformer-based encoder.
- Let $TFMdecoder()$ stand for the process of the Transformer-based decoder.
- Let $DeInput$ stand for the input of the Transformer-based decoder
- Let $FC()$ denote the process of a fully connection layer

Input:

- Track segments that have been completed by TCM.

Output:

- The final track fusion results of each target.
- 1, Seg_1 = the completed track segment corresponding to the first row of AM .
 - 2, $MarkedRows = 1$;
 - 3, for ($i = 2$; Seg_i and Seg_1 originate from the same target & $i \leq n$; $k++$) do
 - 4, $F_1 = TFMencoder(Seg_1)$;
 - 5, $F_2 = TFMencoder(Seg_i)$;
 - 6, $DeInput = \text{reshape}(FC(FC(\text{reshape}(\text{concatenate}(F_1, F_2)))))) \in \mathbb{R}^{S_L \times S_{model}}$;
 - 7, $FusedTrack = TFMdecoder(DeInput)$;
 - 8, $Seg_1 = DeInput$;
 - 9, $MarkedRows = MarkedRows \cup i$;
 - 10, end
 - 11, Remove the rows from AM whose row index has been marked by $MarkedRows$.
 - 12, Output the $FusedTrack$ as the final fused track of the current target.
 - 13, if $AM \neq \emptyset$ do
 - 14, go to step 1
 - 15, endif
-

To address this issue, we adopt homoscedastic uncertainty-based adaptive weighting [44] to dynamically balance multi-task losses during training. The overall loss function is defined as follows:

$$L_{TFM} = \sum_{i=1}^N \frac{1}{2\sigma_i^2} L_i(\hat{y}_i, y_i) + \log(1 + \sigma_i^2) \quad (18)$$

where N denotes the number of predicted features, $\hat{y}_i$ and y_i represent the predicted and ground-truth values of the i th feature, respectively, L_i is the task-specific loss function tailored for the corresponding feature distribution and value scope, and σ_i denotes the learnable uncertainty parameter for each feature prediction task. The network automatically adjusts the contribution of each loss term by optimizing task-dependent uncertainty, thereby preventing any single objective from dominating the optimization process. The regularization term $\log(1 + \sigma_i^2)$ is incorporated to avoid trivial solutions and ensure numerical stability. Compared with naive multi-loss superposition, the proposed composite loss strategy fully adapts to heterogeneous feature scales and statistical characteristics, realizes consistent gradient update magnitudes across all prediction branches, and significantly improves the overall convergence quality and generalization performance of multi-output neural networks. For the heading, the Circular Angle Loss is adopted, which is calculated as the following:

$$L_{heading} = 1 - \cos\left(\frac{\pi}{180}(\hat{y} - y)\right) \quad (19)$$

For longitude and latitude, the SmoothL1 loss function is employed, and its expression is given by the following:

$$L_{position} = \begin{cases} 0.5(\hat{y} - y)^2, & |\hat{y} - y| < 10 \times e^{-4} \\ |\hat{y} - y| - 5 \times e^{-4}, & |\hat{y} - y| \geq 10 \times e^{-4} \end{cases} \quad (20)$$

For other features, which are large-scale continuous variables with wide value ranges, we employ the logarithmic mean squared error (LogMSE). This loss function compresses extreme numerical magnitudes and optimizes relative prediction deviation instead of absolute error, as computed below:

$$L_{LogMSE} = (\log(\hat{y} + \epsilon) - \log(y + \epsilon))^2 \quad (21)$$

where $\epsilon = 10^{-6}$ is introduced to avoid numerical singularity when the ground-truth value is zero.

2.5. Model Training

During the early training phase, the track points completed by the TCM are usually not the same as the real track points collected by sensors. Similarly, the proposed TAM may occasionally produce incorrect association judgments for track segments. Therefore, if the output of TCM is directly fed into TAM, and subsequently the output of TAM is further transmitted to the proposed TFM, the performance of TAM and TFM will be artificially degraded since the incorrect input data will affect the accuracy of track association and fusion results, and result in longer training time overhead. To address this issue, this study adopts a two-stage training strategy. Concretely, in the early training phase, we perform module-wise training to enable fast and stable convergence of individual sub-modules. After the performance of TCM, TAM and TFM are sufficiently improved, we further conduct holistic joint training to globally refine the entire network. This two-stage training strategy can usually shorten the overall training time overhead and maintain competitive or even superior model performance compared with pure joint training. Furthermore, to enhance the generalization performance of the proposed TMTAFF, a data augmentation strategy is employed. In detail, partial track segments are randomly truncated at arbitrary time steps from long continuous track segments to enrich the diversity of training samples and mitigate potential overfitting risks.

In the proposed TCM, TAM and TFM, the input track length is set as $S_L = 64$, the hidden layer size is set as $S_{model} = 128$, the number of Transformer blocks is set as $N = 6$, and each attention mechanism contains 8 heads. During the training phase, the dataset is trained for 80 epochs. In addition, we employ the LeakyRelu function [45] to activate neurons in order to retain negative values, so as to take full use of the useful information from the input. Moreover, the weights and biases of the TMTAFF are initialized using random values with a fixed random seed (seed = 20) to ensure reproducibility and updated using an Adam optimizer [46] with $\beta_1 = 0.85$, $\beta_2 = 0.99$, and $\epsilon = 10^{-6}$. Furthermore, to avoid over-fitting, a dropout layer with a dropout probability set to 0.1 was added before the final output layer. In addition, the learning rate is initialized as 0.002 and then decreased following a half-cycle cosine after 20 epochs of training as a regularization method. Intermediate model snapshots were taken every 50 iterations with a batch size of 64, and the snapshot that performed best on the validation test was selected as the final model. To obtain the optimal parameter configuration mentioned above, we set several groups of parameter values based on the parameter values of the existing relevant literature. After iterative experimental tuning, the parameter configuration yielding optimal model performance was selected and adopted for this study.

Experiments for the TCM, TAM and TFM are implemented on a 64-bit high-performance workstation that runs the Ubuntu 22.04.3 LTS operating system and is equipped with 64 GB RAM, an Intel Xeon E5-2690 central processing unit running at 3.7 GHz, and an NVIDIA GeForce RTX 4070 graphics card. In terms of software deployment, the PyTorch 2.0 deep learning framework and Python 3.8 are adopted to build, train and optimize the TMTAFF.

3. Experiment Evaluation

3.1. Experimental Setup

3.1.1. Dataset and Data Preprocessing

To evaluate the effectiveness of the TMTAFF, the Multi-source Track Association Dataset (MTAD) provided by Naval Aviation University is employed [47], which is based on automatic identification system (AIS) trajectory data of global ships. The MTAD contains more than 1 million tracks. Preprocessing steps like grid division, automatic interruption, noise injection, error adding, etc., are conducted on AIS trajectory data, where the data preprocessing settings are strictly consistent with those adopted in the existing reference literature [48]. Such unified preprocessing settings eliminate performance biases caused by inconsistent data processing strategies, thereby ensuring a fair and credible comparison between the proposed model and baseline methods. Afterwards, we randomly select 3000 unique scenes and then randomly divide them at a ratio of 0.7:0.1:0.2 for training, validation, and test, respectively. For each scene, the number of tracks is not fixed, ranging from several to hundreds. In addition, the ground truth is the original AIS trajectory data, and the track segments collected by two sensors are actually derived from the original AIS trajectories by conducting a series of transformation operations, including trajectory translation, segment partitioning, random noise injection, data interpolation, and so on. Such artificial modification and reconstruction of original AIS data effectively simulate complex and diverse real-world track observation scenarios, which further improve the robustness and generalization capability of the TMTAFF. The sampling periods of the track segments reported by the two sensors are 10 s and 20 s. Each track segment utilizes longitude, latitude, speed, and heading to characterize the motion of the target.

Figure 8 illustrates the track segments of four typical scenes in MTAD. These scenes show quite different characteristics; some scenes have a few targets, while others have many densely clustered targets. Moreover, the motion patterns of these targets differ significantly: some move straight, some make turns, while others follow highly unpredictable trajectories. In addition, these trajectories differ noticeably in length. In particular, MTAD can cover ITSA and MSTSA tasks. Therefore, MTAD features diverse data characteristics, which render it an excellent benchmark for track association and fusion research.

It should be noted that the MTAD dataset is divided into distinct scenes based on geographic locations. Track segments from different scenes generally correspond to different targets. Therefore, this study does not shuffle and mix track segments across all scenes when constructing the training set, the validation set and the test set, since random mixing would lead to a situation where most track segments within a single batch cannot be associated, which increases computational overhead and fails to improve the performance of TMTAFF in target association and fusion. Instead, we concatenate data from different scenes in sequence. For each training iteration, batches of track segments are selected following the original order of the dataset. During testing, the data of each scene is evaluated separately, and the final association and fusion results are generated for every individual scene.

