

Article

Online Multi-Sensor Calibration Method for Unmanned Surface Vehicle Swarms in Complex and Contested Environments

Zhaoqiang Gao ^{1,2,*}, Xixiang Liu ² and Jiazhou He ¹¹ Jiangsu Automation Research Institute, Lianyungang 222001, China; 18036676276@189.cn² School of Instrument Science and Engineering, Southeast University, Nanjing 210018, China; scliuseu@163.com

* Correspondence: 230209098@sdu.edu.cn

Highlights

What are the main findings?

- **A unified targetless calibration framework** is established for USV swarms, which simultaneously estimates sensor extrinsics and swarm poses by integrating navigation radar, EOS, IMU, and GNSS into a tightly coupled factor graph.
- **An anti-interference and degradation handling mechanism** is proposed, where the system achieves an 84.5% improvement in localization accuracy under simulated severe GNSS jamming conditions and robustly calibrates extrinsics without artificial targets.

What are the implications of the main findings?

- **Enabling resilient swarm operations:** The proposed method allows USV swarms to maintain high-precision formation and navigation in GNSS-denied or signal-challenged environments, significantly enhancing their reliability and mission capability.
- **Simplifying deployment logistics:** By eliminating the need for pre-deployment manual calibration with artificial targets, the “calibrate-as-you-go” capability reduces operational costs and allows for flexible, rapid reconfiguration of modular sensor payloads at sea.

Abstract

In complex maritime environments and scenarios with severe signal interference, unmanned surface vehicle (USV) swarms face dual challenges: unreliable GNSS signals due to interference and difficulties in accurately calibrating multi-sensor installation errors. These issues severely constrain the capability for high-precision cooperative formation operations. To address these problems, this paper proposes a cooperative localization and all-source online calibration algorithm based on a unified factor graph optimization framework. First, a tightly coupled all-source graph framework is established, integrating navigation radar, electro-optical systems (EOSs) with laser rangefinders, IMU, and GNSS into a sliding window. By leveraging high-precision mutual observations among the swarm, strong geometric constraints are constructed to mitigate the drift of individual inertial navigation systems. Second, an adaptive GNSS weighting mechanism based on signal quality and a degradation detection strategy based on eigenvalue analysis of the Fisher Information Matrix (FIM) are designed. These mechanisms enable online identification and robust estimation of extrinsic parameters, effectively resolving calibration divergence under weak excitation conditions such as straight-line sailing. Finally, the proposed algorithm is validated using field data from three USVs combined with simulated interference experiments.



Academic Editor: Pablo Rodríguez-Gonzálvez

Received: 14 January 2026

Revised: 19 February 2026

Accepted: 19 February 2026

Published: 27 February 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.

Results demonstrate that the algorithm can rapidly converge to high-precision calibration parameters without artificial targets (radar translation error < 0.2 m, EOS rotation error $< 0.05^\circ$). During periods of **simulated** GNSS interference, the cooperative localization root mean square error (RMSE) is reduced to 2.85 m, representing an accuracy improvement of approximately 84.5% compared to traditional methods. This study achieves a “more accurate as it runs” cooperative navigation effect, providing reliable technical support for USV swarm applications in GNSS-denied environments.

Keywords: USV swarm; cooperative localization; online calibration; factor graph optimization; GNSS-denied environment

1. Introduction

As core nodes in future maritime systems for civilian and scientific applications, autonomous unmanned surface vehicle (USV) swarms rely heavily on the deep fusion of multi-source heterogeneous sensor information for situational awareness and collaborative decision-making [1]. A typical USV platform integrates a suite of sensors, including electro-optical systems (EOSs), navigation radar, search radar, LiDAR, Doppler logs, and crucially, the Strapdown Inertial Navigation System (SINS). Collectively, these sensors form a robust perception network. Through multi-sensor fusion, they provide the swarm with comprehensive situational awareness, precise self-localization, and environmental perception capabilities, forming the foundation for autonomous navigation, collision avoidance, target recognition, and tracking [2].

To achieve cross-platform collaborative perception, a fundamental prerequisite is the precise knowledge of the spatiotemporal relationships among all sensors within the swarm. Specifically, this entails determining the spatial mounting parameters (i.e., extrinsic parameters, including lever arms and boresight angles) between sensors and the SINS, as well as the temporal synchronization across various sensors. Failure to adequately calibrate these parameters can lead to severe fusion errors. For instance, an uncompensated millisecond-level time offset can translate into decimeter-level spatial projection errors on a high-speed maneuvering hull; similarly, minute angular mounting errors can result in significant localization deviations when observing distant targets. Such errors directly undermine the consistency of the swarm’s shared situational picture and may even lead to mission failure [3]. Although factory calibration is mandatory, at-sea operations require rapid, online calibration due to structural deformation, sensor replacement, and modular payloads. This ensures that USV swarms can quickly complete calibration and error compensation prior to deployment [4].

While multi-sensor calibration has been extensively studied over the past decades, most research has focused on offline, target-based methods. For example, the well-known “Zhang’s method” proposed by Zhang et al. [5] utilized a 2D checkerboard to calibrate camera intrinsics via planar homography. Dhall et al. [6] proposed a 3D–3D point matching method that extracts the corners or centers of calibration boards from both images and point clouds, establishing 3D–3D correspondences and solving for rigid transformation using ICP or SVD to calibrate LiDAR and cameras. Similarly, Frey et al. [7] developed a LiDAR–IMU calibration method using point cloud feature extraction, and Liu et al. [8] proposed a camera–IMU calibration method based on visual odometry. However, these methods rely heavily on specific calibration targets, limiting their applicability to laboratory environments or pre-factory settings.

To enhance the applicability of calibration algorithms, numerous targetless online calibration methods have been proposed. Ishikawa et al. [9] introduced a LiDAR–camera

calibration method based on motion consistency. Persic et al. [10] proposed an online multi-sensor calibration method based on detecting and tracking moving objects. Chen et al. [11] presented a general multi-sensor calibration framework. While effective, these approaches are primarily designed for land-based environments, exploiting rich environmental features for calibration. They are ill-suited for maritime environments where observable static features are scarce. Furthermore, most existing methods focus on single-platform calibration. At the swarm level, ensuring that sensor data from all USVs are unified into a global coordinate system while maintaining consistent online calibration accuracy remains an unaddressed research gap.

To address these challenges, this paper proposes a unified, targetless, and swarm-oriented calibration framework. Our core idea is to eliminate the dependence on specific calibration targets and absolute positioning information. Instead, we fully exploit the rich constraints inherent in the USV's own motion and the relative observations between swarm members to achieve multi-sensor calibration.

Unlike existing swarm-level approaches [12] that focused primarily on localization while assuming pre-calibrated and fixed sensor extrinsics, this work introduces a unified framework that treats extrinsic parameters as state variables to be optimized online. Furthermore, we address the critical issue of observability degradation in open-sea environments—a gap often overlooked in the current cooperative calibration literature. The main contributions of this paper are as follows:

- (1) A unified multi-sensor calibration framework for maritime environments: We propose a joint optimization framework tailored for maritime conditions capable of synchronously estimating sensor extrinsics, and relative poses of the swarm, enabling rapid multi-sensor calibration at sea.
- (2) A swarm cooperative calibration mechanism: We introduce a swarm calibration paradigm based on relative motion constraints. Even in the absence of absolute GPS information, a robust calibration constraint network is constructed through mutual observations among USVs, achieving globally consistent calibration.

The remainder of this paper is organized as follows: Section 2 reviews related work; Section 3 introduces system modeling and problem formulation; Section 4 details the proposed swarm calibration algorithm; Section 5 presents the experimental setup and analysis of results; and Section 6 concludes the paper and outlines future research directions.

2. Related Work

2.1. Target-Based Sensor Calibration

Early calibration methodologies predominantly relied on artificial targets with well-defined geometric features to estimate intrinsic and extrinsic parameters. Planar patterns, such as checkerboards, laid the foundation for high-precision photogrammetry-based calibration [5,13,14]. These approaches were subsequently extended to multi-camera and camera–inertial measurement unit (IMU) systems by integrating temporal alignment strategies [13] and non-linear distortion modeling [14,15].

In the domain of LiDAR–camera calibration, physical targets with distinct geometric structures or reflective properties have been widely employed. For instance, planar boards with retroreflective surfaces [16] and checkerboards [17] were used to align 3D point clouds with 2D images by optimizing geometric correspondences. Similarly, 3D–3D point cloud matching methods [6] and line-feature-based registration [18] have proven effective in structured environments. Regarding joint IMU–vision calibration, optical motion capture systems [19] often serve as ground truth providers, while robust estimators [8] have been developed to handle scenarios with incomplete inertial information.

Although target-based methods offer high precision and repeatability, their dependence on visible targets and controlled environments severely limits scalability in large-scale or open-field applications. As autonomous systems evolve toward outdoor operations, the requirement for meticulous setup and manual intervention becomes impractical, driving the shift toward targetless calibration techniques.

2.2. Targetless Sensor Calibration

To overcome the limitations of artificial targets, targetless calibration methods leverage natural scene features or sensor motion constraints. Motion-based approaches for camera–IMU systems exploit kinematic constraints between inertial and visual measurements, using closed-form solutions [20], probabilistic optimization [21], or visual odometry [22] to refine calibration parameters online. Visual-only methods often rely on vanishing points extracted from structural scenes [23,24].

In LiDAR–camera systems, mutual information-based methods [16,17] calibrate sensors by maximizing the correlation between environmental intensity and depth, eliminating the need for specific targets. Probabilistic registration frameworks based on point-to-edge and point-to-plane constraints [25] further improve robustness in unstructured environments. Additionally, edge alignment strategies [16] and Gaussian Mixture Models [26] have been introduced to enhance feature matching accuracy.

Compared to target-based approaches, targetless calibration offers greater flexibility and autonomy, making it particularly suitable for field-deployed systems. However, these methods often face observability and degradation issues: sufficient excitation signals and environmental textures are required to make all parameters identifiable. Moreover, drift and bias in inertial sensors can propagate into calibration results.

The emergence of factor graph optimization frameworks, such as **GTSAM** [27], has enabled unified modeling of calibration and state estimation. Seminal works in visual–inertial odometry (VIO), such as **VINS-Mono** [28], further demonstrated the efficacy of tightly coupled optimization for real-time state estimation. Our work extends these foundational concepts to the domain of multi-USV cooperative calibration.

2.3. Joint Spatiotemporal Calibration

Accurate sensor fusion relies not only on spatial alignment but also on precise temporal synchronization. Asynchronous sensor data can introduce significant errors, especially during high-dynamic maneuvers. Joint spatiotemporal calibration methods have been developed to estimate both spatial extrinsics and temporal offsets simultaneously.

