Journal of
Marine Science and Engineering

MDPI

Article

Multi-Source Sensor Fusion Localization Method for Autonomous Underwater Vehicles Based on Deep Learning

Xin Pan [1], Guoli Feng [1], Haiyan Zeng [1] and Qunhong Tian [2,*]

[1] College of Electrical Engineering, Naval University of Engineering, Wuhan 430033, China; 1920191171@nue.edu.cn (X.P.); 0902121001@nue.edu.cn (G.F.); 1610061131@nue.edu.cn (H.Z.)
[2] College of Ocean Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China
* Correspondence: tianqunhong@sdust.edu.cn

Abstract

Autonomous Underwater Vehicles (AUVs) are increasingly used in deep-sea exploration, environmental monitoring, and marine engineering. Their operational safety and mission performance rely heavily on accurate and long-endurance underwater localization. However, both single-sensor localization methods and existing multi-sensor fusion approaches have inherent limitations, making it difficult to achieve high-precision localization during long-duration missions. To address this issue, this study develops a deep-learning-based multi-source sensor fusion framework for AUV localization. In the proposed framework, high-frequency data from the Inertial navigation system (INS) and Doppler velocity log (DVL) are used for continuous position propagation, while low-frequency absolute position observations from the Ultra-short baseline (USBL) system and Sonar are used to periodically correct the propagated results. Based on this framework, three instantiated models are developed using a Deep neural network (DNN), a Long short-term memory (LSTM) network, and a Bayesian semi-supervised mixed shallow-layer neural network (BSsMSLNN), respectively. Comparative experiments are conducted against the Extended Kalman filter (EKF) and Simultaneous localization and mapping system using Sonar, Visual, Inertial, and Depth sensor (SVIn2). The results show that the proposed framework effectively suppresses long-term error accumulation and significantly improves localization accuracy. Among the evaluated models, the BSsMSLNN-based method achieves the best performance in terms of trajectory fitting, root mean square error (RMSE), and coefficient of determination (R^2). The proposed method provides a feasible solution for high-precision autonomous navigation of AUVs in GPS-denied environments.

Keywords: autonomous underwater vehicle; multi-source sensor fusion; underwater localization; integrated navigation; deep learning

Check for updates

Academic Editor: Eugen Rusu

Received: 29 April 2026
Revised: 31 May 2026
Accepted: 4 June 2026
Published: 5 June 2026

1. Introduction

The ocean contains abundant biological, mineral, and energy resources and has therefore become a strategic focus for many maritime nations [1]. With the continued advancement of marine resource exploration and scientific investigation, Autonomous Underwater Vehicles (AUVs), owing to their flexibility and high degree of autonomy, have become important platforms for deep-sea exploration, environmental monitoring, and marine engineering operations [2,3]. Because of the complexity and variability of the underwater environment, the operational safety of AUVs depends critically on navigation and positioning systems that can provide high accuracy over long-duration missions [4]. Since electromagnetic waves attenuate rapidly in water, the Global navigation satellite system (GNSS) cannot be directly used for underwater localization [5]. As a result, accurate AUV localization mainly depends on the acquisition and fusion of information from onboard and external sensors.

Existing AUV localization methods based on a single sensor can generally be divided into three categories: inertial-navigation-based methods, acoustic-navigation-based methods, and geophysical-navigation-based methods. Inertial navigation relies on an Inertial navigation system (INS), which uses onboard accelerometers and gyroscopes and operates independently of external infrastructure while providing a high update rate. However, because measurement errors are integrated over time through the navigation equations, INS-based localization inevitably suffers from cumulative drift [6]. Acoustic navigation achieves localization by transmitting and receiving acoustic signals via sonar equipment, including specifically Doppler velocity log (DVL)-based methods [7] and underwater acoustic positioning system-based methods [8]. The DVL uses the Doppler effect to calculate the velocity of the vehicle relative to the seafloor. However, this method is highly sensitive to changes in AUV attitude and is also subject to the risk of beam failure [9]. Underwater acoustic positioning systems, such as Ultra-short baseline (USBL) and sonar-based positioning systems, can provide relatively accurate position information [10]. Nevertheless, their update rates are much lower than those of INS and DVL, and their accuracy usually degrades as the distance between the AUV and the beacon or transponder increases [11]. Geophysical navigation estimates the current position by matching measured physical-field information, such as magnetic or gravity data, to a pre-surveyed reference map. However, such methods are difficult to apply reliably in regions where the field gradients are weak.

Because each single-sensor localization method has inherent limitations, it is difficult for any one of them to satisfy the requirements of long-term, autonomous, and high-precision underwater localization. Consequently, multi-source sensor fusion has become a major research direction in AUV navigation [12]. Existing multi-source sensor fusion approaches can generally be broadly grouped into filter-based fusion ones, optimization-based fusion ones, and deep-learning-based fusion ones.

Among filter-based methods, the Extended Kalman Filter (EKF) extends the classical Kalman filter to nonlinear systems by using first-order linearization [13]. However, this approximation makes the EKF sensitive to initial-state errors and limits its estimation accuracy in highly nonlinear scenarios. The Unscented Kalman Filter (UKF) improves nonlinear approximation by propagating sigma points through the nonlinear model [14], while the Cubature Kalman Filter (CKF) uses cubature points to approximate high-dimensional Gaussian integrals [15]. Although these methods can improve nonlinear state estimation to some extent, their performance may still degrade in the presence of non-Gaussian noise, acoustic outliers, and ocean-current disturbances. For example, a robust Kalman filtering method that integrates filtering with M-estimation is proposed, thereby improving robustness to model errors and uncertainty in noise statistics [16]. Huang et al. proposed the Statistical regression adaptive Kalman filter (SRAKF), which combines the Expectation-Maximization algorithm with the unscented transform to improve localization performance under time-varying noise [17]. Shi et al. designed a variational Bayesian robust adaptive Kalman filter (VBRAKF) to improve robustness to non-Gaussian noise [18]. Despite these advances, filter-based methods still face significant challenges in underwater localization scenarios characterized by both strong nonlinearity and time-varying noise.

Optimization-based methods formulate AUV state estimation as a mathematical optimization problem and obtain the maximum a posteriori estimate by minimizing a global or local cost function. These methods have been widely applied in simultaneous localization and mapping (SLAM). For example, Dellaert et al. transformed state estimation into a sparse nonlinear least-squares problem by constructing sparse factor graphs and proposed the square root SAM framework for large-scale SLAM [19]. However, SAM is computationally expensive and is therefore not well suited to online applications. To improve efficiency, Kaess et al. proposed iSAM, which performs incremental optimization [20]. Nevertheless, SAM-type methods still rely on least-squares optimization and are sensitive to outliers. To address this issue, Yang et al. proposed a general global estimation framework that combines graduated non-convexity (GNC) with non-minimal solvers [21]. In addition, the above underwater SLAM systems suffer from localization drift and localization failure. To address this problem, Rahman et al. proposed an underwater SLAM system based on sonar, vision, inertial, and depth sensors, namely SVIn2, which is equipped with loop-closure detection and relocalization capabilities [22]. Bathymetric and magnetic-beacon-assisted SLAM methods have also been investigated to reduce horizontal drift in INS/DVL navigation [23,24]. Although optimization-based methods are effective in many scenarios, they usually involve heavy computational costs, and their performance may degrade when the optimization problem is non-convex or when strict real-time requirements must be satisfied.

Deep learning has been widely applied in industrial processes [25,26], and advanced models have been developed to address the measurement noise and multimodal characteristics of industrial data, such as the Bayesian semi-supervised mixed shallow-layer neural network (BSsMSLNN) [27]. In the early stage, deep learning was mainly used as an auxiliary component in single-sensor localization methods. Deep neural network (DNN) has been used to estimate AUV velocity or compensate for navigation errors when some measurements are unavailable [28]. With the development of models capable of learning temporal dependencies, such as Recurrent neural networks (RNNs) and Long short-term memory (LSTM) networks, researchers began to apply end-to-end learning methods to AUV localization. For example, Mu et al. proposed a hybrid-RNN-based AUV localization framework that processes multi-source sensor data in parallel using unidirectional and bidirectional LSTMs, achieving favorable accuracy while satisfying real-time requirements [29]. However, this method still relies implicitly on system modeling and state estimation, and therefore cannot completely eliminate model-related approximation errors. To address this limitation, Zhang et al. proposed NavNet, which combines an RNN with an attention mechanism to directly regress displacement from raw data [30]. Building on this idea, Zhang et al. further proposed a sequence-learning localization method based on a Temporal convolutional network and self-attention to compensate for the errors of state-estimation algorithms such as the EKF and UKF [31]. However, these methods generally process inputs from different sensors in a unified manner, which may overlook sensor-specific local characteristics. To address this issue, Zhang et al. proposed a hybrid multi-branch framework that uses one-dimensional convolutional neural networks to extract local features from each sensor independently, thereby enabling more refined heterogeneous feature learning [32]. In addition, underwater SLAM systems based on image enhancement and multi-sensor coupling have been reported [33]. Despite these advances, most deep-learning-based multi-source fusion methods still rely on recursively accumulated displacement estimates and therefore remain vulnerable to long-term error accumulation.

Recent studies and reviews have also emphasized the importance of sensor-fusion robustness and learning-assisted perception in underwater SLAM and AUV navigation [34,35].

In summary, existing multi-source sensor fusion methods still exhibit several limitations in AUV localization with heterogeneous sensor update rates and intermittent low-frequency observations. It should be noted that the general idea of high-frequency propagation and low-frequency correction is related to classical integrated-navigation principles. The contribution of this work lies in developing a learning-based multi-source fusion localization framework for heterogeneous AUV sensor data. The framework consists of two modules: a high-frequency recursive prediction module and a low-frequency observation correction module. The first module performs continuous position propagation using INS and DVL data, whereas the second periodically corrects the propagated results using low-frequency absolute position observations from USBL and sonar. In this way, the framework is specifically designed to maintain continuous localization output under sparse acoustic observations while suppressing accumulated propagation errors. Within this framework, three representative models based on DNN, LSTM, and BSsMSLNN are instantiated and evaluated against EKF and SVIn2 through comparative experiments. In addition, depth-related experiments are conducted to analyze the influence of operating-depth variation on model generalization. The real-water experimental scenarios adopted in this study are controllable and repeatable, which helps acquire reliable multi-source data and provides a clear basis for validating the effectiveness of the proposed fusion localization framework.

The main contributions of this study are summarized as follows: (1) a deep-learning-based multi-source fusion localization framework is developed for AUVs under heterogeneous sensor update rates, where high-frequency INS/DVL measurements are combined with sparse low-frequency USBL/sonar observations for continuous localization; (2) three representative learning architectures, namely DNN, LSTM, and BSsMSLNN, are systematically instantiated and compared under the same fusion framework and experimental protocol; and (3) a depth-related generalization analysis is conducted to study the influence of operating-depth variation on localization performance and to evaluate the benefit of incorporating depth-aware information into underwater localization models.