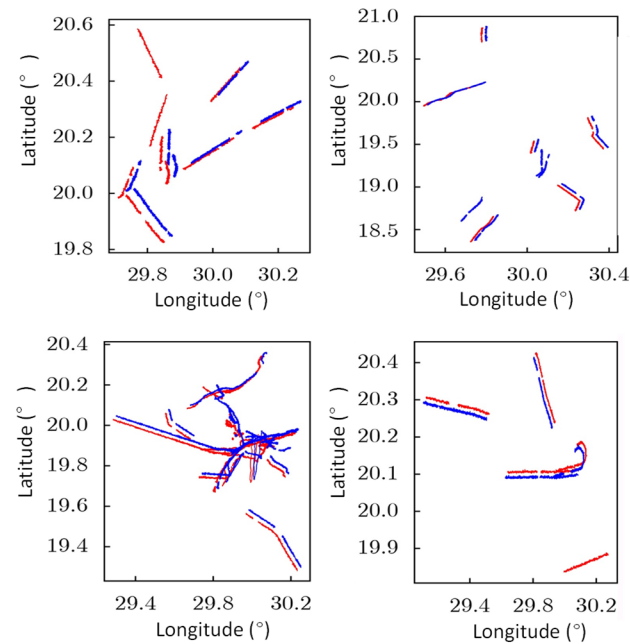


Figure 8. Illustrations of the track segments in typical scenes. Different colors represent trajectories obtained from different AIS devices.

3.1.2. Metrics

In this work, the TCM is employed to reconstruct incomplete track segments, serving as an auxiliary module for track association and track fusion. Therefore, in this study, we do not evaluate the performance of TCM independently. Instead, the capability of TCM is reflected via the performance evaluation of TAM and TFM.

For the track association task, the number of actual track segment pairs in a given scene s is denoted as T_{AP}^s ; similarly, the number of track segment pairs produced by the TAM in the scene s is denoted as O_{AP}^s . To evaluate the performance of TAM, the association accuracy of the track segments in scene s is defined as P_{CA}^s , as below:

$$P_{CA}^s = \frac{|O_{AP}^s \cap T_{AP}^s|}{|T_{AP}^s|} \times 100\% \quad (22)$$

where $|\cdot|$ indicates the number of association pairs in the set and $\cap$ represents the intersection operation between sets. Likewise, the association fault rate of the track segments in the scene s is defined as P_{FA}^s , as below:

$$P_{FA}^s = \frac{|O_{AP}^s - O_{AP}^s \cap T_{AP}^s|}{|O_{AP}^s|} \times 100\% \quad (23)$$

Since the dataset contains multiple scenes, we further define the average association accuracy mP_{CA} and the average association fault rate mP_{FA} for all scenes, respectively, as follows:

$$mP_{CA} = \frac{1}{S} \sum_{s=1}^S P_{CA}^s \quad (24)$$

$$mP_{FA} = \frac{1}{S} \sum_{s=1}^S P_{FA}^s \quad (25)$$

where S represents the quantity of scenes. Additionally, as each scene contains a distinct number of targets, to mitigate the corresponding effects, the average weighted association accuracy mwP_{CA} and average weighted association fault rate mwP_{FA} are formulated as below:

$$mwP_{CA} = \sum_{s=1}^S \frac{|O_{AP}^s|}{O_{AP}} P_{CA}^s \quad (26)$$

$$mwP_{FA} = \sum_{s=1}^S \frac{|O_{AP}^s|}{O_{AP}} P_{FA}^s \quad (27)$$

where $O_{AP} = \sum_{s=1}^S |O_{AP}^s|$.

For the track segment fusion task, the mean error (ME) and error standard deviation (ESD) are adopted as metrics to quantify the track fusion performance. For a target t_i in scene s , ME and ESD are formulated, respectively, as follows:

$$ME_{t_{i,s}} = \frac{1}{n} \sum_{i=1}^n \varepsilon_i - \hat{\varepsilon}_i \quad (28)$$

$$ESD_{t_{i,s}} = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (\varepsilon_i - ME_{t_{i,s}})^2} \quad (29)$$

where n is the number of track points of t_i , and $\hat{\varepsilon}_i$ and ε_i represent the predicted and real feature values of the i th track point, respectively. Furthermore, since the dataset consists of a larger number of scenes and each scene contains multiple targets, two new metrics, namely average mean error and average error standard deviation, are defined to quantify the performance of TFM comprehensively, as below.

$$mME = \frac{1}{S} \sum_{s=1}^S \frac{1}{N} \sum_{i=1}^{i=N} ME_{t_{i,s}} \quad (30)$$

$$mESD = \frac{1}{S} \sum_{s=1}^S \frac{1}{N} \sum_{i=1}^{i=N} ESD_{t_{i,s}} \quad (31)$$

where N represents the number of targets in scene s . In this study, longitude, latitude, speed, and heading are utilized as the features of the targets; therefore, we compute mME and $mESD$ for each of the four features, respectively, and the corresponding results are denoted as mME_{lo} , mME_{la} , mME_{sp} , mME_{he} and $mESD_{lo}$, $mESD_{la}$, $mESD_{sp}$, $mESD_{he}$, respectively.

3.1.3. Baselines

Track Segment Association

For the track segment association problem, we select three traditional methods and three neural network-based methods as baselines. Traditional methods depend on the prediction of the target tracks and subsequently assessing the degree of similarity between two tracks. If the obtained distance or the similarity of two tracks meets the predefined criteria, the two tracks are then regarded as originating from the same target. Approaches based on neural networks are capable of learning high-level representations from tracks and yielding track segment association results end-to-end. Owing to their excellent generalization across different scenes, they represent a major research trend for subsequent studies. The six baselines are presented in turn, as below.

- MF 2-D TSA: The Multi-Frame two-dimensional track segment association method (MF 2-D TSA) proposed in [11] is a traditional method that aims to solve the ITSA problem. It first releases incorrectly associated measurements by going backward and forward in time along the old and young track segments, respectively, and then performs the prediction or retrodiction using the combination of constant velocity

and coordinated turn models. By exploring various turning rates, if the distance between the predicted state of the old track and the state of the young track meets specific criteria, this correlation is deemed successful.

- **ED-based TSA:** The Euclidean Distance-based track segment association (ED-based TSA) proposed in [48] aims to tackle the MSTSA problem by judging whether the distance between two track segments of two different sensors is less than a predefined threshold. To compute Euclidean distance, all trajectory samples must be aligned to the same length. This constraint greatly reduces the method's versatility. When target trajectories show prominent length discrepancies, the cropping operation inevitably discards a large amount of valuable data.
- **DTW-based TSA:** In order to overcome the shortcomings of Euclidean Distance-based TSA, we also design a baseline method named DTW-based TSA by employing the dynamic time warping (DTW) [49] as the distance measurement criterion of multi-source tracks to tackle the MSTSA problem. DTW can align a point of one trajectory with one or more consecutive points of another trajectory, which makes it suitable for time series, especially for track segments whose data length is not fixed and unpredictable.
- **TCN-GRU-GNN:** TCN-GRU-GNN comprises dual-channel temporal convolutional networks (TCNs), gated recurrent units (GRUs), and a global nearest neighbor (GNN)-based association module [50]. The TCN is employed to independently extract the deep-level temporal features from asynchronous radar and infrared track data. The GRU classifier is employed to rectify the insensitivity of the TCN to temporal data edge information and short-term dependencies. The GNN-based association module is designed to refine the scene perception for the association task. TCN-GRU-GNN is suitable for track association tasks that have two distinct sensor modalities.
- **CA-LSTM:** CA-LSTM in [51] establishes a track segment association method by utilizing collaborative optimization of multi-scale dilated convolution, characteristic-aware attention (CA), and gated adaptive LSTM, which realizes the dynamic selection of characteristic dimensions and dynamic perception of time dimensions and solves the challenges of spatiotemporal characteristic fusion and low utilization of multi-dimensional data in track segment association. CA-LSTM reduces missed detections while controlling false detections, verifying its applicability in complex environments.
- **TSA-cTFER:** TSA-cTFER [7] utilizes contrastive learning and a Transformer encoder to accurately assess trajectory similarity through learned representations by computing distances between high-dimensional feature vectors. Additionally, to adapt to dynamic association scenes where targets continuously appear or disappear, TSA-cTFER employs a two-stage online algorithm to manage tracks that appear or disappear at any time. Notably, TSA-cTFER adopts different strategies to cope with ITSA tasks and MSTSA tasks, which leads to limited generalizability.

Track Segment Fusion

For the track segment fusion problem, we select three traditional methods and two neural network-based methods as baselines. The five baselines are presented in turn, as below.

- **TFWFB:** Track-to-track fusion with feedback (TFWFB) proposed in [52] is a multi-sensor track-to-track fusion algorithm based on traditional iterative Joint Integrated Probabilistic Data Association (iJPDA). It feeds the fusion results back to the fusion center in order to make each local sensor tracker able to continuously follow all true

targets via feedback system tracks. Compared with the other two fusion architectures, namely, track-to-track fusion without memory (TFWOM) and track-to-track fusion with memory (TFWM), TFWFB usually exhibits better track fusion performance in cluttered environments.