Early works explicitly modeled time delays in visual–inertial systems [29] or utilized motion-based constraints for multi-modal sensor arrays [30]. The Kalibr framework [13] formulated calibration as a maximum likelihood estimation problem, unifying the optimization of intrinsics, extrinsics, and time offsets. This concept was extended to multi-camera–IMU systems [31] and continuous-time trajectory estimation [32], enabling the handling of rolling shutter effects and high-frequency inertial data. Recent B-spline-based approaches [11] further generalized this framework to targetless scenarios.

While effective, batch optimization methods like Kalibr require sufficient motion excitation and are sensitive to initialization. This motivates the need for robust adaptive calibration strategies capable of maintaining accuracy in dynamic or harsh environments.

2.4. Online Dynamic Calibration

Real-world systems, particularly USV swarms operating in complex maritime conditions, are subject to mechanical vibrations and thermal expansion, which degrade calibration accuracy over time. Online dynamic calibration frameworks address this by continuously refining parameters during operation.

Keyframe-based visual-inertial SLAM methods [31] treat extrinsics as state variables to be optimized online. Similarly, tightly coupled estimators [33] collaboratively estimate pose, landmarks, and calibration parameters, ensuring observability-aware updates to prevent overfitting. Recent research has expanded these capabilities to targetless LiDAR-camera calibration [34], dual-quaternion-based motion estimation [35], and moving object tracking [10]. Additionally, non-holonomic constraints [36] and unified motion-based frameworks [37] have been proposed for mobile robots. Hybrid approaches combining analytical models with deep learning [38] are also emerging to adaptively compensate for minor misalignments.

Despite these advancements, existing frameworks often face trade-offs between accuracy, computational cost, and robustness in partially observable or dynamic environments. Moreover, few studies have investigated enhancing system calibration robustness through cooperative calibration across multiple platforms.

These gaps motivate this paper to propose a cooperative calibration framework for USV swarms, aiming to provide consistent calibration for diverse sensor configurations while maintaining robustness against disturbances in complex maritime environments.

3. System Modeling and Problem Formulation

This section presents the mathematical notation and coordinate frame definitions used throughout this paper, followed by detailed descriptions of the corresponding sensor models and the swarm cooperative observation model.

3.1. Notation and Coordinate Frames

We consider a swarm consisting of n ($n > 2$) unmanned surface vehicles (USVs). Each USV is equipped with three typical sensors: a SINS/GNSS integrated navigation system, a navigation radar, and an electro-optical system (EOS). We denote the world frame, the body frame, the EOS frame, and the radar frame of the i -th USV as W , B_i , C_i , and R_i , respectively. We represent the 6-degree-of-freedom (DoF) rigid transformation from frame a to frame b using a Euclidean matrix $\mathbf{T}_a^b \in SE(3)$, defined as

$$\mathbf{T}_a^b = \begin{bmatrix} \mathbf{R}_a^b & \mathbf{t}_a^b \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \quad (1)$$

where $\mathbf{R}_a^b \in SO(3)$ represents the rotation part, and $\mathbf{t}_a^b \in \mathbb{R}^3$ represents the translation part.

The state of the i -th USV at time k is defined as

$$\mathbf{x}_{i,k} = [\mathbf{p}_{i,k}^W, \mathbf{q}_{i,k}^W, \mathbf{v}_{i,k}^W, \mathbf{b}_{a_i,k}, \mathbf{b}_{g_i,k}]^T \quad (2)$$

where $\mathbf{p}_{i,k}^W$ and $\mathbf{q}_{i,k}^W$ denote the position and orientation (represented by a quaternion) in the world frame, $\mathbf{v}_{i,k}^W$ is the velocity, and $\mathbf{b}_{a_i,k}$, $\mathbf{b}_{g_i,k}$ are the biases of the accelerometer and gyroscope, respectively.

Let the extrinsic parameters of the EOS and the navigation radar for the i -th USV be $\mathbf{T}_{b_i}^{c_i} = \begin{bmatrix} \mathbf{R}_{b_i}^{c_i} & \mathbf{t}_{b_i}^{c_i} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}$, $\mathbf{T}_{b_i}^{r_i} = \begin{bmatrix} \mathbf{R}_{b_i}^{r_i} & \mathbf{t}_{b_i}^{r_i} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}$, respectively. The objective is to estimate the extrinsic parameters $\{\mathbf{T}_{b_i}^c, \mathbf{T}_{b_i}^r\}_{i=1 \dots N}$ for all USVs, where N is the number of USVs.

3.2. Unified Factor Graph Optimization Model

We formulate the multi-sensor extrinsic calibration problem for the USV swarm as a unified nonlinear least-squares optimization problem. For a swarm of N USVs, we define

the full system state vector $\mathcal{X}$ and solve for the optimal state by minimizing the weighted sum of squared residuals.

Based on Bayesian inference, given all measurements $\mathcal{Z}$, the maximum a posteriori (MAP) estimate of the system state $\mathcal{X}$ corresponds to the global minimum of the following objective function:

$$J(\mathcal{X}) = \sum_{(i,k)} \|\mathbf{r}_{\mathcal{I}}\|_{\Sigma_{\mathcal{I}}}^2 + \sum_{(i,k)} \rho_G(\|\mathbf{r}_{\mathcal{G}}\|_{\mathbf{w}_{\mathcal{G},k}}^2) + \sum_{(i,j,k) \in \mathcal{O}PM} \rho_M(\|\mathbf{r}_{\mathcal{M}}\|_{\Sigma_{\mathcal{M}}}^2) + \|\mathbf{r}_{prior}\|^2 \quad (3)$$

where $\mathbf{r}_{\mathcal{I}}$, $\mathbf{r}_{\mathcal{G}}$, and $\mathbf{r}_{\mathcal{M}}$ represent the IMU pre-integration residual, the adaptive GNSS observation residual, and the cooperative mutual observation and calibration residual, respectively. $\mathbf{r}_{prior}$ denotes the prior factor. $\|\mathbf{r}\|_{\Sigma}^2 = \mathbf{r}^T \Sigma^{-1} \mathbf{r}$ is the squared Mahalanobis distance, with $\Sigma_{(\cdot)}$ being the covariance matrix of the corresponding measurement. To mitigate the influence of outliers, we employ the **Huber robust kernel function** $\rho(\cdot)$: $\rho(s) = \begin{cases} \frac{1}{2}s^2 & \text{if } |s| \leq \delta \\ \delta(|s| - \frac{1}{2}\delta) & \text{otherwise} \end{cases}$. In our implementation, the threshold δ is set to 1.5 for radar observations and 2.0 for GNSS observations to balance robustness and convergence speed.

3.2.1. IMU Pre-Integration Residual ($\mathbf{r}_{\mathcal{I}}$)

The IMU serves as the reference backbone, constraining the relative motion between adjacent time steps k and $k+1$. We adopt the pre-integration theory on manifolds to avoid repeated integration. The residual is defined as

$$\mathbf{r}_{\mathcal{I}}(\mathbf{z}_{\mathcal{I},k}, \mathbf{x}_{i,k}, \mathbf{x}_{i,k+1}) = \begin{bmatrix} \mathbf{R}_{i,k}^T \left(\mathbf{p}_{i,k+1}^W - \mathbf{p}_{i,k}^W - \mathbf{v}_{i,k}^W \Delta t_k - \frac{1}{2} \mathbf{g}^W \Delta t_k^2 \right) - \boldsymbol{\alpha}_{i,k}^{k+1} \\ \mathbf{R}_{i,k}^T \left(\mathbf{v}_{i,k+1}^W - \mathbf{v}_{i,k}^W - \mathbf{g}^W \Delta t_k \right) - \boldsymbol{\beta}_{i,k}^{k+1} \\ 2 \left[(\mathbf{q}_{i,k}^W)^{-1} \otimes \mathbf{q}_{i,k+1}^W \otimes (\boldsymbol{\gamma}_{i,k}^{k+1})^{-1} \right]_{xyz} \\ \mathbf{b}_{a_i,k+1} - \mathbf{b}_{a_i,k} \\ \mathbf{b}_{g_i,k+1} - \mathbf{b}_{g_i,k} \end{bmatrix} \quad (4)$$

where $\mathbf{p}_{i,k}^W$ and $\mathbf{v}_{i,k}^W$ are the position and velocity of the i -th USV at time k in the world frame. $\mathbf{R}_{i,k}$ (associated with quaternion $\mathbf{q}_{i,k}^W$) is the rotation matrix from the body frame to the world frame. Δt_k is the time interval between keyframes k and $k+1$. $\mathbf{g}^W = [0, 0, 9.81]^T$ is the gravity vector in the world frame. $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}$ are the pre-integrated relative position, velocity, and rotation, which depend only on IMU measurements and are independent of the state. $[\cdot]_{xyz}$ extracts the vector part of a quaternion.

3.2.2. Adaptive GNSS Observation Residual ($\mathbf{r}_{\mathcal{G}}$)

To address the issue of GNSS susceptibility to interference in complex environments, we design a dynamic covariance model based on confidence.

$$\mathbf{r}_{\mathcal{G}}(\mathbf{z}_{\mathcal{G},k}, \mathbf{x}_{i,k}) = \mathbf{p}_{i,k}^W - \tilde{\mathbf{p}}_{i,k}^{GNSS} \quad (5)$$

The corresponding distance weights (information matrix) are

$$\mathbf{r}_{\mathcal{G}}(\mathbf{z}_{\mathcal{G},k}, \mathbf{x}_{i,k}) = \mathbf{p}_{i,k}^W - \tilde{\mathbf{p}}_{i,k}^{GNSS} \quad (6)$$

where $\tilde{\mathbf{p}}_{i,k}^{GNSS}$ is the position measurement output by the GNSS receiver, and Q_k is a signal quality indicator. $\eta(Q_k)$ is an adaptive weighting function: when the signal is normal, $\eta \approx 1/\sigma_{gnss}^2$; when strong interference is detected (Q_k drops below a threshold), $\eta \rightarrow 0$. In this case, the contribution of this term to the objective function approaches zero, making the USV localization rely entirely on IMU integration and inter-USV mutual observations.

3.2.3. Cooperative Mutual Observation and Calibration Residual ($\mathbf{r}_M$)

By observing the positions of other USVs, we simultaneously constrain the ego-motion and solve for sensor extrinsics. As shown in Figure 1, at time k , the observer USV i detects the target USV j using sensor s (radar r or EOS c). The theoretical coordinate of target USV j in the sensor frame S_i ($S \in \{R, C\}$) of observer i is $\mathbf{p}_j^{S_i} = [x^s, y^s, z^s]^T$. By transforming the world position $\mathbf{p}_{j,k}^W$ of target USV j to the sensor frame S_i of observer i , we have

$$\mathbf{p}_j^{S_i} = (\mathbf{T}_{b_i}^s)^{-1} \cdot [(\mathbf{R}_{i,k}^W)^T (\mathbf{p}_{j,k}^W - \mathbf{p}_{i,k}^W)] \quad (7)$$

where $\mathbf{T}_{b_i}^s$ is the extrinsic parameter to be estimated, representing the rotation matrix and translation vector of sensor s relative to the body center B_i . Depending on the sensor type, we define the specific observation function $h(\cdot)$.