2. Deep-Learning-Based Multi-Source Sensor Fusion Framework

2.1. Multi-Source Sensor Fusion Framework

The underwater motion of an AUV is highly nonlinear and is affected by ocean-current disturbances, sensor noise, and variations in operating conditions, which introduce considerable uncertainty into motion-state estimation. Meanwhile, different sensors exhibit substantial differences in measurement principles, update frequencies, and error characteristics, making it difficult for any single sensor to simultaneously ensure localization continuity and accuracy. To address these problems, a multi-source sensor fusion framework for AUV localization that combines high-frequency recursive propagation with low-frequency observation correction is proposed.

The data sources used in this study include the INS, DVL, USBL, Sonar, and the Global positioning system (GPS). It is worth mentioning that the GPS is used only to provide the reference trajectory during the experimental stage and is not involved in the online prediction process. Specifically, at time step t, the INS input is denoted by $\boldsymbol{I}_t \in \mathbb{R}^9$, including heading angle, pitch angle, roll angle, three-axis angular velocities, and three-axis accelerations. The DVL input is denoted by $\boldsymbol{D}_t \in \mathbb{R}^3$, including the transverse, longitudinal, and normal velocities relative to the seabed. The USBL input is denoted by $\boldsymbol{U}_t \in \mathbb{R}^2$, and the Sonar input is denoted by $\boldsymbol{S}_t \in \mathbb{R}^2$, both of which provide external position observations. The GPS data are used only for ground-truth annotation and performance evaluation and are not involved in the online prediction process.

In terms of signal characteristics, INS and DVL provide continuous measurements at relatively high update frequencies, making them suitable for short-term position propagation. By contrast, USBL and sonar operate at lower update frequencies because of acoustic propagation delays, signal-processing latency, and environmental constraints. However, they are able to provide absolute position observations and are therefore used to suppress error accumulation during recursive propagation. After temporal alignment, USBL and Sonar observations are unavailable at some time instants, which increases the difficulty of unified multi-source fusion. To make the heterogeneous-frequency data relationship clearer, Figure 1 has been revised to include the full names of the sensor signals, their sampling frequencies, and the availability of observations at different time instants.

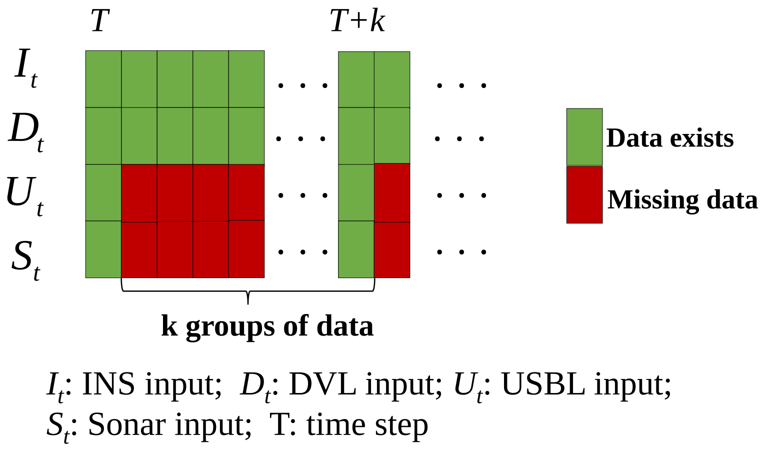


Figure 1. Correspondence diagram of heterogeneous-frequency sensor signals. INS and DVL provide high-frequency measurements, whereas USBL and sonar provide low-frequency absolute observations at discrete correction instants. Green blocks indicate available observations, and red blocks indicate missing observations.

As shown in Figure 1, under a unified time axis, high-frequency sensors such as INS and DVL provide observations at every sampling instant, whereas low-frequency sensors such as USBL and Sonar provide valid position observations only at discrete intervals. Assume that the sampling period of the low-frequency signals is K times that of the high-frequency signals. Then, between two adjacent low-frequency observation instants T and $T+K$, only high-frequency signals are available from time $T+1$ to time $T+K-1$, while low-frequency absolute position observations are absent. Therefore, the key challenge in multi-source sensor fusion is how to maintain continuous position output during intervals without low-frequency observations while effectively suppressing the accumulated propagation error once low-frequency observations become available.

To solve this problem, the proposed multi-source fusion framework is constructed as shown in Figure 2. The framework consists of two modules: a high-frequency recursive prediction module and a low-frequency observation correction module. The first module is designed for periods when low-frequency observations are unavailable and uses high-frequency INS and DVL data for continuous position propagation. The second module is activated when USBL and Sonar observations are available and introduces low-frequency absolute position information to correct the propagated result. In this way, the framework forms a coordinated localization mechanism of high-frequency propagation and low-frequency correction.

During the recursive prediction stage, only high-frequency INS and DVL data are used, and the input structure is defined as

$$\boldsymbol{x}_t^h = [\boldsymbol{I}_t, \boldsymbol{D}_t], \tag{1}$$

where x_t^h denotes the high-frequency input vector at time instant t, I_t represents the INS measurements, and D_t represents the DVL measurements.

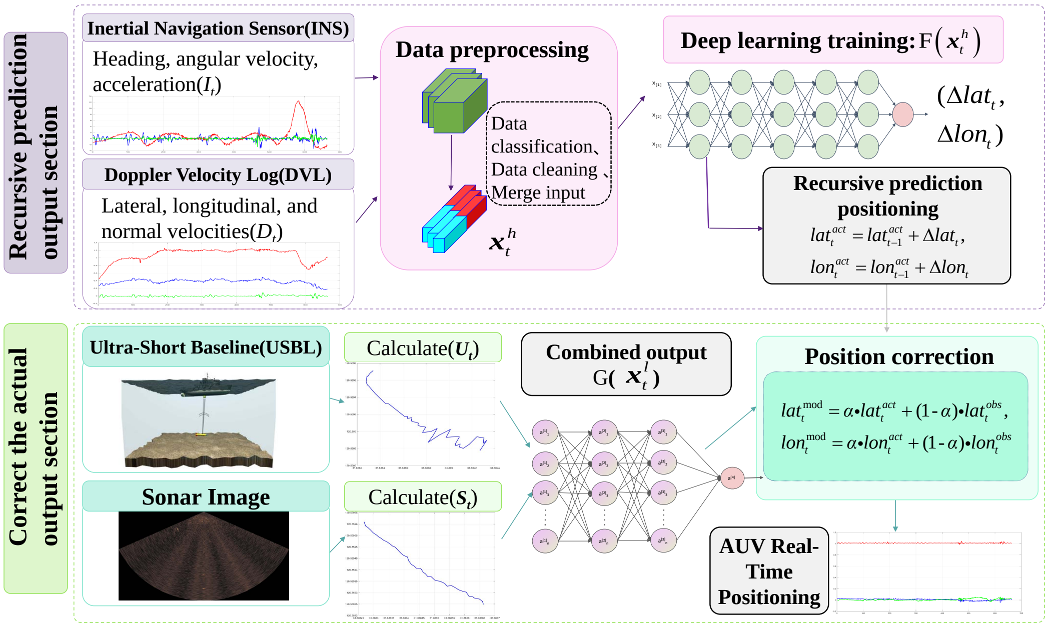


Figure 2. Schematic diagram of the fusion framework.

Since no position constraints from USBL or Sonar are available during this stage, the current AUV position must be continuously estimated from high-frequency motion information alone. Specifically, the INS and DVL measurements collected at the same time step are combined into an input vector, which is then fed into the selected deep learning model to predict the latitude and longitude increments between two consecutive time steps:

$$\Delta \mathrm{lat}_t, \Delta \mathrm{lon}_t = \mathrm{F}(\boldsymbol{x}_t^h), \tag{2}$$

where $F(\cdot)$ denotes the instantiated learning model, and Δlat_t and Δlon_t denote the predicted latitude and longitude increments between two adjacent time steps, respectively.

Subsequently, the predicted increments are combined with the corrected output position from the previous time step to obtain the recursively predicted position at the current time step,

$$\mathrm{lat}_t^{\mathrm{act}} = \mathrm{lat}_{t-1}^{\mathrm{act}} + \Delta \mathrm{lat}_t, \tag{3}$$

$$\mathrm{lon}_t^{\mathrm{act}} = \mathrm{lon}_{t-1}^{\mathrm{act}} + \Delta \mathrm{lon}_t, \tag{4}$$

where $\mathrm{lat}_t^{\mathrm{act}}$ and $\mathrm{lon}_t^{\mathrm{act}}$ denote the predicted latitude and longitude at the current time step, respectively, and $\mathrm{lat}_{t-1}^{\mathrm{act}}$ and $\mathrm{lon}_{t-1}^{\mathrm{act}}$ denote the predicted latitude and longitude at the previous time step.

When low-frequency observations become available, the framework enters the correction stage. In addition to the high-frequency recursive prediction result, low-frequency absolute position observations from USBL and Sonar are introduced to correct the current position. The low-frequency input at this stage is defined as

$$\boldsymbol{x}_t^l = [\boldsymbol{U}_t, \boldsymbol{S}_t], \tag{5}$$

where x_t^l denotes the low-frequency observation vector at time instant t, U_t represents the USBL observation, and S_t represents the sonar observation.

Because both USBL and Sonar provide absolute position references, while differing in measurement accuracy and environmental adaptability, these two sources are first combined into a unified low-frequency position observation:

$$\mathrm{lat}_t^{\mathrm{obs}}, \mathrm{lon}_t^{\mathrm{obs}} = \mathrm{G}(\boldsymbol{x}_t^l) = \beta \boldsymbol{U}_t + (1-\beta)\boldsymbol{S}_t, \tag{6}$$

where $\mathrm{G}(\cdot)$ denotes the weighted fusion function for USBL and sonar observations; $\mathrm{lat}_t^{\mathrm{obs}}$ and $\mathrm{lon}_t^{\mathrm{obs}}$ denote the fused low-frequency latitude and longitude observations, respectively; and $\beta \in [0,1]$ is the weight used to balance the USBL and sonar observations. Similar to α, β is determined by the differential evolution algorithm using K-fold cross-validation on the training set with the objective of minimizing the average validation RMSE. Once the optimal value of β is obtained, it remains fixed during the testing stage and is not updated online with time. The sonar-derived position is obtained by extracting the target echo or feature point from the sonar observation, converting it into a range-bearing measurement according to the sonar imaging geometry, and transforming it into the local navigation coordinate system before being fused with the USBL observation.