- **T2TF-DC:** T2TF-DC in [53] is a novel track fusion solution that imports a dynamic consensus protocol to make every local tracker able to perceive the targets estimated by neighboring sensors. Concretely, each sensor first independently processes its own observations and performance estimation using the nearly coordinate turn model with an extended Kalman filter (EKF). Next, the estimates generated by each sensor are shared with neighboring sensors to improve the quality of the generated tracks. Afterwards, the tracks generated are shared once again in order to produce the final global fused tracks; in other words, these sensors follow a global consensus protocol.
- **MHT-EnKF:** MHT-EnKF substitutes the traditional Kalman filter in the multiple hypotheses tracking (MHT) by using a modified ensemble Kalman filter (EnKF) to cope with the high nonlinearity in multiple target tracking and multi-sensor fusion problems [18]. The utilization of the modified EnKF makes MHT-EnKF suitable for scenarios where the acceleration and velocity of the targets comply with nonlinear kinematic equations in each time interval.
- **PTFM:** The parallel track fusion model (PTFM) proposed in [19] employs an attention mechanism and a long short-term memory (LSTM) network to extract the spatiotemporal features from asynchronous target data. Afterwards, the extracted features are combined and further processed by an adaptive multitask loss mechanism to adaptively select more reasonable feature weights, by which the combined features can be reconstructed into high-precision fused tracks.
- **DeepAF:** DeepAF is a Transformer-based end-to-end track segment fusion model [34]. It first utilizes a basic Transformer unit (named DeepAF-B) to extract features from track segments, and then it employs two Transformer units (named DeepAF-P and DeepAF-V) to separately produce the estimated position and velocity of the fused tracks. In this study, the fused tracks produced by TMTAFF are characterized by four parameters, i.e., longitude, latitude, speed, and heading; therefore, we trained four Transformer units to respectively produce these four parameters so as to facilitate comparative analysis when reproducing this baseline method.

3.2. Evaluation Results

In this section, we first evaluate the performance of TMTAFF in track association and track fusion in order. Then, we conduct comprehensive ablation studies to figure out the contributions of the key components of TMTAFF. All experiments were carried out under identical computational conditions introduced in Section 2.5 to enable valid comparative analysis.

3.2.1. Performance of Track Segment Association

Since some of the baselines introduced in Section Track Segment Association are not versatile for the ITSA task and the MSTSA task simultaneously, we evaluate the baselines based on different types of tasks separately.

Table 1 quantitatively summarizes the experimental outcomes derived from diverse baselines for the ITSA task. All four neural-network-driven approaches yield superior track association performance relative to the MT 2-D TSA that is fundamentally built upon the conventional constant velocity (CV) kinematic model, demonstrating that it is very difficult for rudimentary kinematic modeling schemes to precisely characterize real-world

target trajectory evolutions. Although the TCN-GRU-GNN architecture achieves prominent performance improvements over MF 2-D TSA, its overall accuracy is inferior to the three competing alternatives: CA-LSTM, TSA-cTFER and the proposed TMTAFF. Such performance disparity is primarily attributed to the adoption of multi-scale feature extraction modules embedded within the latter three methods, which facilitate the excavation of discriminative long-range contextual clues embedded in sequential track data. Specifically, TSA-cTFER and TMTAFF integrate Transformer-based multi-head attention modules, a well-validated paradigm for multi-scale feature mining, which enables the model to comprehensively quantify the contribution weight of disparate track sampling points toward the prediction of values of every single timestep. Relative to TSA-cTFER, the developed TMTAFF obtains performance increments of 2.72% and 5.46% regarding the mP_{CA} and mwP_{CA} metrics separately, alongside 3.92% and 2.94% reductions in false association indicators mP_{FA} and mwP_{FA} . When benchmarked against the MF 2-D TSA, TMTAFF explicates further performance gains reaching 20.35% and 20.02% for mP_{CA} and mwP_{CA} , while cutting the corresponding false association errors by 7.87% and 7.33% on mP_{FA} and mwP_{FA} . The aforementioned performance superiority can be reasonably explained by the embedded TCM trajectory completion component within TMTAFF, which completes the interrupted track segments, thereby extends the valid temporal observation window, and empowers the model to conduct cross-segment track matching over prolonged time intervals to judge whether disjoint track fragments originate from an identical target.

Table 1. Comparison of the performance of different methods on ITSA task.

Baselines	mP_{CA} (%)	mwP_{CA} (%)	mP_{FA} (%)	mwP_{FA} (%)
MF 2-D TSA	74.23	71.69	11.46	13.81
TCN-GRU-GNN	86.86 ± 0.48	79.34 ± 0.41	8.73 ± 0.26	10.23 ± 0.35
CA-LSTM	87.72 ± 0.42	81.29 ± 0.52	8.37 ± 0.19	10.87 ± 0.28
TSA-cTFER	91.86 ± 0.36	86.25 ± 0.47	7.51 ± 0.24	9.42 ± 0.33
TMTAFF (ours)	94.58 ± 0.21	91.71 ± 0.24	3.59 ± 0.07	6.48 ± 0.09

The comparative performance of the state-of-the-art baselines on the MSTSA task is quantitatively summarized in Table 2. Benefiting from the Dynamic Time Warping algorithm, the DTW-based TSA mitigates erroneous distance calculation stemming from spatial coordinate misalignment, an inherent defect induced by inconsistent sampling lengths of target observations across heterogeneous sensors. Such optimization yields performance increments of 4.5% for mP_{CA} and 5.82% for mwP_{CA} against the ED-based TSA. Nevertheless, the performance of the DTW-based TSA is still lower than that of all four neural network-based approaches, empirically verifying the superior feature extraction, feature representation, and data modeling capabilities obtained from deep learning architectures. Furthermore, it is observed that all three deep learning baselines, i.e., TCN-GRU-GNN, CA-LSTM and TSA-cTFER, show marginal accuracy degradation (approximately 2%) when compared with their performance produced for the ITSA task. This performance regression can be attributed to divergent data distribution patterns of target measurements collected from multi-source sensors, which impose extra difficulties on the discrimination of track fragments originating from identical targets. Despite the proposed TMTAFF also suffering mild performance drops under multi-sensor test conditions, its decay margins are restricted to merely 0.63% (mP_{CA}) and 0.33% (mwP_{CA}). This robust anti-degradation property of TMTAFF can be accounted for by the following two core module designs. First, the designed TCM supplements incomplete track segments and unifies asynchronous sampling timestamps among cross-sensor measurements, which is helpful

for effectively eliminating performance penalties caused by track fragmentation and unsynchronized data acquisition. Second, the synergy of the Siamese network backbone paired with contrastive loss strengthens discriminative modeling for disparity quantification between paired track samples. Overall, the proposed TMTAFF achieves mP_{CA} and mwP_{CA} scores of 94.58% and 91.71%, alongside mP_{FA} and mwP_{FA} values of 3.59% and 6.48%. Compared with ED-TSA, the TMTAFF produces prominent performance gains of 17.13%, 16.81%, 7.25% and 6.76% on mP_{CA} , mwP_{CA} , mP_{FA} and mwP_{FA} , respectively. When benchmarked against TSA-cTFER, the TMTAFF still retains gains of 4.07%, 3.81%, 4.65% and 3.71% on the four metrics. The above numerical contrasts comprehensively demonstrate the outstanding environmental adaptability of TMTAFF toward complicated multi-sensor multi-target operational scenarios.

Table 2. Comparison of the performance of different methods on MSTSA task.

Baselines	mP_{CA} (%)	mwP_{CA} (%)	mP_{FA} (%)	mwP_{FA} (%)
ED-based TSA	76.78	74.57	11.72	13.63
DTW-based TSA	81.28	80.39	10.70	12.83
TCN-GRU-GNN	84.56 ± 0.42	78.07 ± 0.38	9.21 ± 0.39	11.28 ± 0.49
CA-LSTM	85.23 ± 0.32	84.91 ± 0.45	9.83 ± 0.32	10.75 ± 0.41
TSA-cTFER	89.84 ± 0.41	87.57 ± 0.37	9.12 ± 0.27	10.58 ± 0.33
TMTAFF (ours)	93.91 ± 0.26	91.38 ± 0.21	4.47 ± 0.15	6.87 ± 0.18

3.2.2. Performance of Track Segment Fusion

Table 3 lists the quantitative performance of all comparative baselines for track segment fusion. As revealed from experimental results, the developed TMTAFF achieves optimal overall accuracy with $6.24 \times 10^{-4^\circ}$, $5.88 \times 10^{-4^\circ}$, 1.03 kn and 0.84° for mME_{lo} , mME_{la} , mME_{sp} and mME_{he} separately, surpassing the TFWFB by margins of $17.19 \times 10^{-4^\circ}$, $20.08 \times 10^{-4^\circ}$, 2.88 kn and 2.83° , correspondingly. It is notable that the conventional baselines (namely, TFWFB, T2TF-DC and MHT-EnKF), essentially based on Probabilistic Data Association (PDA) and the Kalman filter, yield larger mean errors than neural network-based approaches. Such performance gaps mainly originate from rigid mathematical prerequisites embedded in classic filtering algorithms: target motions are presumed to follow constant or constant-acceleration kinematic models, and measurement noise complies with a zero-mean Gaussian distribution. However, intensive target maneuvers, heterogeneous multi-sensor input, outlier contamination, and dense clutter prevail in practical complicated environments, making actual motion characteristics deviate sharply from pre-set assumptions. In contrast, data-driven neural networks are capable of removing predefined kinematic and noise constraints and learning intricate motion patterns and noise statistics automatically from abundant training samples. Moreover, DeepAF and the proposed TMTAFF attain remarkable accuracy improvements over PTFM. The underlying cause lies in the superior fitting capability of Transformer units without long-range information attenuation during sequential modeling, whereas the LSTM gradually loses early track information as time prolongs. Beyond that, the multi-head attention mechanisms embedded in the Transformer enable independent feature extraction across diverse feature subspaces. By contrast, the attention module adopted by PTFM only generates a unified global weighting coefficient, limiting its capability to process multi-dimensional sequential track data properly. In particular, in comparison with Transformer-based DeepAF, the proposed TMTAFF further reduces mean errors by $5.66 \times 10^{-4^\circ}$, $7.32 \times 10^{-4^\circ}$, 0.92 kn and 0.56° in terms of mME_{lo} , mME_{la} , mME_{sp} and mME_{he} , respectively, which can be attributed to the embedded TCM module that completes interrupted track segments and thereby supplies more useful information for the subsequent track fusion step. Furthermore, TMTAFF attains the minimum standard deviation of mean error ($2.64 \times$