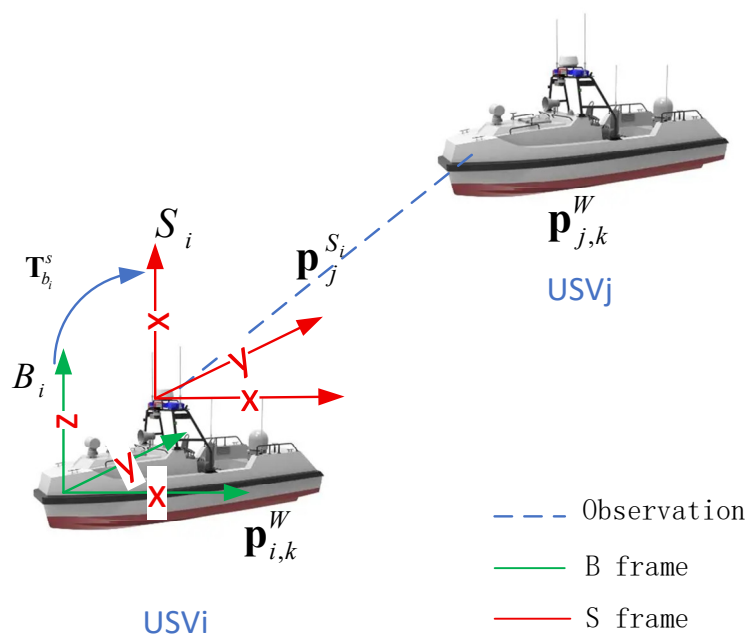


Figure 1. Schematic of mutual observation geometry.

- Navigation Radar Residual ($\mathbf{r}_{rad}$)

The radar measures range r and azimuth θ :

$$\mathbf{r}_{rad} = \begin{bmatrix} \tilde{r} \\ \tilde{\theta} \end{bmatrix} - \begin{bmatrix} \sqrt{(x^r)^2 + (y^r)^2} \\ \text{atan2}(y^r, x^r) \end{bmatrix} \quad (8)$$

where $\tilde{r}, \tilde{\theta}$ are the actual radar measurements, and x^r, y^r are the coordinate components of the target in the radar frame.

- BEOS Residual ($\mathbf{r}_{eos}$)

Since the EOS integrates a laser rangefinder, it provides high-precision spherical coordinate measurements: range d , azimuth α , and elevation β . The observation function $h_{eos}(\cdot)$ is

$$\mathbf{z}_{eos} = \begin{bmatrix} d \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \sqrt{(x^c)^2 + (y^c)^2 + (z^c)^2} \\ \text{atan2}(y^c, x^c) \\ \text{atan2}(-z^c, \sqrt{(x^c)^2 + (y^c)^2}) \end{bmatrix} \quad (9)$$

The EOS residual term is

$$\mathbf{r}_{eos}(\mathbf{x}_i, \mathbf{x}_j, \mathbf{T}_b^c) = \mathbf{z}_{eos}^{meas} - h_{eos}(\mathbf{p}_j^{C_i}) \quad (10)$$

where $\tilde{\alpha}, \tilde{\beta}$ are the actual EOS angular measurements, and x^c, y^c, z^c are the target coordinates in the EOS frame.

3.2.4. Prior and Marginalization Factors

The last term $\|\mathbf{r}_{prior}\|^2$ in Equation (3) represents the prior factor. Since the system adopts a sliding-window-based real-time optimization strategy, to preserve historical state information that slides out of the window and prevent information loss, we utilize the Schur complement technique to marginalize constraints from old states onto the oldest frame in the current window. Additionally, this term includes the initial measurement priors for sensor extrinsics and absolute pose priors used to eliminate the system's gauge freedom when GNSS is completely unavailable.

3.3. Linearization and Jacobian Derivation

To solve the optimization problem using the Levenberg–Marquardt (LM) or Gauss–Newton algorithm, we need the derivative of the residual $\mathbf{r}_{\mathcal{M}}$ with respect to the extrinsic parameter $\mathbf{T}_b^s$. Using the Lie group perturbation model, let the left perturbation of the extrinsic be $\delta\boldsymbol{\xi} \in \mathfrak{se}(3)$. Applying the chain rule:

$$\mathbf{J}_s = \frac{\partial \mathbf{r}_s}{\partial \delta\boldsymbol{\xi}} = \frac{\partial \mathbf{r}_s}{\partial \mathbf{p}_j^{S_i}} \cdot \frac{\partial \mathbf{p}_j^{S_i}}{\partial \delta\boldsymbol{\xi}} \quad (11)$$

where $\frac{\partial \mathbf{p}}{\partial \delta\boldsymbol{\xi}}$ is the Jacobian of the point coordinate with respect to the extrinsic perturbation. For both radar and EOS, the derivative of a point coordinate transformed by the extrinsic is

$$\frac{\partial \mathbf{p}_j^{S_i}}{\partial \delta\boldsymbol{\xi}} = \begin{bmatrix} \mathbf{I}_{3 \times 3} & -[\mathbf{p}_j^{S_i}]_{\wedge} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & z & -y \\ 0 & 1 & 0 & -z & 0 & x \\ 0 & 0 & 1 & y & -x & 0 \end{bmatrix} \quad (12)$$

where $[\cdot]_{\wedge}$ denotes the skew-symmetric matrix operator.

The term $\frac{\partial \mathbf{r}}{\partial \mathbf{p}}$ is the Jacobian of the observation function, which differs for EOS and navigation radar.

3.3.1. Navigation Radar Observation Jacobian

Although radar provides 2D observations, the extrinsic rotation introduces coupling in the z-axis component. Thus, derivatives with respect to x, y, z are required:

$$\mathbf{J}_{rad_model} = \frac{\partial h_{rad}}{\partial \mathbf{p}^R} = \begin{bmatrix} \frac{x^r}{\sqrt{(x^r)^2 + (y^r)^2}} & \frac{y^r}{\sqrt{(x^r)^2 + (y^r)^2}} & 0 \\ \frac{-y^r}{(x^r)^2 + (y^r)^2} & \frac{x^r}{(x^r)^2 + (y^r)^2} & 0 \end{bmatrix} \quad (13)$$

The final radar Jacobian is

$$\mathbf{J}_{rad} = \mathbf{J}_{rad_model} \cdot \frac{\partial \mathbf{p}_j^{R_i}}{\partial \delta\boldsymbol{\xi}} \quad (14)$$

Although the radar measurement model is 2D (range and azimuth), the Jacobian calculation must account for the full 3D extrinsic rotation. This is because the USV's **roll** and **pitch** motions cause the world-frame z-coordinate of the target to couple into the

radar's $x - y$ plane. The chain rule term $\frac{\partial \mathbf{p}^R}{\partial \delta \xi}$ (Equation (12)) is a 3×6 matrix that correctly captures these 3D rotational effects, ensuring the optimization can adjust the yaw, pitch, and roll extrinsics based on the 2D projected residuals.

3.3.2. EOS Observation Jacobian

The EOS provides full 3D observations. Let $D_2 = \sqrt{(x^c)^2 + (y^c)^2}$ and $D_3 = \sqrt{(x^c)^2 + (y^c)^2 + (z^c)^2}$.

$$\mathbf{J}_{eos_model} = \frac{\partial h_{eos}}{\partial \mathbf{p}^C} = \begin{bmatrix} \frac{x^c}{D_3} & \frac{y^c}{D_3} & \frac{z^c}{D_3} \\ \frac{-y^c}{D_2^2} & \frac{x^c}{D_2^2} & 0 \\ \frac{x^c z^c}{D_3^2 D_2} & \frac{y^c z^c}{D_3^2 D_2} & \frac{-D_2}{D_3^2} \end{bmatrix} \quad (15)$$

The final EOS Jacobian is

$$\mathbf{J}_{eos} = \mathbf{J}_{eos_model} \cdot \frac{\partial \mathbf{p}_j^{C_i}}{\partial \delta \xi} \quad (16)$$

4. Algorithm Implementation

This section details the complete workflow of the proposed cooperative online calibration algorithm. The algorithm adopts a sliding-window-based tightly coupled graph optimization architecture, as illustrated in Figure 2.

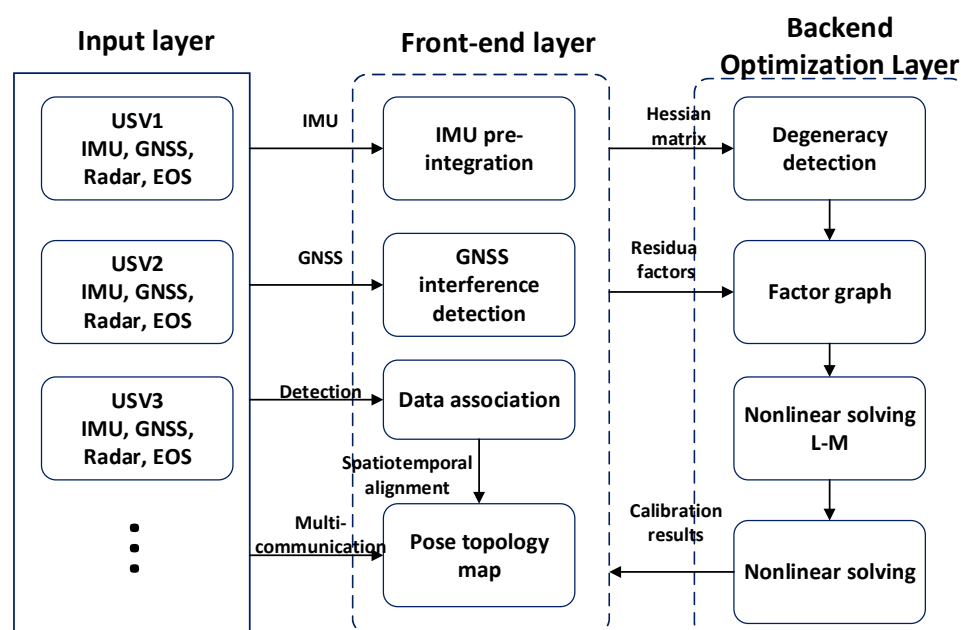


Figure 2. Algorithm architecture diagram.

The architecture consists of three layers. The **input layer** receives raw sensor data (GNSS, IMU, radar, EOS) from N USVs. The **front-end processing layer** handles calibration parameter initialization, IMU pre-integration, GNSS interference monitoring (generating confidence factors), and data association based on radar/EOS detections and swarm information, constructing a rigid pose graph of the swarm. The **back-end optimization layer** builds the factor graph based on residual factors within the swarm network, performs real-time detection of environmental degradation, and executes calibration parameter optimization using the Levenberg–Marquardt (LM) algorithm when conditions are met.