Subsequently, the fused low-frequency observation is integrated with the high-frequency recursive prediction result through weighted fusion to obtain the final output at the current time step:

$$\mathrm{lat}_t^{\mathrm{mod}} = \alpha \cdot \mathrm{lat}_t^{\mathrm{act}} + (1-\alpha) \cdot \mathrm{lat}_t^{\mathrm{obs}}, \tag{7}$$

$$\mathrm{lon}_t^{\mathrm{mod}} = \alpha \cdot \mathrm{lon}_t^{\mathrm{act}} + (1-\alpha) \cdot \mathrm{lon}_t^{\mathrm{obs}}, \tag{8}$$

where $\alpha \in [0,1]$ denotes the fusion weight used to balance the contributions of recursive prediction and low-frequency observations, and $\mathrm{lat}_t^{\mathrm{mod}}$ and $\mathrm{lon}_t^{\mathrm{mod}}$ denote the corrected latitude and longitude outputs, respectively. The value of α is determined by the differential evolution algorithm using K-fold cross-validation on the training set, with the objective of minimizing the average RMSE over the validation folds. The testing trajectories are not involved in determining α. Once the optimal α is obtained, it is fixed during the testing stage and is not updated online at each sampling instant.

It should be noted that the final output obtained in the correction stage is used not only as the localization result at the current time step but also as the reference input for recursive updating at the next time step. In this way, the framework forms a closed-loop recursive process along the temporal dimension. During intervals in which low-frequency observations are unavailable, continuous localization is achieved through high-frequency propagation; when low-frequency observations become available, absolute position information is used to periodically correct the accumulated error, and the corrected result is then fed back into the next round of recursive propagation. This mechanism enables the framework to maintain both short-term continuity and long-term stability.

Based on the above design, Algorithm 1 summarizes the implementation procedure of the proposed multi-source sensor fusion localization framework. First, the initial true position of the AUV is used as the starting point for recursive propagation. At each time step, the available INS, DVL, USBL, and Sonar data are collected to construct the corresponding input vectors. If the current time step falls within an interval in which low-frequency observations are unavailable, only INS and DVL inputs are used, and the deep learning model predicts the position increment for recursive propagation. If low-frequency observations are available, the framework further fuses USBL and Sonar observations to obtain the corrected final output. This output is then used as the reference for the next recursive update, and the above process is repeated until the localization of the entire trajectory is completed.

Algorithm 1 Deep Learning-Based Multi-Sensor Fusion Localization Framework

Input High-frequency sensor data $\{\boldsymbol{I}_t, \boldsymbol{D}_t\}_{t=1}^{N}$, low-frequency sensor data $\{\boldsymbol{U}_t, \boldsymbol{S}_t\}_{t=1}^{N}$, fusion weights α and β.

Output Estimated longitude and latitude positions of the AUV at each time instant $\left\{\mathrm{lat}_t^{\mathrm{pre}}, \mathrm{lon}_t^{\mathrm{pre}}\right\}_{t=1}^{N}$.

Initialize the initial position of the AUV: $(\mathrm{lat}_0, \mathrm{lon}_0)$.
for $t = 1, 2, \cdots, N$ **do**
  Read the high-frequency sensor data at the current time instant $\boldsymbol{I}_t, \boldsymbol{D}_t$, construct the high-frequency input vector $\boldsymbol{x}_t^h = [\boldsymbol{I}_t, \boldsymbol{D}_t]$
  Predict the position increment between consecutive time instants $\Delta\mathrm{lat}_t, \Delta\mathrm{lon}_t$ according to Equation (2)
  Calculate the recursively predicted position at the current time instant $\mathrm{lat}_t^{\mathrm{act}}, \mathrm{lon}_t^{\mathrm{act}}$ according to Equations (3) and (4) is available
  if Low-frequency observation available at current time instant **then**
    Read the low-frequency sensor data $\boldsymbol{U}_t, \boldsymbol{S}_t$ at the current time instant and construct the low-frequency input vector $\boldsymbol{x}_t^l = [\boldsymbol{U}_t, \boldsymbol{S}_t]$
    Combine the USBL and sonar observations to obtain the low-frequency position measurement $\mathrm{lat}_t^{\mathrm{obs}}, \mathrm{lon}_t^{\mathrm{obs}}$
    Fuse the recursively predicted result with the low-frequency observation according to Equations (7) and (8) to obtain the corrected output $\mathrm{lat}_t^{\mathrm{mod}}, \mathrm{lon}_t^{\mathrm{mod}}$
    Obtain the final predicted position $\mathrm{lat}_t^{\mathrm{pre}} = \mathrm{lat}_t^{\mathrm{mod}}, \quad \mathrm{lon}_t^{\mathrm{pre}} = \mathrm{lon}_t^{\mathrm{mod}}$ at time t
  else
    The final predicted position $\mathrm{lat}_t^{\mathrm{pre}} = \mathrm{lat}_t^{\mathrm{act}}, \mathrm{lon}_t^{\mathrm{pre}} = \mathrm{lon}_t^{\mathrm{act}}$ at time t
  end if
end for

2.2. Instantiated Model Configuration

In the proposed multi-source fusion framework, the learning model is responsible for estimating the position increment of the AUV from high-frequency INS/DVL measurements. Three representative models, namely DNN, LSTM, and BSsMSLNN, are implemented under the same fusion framework for comparison. It should be noted that the three models share the same input–output definition and evaluation protocol, but they do not completely share the same training objective. The DNN and LSTM models are trained by minimizing the mean squared error (MSE), whereas the BSsMSLNN model is trained by maximizing the evidence lower bound (ELBO) within a semi-supervised Bayesian inference framework.

For all instantiated models, the high-frequency input at time instant t is denoted as

$$x_t^h = [I_t, D_t], \tag{9}$$

where I_t and D_t denote the INS and DVL measurements, respectively. The output of the learning model is the two-dimensional position increment in the local metric coordinate system, denoted as

$$\Delta\hat{\mathbf{p}}_t = [\Delta\hat{x}_t, \Delta\hat{y}_t]^T. \tag{10}$$

The recursively predicted position is then obtained by accumulating the predicted position increments over time. The detailed network structures, training parameters, and hyperparameter settings used in the experiments are summarized in Section 3.1.

2.2.1. DNN-Based Implementation

The DNN-based implementation is adopted as a representative feedforward neural network model. It directly maps the current high-frequency input x_t^h to the position increment $\Delta\hat{\mathbf{p}}_t$ through multiple fully connected layers. The hidden layers are used to approximate the nonlinear relationship between the INS/DVL measurements and the AUV motion increment, while the output layer contains two neurons corresponding to the predicted increments in the two horizontal directions.

The DNN can be expressed as

$$\Delta\hat{\mathbf{p}}_t = f_{\mathrm{DNN}}(x_t^h; \Theta_{\mathrm{DNN}}), \tag{11}$$

where $f_{\mathrm{DNN}}(\cdot)$ denotes the nonlinear mapping represented by the fully connected neural network, and Θ_{DNN} denotes the trainable network parameters. The DNN model is trained by minimizing the MSE between the predicted and reference position increments:

$$\mathcal{L}_{\mathrm{DNN}} = \frac{1}{N}\sum_{t=1}^{N} \|\Delta\hat{\mathbf{p}}_t - \Delta\mathbf{p}_t\|_2^2, \tag{12}$$

where N is the number of training samples, $\Delta\hat{\mathbf{p}}_t$ is the predicted position increment, and $\Delta\mathbf{p}_t$ is the reference position increment calculated from the reference trajectory.

2.2.2. LSTM-Based Implementation

The LSTM-based implementation is adopted as a representative recurrent neural network model. Different from the DNN, the LSTM uses a sliding-window sequence of high-frequency INS/DVL measurements as input, so that temporal dependencies among consecutive samples can be considered. The input sequence at time instant t is written as

$$\mathcal{X}_t^h = \{x_{t-L+1}^h, x_{t-L+2}^h, \ldots, x_t^h\}, \tag{13}$$

where L denotes the sequence length. Through its memory cell and gating mechanisms, the LSTM extracts temporal features related to AUV motion dynamics. The final hidden representation is then mapped to the two-dimensional position increment through a fully connected output layer:

$$\Delta\hat{\mathbf{p}}_t = f_{\mathrm{LSTM}}(\mathcal{X}_t^h; \Theta_{\mathrm{LSTM}}), \tag{14}$$

where $f_{\mathrm{LSTM}}(\cdot)$ denotes the LSTM-based mapping function, and Θ_{LSTM} denotes the trainable parameters. Similarly to the DNN, the LSTM model is trained by minimizing the MSE loss:

$$\mathcal{L}_{\mathrm{LSTM}} = \frac{1}{N}\sum_{t=1}^{N} \|\Delta\hat{\mathbf{p}}_t - \Delta\mathbf{p}_t\|_2^2. \tag{15}$$

2.2.3. BSsMSLNN-Based Implementation

The BSsMSLNN-based implementation is adopted to enhance local nonlinear modeling capability under limited labeled trajectory data. BSsMSLNN is a semi-supervised Bayesian mixture of multiple shallow neural networks. Each shallow neural network acts as a local expert for a specific region of the input space, and the final prediction is obtained by adaptively combining the outputs of different local experts. This "divide-and-conquer" structure is suitable for AUV localization because the relationship between sensor measurements and position increments may vary under different motion states and trajectory segments.

Let K denote the number of local shallow neural networks. For the k-th local shallow network, the predicted position increment is denoted as $\Delta\hat{\mathbf{p}}_t^{(k)}$. A latent variable is introduced to describe the contribution of the current sample to each local network. The posterior probability of the k-th local network for the sample at time instant t is denoted as δ_{tk}, satisfying $\sum_{k=1}^{K} \delta_{tk} = 1$. The final prediction of BSsMSLNN is then obtained by posterior-weighted fusion:

$$\Delta \hat{\mathbf{p}}_t = \sum_{k=1}^{K} \delta_{tk} \Delta \hat{\mathbf{p}}_t^{(k)}. \tag{16}$$

Differently from the DNN and LSTM models, BSsMSLNN is not trained by directly minimizing the MSE loss. Instead, it is trained within a semi-supervised Bayesian learning framework. The model contains latent variables and Bayesian random parameters, whose posterior distributions are estimated by variational inference. Meanwhile, the connection weights of the shallow neural networks are updated by gradient ascent. The training objective is to maximize the evidence lower bound (ELBO), which is expressed as

$$\mathcal{L}_{\text{ELBO}} = \mathbb{E}_{q(\Omega)}[\ln p(\mathcal{D}, \Omega \mid W)] - \mathbb{H}_{q(\Omega)}[\Omega], \tag{17}$$

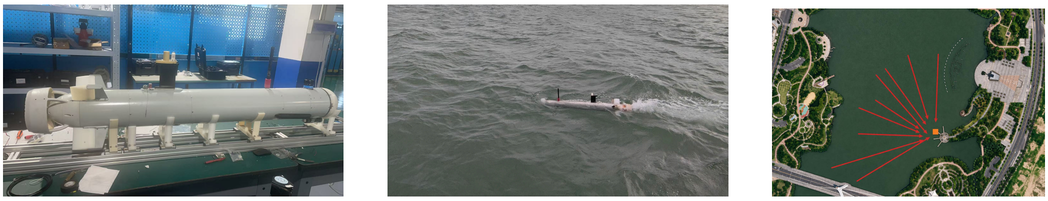
where $\mathcal{D}$ denotes the training data, Ω denotes the set of latent variables and Bayesian random parameters, W denotes the connection weights of the shallow neural networks, $q(\Omega)$ is the variational posterior distribution, and $\mathbb{H}_{q(\Omega)}[\Omega]$ is the entropy term. During training, the posterior distributions $q(\Omega)$ are updated by variational inference, while the network weights W are updated by gradient ascent to increase the ELBO. When reliable prior information is unavailable, the hyperparameters of the prior distributions are set to small values, corresponding to non-informative priors. This setting reduces the dependence on subjective prior assumptions and allows the local model contributions and network parameters to be mainly learned from the experimental data.