10^{-4° , $2.39 \times 10^{-4^\circ}$, 0.47 kn and 0.42° in terms of $mESD_{lo}$, $mESD_{la}$, $mESD_{sp}$ and $mESD_{he}$, respectively) compared with all the other state-of-the-art baselines, demonstrating outstanding stability and smoothness of the final fused tracks.

Table 3. Comparison of the performance of different methods on track segment fusion.

Baselines	mME_{lo} (10^{-4°)	mME_{la} (10^{-4°)	mME_{sp} (kn)	mME_{he} ($^\circ$)	$mESD_{lo}$ (10^{-4°)	$mESD_{la}$ (10^{-4°)	$mESD_{sp}$ (kn)	$mESD_{he}$ ($^\circ$)
TFWFB	23.43	25.96	3.91	3.67	8.94	7.87	2.55	2.38
T2TF-DC	27.92	28.08	3.29	2.73	8.12	6.78	2.13	2.07
MHT-EnKF	18.49	17.08	2.6	2.17	5.24	4.81	1.49	1.54
PTFM	17.07 ± 0.72	18.29 ± 0.67	2.84 ± 0.16	2.46 ± 0.22	6.20 ± 0.37	5.83 ± 0.41	1.82 ± 0.28	1.97 ± 0.23
DeepAF	11.9 ± 0.35	13.2 ± 0.23	1.95 ± 0.19	1.40 ± 0.24	4.31 ± 0.19	4.59 ± 0.21	1.28 ± 0.14	1.17 ± 0.12
TMTAFF (ours)	6.24 ± 0.13	5.88 ± 0.11	1.03 ± 0.06	0.84 ± 0.05	2.64 ± 0.07	2.39 ± 0.08	0.47 ± 0.05	0.42 ± 0.04

3.2.3. Ablation Study

In this section, ablation experiments are performed in order to investigate the effects of the key components of TMTAFF.

Effects of the proposed TCM: Figure 9 visualizes the effects of TCM on the performance of TAM on TSA tasks and MSTSA tasks. Across all cases, the TMTAFF equipped with TCM achieves superior mP_{CA} and mwP_{CA} in comparison with their counterparts stripped of TCM, which indicates that completing the interrupted track segments is helpful for obtaining more accurate target association relations. Quantitatively, the performance gains brought by TCM are quantified as follows: for the ITSA task, mP_{CA} and mwP_{CA} significantly increase by 13.12% and 11.03%, respectively; for MSTSA tasks, these two metrics rise by 7.50% and 7.29%, correspondingly. The larger performance boost reported in the ITSA task can be attributed to the inherent discrepancy in temporal gaps between paired track fragments. In the MSTSA task, the temporal offset between two target segments remains relatively marginal, whereas the ITSA task usually shows a much longer time interval between the old and the new track segments. Under such circumstances, TCM is capable of effectively mitigating the degradation of association accuracy induced by large temporal discontinuities to a more remarkable extent. In particular, the TMTAFF with TCM embedded delivers overall improvements of 9.81% in mP_{CA} and 8.21% in mwP_{CA} when tested on the mixed scenarios that have ITSA and MSTSA tasks simultaneously. Such consistent performance gains demonstrate the strong generalization of TCM toward complicated multi-target association scenarios.

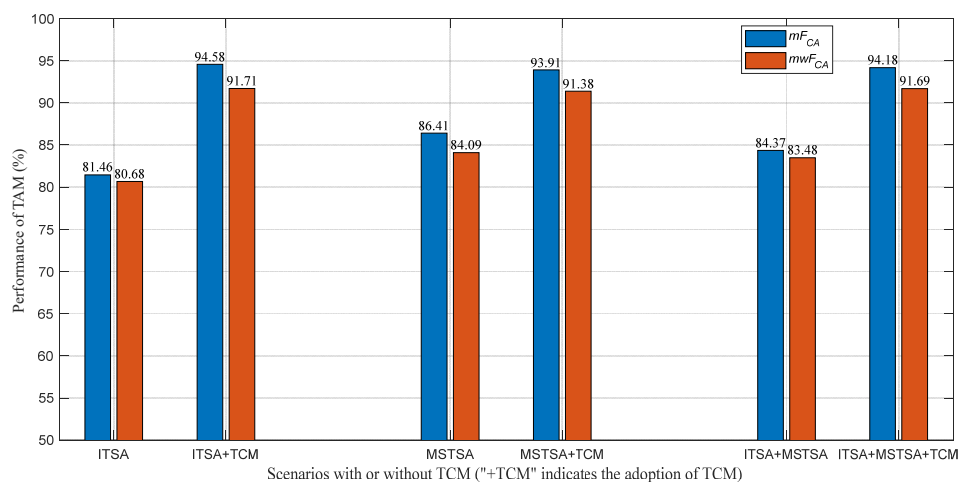


Figure 9. Effect of TCM on track segment association under different scenarios with or without TCM. “+TCM” signifies the employment of TCM.

The effects of TCM on the performance of TFM are comprehensively shown in Figure 10. Obviously, integrating the TCM yields conspicuous reductions in both the average mean error and average error standard deviation of fused tracks. Quantitatively, the TMTAFF equipped with TCM brings average mean error reductions of 4.19×10^{-4} , 5.51×10^{-4} , 0.55 kn and 0.84° for mME_{lo} , mME_{la} , mME_{sp} and mME_{he} , respectively. At the same time, the corresponding $mESD_{lo}$, $mESD_{la}$, $mESD_{sp}$ and $mESD_{he}$ shrink by 1.27×10^{-4} , 1.79×10^{-4} , 0.45 kn and 0.43° , respectively, which signifies that the systematic bias of the final fused target tracks is drastically suppressed and the trajectory smoothness of fused tracks is significantly improved. The above performance gains can be attributed to the trajectory completion capability of TCM. The supplementary track points recovered by TCM furnish the fusion model with precise temporal motion characteristics of targets, which further enables the TFM to implement more effective processing on trajectory measurements collected from the other sensors.

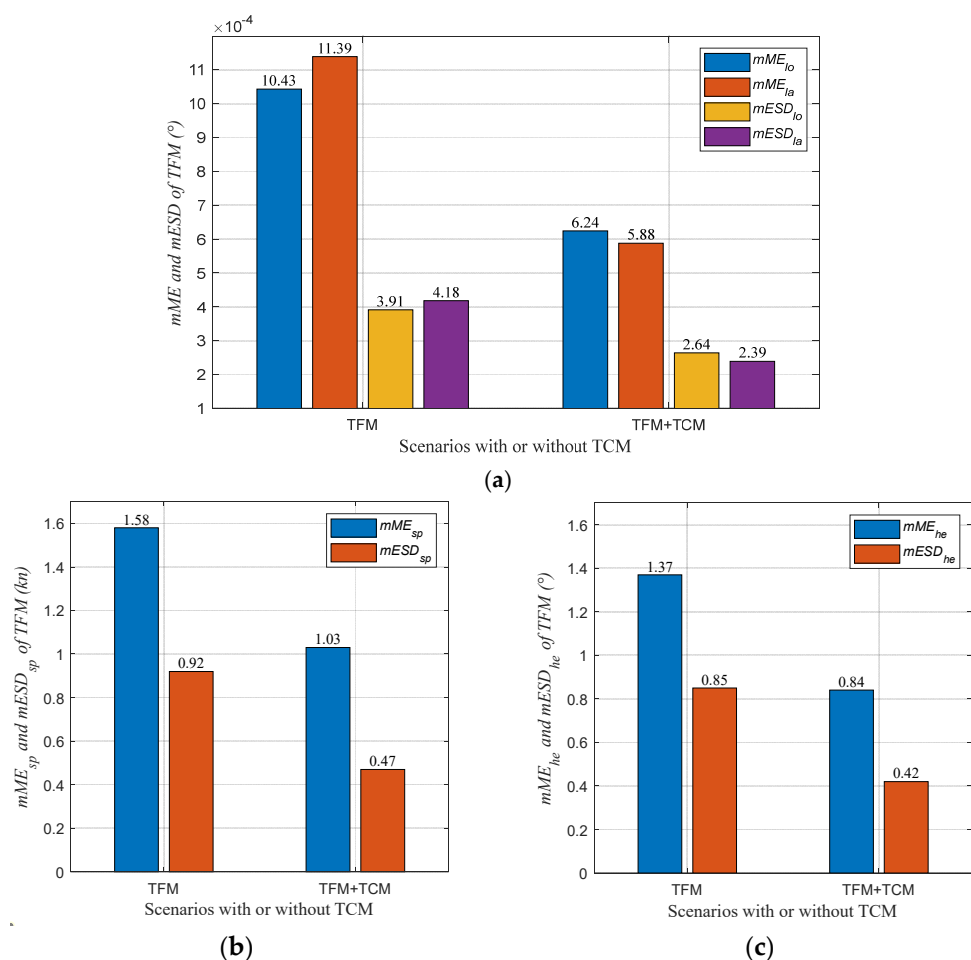


Figure 10. Effect of TCM on track segment fusion. (a) mME_{lo} , mME_{la} , $mESD_{lo}$, and $mESD_{la}$ variation for TFM with or without TCM. (b) mME_{sp} and $mESD_{sp}$ variation for TFM with or without TCM. (c) mME_{he} and $mESD_{he}$ variation for TFM with or without TCM.