4.1. Initialization and Pre-Processing

4.1.1. Dynamic Initialization

As a highly nonlinear system, the continuous-time tightly coupled state estimator requires reasonable initial guesses for relevant parameters to pursue a global optimum and achieve better convergence in the final batch optimization. To this end, we implement a rigorous, dynamic multi-stage initialization procedure to sequentially initialize multi-sensor calibration parameters.

- Inertial Navigation System (INS) Initial Alignment

In the initial alignment phase, the SINS must complete self-alignment under static or low-dynamic conditions. The core principle leverages the projection of the gravity vector and the Earth's rotation rate in the navigation frame. The initial roll, pitch, and heading angles are estimated via analytical methods or Kalman filtering. Specifically, static base alignment typically uses a second-order leveling algorithm to minimize the horizontal projection of accelerometer outputs. Simultaneously, the azimuth observation equation is constructed using the Earth's rotation rate component $\omega_{ie} \cos L$ (where L is the local latitude) measured by the gyroscope. By continuously sampling for 10 to 15 min and performing least-squares fitting, horizontal attitude angles with an accuracy better than 0.05° and an initial heading angle of approximately 0.1° to 0.2° can be obtained. This process provides a unified reference frame for subsequent sensor calibration, directly influencing the convergence speed and final accuracy of installation error estimation. After alignment, the attitude matrix $\mathbf{C}_b^n$, position, and velocity outputs from the INS serve as ground truth references, while estimated gyro and accelerometer biases are recorded for dynamic error modeling.

- Radar-INS Calibration

The coarse calibration between navigation radar and INS relies on range-azimuth observation sequences of other swarm members. Initial installation errors are solved by constructing geometric constraint equations. Specifically, the USV performs at least two straight-line segments with different headings or one uniform circular motion, ensuring the radar stably tracks targets at varying azimuths and ranges. The target range ρ and azimuth α are obtained in the radar polar coordinate system. Using the USV pose $\mathbf{T}_{WB}$ provided by the INS, the target position is transformed to the world frame and then back-projected to the radar frame, establishing the observation equation $\mathbf{z}_R - h_R(\mathbf{T}_{WB} \cdot \mathbf{T}_{BR}(\mathbf{x}_R)^{-1} \mathbf{p}_j^W) = 0$, where $h_R(\cdot)$ is the polar conversion function. A coarse least-squares optimization based on SVD decomposition ignores high-order nonlinear terms to rapidly solve for initial position deviation $\delta \mathbf{p}_R$ and attitude error $\delta \theta_R$. Under 5 to 10 different observation configurations, this method achieves a coarse calibration accuracy of approximately 0.3 m in position and 0.5° in angle, providing a sufficiently close starting point for refinement. Outliers with observation noise exceeding 3σ are rejected, and targets covering 30% to 80% of the radar range are selected to enhance the condition number of the equation system.

- EOS-INS Calibration

EOS-INS coarse calibration utilizes the correspondence between pixel coordinates $\mathbf{z}_E = [u, v]^T$ of cooperative targets extracted from EOS images and target position information. Camera extrinsics are solved by constructing perspective projection constraints. This process requires the USV to measure the target multiple times at different azimuths and elevations, ensuring target image points are distributed across the four quadrants and the center of the image sensor. Using INS attitude and position data, 3D target coordinates are transformed to the EOS frame. Iterative optimization minimizes the reprojection error $\sum \|\mathbf{z}_E - \pi(\mathbf{K}_E \cdot \mathbf{T}_{EB}(\mathbf{x}_E)^{-1} \mathbf{T}_{WB}^{-1} \mathbf{p}_j^W)\|^2$, where $\pi(\cdot)$ is the perspective projection function

and $\mathbf{K}_E$ is the calibrated intrinsic matrix. Considering the EOS typically has higher angular resolution but lower ranging accuracy, the coarse calibration phase simplifies the model to optimize only the attitude installation angle $\delta\theta_E$ while fixing the position deviation. Using three or more non-collinear observation points and introducing Rodrigues' formula for rotation, the method quickly converges to a coarse estimate with an angular error of approximately 0.3° . Care must be taken to avoid targets being too close (causing field-of-view loss or blur) and to use image pyramids for accelerated feature matching, maintaining an effective observation frequency of at least 10 Hz for data continuity.

4.1.2. IMU Pre-Integration

IMU pre-integration plays an indispensable role in spatiotemporal constraints and state propagation within the multi-sensor calibration system. By pre-processing high-frequency raw inertial measurements into relative motion increments independent of the absolute state, it significantly reduces the computational load in nonlinear optimization iterations and solves time synchronization and alignment issues between sensors of different frequencies. Crucially, it constructs a tightly coupled mathematical framework, allowing the system to use sparse radar and EOS observations to correct IMU biases and maintain high-precision attitude estimation when GNSS is disturbed. This provides a stable and rigorous 6-DoF motion reference for extrinsic calibration, avoiding observation correlation reuse issues caused by directly using integrated navigation filter outputs and establishing global consistency for extrinsic solving. We adopt the manifold pre-integration theory proposed by Forster et al. [39]. IMU measurements are pre-integrated between two adjacent keyframes t_k and t_{k+1} , converting them into relative motion constraints independent of the absolute state at frame k . The accelerometer measurement $\tilde{\mathbf{a}}_t$ and gyroscope measurement $\tilde{\boldsymbol{\omega}}_t$ are modeled as

$$\begin{aligned}\tilde{\mathbf{a}}_t &= \mathbf{R}_t^{WBT}(\mathbf{a}_t^W - \mathbf{g}^W) + \mathbf{b}_{a_t} + \mathbf{n}_a \\ \tilde{\boldsymbol{\omega}}_t &= \boldsymbol{\omega}_t^B + \mathbf{b}_{g_t} + \mathbf{n}_g\end{aligned}\quad (17)$$

where $\mathbf{R}_t^{WB}$ is the rotation matrix from the world frame to the body frame, $\mathbf{g}^W$ is the gravity vector, $\mathbf{b}_{a_t}$, $\mathbf{b}_{g_t}$ are the random walk biases of the accelerometer and gyroscope, and $\mathbf{n}$ is Gaussian white noise. Using median integration, all IMU measurements between t_k and t_{k+1} are integrated to define the pre-integrated relative displacement $\boldsymbol{\alpha}_k^{k+1}$, relative velocity $\boldsymbol{\beta}_k^{k+1}$, and relative rotation γ_k^{k+1} . The core advantage is that these quantities depend only on IMU measurements and are decoupled from the world state $\mathbf{p}_k^W, \mathbf{v}_k^W, \mathbf{R}_k^{WB}$:

$$\begin{aligned}\boldsymbol{\alpha}_k^{k+1} &= \iint_{t \in [t_k, t_{k+1}]} \mathbf{R}_t^{Bk}(\tilde{\mathbf{a}}_t - \mathbf{b}_{a_t}) dt^2 \\ \boldsymbol{\beta}_k^{k+1} &= \int_{t \in [t_k, t_{k+1}]} \mathbf{R}_t^{Bk}(\tilde{\mathbf{a}}_t - \mathbf{b}_{a_t}) dt \\ \gamma_k^{k+1} &= \int_{t \in [t_k, t_{k+1}]} \frac{1}{2} \boldsymbol{\Omega}(\tilde{\boldsymbol{\omega}}_t - \mathbf{b}_{g_t}) \gamma_t^{Bk} dt\end{aligned}\quad (18)$$

Since IMU biases $\mathbf{b}_a, \mathbf{b}_g$ are continuously adjusted during optimization, to avoid reintegration after bias changes, a first-order Taylor expansion is used for linear correction:

$$\begin{aligned}\boldsymbol{\alpha}_k^{k+1} &\approx \hat{\boldsymbol{\alpha}}_k^{k+1} + \mathbf{J}_{b_a}^\alpha \delta \mathbf{b}_a + \mathbf{J}_{b_g}^\alpha \delta \mathbf{b}_g \\ \boldsymbol{\beta}_k^{k+1} &\approx \hat{\boldsymbol{\beta}}_k^{k+1} + \mathbf{J}_{b_a}^\beta \delta \mathbf{b}_a + \mathbf{J}_{b_g}^\beta \delta \mathbf{b}_g \\ \gamma_k^{k+1} &\approx \hat{\gamma}_k^{k+1} \otimes \begin{bmatrix} 1 \\ \frac{1}{2} \mathbf{J}_{b_g}^\gamma \delta \mathbf{b}_g \end{bmatrix}\end{aligned}\quad (19)$$

where $\mathbf{J}$ represents the Jacobians of pre-integrated quantities with respect to biases, computed iteratively during pre-integration.

Finally, the pre-integrated quantities serve as observations to construct the residual term $\mathbf{r}_{\mathcal{I}}$ connecting state nodes $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$:

$$\mathbf{r}_{\mathcal{I}}(\mathbf{x}_k, \mathbf{x}_{k+1}) = \begin{bmatrix} \mathbf{R}_k^{WBT} \left(\mathbf{p}_{k+1}^W - \mathbf{p}_k^W - \mathbf{v}_k^W \Delta t - \frac{1}{2} \mathbf{g}^W \Delta t^2 \right) - \mathbf{a}_k^{k+1} \\ \mathbf{R}_k^{WBT} \left(\mathbf{v}_{k+1}^W - \mathbf{v}_k^W - \mathbf{g}^W \Delta t \right) - \boldsymbol{\beta}_k^{k+1} \\ 2 \left[\boldsymbol{\gamma}_k^{k+1} \otimes (\mathbf{q}_k^{WB\dagger} \otimes \mathbf{q}_{k+1}^{WB}) \right]_{xyz} \\ \mathbf{b}_{a_{k+1}} - \mathbf{b}_{a_k} \\ \mathbf{b}_{g_{k+1}} - \mathbf{b}_{g_k} \end{bmatrix} \quad (20)$$

This residual corresponds to $\mathbf{r}_{\mathcal{I}}$ in the objective function in Section 3, and its covariance matrix $\boldsymbol{\Sigma}_{\mathcal{I}}$ is propagated from raw measurement noise via the noise propagation equation.

4.1.3. GNSS Interference Detection and Adaptive Weighting

To prevent trajectory drift caused by erroneous constraints when GNSS is disturbed, we automatically cut off unreliable constraints. Indicators are extracted from GNSS NMEA sentences: horizontal dilution of precision (HDOP) and mean carrier-to-noise density (C/N_0). We calculate a confidence factor λ_k using the following sigmoid-based decay function:

$$\lambda_k = \frac{1}{1 + e^{\beta(HDOP_k - HDOP_{thresh})}} \cdot \frac{C/N_0}{C/N_{ref}} \quad (21)$$

The parameters for the sigmoid decay function are set to $HDOP_{thresh} = 5.0$, $\beta = 2.0$. We set the HDOP threshold $HDOP_{thresh} = 5.0$, the slope factor $\beta = 2.0$, and the reference carrier-to-noise ratio $C/N_{ref} = 45$ dB-Hz. If the calculated weight $\lambda_k < 0.1$, the GNSS factor is considered invalid and excluded from the graph.