3. Experiments and Analysis

3.1. Experimental Setup and Evaluation Metrics

To verify the effectiveness of the proposed deep-learning-based multi-source sensor fusion framework and to compare the localization performance of different instantiated models, AUV localization experiments were conducted under different trajectory scenarios and water-depth conditions. The experimental scenarios were designed in a controllable real-water environment, which facilitates reliable acquisition of synchronized multi-source sensor data and provides a clear basis for evaluating the proposed high-frequency propagation and low-frequency correction mechanism. To provide a clearer description of the experimental conditions, the physical AUV platform and the field-test arrangement are shown in Figure 3. Figure 3a shows the AUV platform used in the experiment, where the onboard navigation sensors and communication modules were integrated into the cylindrical vehicle body. Figure 3b presents the AUV during the field test, indicating that the vehicle operated in a real water environment rather than under purely laboratory conditions. Figure 3c illustrates the voyage arrangement and observation layout during the experiment. The red arrows indicate the planned navigation directions of different voyages, while the marked region represents the reference observation area used for data acquisition and trajectory evaluation. These experimental settings provide the basis for collecting heterogeneous multi-source navigation data under practical operating conditions. It is should be noted that the background map in Figure 3c is used only for visualization of the experimental site and planned route; it is not used as an input to the proposed localization algorithm.

During the experiments, multiple types of data were collected: INS, DVL, USBL, Sonar, and GPS. Among them, INS and DVL data were recorded in real time by the AUV platform and were mainly used for position estimation during the high-frequency propagation stage. USBL and sonar data were obtained from external observation systems and were used for position correction during the low-frequency correction stage. RTK-GPS was used to obtain the reference latitude and longitude coordinates of the AUV during the experiments. Compared with standard GPS, RTK-GPS provides higher positioning accuracy and a higher update frequency, which makes it more suitable for constructing the reference trajectory used for quantitative evaluation. The RTK-GPS data were used only for reference trajectory construction and performance evaluation and were not involved in the online localization process. The operating depth of the vehicle was measured by the depth sensor, while the underwater motion states, including velocity, attitude, angular velocity, and acceleration, were obtained from the INS and DVL sensors. According to the current experimental platform configuration, INS and DVL were recorded at 20 Hz, whereas the low-frequency acoustic observations were recorded at approximately 1 Hz. Therefore, the experimental data effectively reflect the heterogeneous-frequency multi-source fusion scenario considered in this study.



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

Figure 3. Experimental platform and field-test trajectory design: (**a**) physical AUV platform and sensor installation; (**b**) AUV field-test process in the water; (**c**) schematic diagram of the planned voyage for visualizing the experimental site and planned route.

Based on the above sensor data, this study collected voyage data at three water depths: 0.3 m, 0.6 m, and 0.9 m. The raw data were preprocessed before model training and testing. These values refer to the immersion depth of the AUV, namely the vertical distance from the water surface to the reference point of the vehicle body, rather than the bottom depth of the experimental area. The bottom depth of the test area was sufficient to ensure safe vehicle operation and valid sensor measurement during the tests.

The raw data were preprocessed before model training and testing. Abnormal records and invalid samples were removed, and the remaining data were divided at the trajectory level. Complete trajectories were treated as independent units, where approximately 80% of the trajectories were selected as the training set and the remaining 20% were used as the testing set. K-fold cross-validation was adopted on the training trajectories for parameter selection. The testing trajectories were only used for final localization-performance evaluation. Data collected at an operating depth of 0.3 m were mainly used for performance comparison among different models, whereas data collected at depths of 0.6 m and 0.9 m were used to investigate the effect of depth variation on model prediction performance.

To demonstrate the effectiveness of the proposed framework, two categories of comparison methods were selected in the experiments. The first category consists of three data-driven instantiated models constructed within the proposed fusion framework, namely DNN [26], LSTM [30], and BSsMSLNN [27]. The second category comprises representative conventional fusion methods, including the EKF [13] and the multi-sensor fusion SLAM system SVIn2 [22]. The former was used to compare the adaptability of different deep learning models under a unified framework, whereas the latter was used to evaluate the performance difference between the proposed framework and conventional fusion methods in complex navigation scenarios.

To ensure reproducibility, the network architectures and training settings of the three instantiated learning models are summarized in Table 1. All models use the same training/testing trajectory split and the same data normalization strategy. The input variables are normalized using the mean and standard deviation calculated from the training set, and the same normalization parameters are applied to the validation and testing sets. K-fold cross-validation is performed on the training trajectories for parameter selection. The testing trajectories are not involved in model training, hyperparameter selection, or fusion-weight optimization.

For the BSsMSLNN-based implementation, the hyperparameters of the prior distributions are set to small values when no reliable prior information is available, corresponding to non-informative priors. This setting reduces the dependence on subjective prior assumptions and allows the model parameters and local-mode weights to be mainly learned from the experimental data.

Table 1. Network architectures and training hyperparameters of the instantiated learning models.

Item	DNN	LSTM	BSsMSLNN
Input	INS + DVL measurements	INS + DVL sequence	INS + DVL measurements
Output	2-D position increment	2-D position increment	2-D position increment
Input dimension	d_h	$L \times d_h$	d_h
Network type	Fully connected neural network	Recurrent neural network	Bayesian mixture of shallow NNs
Network architecture	d_h–128–64–32–2	LSTM layer + FC layer	3 local single-hidden-layer NNs
Number of hidden layers	3	1 LSTM layer	1 for each local NN
Hidden units	128, 64, and 32	64	20 for each local NN
Activation function	ReLU	Tanh/Sigmoid gates	Sigmoid
Sequence length	–	10	–
Initial number of local models	–	–	3
Prior setting	–	–	Non-informative priors
Training objective	MSE minimization	MSE minimization	ELBO maximization
Training algorithm	Adam optimizer	Adam optimizer	Variational inference + gradient ascent
Learning rate	1.0×10^{-3}	1.0×10^{-3}	1.0×10^{-3}
Batch size	64	64	–
Maximum epochs/iterations	300	300	300
Stopping criterion	Validation RMSE	Validation RMSE	ELBO convergence
Parameter selection	K-fold cross-validation	K-fold cross-validation	K-fold cross-validation

Quantitative assessments of localization efficacy were conducted through the application of the root mean square error (RMSE) and coefficient of determination (R^2), which served as diagnostic tools to measure prediction accuracy and model fit, respectively. To provide a more intuitive and practical evaluation of localization accuracy, the starting point of each AUV trajectory was used as the origin of a local planar coordinate system. The latitude and longitude coordinates of both the reference trajectory and the predicted trajectory were then converted into local metric coordinates relative to this origin. Based on the converted coordinates, the localization errors and RMSE values were calculated in meters.

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(y_n - \hat{y}_n)^2}, \tag{18}$$

$$\mathrm{R}^2 = 1 - \frac{\sum\limits_{n=1}^{N}(y_n - \hat{y}_n)^2}{\sum\limits_{n=1}^{N}\left(y_n - \frac{1}{N}\sum\limits_{n=1}^{N} y_n\right)^2}, \tag{19}$$

where y_n denotes the true value of the sample at the n-th sampling instant in the testing stage, $\hat{y}_n$ denotes the estimated value of the sample at the n-th sampling instant in the testing stage, and N denotes the total number of samples in the testing stage. RMSE is used to measure the overall deviation between the predicted trajectory and the ground-truth trajectory; the smaller its value, the smaller the localization error of the model. R^2 is used to evaluate the degree to which the predicted results fit the variation trend of the ground-truth trajectory; the attainment of values nearing 1 is indicative of optimal model calibration and data correspondence.

The summary of the experimental scenario and dataset division is shown in Table 2.

Table 2. Summary of the experimental scenario and data usage.

Item	Description
Test environment	Shallow real-water experimental area
Platform	AUV experimental platform shown in Figure 3
Operating depths	0.3 m, 0.6 m, and 0.9 m immersion depths
High-frequency sensors	INS and DVL, recorded at 20 Hz
Low-frequency sensors	USBL and sonar, recorded at approximately 1 Hz
Reference data	RTK-GPS for reference latitude/longitude and depth sensor for operating depth
Data split	Trajectory-level split, with approximately 80% of trajectories for training and 20% for testing
Evaluation metrics	RMSE in meters and dimensionless R^2

3.2. AUV Localization Results and Discussion

To first evaluate the recursive prediction capability of the instantiated models under the proposed framework, no low-frequency correction information from USBL or Sonar was introduced (i.e., α is set to 1). Under this setting, the predicted trajectory was obtained entirely through high-frequency propagation based on INS and DVL data. Figures 4 and 5 present the recursive prediction results of DNN, LSTM, and BSsMSLNN under two representative trajectory scenarios.