Effects of the loss function of TAM: In TAM, the selection of the loss function exerts a substantial influence on the performance of track segment association. Since the outputs yielded by TAM serve as the input for the subsequent TFM, the loss function of TAM also

imposes an indirect effect on the track segment fusion accuracy of TFM. Figure 11 quantitatively compares the track association performance of TAM equipped with a contrastive loss function and a cross-entropy loss function. Substituting cross-entropy loss with contrastive loss delivers a 4.39% and 5.43% improvement in mP_{CA} and mwP_{CA} , separately. Concurrently, mP_{FA} and mwP_{FA} drop by 2.23% and 1.88%, respectively. Such statistical contrasts verify that the contrastive loss function possesses strong discriminative capability to measure the similarity degree between paired track segments. Further, Figure 12 illustrates the degradation of the final fused track quality of TFM after switching to the contrastive loss function. Clear deterioration can be observed in both the average mean error and average error standard deviation of the final fused tracks. In detail, mME_{lo} , mME_{la} , mME_{sp} and mME_{he} increase by $2.95 \times 10^{-4}^\circ$, $2.9 \times 10^{-4}^\circ$, 0.31 kn, 0.29° , respectively. For $mESD_{lo}$, $mESD_{la}$, $mESD_{sp}$ and $mESD_{he}$, their increments reach 1.73×10^{-4} , 2.54×10^{-4} , 0.19 kn and 0.25° . Overall, the performance degradation magnitude of TFM exceeds that of TAM when the cross-entropy loss function is adopted instead of the contrastive loss function. The underlying reason lies in incorrect track segment association: mismatched track segments will cause the final fused trajectory of each target to deviate drastically from its real motion path.

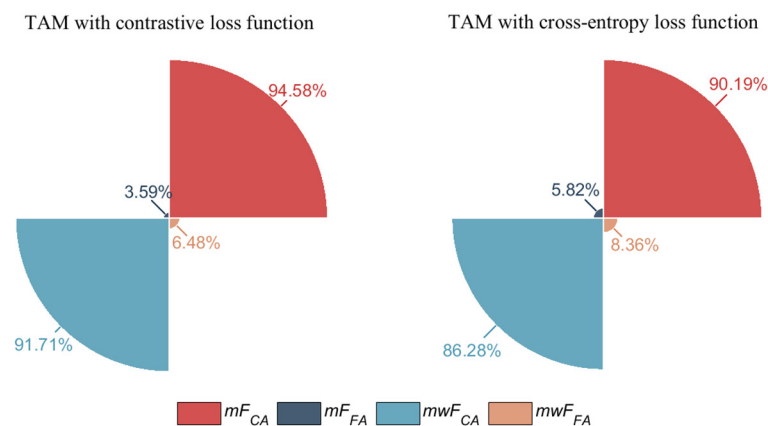


Figure 11. Effects of the loss function of TAM on track segment association.

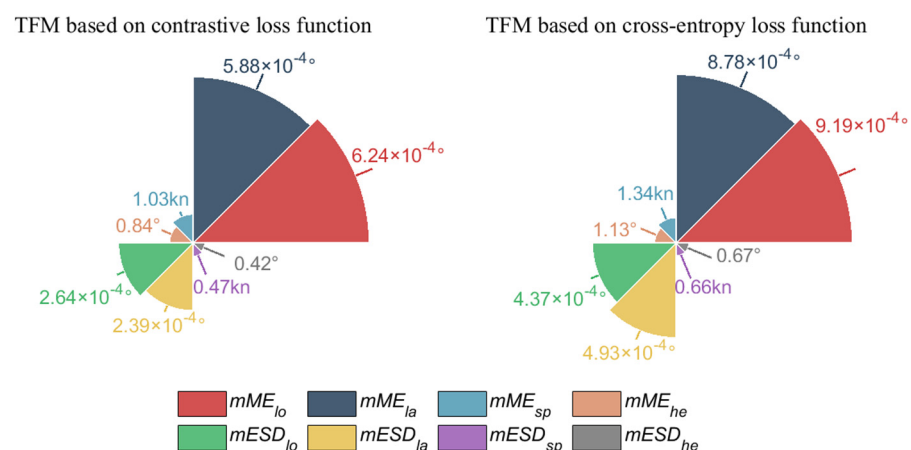


Figure 12. Effects of the loss function of TAM on track segment fusion.

3.2.4. Parameter Sensitivity Analysis

In this section, how the key hyperparameter of the TMTAFF affects the performance of track segment association and fusion is analyzed in detail.

Effects of the length of track segments: Figure 13 illustrates how the length of track segments affects the performance of track segment association and fusion. As observed, the track association capability peaks at a track segment length of 64, where mP_{CA} and mwP_{CA} reach 94.18% and 91.69%, respectively. Further increment in the track segment length leads to marginal performance degradation. In contrast, shortening the track segment length brings a notable drop in association accuracy. Specifically, reducing the length from 64 to 16 results in a 9.36% and 8.16% decline in mP_{CA} and mwP_{CA} , respectively. This phenomenon can be attributed to excessively short track segments, which fail to adequately characterize the overall motion patterns of targets and render the algorithm vulnerable to random data fluctuations. For track segment fusion, optimal performance is achieved at a track segment length of 128, with mME_{lo} , mME_{la} , $mESD_{lo}$ and $mESD_{la}$ equal to 6.31×10^{-4} , 5.75×10^{-4} , 2.59×10^{-4} and 2.33×10^{-4} . The fusion performance under this configuration is highly similar to that obtained with a segment length of 64. Moreover, decreasing the track segment length from 128 to 16 yields substantial growth in positional errors of fused tracks: mME_{lo} , mME_{la} , $mESD_{lo}$ and $mESD_{la}$ rise by 3.27×10^{-4} , 5.18×10^{-4} , 1.73×10^{-4} and 2.63×10^{-4} , correspondingly. According to the above analytical results, the track segment length is finally set to 64, which is able to obtain favorable tracking performance and alleviate the computational overhead of the TMTAFF as well.

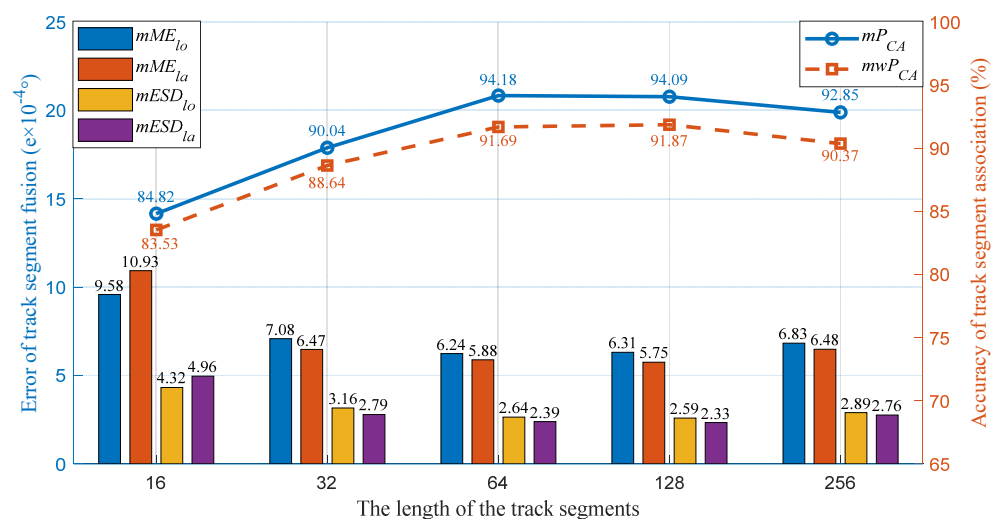


Figure 13. Effects of the length of track segments on track segment association and fusion.