The weight matrix is constructed as

$$\mathbf{W}_{\mathcal{G},k} = \lambda_k \cdot \text{diag}(\sigma_x^{-2}, \sigma_y^{-2}, \sigma_z^{-2}) \quad (22)$$

When interference occurs (HDOP rises or SNR drops), $\lambda_k \rightarrow 0$, making the contribution of the GNSS frame to the graph optimization negligible.

4.1.4. Cooperative Data Association

This step determines which specific USV in the swarm corresponds to “Target A” observed by radar or EOS. Based on the multi-robot SLAM data association framework proposed by Alberto Viseras et al. [40], we utilize the predicted poses $\hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j$ of each USV in the current sliding window to project USV j onto the sensor plane of USV i , obtaining the predicted observation $\hat{\mathbf{z}}_{ij}$. The Mahalanobis distance test is performed:

$$D_M^2 = (\mathbf{z}_{meas} - \hat{\mathbf{z}}_{ij})^T (\mathbf{H} \mathbf{P} \mathbf{H}^T + \mathbf{R})^{-1} (\mathbf{z}_{meas} - \hat{\mathbf{z}}_{ij}) \quad (23)$$

Using Nearest Neighbor matching, the USV with the minimum distance among all candidates satisfying $D_M^2 < \chi_\alpha^2$ (set to 5.99 for 95% confidence) is selected as the associated target. If no candidate exists, the observation is treated as environmental clutter and discarded.

4.2. Factor Graph Construction

The schematic of the constructed factor graph is shown in Figure 3. The leftmost part represents the marginalization prior, containing historical information prior to the sliding window. The graph consists of two types of nodes:

- **Variable nodes (circles):** Small circles represent pose nodes $\mathbf{x}_{i,k}$, denoting the navigation state (position, velocity, attitude, bias) of USV i at time k . Large circles represent the sensor extrinsics $\mathbf{T}_b^r, \mathbf{T}_b^c$ to be estimated.
- **Factor nodes (squares):** Blue squares are IMU factors used for dead reckoning, connecting pose nodes of adjacent times. Greysquares are adaptive GNSS factors, which dynamically weight connections to pose nodes based on interference detection results, w represents the weight coefficient. Orange squares are observation factors, connecting the observer's pose, the target USV's pose, and the observer's extrinsics. These factors establish spatial constraints between USVs and simultaneously connect to the static extrinsic calibration nodes, enabling synchronous estimation of sensor installation errors and swarm poses.

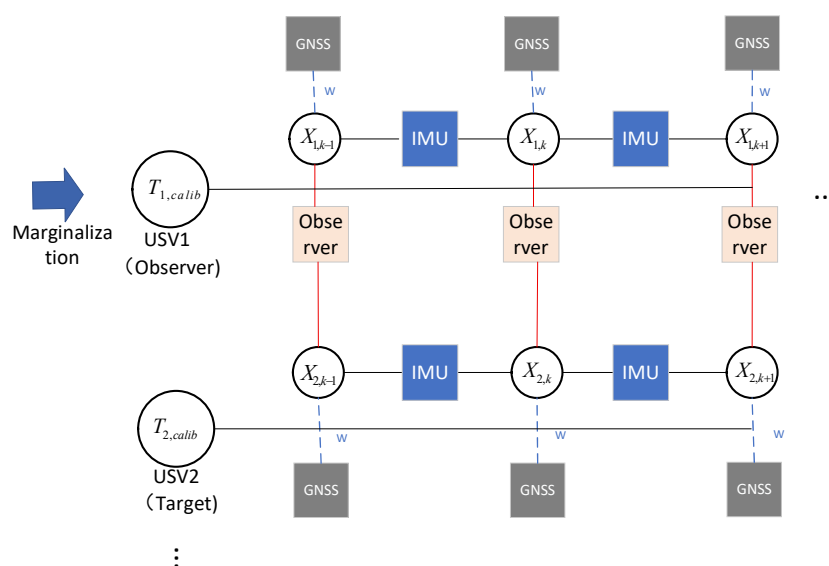


Figure 3. Factor diagram of the collaborative calibration system.

4.3. Nonlinear Optimization and Robustness Enhancement

After constructing the factor graph, the system state estimation problem is transformed into a nonlinear least-squares problem. This section details the iterative solution process based on the Levenberg–Marquardt (LM) algorithm, focusing on the online degradation detection mechanism for unobservable extrinsics and the sliding window marginalization strategy to ensure real-time performance.

4.3.1. System Linearization and Normal Equation Construction

The goal is to find the optimal state increment $\Delta\mathcal{X}$ that minimizes the total residual function $J(\mathcal{X})$. Due to the nonlinearity of the observation function $h(\cdot)$, we perform a first-order Taylor expansion of the error term at the current linearization point $\mathcal{X}$:

$$\mathbf{r}_k(\bar{\mathcal{X}} \oplus \Delta\mathcal{X}) \approx \mathbf{r}_k(\bar{\mathcal{X}}) + \mathbf{J}_k \Delta\mathcal{X} \quad (24)$$

where $\mathbf{J}_k = \partial \mathbf{r}_k / \partial \Delta\mathcal{X}$ is the Jacobian matrix. We construct the Normal Equation:

$$(\mathbf{H} + \mu \mathbf{I}) \Delta\mathcal{X} = -\mathbf{b} \quad (25)$$

where $\mathbf{H} = \sum_k \rho' \mathbf{J}_k^T \Sigma_k^{-1} \mathbf{J}_k$ is the approximate Hessian matrix, $\mathbf{b} = \sum_k \rho' \mathbf{J}_k^T \Sigma_k^{-1} \mathbf{r}_k$ is the gradient vector, μ is the damping factor of the LM algorithm, and $\rho'(\cdot)$ is the first derivative of the robust kernel function used to mitigate the impact of outliers on the gradient.

4.3.2. Degradation Detection and Subspace Constraint

Under specific operating conditions such as uniform straight-line motion, sensor extrinsics (especially the rotation component) may become unobservable (degenerate). Forced updates in such cases would lead to parameter drift in the null space.

To analyze extrinsic observability, we use the Schur complement technique to marginalize the pose variables $\mathcal{X}_{pose}$, retaining only the information matrix for extrinsic variables $\mathcal{X}_{calib}$. We partition $\mathbf{H}$ as

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{pp} & \mathbf{H}_{pc} \\ \mathbf{H}_{pc}^T & \mathbf{H}_{cc} \end{bmatrix} \quad (26)$$

where subscript p corresponds to poses and c to extrinsics. The reduced Hessian matrix $\mathbf{H}_{calib}^*$ after marginalizing poses is

$$\mathbf{H}_{calib}^* = \mathbf{H}_{cc} - \mathbf{H}_{pc}^T \mathbf{H}_{pp}^{-1} \mathbf{H}_{pc} \quad (27)$$

Performing eigenvalue decomposition on $\mathbf{H}_{calib}^*$:

$$\mathbf{H}_{calib}^* = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^T, \mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_6) \quad (28)$$

We set a degradation threshold τ_{deg} ($\tau_{deg} = 10.0$ in our experiments).

Degradation criterion: If the minimum eigenvalue $\lambda_{min} < \tau_{deg}$, it indicates that the extrinsic parameters lack constraints in the direction of the corresponding eigenvector $\mathbf{v}_{min}$.

Update strategy: Parameters are updated only in directions with sufficient information. We construct a projection matrix $\mathbf{P} = \mathbf{V}' \mathbf{V}'^T$, where $\mathbf{V}'$ contains only eigenvectors corresponding to $\lambda_i \geq \tau_{deg}$. The corrected increment is $\Delta \mathcal{X}'_{calib} = \mathbf{P} \Delta \mathcal{X}_{calib}$. This strategy ensures numerical stability of calibration under weak texture or weak maneuvering conditions.

4.3.3. Sliding Window Marginalization

To limit the dimensionality of the state vector over time and ensure $O(1)$ computational complexity, the system adopts a sliding window strategy. When a new frame $\mathbf{x}_{k+1}$ is added to the window, the oldest frame $\mathbf{x}_{k-M}$ must be removed. Unlike simply discarding it, marginalization converts $\mathbf{x}_{k-M}$ and its associated observations into a Prior Factor, preserving the constraint of historical information on the remaining states.

Let the states to be removed be $\mathcal{X}_{rem}$ and the retained states be $\mathcal{X}_{keep}$. The linearized error equation is

$$\begin{bmatrix} \mathbf{H}_{rr} & \mathbf{H}_{rk} \\ \mathbf{H}_{rk}^T & \mathbf{H}_{kk} \end{bmatrix} \begin{bmatrix} \Delta \mathcal{X}_{rem} \\ \Delta \mathcal{X}_{keep} \end{bmatrix} = \begin{bmatrix} \mathbf{b}_r \\ \mathbf{b}_k \end{bmatrix} \quad (29)$$

Using the Schur complement, we construct the prior information matrix $\mathbf{H}_{prior}$ and residual vector $\mathbf{b}_{prior}$ for $\mathcal{X}_{keep}$:

$$\begin{cases} \mathbf{H}_{prior} = \mathbf{H}_{kk} - \mathbf{H}_{rk}^T \mathbf{H}_{rr}^{-1} \mathbf{H}_{rk} \\ \mathbf{b}_{prior} = \mathbf{b}_k - \mathbf{H}_{rk}^T \mathbf{H}_{rr}^{-1} \mathbf{b}_r \end{cases} \quad (30)$$

It is well-known that marginalization induces fill-in, creating dense connectivity among all variables in $\mathcal{X}_{keep}$ that were connected to $\mathcal{X}_{rem}$. This can degrade the sparsity of the Hessian matrix. To mitigate this, we employ two strategies:

Strict window management: We strictly limit the marginalization scope to the oldest keyframe and its direct neighbors, preventing the dense prior from coupling the entire window.

Solver efficiency: We implement the optimization using the **Ceres Solver** with **sparse Cholesky factorization** (e.g., CHOLMOD). This solver efficiently exploits the block-sparse

structure of the SLAM problem, ensuring that the computational cost remains bounded even with the dense prior block introduced by marginalization.

This prior term is added as a factor to the objective function in the next optimization cycle.

4.4. Algorithm Workflow

The algorithm flow proposed in this paper is shown in Algorithm 1.