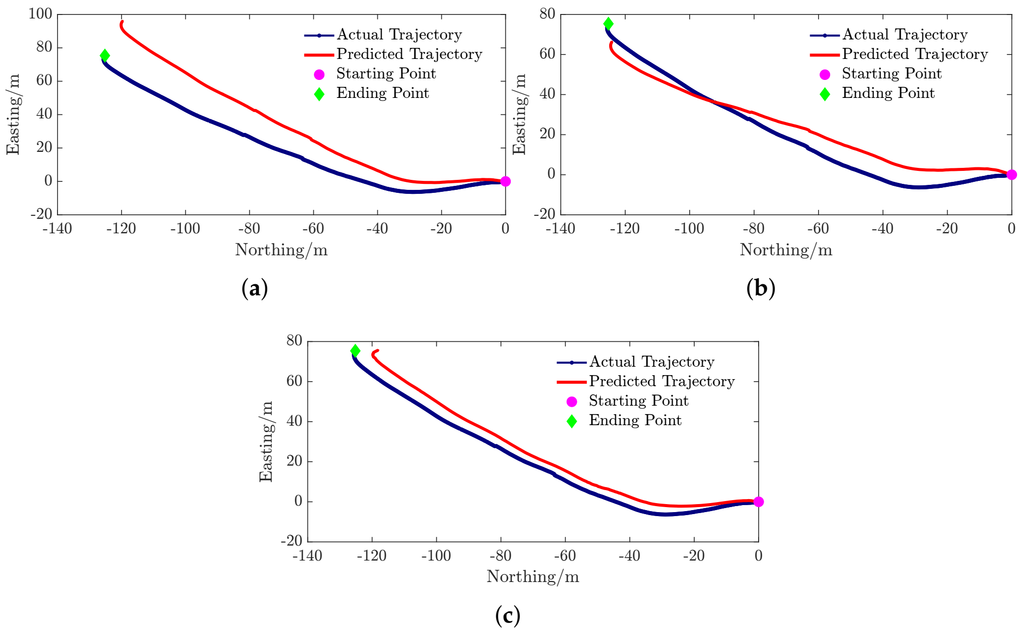
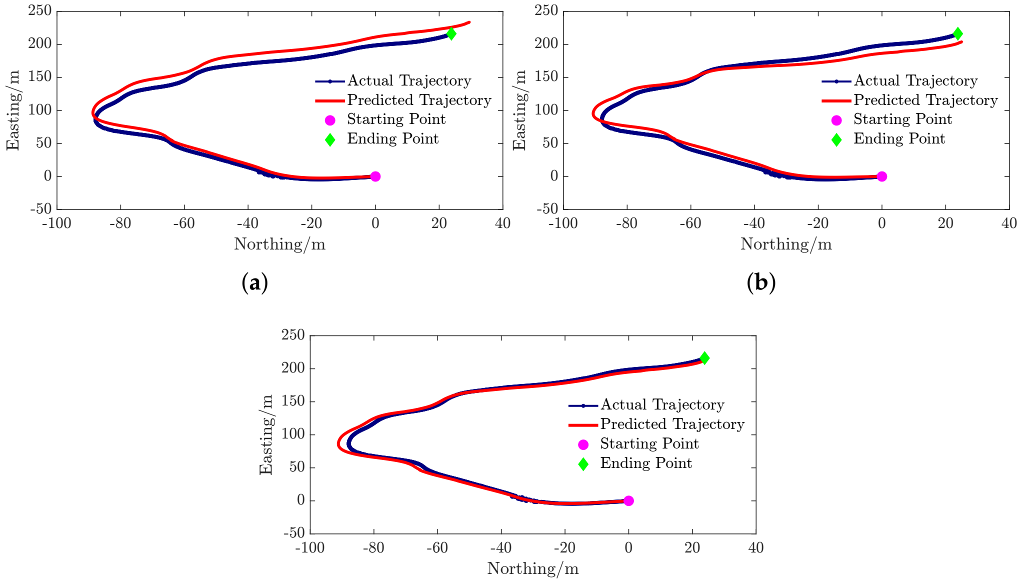


Figure 4. The recursive prediction results of each model in Scenario 1 when no low-frequency signal is introduced: (**a**) DNN; (**b**) LSTM; (**c**) BSsMSLNN.

In Scenario 1, the ground-truth trajectory consists of two approximately straight segments and one small-angle turn. As can be seen from Figure 4a, the DNN model gradually exhibits a pronounced deviation as the navigation distance increases, indicating strong susceptibility to long-term error accumulation. Figure 4b shows that the LSTM model can exploit temporal information to alleviate error growth to some extent; however, an obvious discrepancy between the predicted and true trajectories still remains. By contrast, Figure 4c shows that the BSsMSLNN-based model achieves the highest degree of agreement with the true trajectory, demonstrating stronger local dynamic modeling capability and better recursive stability.

https://doi.org/10.3390/jmse14111064

In Scenario 2, the ground-truth trajectory includes large-angle turns and multiple local bends, resulting in a more complex motion pattern. As can be seen from Figure 5a, the DNN model exhibits clear lag in turning regions, and its error continues to increase with the traveled distance. Figure 5b shows that the LSTM model is able to follow the overall trajectory trend reasonably well, but still shows fluctuations and local deviations in tortuous regions. By contrast, as shown in Figure 5c, the BSsMSLNN model remains highly consistent with the true trajectory throughout most of the navigation process, except for slight deviations in a few sharp-turn regions, indicating that this model still possesses strong adaptability under complex trajectory conditions.



(a) (b) (c)

Figure 5. The recursive prediction results of each model in Scenario 2 when no low-frequency signal is introduced: (**a**) DNN; (**b**) LSTM; (**c**) BSsMSLNN.

Table 3 further quantifies the recursive prediction performance of the three instantiated models under the two scenarios. It can be seen that, in both navigation scenarios, the BSsMSLNN-based model within the framework outperforms the other models. Specifically, compared with the DNN- and LSTM-based models within the framework, the BSsMSLNN-based model reduces the RMSE by 61.4% and 16.3%, respectively, and increases the R^2 value by 15.2% and 2.9%, respectively, in Scenario 1. In Scenario 2, it reduces the RMSE by 73% and 61.1%, respectively, and improves the R^2 value by 2.48% and 0.55%, respectively. These results indicate that, under the condition of relying solely on high-frequency propagation, BSsMSLNN achieves the best performance among the three models. However, all models still exhibit long-term cumulative errors to varying degrees, suggesting that high-frequency signals alone are insufficient to meet the requirements of high-precision localization over long-duration missions.

Table 3. Quantitative evaluation metric values of each model in Scenario 1 and Scenario 2 when no low-frequency signal is introduced.

		DNN	**LSTM**	**BSsMSLNN**
Scenario 1	RMSE (m)	8.2385	3.8006	3.1826
	R^2	0.8594	0.9621	0.9898
Scenario 2	RMSE (m)	7.2022	4.9894	1.9427
	R^2	0.9833	0.9920	0.9975

Based on the above observations, low-frequency absolute position observations from USBL and Sonar were then incorporated into the proposed fusion framework to correct the propagated results. The fusion weight parameter α is determined by the differential evolution algorithm. Meanwhile, to enhance the comparability of the results, EKF and SVIn2 were selected as conventional baselines and compared with the three deep-learning-based instantiated models under the same fusion framework.

Figure 6 presents the localization results of the five models in Scenario 1 after low-frequency correction is introduced. With the incorporation of low-frequency absolute observations, the predicted trajectories of all models generally follow the ground-truth trajectory more closely. However, clear differences remain in localization accuracy and trajectory smoothness. As shown in Figure 6a,b, EKF and SVIn2 still exhibit relatively large deviations and noticeable trajectory fluctuations. By contrast, the three deep-learning-based models shown in Figure 6c–e achieve a higher degree of consistency with the true trajectory and more stable overall performance. Furthermore, a closer comparison reveals that the BSsMSLNN-based model yields lower localization deviation and a higher degree of agreement between the predicted and true trajectories, indicating that this model achieves the best localization prediction performance in Scenario 1.

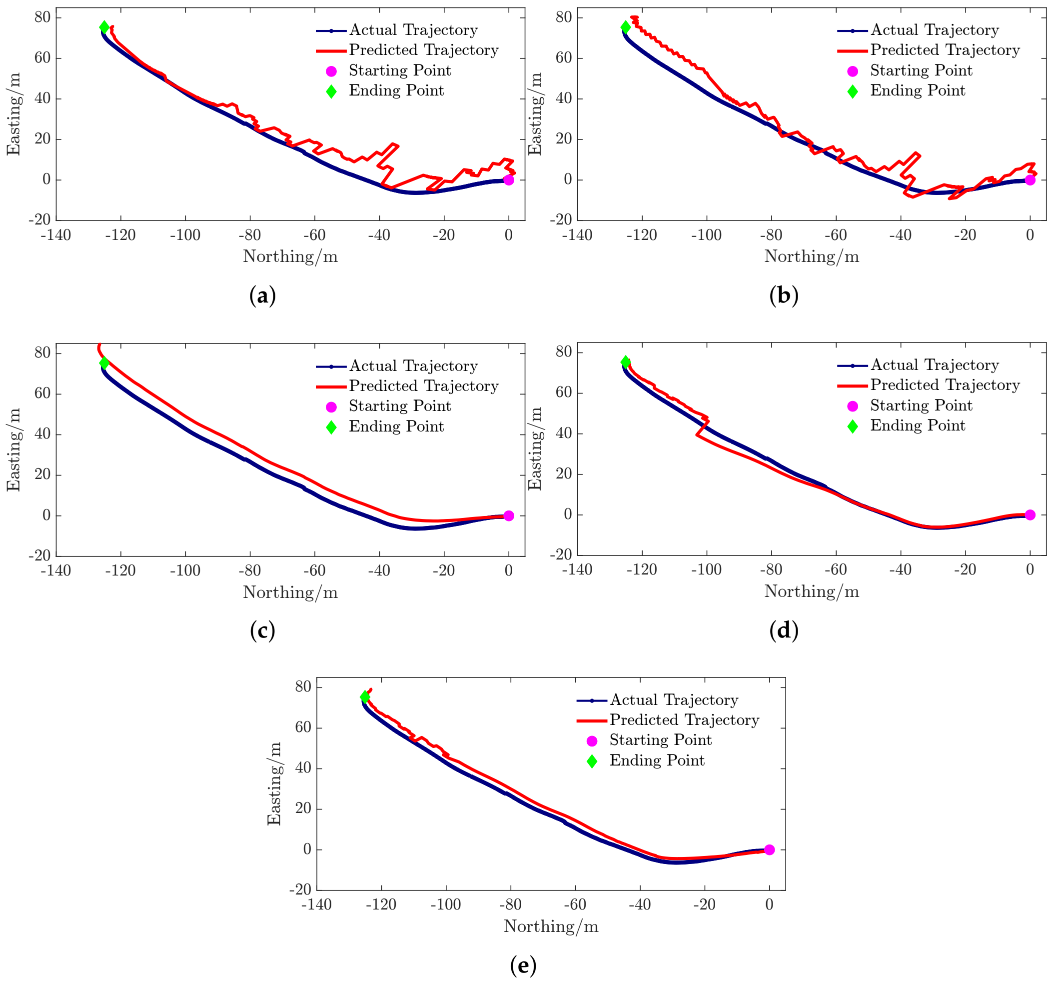


Figure 6. The prediction results of each model in Scenario 1: (**a**) EKF; (**b**) SVIn2; (**c**) DNN; (**d**) LSTM; (**e**) BSsMSLNN.

Figure 7 illustrates the localization prediction trends of five models in Scenario 2 after low-frequency correction is introduced. As can be seen from Figure 7a,b, under this more complex trajectory condition, the EKF exhibits obvious deviations at large-angle turns, whereas the SVIn2 shows multiple sharp fluctuations and poor overall smoothness. By contrast, the three deep-learning-based models within the proposed framework demonstrate clearly better performance, as shown in Figure 7c–e. As shown in Figure 7c, the DNN-based model produces a relatively smooth trajectory overall but still deviates in several regions. Figure 7d indicates the LSTM-based model captures the global trend effectively and remains stable in both straight and turning segments, but slight deviations are still visible in locally tortuous regions. A further comparison of Figure 7c–e shows that the BSsMSLNN-based model achieves the smallest localization deviation in Scenario 2 and exhibits the highest degree of agreement between the predicted and true trajectories. Whether in straight segments, at large-angle turns, or in locally tortuous regions, it demonstrates a very high level of consistency. This further verifies the superiority of the BSsMSLNN model in complex navigation scenarios, indicating that it can fully exploit the collaborative effects of multiple local models to more accurately capture the dynamic variations in the AUV trajectory.