Effect of the number of Transformer units: The quantity of Transformer units deployed within the Transformer-based encoder and decoder imposes a tangible impact on the feature extraction and fusion capacity of the proposed TMTAFF. As illustrated in Figure 14, the overall performance of track segment association and fusion converges to a relatively stable state when the number of Transformer units is set to six. Reducing the Transformer unit number from six to two triggers dramatic performance degradation: mP_{CA} and mwP_{CA} drop by 15.45% and 15.28%, respectively. Simultaneously, mME_{lo} and mME_{la} surge to 24.67×10^{-4} and 24.48×10^{-4} . This observation reveals that fewer Transformer units fail to extract sufficient discriminative high-dimensional features from the input track segments. Accordingly, the number of Transformer units is finalized as six in our TMTAFF.

Effects of the hidden state size: Figure 15 illustrates the track association and fusion performance under varying hidden state sizes. Notably, we conducted multiple groups of experiments, all of which exhibited consistent variation trends. In other words, the experimental results are repeatable. As the hidden state size increases from 32 to 128, the track

segment association accuracy improves remarkably. Specifically, mP_{CA} and mwP_{CA} rise by 14.08% and 12.61%, respectively, while the fused track errors mME_{lo} and mME_{la} drop sharply by $11.55 \times 10^{-4}^\circ$, $12.47 \times 10^{-4}^\circ$. Further expanding the hidden state size to 512 yields negligible performance gains in both track association and fusion tasks, yet drastically enlarges the total parameter count of the model. Accordingly, 128 is selected as the optimal hidden state size for our framework.

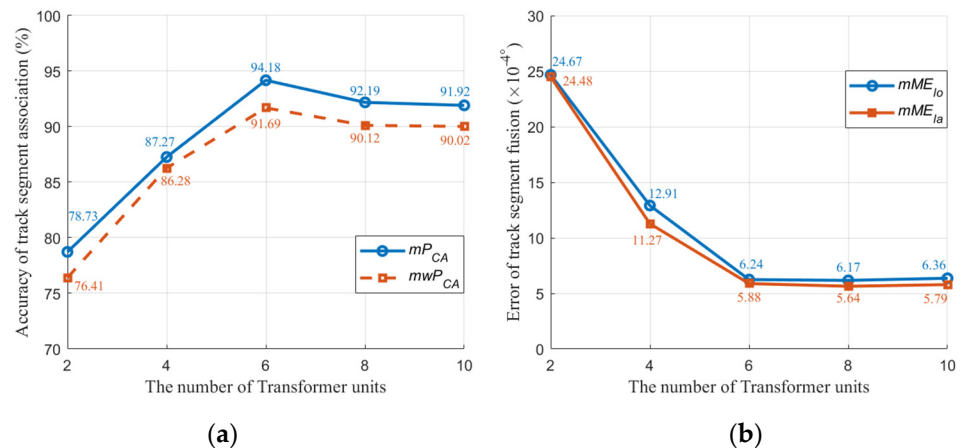


Figure 14. Effects of the number of Transformer units on track segment association and fusion. (a) Variations in mP_{CA} and mwP_{CA} as the number increases. (b) Variations in mME_{lo} and mME_{la} as the number increases.

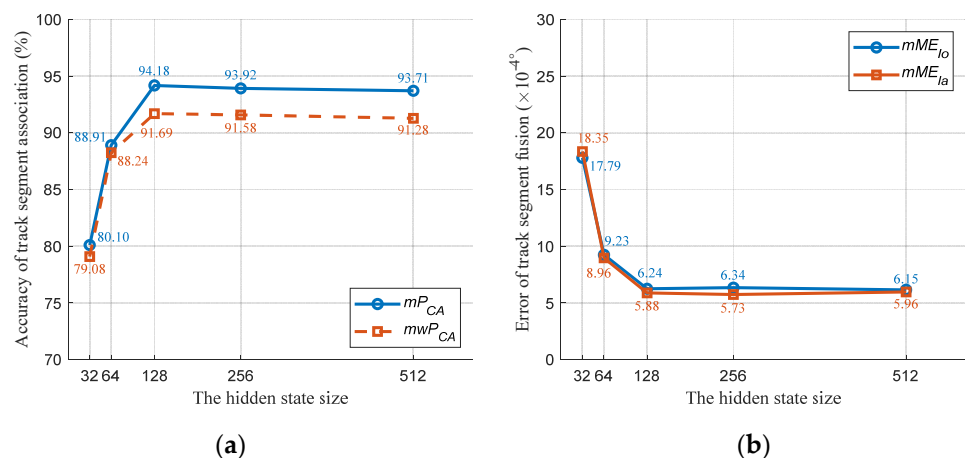


Figure 15. Effects of the hidden state size on track segment association and fusion. (a) Variations in mP_{CA} and mwP_{CA} as the size increases. (b) Variations in mME_{lo} and mME_{la} as the size increases.

Effect of the number of epochs: Figure 16 shows the evolution of model loss, as well as track segment association and fusion performance, as the number of training epochs increases. It can be observed that both the training and validation losses start to converge at the 30th epoch. After 60 epochs, the model loss hardly decreases further. Correspondingly, the track association accuracy metrics on the test set, namely mP_{CA} and mwP_{CA} , alongside the average mean fusion errors of fused tracks, i.e., mME_{lo} and mME_{la} , reach a steady state with negligible performance improvement. Thereby, to guarantee the generalization capability of the model across diverse operational scenarios, we finally fix the number of training epochs to 80 in our TMTAFF. Notably, in line with neural network theory, the training loss decreases continuously throughout the training process. By contrast, given a fixed network topology, the validation error will rise slightly once training

proceeds beyond a certain number of epochs. This trend corresponds to the classic over-fitting effect. As the training iterations increase, the model gradually memorizes noise and sample-specific features within the training set, which continuously reduces training loss. However, such memorization impairs the model's generalization ability on unseen validation data, thereby leading to a slow upward drift of validation error after a specific epoch threshold.

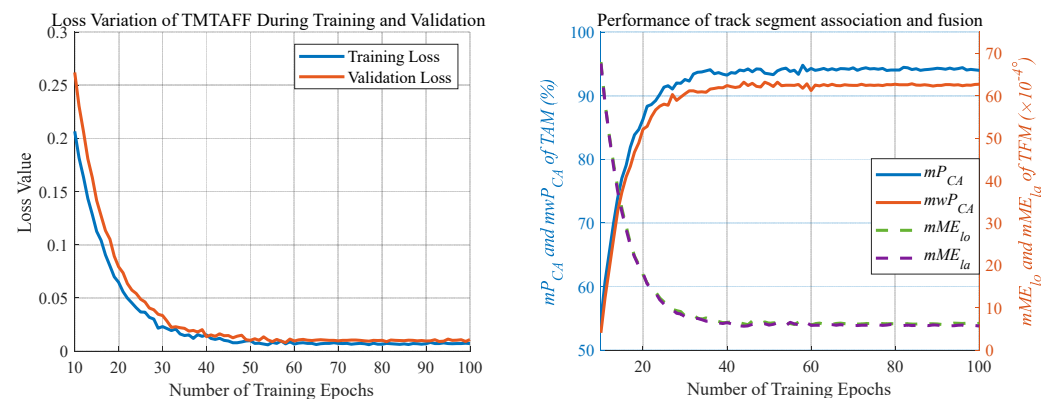


Figure 16. Effects of the number of epochs on model loss and track segment association and fusion.

4. Discussion

In this study, we present a unified computational framework called TMTAFF based on a Transformer encoder and Transformer decoder, which can tackle the track segment association problem and the track segment fusion problem simultaneously and obtain higher performance compared with several state-of-the-art baselines. Furthermore, the proposed TMTAFF is compatible with both the ITSA task and the MSTSA task simultaneously due to the designed TCM, making itself more adaptable to complicated operational scenarios. Although TMTAFF has demonstrated highly promising merits, there are still some limitations needing improvement, as below.

Firstly, during the track association phase, the designed TAM separately determines whether any pair of two arbitrary track segments originates from an identical target. In practice, numerous track segments are geographically distant, and accordingly, it is barely possible for them to correspond to the same target, rendering pairwise matching redundant and inducing unnecessary computational resource consumption and runtime overhead for the model. Although this study has partitioned all the target track data into discrete grids in order to exclude pairing of tracks with excessive spatial separation and hence decrease the number of redundant and invalid input samples partially, track segments with large spatial distance within an identical grid can still be paired into one input sample. Moreover, the grid-based partitioning inevitably splits some track segments into multiple grids, which artificially fractures target information and raises the track fragmentation rate. To tackle the above deficiency, this work proposes deploying an efficient distance metric during input sample construction. Specifically, only candidate track pairs satisfying the predefined distance threshold are screened for subsequent model inference. Such a screening mechanism can fundamentally improve the validity of input samples and enhance the integrity of ultimate track fusion outcomes concurrently.

Secondly, the proposed TMTAFF works in an offline paradigm instead of an online architecture. The entire target measurement set needs to be fully acquired prior to implementing track segment association and fusion, which restricts its real-time execution capability. To mitigate such a limitation, here we put forward several feasible improvement schemes for future optimization. On the one hand, constraining the time span of input samples enables the network to generate preliminary association and fusion outputs

within a short latency. As time proceeds, the first track point belonging to the previous sample is discarded, and the residual track fragments are sequentially concatenated with the newly arrived track point to constitute the updated input sample at the current timestamp. Benefiting from such a sliding sample construction strategy, the framework is capable of updating real-time association and fusion results at each time step and consequently satisfies practical real-time requirements. On the other hand, in order to compensate for potential performance degradation triggered by the shortened length of the input samples, historical track association and fusion results of the preceding time step are fed into the TAM and TFM as prior information separately, which is expected to further promote the performance of subsequent track association and fusion.