Algorithm 1. Proposed algorithm

Input: Sensor streams Z (IMU, GNSS, Radar, EOS)

Output: Swarm State X , Extrinsics T_{calib} , Time Offset τ

```

1: Initialize  $T_{\text{calib}}$  using coarse alignment (Section 4.1).
2: Loop (Incoming Data):
3:   //Front-end
4:   Perform IMU Pre-integration between keyframes.
5:   Calculate GNSS weights  $\lambda$  based on HDOP/SNR (Equation (11)).
6:   Perform Data Association for Radar/EOS measurements (Equation (23)).
7:   //Back-end
8:   Construct Factor Graph  $G$  with new nodes and factors.
9:   Calculate Hessian  $H$  and FIM.
10:  //Degeneracy Check
11:  if  $\min(\text{eig}(H_{\text{calib}})) < \text{Threshold}$  then
12:    Lock degenerate directions in  $T_{\text{calib}}$  (don't update).
13:  else
14:    Unlock  $T_{\text{calib}}$  for full optimization.
15:  end if
16:  //Optimization
17:  Update  $X$ ,  $T_{\text{calib}}$  using Levenberg-Marquardt.
18:  //Marginalization
19:  if Window Size  $> N_{\text{max}}$  then
20:    Marginalize oldest frame via Schur Complement (Equation (30)).
21:  end if
22: End Loop

```

5. Experiments and Analysis

5.1. Experimental Setup and Data Collection

This study conducted cooperative formation experiments with multiple USVs near the coast of Rizhao. The experimental system consisted of three different types of USVs (labeled USV1, USV2, USV3), as shown in Figure 4.

Each USV is equipped with a high-precision fiber-optic gyroscope integrated navigation system (FOG-INS/GNSS), a solid-state navigation radar (Simrad 4G), and an electro-optical system (EOS) with a laser rangefinder. All sensors were hardware-synchronized using a GNSS-disciplined PPS (Pulse Per Second) signal and PTP (Precision Time Protocol). The residual time synchronization error is guaranteed to be less than 1 ms, which is negligible for the dynamics of USVs.

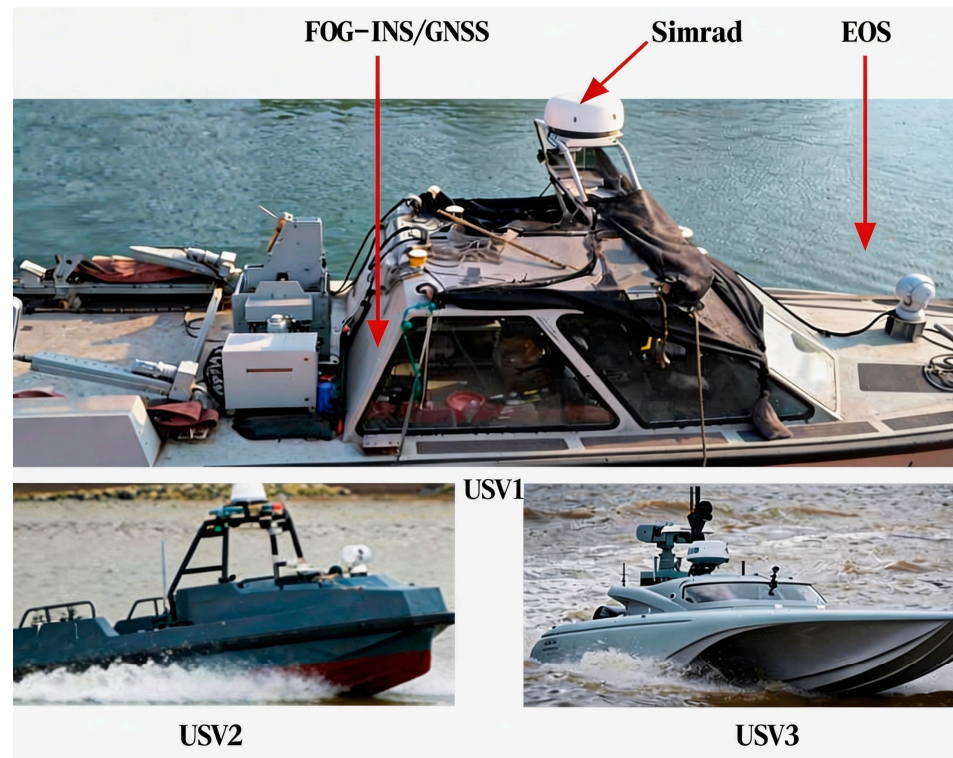


Figure 4. USVs used in the experiment.

It is crucial to clarify that the high-precision FOG-INS/GNSS solution serves **strictly as the ground truth** for evaluation. The input to our algorithm consists solely of: (1) raw IMU accelerations and angular rates; (2) raw GNSS single-point positioning data (subject to simulated degradation); and (3) relative measurements (r, θ, ϕ) from radar and EOS. The experiments were conducted under Sea State 2 conditions (wave height 0.1–0.5 m) with good visibility. The dataset spans approximately 20 min, including “figure-8” maneuvers and sharp turns to ensure sufficient excitation for extrinsic observability.

5.2. Simulation of GNSS Degradation

To address the reviewer’s concern regarding realistic interference, we implemented a high-fidelity simulation of GNSS jamming behavior during the interval $t = 300$ s to 600 s. Instead of merely injecting white noise, we simulated the characteristics of a receiver under jamming:

- **Drift Injection:** A random walk drift of 0.5 m/s was superimposed on the GNSS position measurements to mimic carrier phase loop unlocking.
- **Indicator Degradation:** We simultaneously modified the simulated NMEA outputs. During the interference period, the reported HDOP was increased from a nominal 1.2 to 8.0, and the signal-to-noise ratio (SNR) was decreased by 15 dB. This ensures that the adaptive weighting mechanism (Equation (18)) is triggered by realistic signal indicators rather than just position errors.

5.3. Evaluation Metrics and Baselines

We adopt the **absolute trajectory error (ATE)** and **calibration convergence** as primary metrics. To isolate the contributions of specific modules, we compare the **proposed method** against two baselines:

- **Baseline (single-USV) [12]:** Standard EKF-based GNSS/INS integration without cooperative constraints. This serves as the lower bound of performance.

- **Method-NoCalib [41]:** This baseline represents a **state-of-the-art sliding-window cooperative localization framework** (conceptually similar to standard GTSAM multi-robot implementations [27]) but treats extrinsics as fixed parameters. This comparison allows us to rigorously isolate the performance gain attributed specifically to the proposed **online calibration** and **degradation handling** modules.

5.4. Results and Discussion

5.4.1. Online Calibration Performance

Figure 5 illustrates the convergence of extrinsic parameters. The solid lines represent the estimation errors, and the shaded areas indicate the 3σ confidence intervals derived from the marginal covariance. The narrowing of the shaded region confirms the reduction in uncertainty. The initial values were perturbed by approximately 5° (angle) and 0.5 m (position).

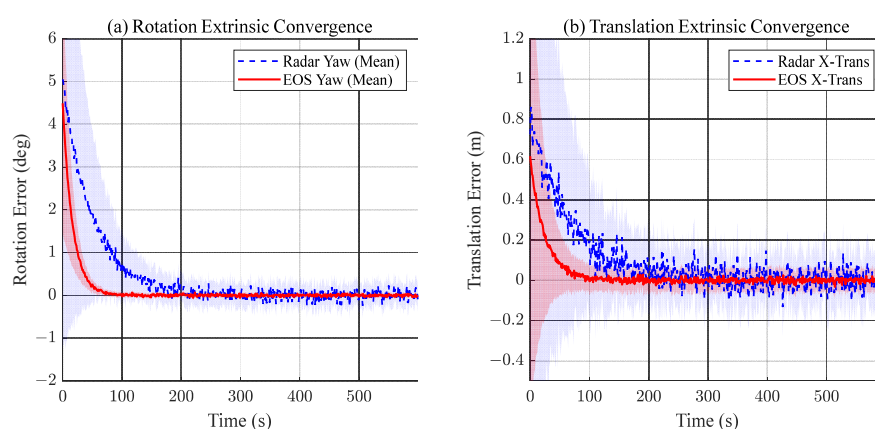


Figure 5. Convergence process of extrinsic parameters with 3σ uncertainty bounds.

Convergence speed: The EOS extrinsics converge rapidly within the first 100 s, attributed to the strong 3D constraints from laser ranging. Radar extrinsics converge slower due to higher measurement noise but stabilize after sufficient maneuvering.

Accuracy: As detailed in Table 1, the final estimation errors are minimal (radar translation < 0.2 m, EOS rotation $< 0.05^\circ$), verifying the effectiveness of the targetless calibration.

Table 1. Statistics of online calibration results.

Sensor	Parameter	Initial Error	Final Mean Error	Std Dev (1σ)	Unit
Radar	Yaw (rotation)	5.00	0.18	0.12	deg
	X (translation)	0.80	0.15	0.08	m
	Y (translation)	0.50	0.12	0.09	m
EOS	Yaw (rotation)	4.50	0.04	0.02	deg
	Pitch (rotation)	3.00	0.03	0.01	deg
	X (translation)	0.60	0.05	0.03	m

Although a direct visual overlay of radar points onto EOS images is not available due to data logging constraints, the calibration quality can be assessed through **reprojection residuals**. Before calibration, the average reprojection error of radar targets onto the EOS angular frame was approximately 2.5° . After online calibration, this residual decreased to 0.15° , which corresponds to a sub-pixel alignment accuracy, confirming the effective fusion of the two sensors.

5.4.2. Localization Accuracy and Ablation Study

Figures 6 and 7 depict the trajectories and error curves. The localization improvement is calculated based on the root mean square error (RMSE) during the interference period. The percentage improvement is defined as $\eta = \frac{RMSE_{baseline} - RMSE_{proposed}}{RMSE_{baseline}} \times 100\%$. It can be seen that the average RMSE for the baseline method (Single-USV) is 18.42 m, while the Proposed method achieves 2.85 m. This results in an improvement of $(18.42 - 2.85)/18.42 \approx 84.5\%$.