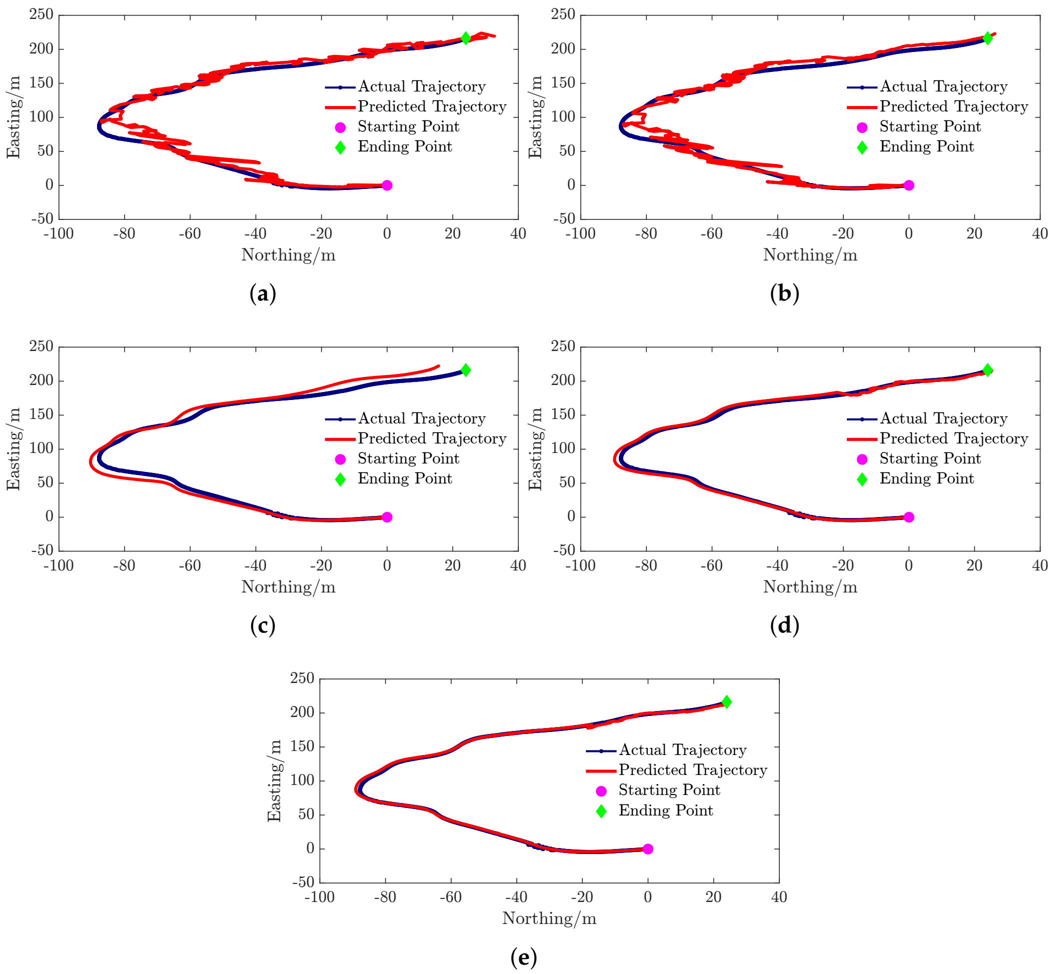


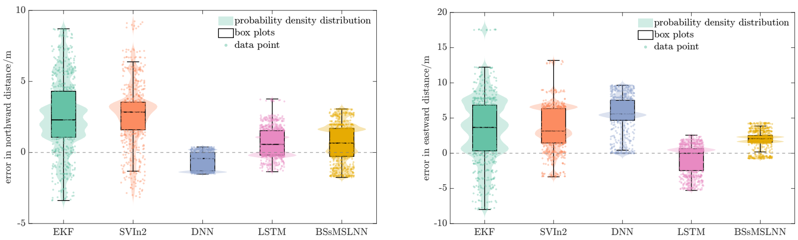
Figure 7. The prediction results of each model in Scenario 2: (**a**) EKF; (**b**) SVIn2; (**c**) DNN; (**d**) LSTM; (**e**) BSsMSLNN.

Table 4 summarizes the quantitative evaluation metrics of localization prediction for the five different models under two different scenarios. It can be seen that, in both navigation scenarios, the BSsMSLNN-based model within the proposed framework outperforms the other models. Specifically, compared with the EKF, SVIn2, the DNN-based model, and the LSTM-based model, the BSsMSLNN-based model reduces the RMSE by 63.8%, 57.7%, 50.9%, and 9.8%, respectively, and improves the R^2 value by 2.78%, 2.62%, 1.72%, and 0.06%, respectively, in Scenario 1. In Scenario 2, it reduces the RMSE by 84.2%, 81.3%, 77.1%, and 20.4%, respectively, and improves the R^2 value by 2.45%, 1.92%, 1.39%, and 0.09%, respectively. These results indicate that low-frequency absolute observations can effectively suppress the accumulated errors caused by high-frequency propagation and that, within the unified fusion framework, the BSsMSLNN offers stronger local modeling capability and more stable prediction performance.

Table 4. Quantitative evaluation metric values of each model in Scenario 1 and Scenario 2 when low-frequency signal is introduced.

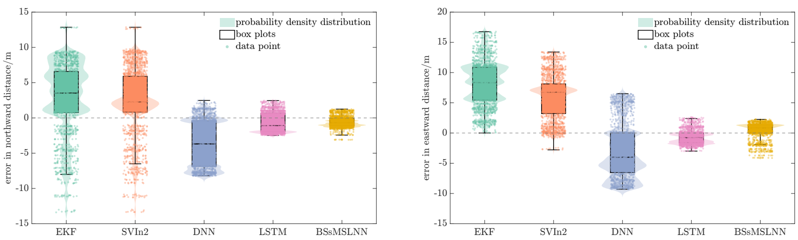
		EKF	SVIn2	DNN	LSTM	BSsMSLNN
Scenario 1	RMSE (m)	4.7299	4.0544	3.4929	1.9003	1.7142
	R^2	0.9689	0.9704	0.9790	0.9952	0.9958
Scenario 2	RMSE (m)	7.3687	6.2280	5.0880	1.4622	1.1641
	R^2	0.9755	0.9806	0.9857	0.9985	0.9994

To further compare the stability of the five methods, Figures 8 and 9 present the distributions of the prediction errors in the local metric coordinate system under Scenarios 1 and 2, respectively. In both scenarios, the three models developed within the proposed framework exhibit narrower interquartile ranges, shorter whiskers, and medians closer to zero than EKF and SVIn2, indicating generally lower prediction errors and better stability. Among the three deep-learning-based models, the error distribution of DNN is relatively dispersed and exhibits noticeable systematic bias in some cases. The LSTM-based model produces a more concentrated error distribution than DNN, but its prediction uncertainty remains non-negligible. By contrast, the BSsMSLNN-based model exhibits the most concentrated error distribution, with most errors clustered near zero, indicating superior error control capability and prediction stability. Taken together with the results in Figures 6–9 and Table 4, these findings confirm that the proposed fusion framework can effectively improve AUV localization accuracy under different trajectory conditions and that BSsMSLNN is the best-performing instantiated model within this framework.



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

Figure 8. The distribution of prediction errors of the five models in Scenario 1: (**a**) northward error; (**b**) eastward error.



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

Figure 9. The distribution of prediction errors of the five models in Scenario 2: (**a**) northward error; (**b**) eastward error.

To evaluate the real-time feasibility of the proposed framework, the average inference times of all instantiated learning models are measured and compared with the sampling interval of the high-frequency INS/DVL data. Since the INS and DVL measurements are updated at 20 Hz, the corresponding sampling interval is 50 ms. As shown in Table 5, the average inference times of the DNN-, LSTM-, and BSsMSLNN-based models are all much lower than the 50 ms sampling interval of the high-frequency INS/DVL measurements. Although the BSsMSLNN-based model requires a longer inference time than the DNN- and LSTM-based models due to its posterior-weighted combination of multiple local shallow networks, its inference time is still sufficient for real-time localization under the current experimental configuration. These results indicate that the proposed fusion localization framework can satisfy the real-time requirement of the present AUV experimental platform.

Table 5. Average inference time of different models.

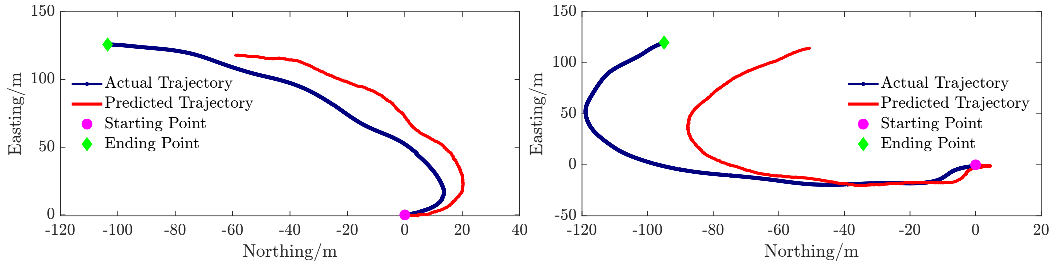
Model	DNN	LSTM	BSsMSLNN
Average inference time (ms)	1.12	1.89	2.32
Sampling interval (ms)	50	50	50

3.3. Analysis of the Effect of Underwater Depth on Model Performance

The above experiments were conducted mainly under the water-depth condition of 0.3 m and verified the localization performance of the proposed fusion framework and its instantiated models under typical trajectory scenarios. However, in practical applications, AUVs are often required to operate under different depth conditions. As the water depth changes, the surrounding environment, hydrodynamic characteristics, and sensor measurement states may also vary, thereby causing a mismatch between the distribution of the training data and the actual testing conditions. To further investigate the effect of operating depth on model performance, the depth-related analysis is designed in two parts. The first part evaluates direct cross-depth generalization, where the model trained at 0.3 m is directly tested on the 0.6 m and 0.9 m data without retraining. The second part investigates whether explicitly introducing depth information and using multi-depth training trajectories can alleviate the above distribution mismatch. Because BSsMSLNN showed the best overall performance in the previous experiments, it was selected for further analysis.