Thirdly, the Transformer decoder embedded in TFM is theoretically capable of sequentially producing fused track estimates until a termination token is triggered. Under such a mechanism, the proposed TMTAFF possesses a predictive capability for future target trajectories to a certain degree. Nevertheless, due to its offline track fusion mechanism, TMTAFF can access future observational measurements when computing the track fusion results of the current timestamp, which fundamentally differs from the online track fusion paradigm that prohibits the utilization of subsequent incoming observations. Furthermore, once the output sequence length of TFM matches that of input samples, the decoder can still extrapolate prospective track points without incorporating newly captured sensor data. Accordingly, the prediction accuracy is anticipated to degrade progressively as time prolongs. To refine the track prediction precision of TFM, continuous real-time subsequent observations can be fed back into the TFM. In this way, the generated predicted tracks can be derived from abundant authentic measurements. The aforementioned optimization approach holds promising application prospects for unmanned surface vehicles (USVs) in autonomous obstacle avoidance scenarios, where precise short-term trajectory prediction of surrounding targets is indispensable to support reliable collision-evasion maneuvers.

Fourthly, all the input track points fed into the TMTAFF share identical dimensional configurations in this study, with each track point consisting of longitude, latitude, velocity and heading. Nevertheless, multiple sensors mounted on a vessel commonly produce measurements of disparate dimensions under realistic conditions. For instance, 3D radars are capable of acquiring target range, azimuth and elevation data, whereas electro-optical sensors fail to provide range-related observations in most cases. Such dimensional inconsistency of heterogeneous track segments prohibits the direct deployment of TMTAFF and narrows its practical applicable scope. To tackle this constraint, two feasible improvement strategies are proposed. The first one aims to supplement missing state components for low-dimensional tracks via pseudo-measurement construction or dynamic motion constraints to unify the dimension of state vectors prior to track association and fusion. As an alternative solution, the second one discards the original physical dimension information, and the temporal, kinematic, and geometric features are extracted from disparate track segments and projected into a shared latent space, where subsequent track association and fusion procedures are implemented uniformly.

Fifthly, the MTAD dataset adopted in this study is constructed based on massive AIS trajectories of maritime targets. However, maritime vessels generally move at low cruising speeds with limited maneuverability and usually show predictable trajectory evolutions, which differ substantially from the motion characteristics of aerial targets. Given that aerial targets constitute a prevalent target category in practical battlefield scenarios, further experiments utilizing airborne track datasets are required in follow-up research to validate the track association and fusion performance of TMTAFF across diversified target types comprehensively.

Sixthly, although discrete attributes have been considered in the network design of TMTAFF, they are not utilized in our experimental validation. This is because the employed public dataset lacks adequate discrete attributes, and several baseline methods also exclude discrete attributes in their experiments. To ensure fair experimental comparisons, we do not introduce discrete attributes in the experimental validation. In the future, we will adopt more suitable datasets and conduct further targeted experiments to explore the effect of discrete attributes.

5. Conclusions and Future Work

In this paper, we propose a Transformer-based track association and fusion model named TMTAFF, which is capable of dealing with the track segment association problem and track segment fusion problem simultaneously in an offline manner. Specifically, the TAM module integrated into the TMTAFF is compatible with both ITSA and MSTSA tasks simultaneously. Concretely, it achieves association accuracies of 94.58% and 91.71% on the ITSA task, as well as 93.91% and 91.38% on the MSTSA task in terms of mP_{CA} and mwP_{CA} , respectively, surpassing six state-of-the-art baselines. However, TAM exhibits a certain level of computational redundancy. Meanwhile, the designed TFM enables precise estimation of targets' real kinematic states. Quantitatively, it yields fusion errors of $6.24 \times 10^{-4}^\circ$, $5.88 \times 10^{-4}^\circ$, 1.03 kn and 0.84° in terms of mME_{lo} , mME_{la} , mME_{sp} and mME_{he} , respectively, outperforming five cutting-edge baselines. Moreover, ablation experiments quantitatively validate that the proposed TCM contributes to boosting the overall performance on track segment association and fusion. In addition, the hyperparameters of the TMTAFF are fine-tuned to their optimal values based on extensive experiments.

For future research, we will prioritize improving the real-time inference capability of the TMTAFF and thoroughly investigate its multi-sensor multi-target track association and fusion performance when confronting highly maneuvering aerial targets.

Author Contributions: The work described in this article is the collaborative development of all authors. Conceptualization of data processing, L.Z., X.W. and F.L.; algorithm design, L.Z. and F.L.; data measurement, X.W. and F.L.; analysis, L.Z. and F.L.; writing—original draft preparation, L.Z., X.W. and F.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work is funded by the National Natural Science Foundation of China (No. 62025205, 62032018, 62472358, 62322601, 62102322).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset used is a public dataset and can be downloaded from the following website. <https://www.scidb.cn/en/detail?dataSetId=c7d8dc56fe854ec2b084d075feb887fd> (accessed on 16 July 2026).

Acknowledgments: The authors would like to thank the support of the laboratory, university, and government.

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

References

1. Yang, Z.; Yao, T.; Hou, Y.; Su, X.; Bai, H.; Geng, X.; Wang, B. Association and Fusion Algorithm in Air Defense Weapon System. In *Proceedings of the 10th Frontier Academic Forum of Electrical Engineering (FAFEE2022): Volume II*; Springer Nature: Berlin/Heidelberg, Germany, 2023; p. 2: 353.
2. Xiong, W.; Xu, P.; Cui, Y. Unsupervised and interpretable track-to-track association based on homography estimation of radar bias. *IET Radar Sonar Navig.* **2024**, *18*, 294–307.
3. Wang, J.; Zeng, Y.; Wei, S.; Wei, Z.; Wu, Q.; Yvon, S. Multi-sensor track-to-track association and spatial registration algorithm under incomplete measurements. *IEEE Trans. Signal Process.* **2021**, *69*, 3337–3350.
4. Ogle, T.L.; Blair, W.D.; Slocumb, B.J.; Dunham, D.T. Assessment of hierarchical multi-Sensor multi-target track fusion in the presence of large sensor biases. In *2019 22th International Conference on Information Fusion (FUSION)*; IEEE: New York, NY, USA, 2019; pp. 1–7.
5. Zhang, J.; Liu, Z.; Gan, Y.; Liu, Y. Dong, J. Track-to-track association from diverse source ship trajectory based on an improved graph neural network. *Appl. Ocean Res.* **2025**, *159*, 104598.
6. Zheng, S.; Wan, X.; Zhang, L.; Ni, P.; Shi, A.; Wei, X.; Yi, X. Transformer-Based Multi-Target Tracking with Unified Association and Filtering. In *2025 IEEE Radar Conference (RadarConf25)*; IEEE: New York, NY, USA, 2025; pp. 449–454.
7. Cao, Z.; Liu, B.; Yang, J.; Tan, K.; Dai, Z.; Lu, X.; Gu, H. Contrastive transformer network for track segment association with two-stage online method. *Remote Sens.* **2024**, *16*, 3380.
8. Yeom, S.W.; Kirubarajan, T.; Bar-Shalom, Y. Track segment association, fine-step IMM and initialization with Doppler for improved track performance. *IEEE Trans. Aerosp. Electron. Syst.* **2004**, *40*, 293–309.
9. Zhang, S.; Bar-Shalom, Y. Track segment association for GMTI tracks of evasive move-stop-move maneuvering targets. *IEEE Trans. Aerosp. Electron. Syst.* **2011**, *47*, 1899–1914.
10. Qi, L.; Wang, H.; Xiong, W.; Dong, K. Track segment association algorithm based on multiple-hypothesis models with priori information. *Syst. Eng. Electron* **2015**, *37*, 732–739.
11. Raghu, J.; Srihari, P.; Tharmarasa, R.; Kirubarajan, T. Comprehensive track segment association for improved track continuity. *IEEE Trans. Aerosp. Electron. Syst.* **2018**, *54*, 2463–2480.
12. Kuhn, H.W. The Hungarian method for the assignment problem. *Nav. Res. Logist. Q.* **1955**, *2*, 83–97.
13. Tian, W.; Wang, Y.; Shan, X.; et al. Track-to-track association for biased data based on the reference topology feature. *IEEE Signal Process. Lett.* **2014**, *21*, 449–453.
14. Zhao, H.; Sha, Z.; Wu, J. An Improved Fuzzy Track Association Algorithm Based on Weight Function. In *2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*; IEEE: New York, NY, USA, 2017; pp. 1125–1128.
15. Senel, N.; Kefferpütz, K.; Doycheva, K.; Elger, G. Multi-sensor data fusion for real-time multi-object tracking. *Processes* **2023**, *11*, 501.
16. Sengupta, A.; Cheng, L.; Cao, S. Robust multiobject tracking using mmwave radar-camera sensor fusion. *IEEE Sens. Lett.* **2022**, *6*, 1–4.
17. Guan, H.; Diao, W. A Sequential Fusion Method of Multi-radar Tracking Orbital Target Based on EKF. In *2023 6th International Conference on Information Communication and Signal Processing (ICICSP)*; IEEE: New York, NY, USA, 2023; pp. 579–584.
18. Zhang, Z.; Fu, K.; Sun, X.; Ren, W. Multiple target tracking based on multiple hypotheses tracking and modified ensemble Kalman filter in multi-sensor fusion. *Sensors* **2019**, *19*, 3118.
19. Qi, J.; Lu, X.; Sun, J. Multi-Radar track fusion method based on parallel track fusion model. *Electronics* **2025**, *14*, 3461.
20. Wang, Y.; Ye, B.; Cai, Z.; Wu, H. A Transformer-Based Tracker Integrating Motion and Representation Information. *IEEE Trans. Multimed.* **2026**, *28*, 4383–4392.
21. Wang, S. Graph neural network-driven text classification for fire-door defect inspection in pre-completion construction. *Sci. Rep.* **2025**, *15*, 44382.
22. Wang, S. Development of an automated transformer-based text analysis framework for monitoring fire door defects in buildings. *Sci. Rep.* **2025**, *15*, 43910.
23. Xiong, W.; Xu, P.; Cui, Y.; Lv, Y.; Gu, X. Track segment association with dual contrast neural network. *IEEE Trans. Aerosp. Electron. Syst.* **2021**, *58*, 247–261.
24. Liu, F.; Zhou, X.; Cao, J.; Wang, Z.; Wang, T.; Wang, H.; Zhang, Y. Anomaly detection in quasi-periodic time series based on automatic data segmentation and attentional LSTM-CNN. *IEEE Trans. Knowl. Data Eng.* **2020**, *34*, 2626–2640.