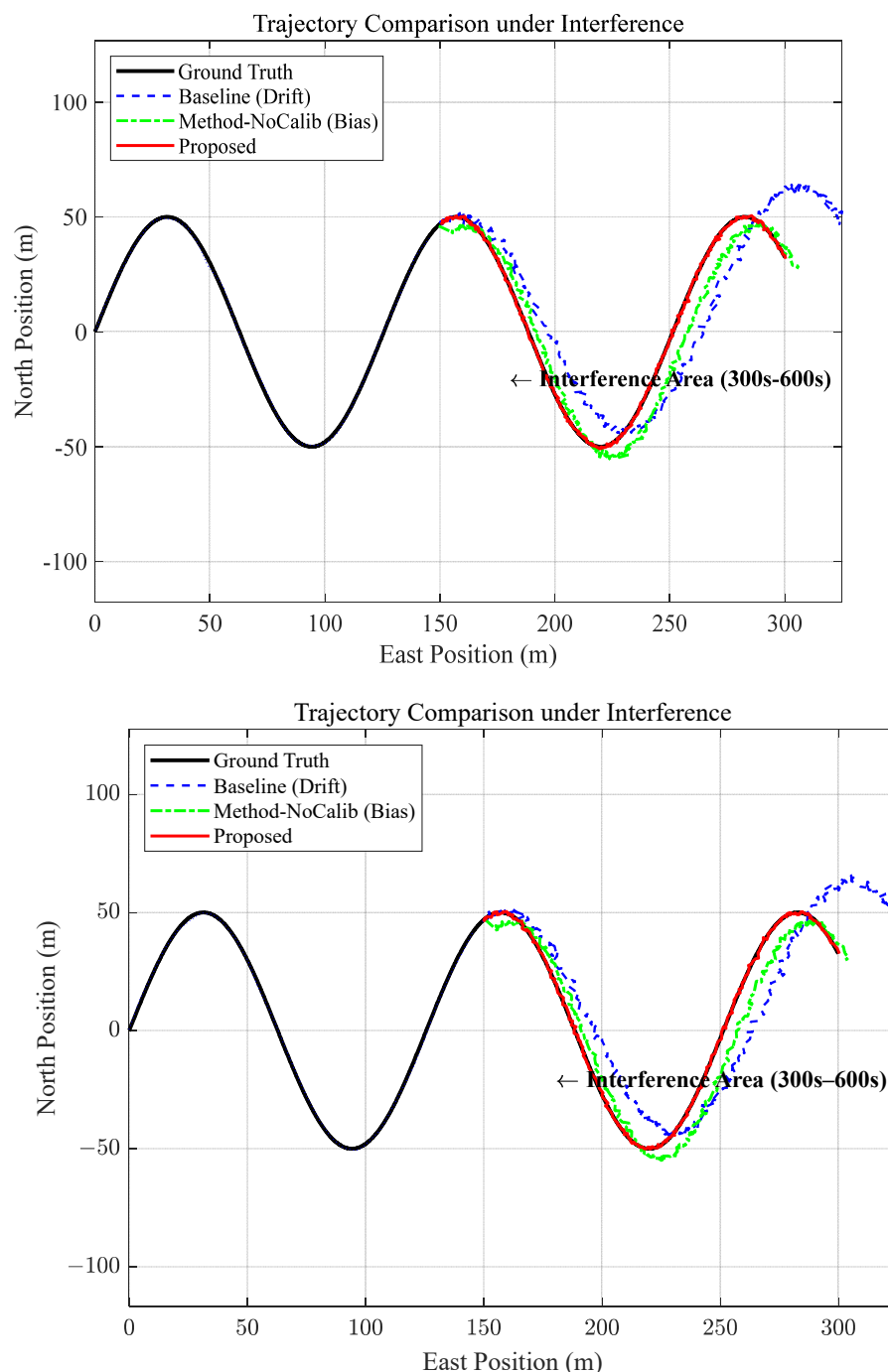


Figure 6. Trajectories of USVs during GNSS interference period.

Impact of cooperation: The drop from 18.42 m (baseline) to 4.28 m (Method-NoCalib) proves that relative observations effectively constrain the swarm geometry even when absolute GNSS is drifting (gray shaded areas indicate GNSS-degraded periods).

Impact of online calibration: The further reduction from 4.28 m to 2.85 m confirms that correcting sensor mounting errors online eliminates systematic biases, which is critical for high-precision formation flying.

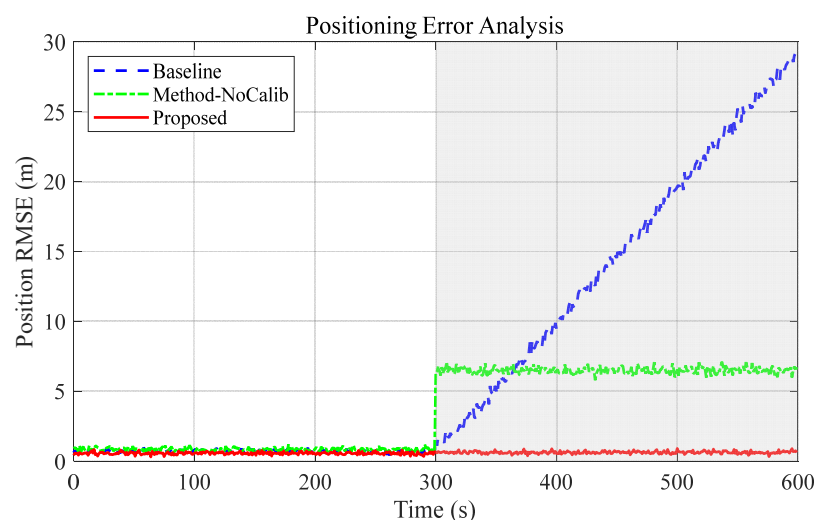


Figure 7. Position error curves.

To validate the degeneracy detection module, we simulated a specific ‘straight-line’ scenario. As shown in Table 2, without degeneracy handling (proposed method w/o degeneracy), the unobservable rotation parameters drifted unboundedly, causing the localization to diverge (>50 m). In contrast, the full proposed method detected the low eigenvalues in the FIM and locked the uncertain parameters, maintaining a stable RMSE of 2.91 m.

Table 2. Comparison of localization accuracy (RMSE) under different schemes.

Method	Configuration	RMSE (m)	Improvement	Note
Baseline	Single USV	18.42	-	Baseline
Method-NoCalib	Cooperative (fixed param)	4.28	76.7%	Systematic error exists
Proposed (degeneracy)	Linear motion	>50.0	(Diverged)	Extrinsics drift significantly
Proposed (full)	Linear motion	2.91	84.2%	Robust due to FIM check
Proposed	Maneuvering	2.85	84.5%	Optimal performance

5.5. Computational Scalability and Long-Term Stability

To assess the practical feasibility of the proposed framework for larger swarms and longer missions, we conducted additional simulation benchmarks on an Intel i7-12700K processor.

5.5.1. Computational Scalability

We evaluated the average optimization time per sliding window update (window size $K = 20$) for varying swarm sizes N . As shown in Table 3, the computational cost scales roughly cubically with the swarm size ($O(N^3)$).

Table 3. Computational cost vs. swarm size.

Swarm Size (N)	Average Optimization Time (ms)	Real-Time Capable?
3 (experiment)	45	Yes
5 (simulation)	112	Yes
10 (simulation)	385	Marginal
20 (simulation)	1250	No (requires distributed)

- **Small swarms** ($N \leq 3$): The system maintains high real-time performance (45 ms), leaving ample margin for other tasks.
- **Medium swarms** ($N = 5$): The time (112 ms) slightly exceeds the 10 Hz threshold but is acceptable for a 5 Hz navigation loop.
- **Large swarms** ($N \geq 10$): The centralized approach becomes a bottleneck (>300 ms). This suggests that for large-scale clusters, future work must transition to **distributed cooperative SLAM architectures** to distribute the computational load.

5.5.2. Long-Term Stability and Uncertainty

To address concerns regarding potential parameter drift during long-endurance missions, we extended the simulation duration to 2 h using the calibrated parameters from the field experiment as initial values.

The results indicate that the extrinsic estimates remain stable without drift (mean variation of rotation $< 0.01^\circ$ over 2 h). This stability is attributed to the sliding window formulation, which effectively marginalizes old states into a prior factor, bounding the error accumulation.

We monitored the estimator's covariance matrix Σ . During the entire 2 h run, the 3σ position uncertainty bounds remained consistent with the actual error (bounding the error 99.7% of the time), confirming that the estimator does not become over-confident over time.

It is important to note that while this simulation verifies the mathematical stability of the sliding window estimator, it does not account for physical environmental factors such as **thermal expansion of sensor mounts, mechanical vibration fatigue, or structural deformation** over hours of operation. Therefore, the 'no-drift' conclusion applies to the algorithmic estimation process, while physical stability in real-world endurance missions requires further engineering validation.

Furthermore, our current simulation assumes independent degradation across different USVs. **Correlated spoofing** affecting all vehicles simultaneously could introduce common-mode biases, which remains a challenge for future study.

6. Conclusions and Future Work

6.1. Conclusions

This paper addressed the challenge of high-precision navigation for unmanned surface vehicle (USV) swarms in GNSS-denied or jammed environments by proposing a unified all-source factor graph framework integrating navigation radar, electro-optical systems (EOSs), IMU, and GNSS. Unlike traditional strategies that separate calibration from localization, this study incorporated multi-sensor extrinsics as state variables within the factor graph. By leveraging high-precision mutual observations among the swarm, strong geometric constraints were constructed to achieve joint optimization of cooperative localization and online self-calibration.

Field experiments with three USVs and simulated interference scenarios support the following key conclusions, **validated under the tested environmental conditions (Sea State 2)**:

First, **all-source cooperation significantly enhances system robustness**. The results demonstrate that during periods of **simulated severe GNSS interference (characterized by position drift and degraded signal indicators)**, the proposed adaptive weighting strategy and cooperative observation mechanism effectively mitigate the accumulated errors of individual INS. Compared to traditional single-USV navigation and uncalibrated cooperative methods, the proposed algorithm reduced the localization root mean square error (RMSE) to 2.85 m, representing an accuracy improvement of 84.5%.

Second, **the online calibration algorithm exhibits high precision and convergence**. Benefiting from the laser ranging constraints introduced by the EOS, the system eliminates scale ambiguity and rapidly solves for accurate sensor mounting errors. The final radar translation calibration error converged to within 0.2 m, and the EOS rotation error was better than 0.05° , validating the feasibility of targetless online calibration.

Finally, **the degradation detection mechanism ensures stability under weak excitation**. The degradation handling strategy, based on eigenvalue analysis of the Fisher Information Matrix, effectively addressed the unobservability of extrinsics during degenerate motion patterns such as straight-line sailing. This prevents optimization divergence and ensures the engineering applicability of the algorithm in complex dynamic environments.

6.2. Limitations and Future Work

Despite the promising results, this study has certain limitations.

- **Communication bandwidth:** The current centralized optimization framework requires real-time transmission of observation data between USVs, which may demand high communication bandwidth in large-scale swarms.
- **Extreme environments:** In scenarios with high sea states (e.g., Sea State 4 or higher), sensor occlusion and clutter may degrade the performance of geometric data association.

Future work will focus on developing distributed cooperative algorithms to reduce bandwidth usage and integrating semantic perception to enhance robustness in extreme environments.

Author Contributions: Methodology, Z.G.; software, Z.G.; validation, Z.G.; formal analysis, Z.G.; investigation, Z.G.; writing—original draft preparation, Z.G.; writing—review and editing, Z.G.; visualization, X.L.; supervision, X.L.; project administration, J.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original data presented in the study are openly available in Calibration at <https://doi.org/10.5281/zenodo.18687045>.