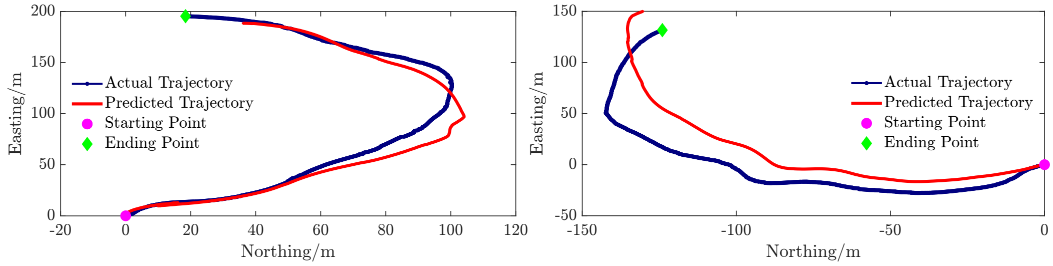
In this part of the study, the BSsMSLNN model trained using data collected at a depth of 0.3 m was directly applied to data collected at depths of 0.6 m and 0.9 m to evaluate its generalization capability under varying depth conditions. Figure 10 presents the prediction results for two voyages at a depth of 0.6 m. As shown in the figure, the predicted trajectories deviate from the true trajectories in both voyages to different extents, especially in turning segments, and a noticeable cumulative offset remains after the turn. These results suggest that when the training depth is inconsistent with the testing depth, the model's ability to characterize changes in navigation state, particularly complex motion patterns such as turning, deteriorates.

Figure 11 further presents the prediction results for two voyages at a depth of 0.9 m. Compared with the results obtained at 0.6 m, the trajectory deviations under the 0.9 m condition are generally more pronounced. In Voyage 1, the predicted trajectory is relatively close to the true trajectory at the initial stage, but the deviation gradually accumulates and becomes especially obvious in the turning region. In Voyage 2, although the predicted trajectory is close to the true trajectory near the starting point, the subsequent segments show a noticeable overall offset and reduced consistency with the ground-truth trajectory shape. These observations indicate that as the water depth increases further, the localization performance of a model trained at a fixed depth deteriorates further when it is directly applied to cross-depth prediction.



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

Figure 10. Prediction results of BSsMSLNN at 0.6 m depth: (**a**) Voyage 1; (**b**) Voyage 2.



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

Figure 11. Prediction results of BSsMSLNN at 0.9 m depth: (**a**) Voyage 1; (**b**) Voyage 2.

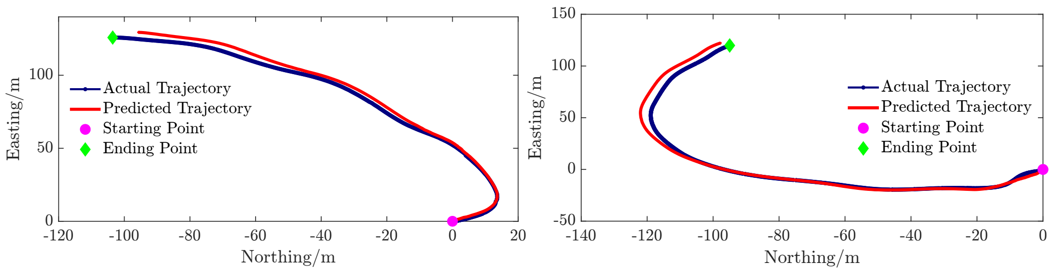
Table 6 quantifies the localization performance obtained when the BSsMSLNN model parameters trained at a depth of 0.3 m are directly applied to data collected at depths of 0.6 m and 0.9 m. As can be seen from Table 6, the RMSE values remain relatively large under direct cross-depth prediction. At a depth of 0.6 m, the RMSE values of Voyage 1 and Voyage 2 are 14.944 and 10.197, respectively, while the corresponding R^2 values are 0.7460 and 0.8945. At a depth of 0.9 m, the RMSE values of Voyage 1 and Voyage 2 are 6.082 and 8.1482, respectively, with corresponding R^2 values of 0.9590 and 0.9373. These results indicate that when the operating depth of the testing data differs from that of the training data, the prediction accuracy of the model decreases, demonstrating the influence of depth-induced distribution mismatch on cross-depth localization performance.

Table 6. Quantitative evaluation results of direct cross-depth prediction using the BSsMSLNN model trained at 0.3 m.

		RMSE (m)	$\mathbf{R^2}$
depth 0.6 m	Voyage 1	14.9441	0.7460
	Voyage 2	10.1973	0.8945
depth 0.9 m	Voyage 1	6.0828	0.9590
	Voyage 2	8.1482	0.9373

To address the above problem, this study further incorporates depth information as an auxiliary input variable and combines it with other sensor features to form the model input, thereby explicitly characterizing the variations in AUV motion states and sensor response characteristics under different depth conditions. In this part, parts of the trajectories collected at 0.3 m, 0.6 m, and 0.9 m are jointly used as the training set to train a depth-aware BSsMSLNN model, while the remaining independent trajectories at 0.6 m and 0.9 m are used for testing. Therefore, the retraining in this part is not intended to artificially improve the results by changing the experimental condition, but to examine whether a model trained with multi-depth data and explicit depth information can better adapt to varying operating depths.

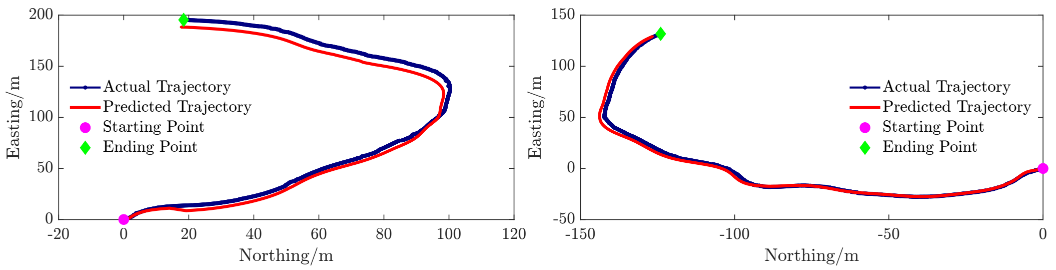
Figure 12 presents the prediction results of the model for two voyages at a depth of 0.6 m after the incorporation of depth information. Compared with Figure 10, the retrained model shows a substantially improved agreement between the predicted and true trajectories. In both voyages, the model is able to track the overall trajectory trend more effectively, and the previously observed cumulative drift and lag are no longer evident in either straight or turning segments. In particular, the tracking consistency at the large-angle turn in Voyage 1 is markedly improved, while Voyage 2 also exhibits high trajectory-tracking accuracy. These results indicate that, after depth is introduced as an input variable, the model is able to learn the trajectory evolution patterns under different depth conditions more effectively and thus alleviate the mismatch caused by depth variation.



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

Figure 12. The prediction results considering depth information at a depth of 0.6 m: (**a**) Voyage 1; (**b**) Voyage 2.

Figure 13 further presents the prediction results of the model for two voyages at a depth of 0.9 m after the incorporation of depth information. Compared with Figure 11, the consistency between the predicted trajectories and the true trajectories is further improved after depth information is introduced. In Voyage 1, the model fits the true trajectory well throughout the entire voyage, with especially notable improvement in the turning region. In Voyage 2, the overall trajectory offset is effectively alleviated, and the predicted trajectory remains highly consistent with the shape of the ground-truth trajectory. These results indicate that even under deeper-water conditions, incorporating depth information still contributes substantially to improving localization accuracy and trajectory-tracking stability.



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

Figure 13. The new prediction results considering depth information at a depth of 0.9 m: (**a**) Voyage 1; (**b**) Voyage 2.

Table 7 presents the quantitative evaluation results after incorporating depth information. A combined examination of Tables 6 and 7 shows that, at a depth of 0.6 m, the RMSE values for Voyage 1 and Voyage 2 decrease by 84.3% and 76%, respectively, relative to the corresponding voyages in Table 6, while the R^2 values increase by 33.4% and 11.3%, respectively. At a depth of 0.9 m, the RMSE values for Voyage 1 and Voyage 2 decrease by 47.5% and 79%, respectively, relative to the corresponding voyages in Table 6, while the R^2 values increase by 4.1% and 6.5%, respectively, compared with those of the corresponding voyages in Table 6. These results indicate that, after the introduction of depth information, the prediction errors of the model under different depth conditions are significantly reduced, and its fitting capability is also markedly improved. This demonstrates that the depth variable can provide the model with important supplementary information related to environmental variations, thereby enhancing its adaptability to different operating conditions.

Table 7. Quantitative evaluation metrics after fusing depth information.

		RMSE (m)	$\mathbf{R^2}$
depth 0.6 m	Voyage 1	2.3492	0.9949
	Voyage 2	2.4495	0.9958
depth 0.9 m	Voyage 1	3.1930	0.9979
	Voyage 2	1.7090	0.9986

Overall, the depth-related experiments demonstrate two important findings. First, a model trained at a single operating depth has limited generalization capability when directly applied to other depths, indicating that depth-induced distribution shift is an important factor affecting AUV localization accuracy. Second, incorporating depth information and multi-depth training data can improve the adaptability of the model to varying operating depths. These results indicate that, in localization modeling for practical underwater missions, it is necessary to incorporate depth information into the multi-source sensor fusion framework so as to enhance the robustness and generalization capability of the model under complex operating conditions.

4. Conclusions

In this paper, a deep-learning-based multi-source sensor fusion framework is proposed for AUV localization. The framework combines high-frequency recursive propagation based on INS and DVL data with periodic low-frequency correction using absolute position observations from USBL and sonar, thereby establishing a coordinated localization mechanism of high-frequency propagation and low-frequency correction. Within this framework, three instantiated models, namely DNN, LSTM, and BSsMSLNN, are developed and compared with EKF and SVIn2 through experiments conducted under different trajectory scenarios and water-depth conditions.

The experimental results lead to the following conclusions. First, the proposed fusion framework effectively coordinates high-frequency propagation and low-frequency correction, thereby suppressing the accumulation of long-term propagation errors while maintaining continuous position output and improving AUV localization accuracy. Compared with localization based solely on high-frequency propagation, the introduction of low-frequency observations from USBL and sonar improves the performance of all evaluated models to different degrees.

Second, under the unified fusion framework, substantial differences are observed among the instantiated models. Among them, the BSsMSLNN-based model achieves better trajectory agreement, lower RMSE, and higher R^2 values than the DNN- and LSTM-based models, as well as the conventional benchmark methods EKF and SVIn2, in both representative trajectory scenarios. These results indicate that BSsMSLNN is more capable of handling the complex nonlinear relationships, heterogeneous multi-source information, and local dynamic variations involved in AUV localization tasks.

Third, water-depth variation has a significant influence on localization performance. When a model trained at a fixed depth is directly transferred to data collected at other depths, its prediction accuracy degrades substantially, indicating the presence of operating-condition mismatch in cross-depth scenarios. After depth information is introduced into the model as an auxiliary input variable, both localization accuracy and prediction stability under different depth conditions are significantly improved. This demonstrates that depth information can effectively enhance the model's adaptability to environmental variation and improve its generalization capability across different operating conditions.