25. Liu, F.; Zhou, X.; Cao, J.; Wang, Z.; Wang, H.; Zhang, Y. Arrhythmias Classification by Integrating Stacked Bidirectional LSTM and Two-dimensional CNN. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*; Springer International Publishing: Cham, Switzerland, 2019; pp. 136–149.
26. Xiong, W.; Xu, P.; Cui, Y.; Xiong, Z.; Gu, X.; Lv, Y. Track Segment Association via track graph representation learning. *IET Radar Sonar Navig.* **2021**, *15*, 1458–1471.
27. Liu, J.; Luo, L.; Gao, J.; Yang, S. Radar Multi-Target Tracking Data Association Method Based on Transformer. In *2025 6th International Conference on Computer Engineering and Intelligent Control (ICCEIC)*; IEEE: New York, NY, USA, 2025; pp. 448–451.
28. Helgesen, Ø.K.; Brekke, E.F.; Helgesen, H.H.; Engelhardtsen, Ø. Sensor Combinations in Heterogeneous Multi-sensor Fusion for Maritime Target Tracking. In *2019 22th International Conference on Information Fusion (FUSION)*; IEEE: New York, NY, USA, 2019; pp. 1–9.
29. Ma, K.; Zhang, H.; Wang, R.; Zhang, Z. Target Tracking System for Multi-sensor Data Fusion. In *2017 IEEE 2nd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*; IEEE: New York, NY, USA, 2017; pp. 1768–1772.
30. Jin, B.; Tang, Y.; Zhang, Z.; Lian, Z.; Wang, B. Radar and AIS track association integrated track and scene features through deep learning. *IEEE Sens. J.* **2023**, *23*, 8001–8009.
31. Meinhardt, T.; Kirillov, A.; Leal-Taixe, L.; Feichtenhofer, C. Trackformer: Multi-object tracking with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: New York, NY, USA, 2022; pp. 8844–8854.
32. Liu, L.; Ji, H.; Zhang, W.; Liao, G. Multi-sensor fusion for multi-target tracking using measurement division. *IET Radar Sonar Navig.* **2020**, *14*, 1451–1461.
33. Xu, X.; Feng, Z.; Cao, C.; Yu, C.; Li, M.; Wu, Z.; Ye, S.; Shang, Y. Stn-track: Multiobject tracking of unmanned aerial vehicles by swin transformer neck and new data association method. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 8734–8743.
34. Cui, Y.; Xu, P.; Sun, W.; Zhang, S.; Li, J. DeepAF: Transformer-Based Deep Data Association and Track Filtering Network for Multi-Target Tracking in Clutter. *Aerospace* **2025**, *12*, 194.
35. Xiang, X.; Zhou, X.; Wang, X.; Zhai, M.; Saddik, A.EI. Multi-object tracking with scale-aware transformer and enhanced association strategy. *Multimed. Syst.* **2025**, *31*, 122.
36. Liu, F.; Zhou, X.; Wang, Z.; Ni, H.; Wang, T. OSA-weigher: An automated computational framework for identifying obstructive sleep apnea based on event phase segmentation. *J. Ambient Intell. Humaniz. Comput.* **2019**, *10*, 1937–1954.
37. Liu, F.; Zhou, X.; Cao, J.; Wang, Z.; Wang, H.; Zhang, Y. A LSTM and CNN Based Assemble Neural Network Framework for Arrhythmias Classification. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*; IEEE: New York, NY, USA, 2019; pp. 1303–1307.
38. Zeng, F.; Dong, B.; Zhang, Y.; Wang, T.; Zhang, X.; Wei, Y. Motr: End-to-end Multiple-object Tracking with Transformer. In *European Conference on Computer Vision*; Springer Nature Switzerland: Cham, Switzerland, 2022; pp. 659–675.
39. Lee, M. Mathematical analysis and performance evaluation of the gelu activation function in deep learning. *J. Math.* **2023**, *2023*, 4229924.
40. Ghogh, B.; Sikaroudi, M.; Shafiei, S.; Tizhoosh, H.R.; Karray, F.; Crowley, M. Fisher discriminant triplet and contrastive losses for training siamese networks. In *2020 International Joint Conference on Neural Networks (IJCNN)*; IEEE: New York, NY, USA, 2020; pp. 1–7.
41. Xue, Z.; Zhou, Y.; Du, P. S3Net: Spectral-spatial Siamese network for few-shot hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–19.
42. Lee, K.H.; Kanzawa, Y.; Derry, M.; James, M.R. Multi-target Track-to-track Fusion Based on Permutation Matrix Track Association. In *2018 IEEE Intelligent Vehicles Symposium (IV)*; IEEE: New York, NY, USA, 2018; pp. 465–470.
43. Zhu, Y.; Wang, T.; Zhu, S. Adaptive multi-pedestrian tracking by multi-sensor: Track-to-track fusion using monocular 3D detection and MMW radar. *Remote Sens.* **2022**, *14*, 1837.
44. Caposova, R.; Blackwell, T. Evaluating Homoscedastic Uncertainty Weighting and Generative Tau Imputation in a Multimodal, Multitask Deep Learning Framework for Alzheimer’s Disease. *Neuroinformatics* **2026**, *24*, 22.
45. Liu, F.; Zhou, X.; Wang, T.; Cao, J.; Wang, Z.; Wang, H.; Zhang, Y. An Attention-based Hybrid LSTM-CNN Model for Arrhythmias Classification. In *2019 International Joint Conference on Neural Networks (IJCNN)*; IEEE: New York, NY, USA, 2019; pp. 1–8.
46. Liu, F.; Li, G.; Wu, C.; Zeng, L.; Wu, G. An Attentional YOLOv4 Model for Target Detection. In *2022 4th International Academic Exchange Conference on Science and Technology Innovation (IAECST)*; IEEE: New York, NY, USA, 2022; pp. 871–875.
47. Liu, T.; Liu, K.; Zeng, X.; Zhang, S.; Zhou, Y. Efficient marine track-to-track association method for multi-source sensors. *IEEE Signal Process. Lett.* **2024**, *31*, 1364–1368.

48. Cui, Y.; Xu, P.; Gong, C.; Yu, Z.; Zhang, J.; Yu, H. Dong, K. Multisource Track Association Dataset Based on the Global AIS. *J. Electron. Inf. Technol.* **2023**, *45*, 746–756.
49. Wan, S.; Mu, H.; Han, K.; Cheong, T.; Xie, C. A fuzzy track-to-track association algorithm with dynamic time warping for trajectory-level vehicle detection. *Int. J. Transp. Sci. Technol.* **2025**, *17*, 95–108.
50. Ling, X.; Chen, Z.; Du, G.; Sheng, Q. A multi-source clustered targets track association method based on dual-channel TCN-GRU. *J. Syst. Eng. Electron.* **2026**, *37*, 367–376.
51. Qi, J.; Lu, X.; Sun, J. An Intelligent Track Segment Association Method Based on Characteristic-Aware Attention LSTM Network. *Sensors* **2025**, *25*, 3465.
52. Lee, E.H.; Song, T.L. Multi-sensor track-to-track fusion with target existence in cluttered environments. *IET Radar Sonar Navig.* **2017**, *11*, 1108–1115.
53. Zhang, S.; Yuan, T.; Caol, W.; Chang, K.C.; Zhang, L.; Krishnan, K. Multi-sensor track-to-track fusion using dynamic consensus. In *2024 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*; IEEE: New York, NY, USA, 2024; pp. 1–7.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.