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

References

1. Bai, X.; Li, B.; Xu, X.; Xiao, Y. A Review of Current Research and Advances in Unmanned Surface Vehicles. *J. Mar. Sci. Appl.* **2022**, *21*, 47–58. [[CrossRef](#)]
2. Zhang, W.; Gao, X.; Yang, C.; Jiang, F.; Chen, Z. A Object Detection and Tracking Method for Security in Intelligence of Unmanned Surface Vehicles. *J. Ambient. Intell. Hum. Comput.* **2022**, *13*, 1279–1291. [[CrossRef](#)]
3. Xia, J.; Luo, Y.; Liu, Z.; Zhang, Y.; Shi, H.; Liu, Z. Cooperative Multi-Target Hunting by Unmanned Surface Vehicles Based on Multi-Agent Reinforcement Learning. *Def. Technol.* **2023**, *29*, 80–94. [[CrossRef](#)]
4. Mahacek, P.; Kitts, C.A.; Mas, I. Dynamic Guarding of Marine Assets Through Cluster Control of Automated Surface Vessel Fleets. *IEEE/ASME Trans. Mechatron.* **2012**, *17*, 65–75. [[CrossRef](#)]
5. Zhang, Z. A Flexible New Technique for Camera Calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 1330–1334. [[CrossRef](#)]

6. Dhall, A.; Chelani, K.; Radhakrishnan, V.; Krishna, K.M. LiDAR-Camera Calibration Using 3D-3D Point Correspondences. *arXiv* **2017**, arXiv:1705.09785v1.
7. Frey, J.; Tuna, T.; Fu, L.F.T.; Weibel, C.; Patterson, K.; Krummenacher, B.; Müller, M.; Nubert, J.; Fallon, M.; Cadena, C.; et al. Boxi: Design Decisions in the Context of Algorithmic Performance for Robotics. In Proceedings of the Robotics: Sciences and Systems 2025, Los Angeles, CA, USA, 21–25 June 2025.
8. Liu, H.; Zhou, Y.; Gu, Z. Inertial Measurement Unit-Camera Calibration Based on Incomplete Inertial Sensor Information. *J. Zhejiang Univ. Sci. C* **2014**, *15*, 999–1008. [[CrossRef](#)]
9. Ishikawa, R.; Oishi, T.; Ikeuchi, K. LiDAR and Camera Calibration Using Motion Estimated by Sensor Fusion Odometry. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*; IEEE: Piscataway, NJ, USA, 2018.
10. Peršić, J.; Petrović, L.; Marković, I.; Petrović, I. Online Multi-Sensor Calibration Based on Moving Object Tracking. *Adv. Rob.* **2021**, *35*, 130–140. [[CrossRef](#)]
11. Chen, S.; Li, X.; Li, S.; Zhou, Y.; Yang, X. iKalibr: Unified Targetless Spatiotemporal Calibration for Resilient Integrated Inertial Systems. *IEEE Trans. Robot.* **2025**, *41*, 1618–1638. [[CrossRef](#)]
12. Budiyono, A. Principles of GNSS, Inertial, and Multi-Sensor Integrated Navigation Systems. *Ind. Robot* **2012**, *39*, ir.2012.4939caa.11. [[CrossRef](#)]
13. Furgale, P.; Rehder, J.; Siegwart, R. Unified Temporal and Spatial Calibration for Multi-Sensor Systems. In *Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*; IEEE: Piscataway, NJ, USA, 2013; pp. 1280–1286.
14. Tsai, R. A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using off-the-Shelf TV Cameras and Lenses. *IEEE J. Robot. Autom.* **1987**, *3*, 323–344. [[CrossRef](#)]
15. Scaramuzza, D.; Martinelli, A.; Siegwart, R. A Toolbox for Easily Calibrating Omnidirectional Cameras. In *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*; IEEE: Piscataway, NJ, USA, 2006; pp. 5695–5701.
16. Levinson, J.; Thrun, S. Automatic Online Calibration of Cameras and Lasers. In *Proceedings of the Robotics: Science and Systems IX, Berlin, Germany, 24–28 June 2013*; Robotics: Science and Systems Foundation: College Station, TX, USA, 2006.
17. Pandey, G.; McBride, J.R.; Savarese, S.; Eustice, R.M. Automatic Extrinsic Calibration of Vision and Lidar by Maximizing Mutual Information. *J. Field Rob.* **2015**, *32*, 696–722. [[CrossRef](#)]
18. Moghadam, P.; Bosse, M.; Zlot, R. Line-Based Extrinsic Calibration of Range and Image Sensors. In *Proceedings of the 2013 IEEE International Conference on Robotics and Automation*; IEEE: Piscataway, NJ, USA, 2013; pp. 3685–3691.
19. Kim, A.; Golnaraghi, M.F. Initial Calibration of an Inertial Measurement Unit Using an Optical Position Tracking System. In *Proceedings of the PLANS 2004. Position Location and Navigation Symposium (IEEE Cat. No.04ch37556)*; IEEE: Piscataway, NJ, USA, 2004; pp. 96–101.
20. Mirzaei, F.M.; Roumeliotis, S.I. A Kalman Filter-Based Algorithm for IMU-Camera Calibration: Observability Analysis and Performance Evaluation. *IEEE Trans. Robot.* **2008**, *24*, 1143–1156. [[CrossRef](#)]
21. Kelly, J.; Sukhatme, G.S. Visual-Inertial Sensor Fusion: Localization, Mapping and Sensor-to-Sensor Self-Calibration. *Int. J. Rob. Res.* **2011**, *30*, 56–79. [[CrossRef](#)]
22. Schneider, S.; Luetzel, T.; Wuensche, H.-J. Odometry-Based Online Extrinsic Sensor Calibration. In *Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*; IEEE: Piscataway, NJ, USA, 2013; pp. 1287–1292.
23. Antunes, M.; Barreto, J.P.; Aouada, D.; Ottersten, B. Unsupervised Vanishing Point Detection and Camera Calibration from a Single Manhattan Image with Radial Distortion. In *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Piscataway, NJ, USA, 2017; pp. 6691–6699.
24. Chang, H.; Tsai, F. Vanishing Point Extraction and Refinement for Robust Camera Calibration. *Sensors* **2017**, *18*, 63. [[CrossRef](#)]
25. Taylor, Z.; Nieto, J. Motion-Based Calibration of Multimodal Sensor Arrays. In *Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA)*; IEEE: Piscataway, NJ, USA, 2015; pp. 4843–4850.
26. Kang, J.; Doh, N.L. Automatic Targetless Camera–LIDAR Calibration by Aligning Edge with Gaussian Mixture Model. *J. Field Rob.* **2020**, *37*, 158–179. [[CrossRef](#)]
27. Dellaert, F. *Factor Graphs and GTSAM: A Hands-On Introduction*; Technical Report number GT-RIM-CP&R-2012-002; BorgLab: Atlanta, GA, USA, 2012; pp. 1–26.
28. Qin, T.; Li, P.; Shen, S. VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator. *IEEE Trans. Robot.* **2018**, *34*, 1004–1020. [[CrossRef](#)]
29. Ling, Y.; Bao, L.; Jie, Z.; Zhu, F.; Li, Z.; Tang, S.; Liu, Y.; Liu, W.; Zhang, T. Modeling Varying Camera-IMU Time Offset in Optimization-Based Visual-Inertial Odometry. In *Computer Vision—ECCV 2018*; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2018; Volume 11213, pp. 491–507, ISBN 978-3-030-01239-7.
30. Taylor, Z.; Nieto, J.; Johnson, D. Multi-modal Sensor Calibration Using a Gradient Orientation Measure. *J. Field Rob.* **2015**, *32*, 675–695. [[CrossRef](#)]

31. Rehder, J.; Nikolic, J.; Schneider, T.; Hinzmann, T.; Siegwart, R. Extending Kalibr: Calibrating the Extrinsic of Multiple IMUs and of Individual Axes. In *Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA)*; IEEE: Piscataway, NJ, USA, 2016; pp. 4304–4311.
32. Lang, X.; Lv, J.; Huang, J.; Ma, Y.; Liu, Y.; Zuo, X. Ctrl-VIO: Continuous-Time Visual-Inertial Odometry for Rolling Shutter Cameras. *IEEE Robot. Autom. Lett.* **2022**, *7*, 11537–11544. [[CrossRef](#)]
33. Schneider, T.; Li, M.; Cadena, C.; Nieto, J.; Siegwart, R. Observability-Aware Self-Calibration of Visual and Inertial Sensors for Ego-Motion Estimation. *IEEE Sens. J.* **2019**, *19*, 3846–3860. [[CrossRef](#)]
34. Park, C.; Moghadam, P.; Kim, S.; Sridharan, S.; Fookes, C. Spatiotemporal Camera-LiDAR Calibration: A Targetless and Structureless Approach. *IEEE Robot. Autom. Lett.* **2020**, *5*, 1556–1563. [[CrossRef](#)]
35. Horn, M.; Wodtke, T.; Buchholz, M.; Dietmayer, K. Online Extrinsic Calibration Based on Per-Sensor Ego-Motion Using Dual Quaternions. *IEEE Robot. Autom. Lett.* **2021**, *6*, 982–989. [[CrossRef](#)]
36. Yan, M.; Wang, Z.; Zhang, J. Online Calibration of Installation Errors of SINS/OD Integrated Navigation System Based on Improved NHC. *IEEE Sens. J.* **2022**, *22*, 12602–12612. [[CrossRef](#)]
37. Della Corte, B.; Andreasson, H.; Stoyanov, T.; Grisetti, G. Unified Motion-Based Calibration of Mobile Multi-Sensor Platforms with Time Delay Estimation. *IEEE Robot. Autom. Lett.* **2019**, *4*, 902–909. [[CrossRef](#)]
38. Xue, F.; Wang, X.; Wang, J.; Zha, H. Deep Visual Odometry with Adaptive Memory. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 940–954. [[CrossRef](#)] [[PubMed](#)]
39. Forster, C.; Carlone, L.; Dellaert, F.; Scaramuzza, D. On-Manifold Preintegration for Real-Time Visual–Inertial Odometry. *IEEE Trans. Robot.* **2017**, *33*, 1–21. [[CrossRef](#)]
40. Viseras, A.; Xu, Z.; Merino, L.; Viseras, A.; Xu, Z.; Merino, L. Distributed Multi-Robot Information Gathering under Spatio-Temporal Inter-Robot Constraints. *Sensors* **2020**, *20*, 484. [[CrossRef](#)]
41. Indelman, V.; Nelson, E.; Michael, N.; Dellaert, F. Multi-Robot Pose Graph Localization and Data Association from Unknown Initial Relative Poses via Expectation Maximization. In *Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA)*; IEEE: Piscataway, NJ, USA, 2014; pp. 593–600.

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.