Overall, the proposed multi-source sensor fusion localization framework combines the continuity of recursive propagation with the effectiveness of periodic correction and therefore provides a promising solution with practical engineering potential for autonomous AUV navigation in complex underwater environments.

Furthermore, the current experiments were conducted in a controlled shallow-water real-water environment, which is suitable for reliably acquiring multi-source sensor data and for clearly verifying the effectiveness of the proposed fusion localization framework. In addition, the RTK-GPS-assisted reference setup is suitable for horizontal trajectory evaluation in the present experiments, but it cannot fully represent prolonged fully submerged AUV navigation conditions. Future work will further enrich the validation of the proposed method in deep-water, long-range trajectories, and more complex ocean environments, and will investigate more realistic underwater reference systems for long-duration AUV missions.

Author Contributions: Conceptualization, X.P.; Methodology, X.P. and Q.T.; Software, G.F. and Q.T.; Validation, G.F., H.Z. and Q.T.; Formal analysis, G.F.; Investigation, X.P. and H.Z.; Resources, H.Z.; Data curation, X.P. and H.Z.; Writing—original draft, X.P.; Writing—review & editing, Q.T.; Visualization, Q.T.; Supervision, Q.T.; Project administration, Q.T.; Funding acquisition, X.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science of China under Grant U24A20144.

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

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

Abbreviations

The following abbreviations are used in this manuscript:

AUV	Autonomous Underwater Vehicle
INS	Inertial Navigation System
DVL	Doppler Velocity Log
USBL	Ultra-Short Baseline
GPS	Global Positioning System
RTK-GPS	Real-Time Kinematic Global Positioning System
GNSS	Global Navigation Satellite System
DNN	Deep Neural Network
LSTM	Long Short-Term Memory
BSsMSLNN	Bayesian Semi-supervised Mixed Shallow-layer Neural Network
RNN	Recurrent Neural Network
EKF	Extended Kalman Filter
UKF	Unscented Kalman Filter
CKF	Cubature Kalman Filter
SLAM	Simultaneous Localization and Mapping
SVIn2	Sonar, Visual, Inertial, and Depth sensor-based SLAM system
RMSE	Root Mean Square Error
R^2	Coefficient of Determination

References

1. Wu, L.; Jing, Z.; Chen, X.; Li, C.; Zhang, G.; Wang, S.; Dong, B.; Zhuang, G. Marine science in China: Current status and future outlooks. *Earth Sci. Front.* **2022**, *29*, 1.
2. Sánchez, P.J.B.; Papaelias, M.; Márquez, F.P.G. Autonomous underwater vehicles: Instrumentation and measurements. *IEEE Instrum. Meas. Mag.* **2020**, *23*, 105–114. [CrossRef]
3. Felemban, E.; Shaikh, F.K.; Qureshi, U.M.; Sheikh, A.A.; Qaisar, S.B. Underwater sensor network applications: A comprehensive survey. *Int. J. Distrib. Sens. Netw.* **2015**, *11*, 896832. [CrossRef]
4. Paull, L.; Saeedi, S.; Seto, M.; Li, H. AUV navigation and localization: A review. *IEEE J. Ocean. Eng.* **2014**, *39*, 131–149. [CrossRef]
5. Zhang, B.; Ji, D.; Liu, S.; Zhu, X.; Xu, W. Autonomous Underwater Vehicle navigation: A review. *Ocean Eng.* **2023**, *273*, 113861. [CrossRef]
6. Angrisano, A. GNSS/INS Integration Methods. Ph.D. Thesis, Università degli Studi di Napoli Parthenope, Naples, Italy, 2011.
7. Wang, D.; Xu, X.; Yao, Y.; Zhang, T. Virtual DVL Reconstruction Method for an Integrated Navigation System Based on DS-LSSVM Algorithm. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–13. [CrossRef]
8. Liu, J.; Yu, T.; Wu, C.; Zhou, C.; Lu, D.; Zeng, Q. A Low-Cost and High-Precision Underwater Integrated Navigation System. *J. Mar. Sci. Eng.* **2024**, *12*, 200. [CrossRef]
9. Wang, C.; Cheng, C.; Cao, C.; Guo, X.; Pan, G.; Zhang, F. An invariant filtering method based on frame transformed for underwater ins/dvl/ps navigation. *J. Mar. Sci. Eng.* **2024**, *12*, 1178. [CrossRef]
10. Peñas, A. Positioning and Navigation Systems for Robotic Underwater Vehicles. Ph.D. Thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa, Lisbon, Portugal, 2009.
11. Lu, Y.; Wang, J.; Zhou, H.; Wang, D.; Xing, Y.; He, Z. Station layout optimization for underwater acoustic positioning system based on combined cone configuration. *Ocean Eng.* **2024**, *294*, 116753. [CrossRef]
12. Zhuang, Y.; Sun, X.; Li, Y.; Huai, J.; Hua, L.; Yang, X.; Cao, X.; Zhang, P.; Cao, Y.; Qi, L.; et al. Multi-sensor integrated navigation/positioning systems using data fusion: From analytics-based to learning-based approaches. *Inf. Fusion* **2023**, *95*, 62–90. [CrossRef]
13. Sheng, G.; Liu, X.; Zhang, Y.; Shao, Q.; Xu, H.; Cheng, X.; Yuan, X. The EKF-based SINS/DVL integrated navigation for AUV on lie group under hovering condition. *Ocean Eng.* **2025**, *325*, 120742. [CrossRef]
14. Julier, S.J.; Uhlmann, J.K. Unscented filtering and nonlinear estimation. *Proc. IEEE* **2004**, *92*, 401–422. [CrossRef]
15. Arasaratnam, I.; Haykin, S. Cubature Kalman filters. *IEEE Trans. Autom. Control* **2009**, *54*, 1254–1269. [CrossRef]
16. Li, Y.; Hou, L.; Yang, Y.; Tong, J. Huber's M-estimation-based cubature Kalman filter for an INS/DVL integrated system. *Math. Probl. Eng.* **2020**, *1*, 1060672. [CrossRef]
17. Huang, H.; Tang, J.; Zhang, B. Positioning parameter determination based on statistical regression applied to autonomous underwater vehicle. *Appl. Sci.* **2021**, *11*, 7777. [CrossRef]
18. Shi, W.; Xu, J.; Li, D.; He, H. Attitude Estimation of SINS on Underwater Dynamic Base with Variational Bayesian Robust Adaptive Kalman Filter. *IEEE Sens. J.* **2022**, *22*, 10954–10964. [CrossRef]
19. Dellaert, F.; Kaess, M. Square Root SAM: Simultaneous Localization and Mapping via Square Root Information Smoothing. *Int. J. Robot. Res.* **2006**, *25*, 1181–1203. [CrossRef]
20. Kaess, M.; Ranganathan, A.; Dellaert, F. iSAM: Incremental smoothing and mapping. *IEEE Trans. Robot.* **2008**, *24*, 1365–1378. [CrossRef]
21. Yang, H.; Antonante, P.; Tzoumas, V.; Carlone, L. Graduated non-convexity for robust spatial perception: From non-minimal solvers to global outlier rejection. *IEEE Robot. Autom. Lett.* **2020**, *5*, 1127–1134. [CrossRef]
22. Rahman, S.; Li, A.Q.; Rekleitis, I. SVIn2: An Underwater SLAM System using Sonar, Visual, Inertial, and Depth Sensor. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 1861–1868.
23. Zhang, D.; Chang, S.; Zou, G.; Wan, C.; Li, H. A Robust Graph-Based Bathymetric Simultaneous Localization and Mapping Approach for AUVs. *IEEE J. Ocean. Eng.* **2024**, *49*, 1350–1370. [CrossRef]
24. Chang, S.; Zhang, D.; Zhang, L.; Zou, G.; Wan, C.; Ma, W.; Zhou, Q. A joint graph-based approach for simultaneous underwater localization and mapping for AUV navigation fusing bathymetric and magnetic-beacon-observation data. *J. Mar. Sci. Eng.* **2024**, *12*, 954. [CrossRef]
25. Chang, P.; Li, Z.Y. Over-complete deep recurrent neutral network based on wastewater treatment process soft sensor application. *Appl. Soft Comput.* **2021**, *105*, 107227. [CrossRef]
26. Pradeep, T.; Bardhan, A.; Burman, A.; Samui, P. Rock strain prediction using deep neural network and hybrid models of anfis and meta-heuristic optimization algorithms. *Infrastructures* **2021**, *6*, 129. [CrossRef]
27. Shao, W.; Li, X.; Xing, Y.; Chen, J. A mixture of shallow neural networks for virtual sensing: Could perform better than deep neural networks. *Expert Syst. Appl.* **2024**, *256*, 124870. [CrossRef]

28. Topini, E.; Fanelli, F.; Topini, A.; Pebody, M.; Ridolfi, A.; Phillips, A.B.; Allotta, B. An experimental comparison of Deep Learning strategies for AUV navigation in DVL-denied environments. *Ocean Eng.* **2023**, *274*, 114034. [CrossRef]
29. Mu, X.; He, B.; Zhang, X.; Song, Y.; Shen, Y.; Feng, C. End-to-end navigation for autonomous underwater vehicle with hybrid recurrent neural networks. *Ocean Eng.* **2019**, *194*, 106602. [CrossRef]
30. Zhang, X.; He, B.; Li, G.; Mu, X.; Zhou, Y.; Mang, T. NavNet: AUV navigation through deep sequential learning. *IEEE Access* **2020**, *8*, 59845–59861. [CrossRef]
31. Zhang, X.; He, B.; Gao, S.; Zhou, L.; Huang, R. Sequential learning navigation method and general correction model for Autonomous Underwater Vehicle. *Ocean Eng.* **2023**, *278*, 114347. [CrossRef]
32. Zhang, X.; Sheng, L.; He, B.; Lu, Y. A data-driven based hybrid multi-branch framework for AUV navigation. *Ocean Eng.* **2025**, *324*, 120675. [CrossRef]
33. Liang, Z.; Wang, K.; Zhang, J.; Zhang, F. An underwater multisensor fusion simultaneous localization and mapping system based on image enhancement. *J. Mar. Sci. Eng.* **2024**, *12*, 1170. [CrossRef]
34. Merveille, F.F.R.; Jia, B.; Xu, Z.; Fred, B. Advancements in Sensor Fusion for Underwater SLAM: A Review on Enhanced Navigation and Environmental Perception. *Sensors* **2024**, *24*, 7490. [CrossRef] [PubMed]
35. Heshmat, M.; Saad Saoud, L.; Abujabal, M.; Sultan, A.; Elmezain, M.; Seneviratne, L.D.; Hussain, I. Underwater SLAM Meets Deep Learning: Challenges, Multi-Sensor Integration, and Future Directions. *Sensors* **2025**, *25*, 3258. [CrossRef] [PubMed]