

Article

A Non-Equal Time Interval Incremental Motion Prediction Method for Maritime Autonomous Surface Ships

Zhijie Zhou ^{1,2} , Haixiang Xu ^{1,2,*}, Hui Feng ^{1,2} and Wenjuan Li ³

¹ Key Laboratory of High-Performance Ship Technology, Wuhan University of Technology, Ministry of Education, Wuhan 430063, China

² School of Naval Architecture, Ocean and Energy Power Engineering, Wuhan University of Technology, Wuhan 430063, China

³ Marine Equipment Technology Institute, Jiangsu University of Science and Technology, Zhenjiang 212003, China

* Correspondence: qukaiyang@163.com; Tel.: +86-13657283428

Abstract: Recent technological advancements facilitate the autonomous navigation of maritime surface ships. The accurate data given by a range of various sensors serve as the primary assurance of a voyage's safety. Nevertheless, as sensors have various sample rates, they cannot obtain information at the same time. Fusion decreases the accuracy and reliability of perceptual data if different sensor sample rates are not taken into account. Hence, it is helpful to increase the quality of the fusion information to precisely anticipate the motion status of ships at the sampling time of each sensor. This paper proposes a non-equal time interval incremental prediction method. In this method, the high dimensionality of the estimated state and nonlinearity of the kinematic equation are taken into consideration. First, the cubature Kalman filter is employed to estimate a ship's motion at equal intervals based on the ship's kinematic equation. Next, a ship motion state predictor based on a long short-term memory network structure is created, using the increment and time interval of the historical estimation sequence as the network input and the increment of the motion state at the projected time as the network output. The suggested technique can lessen the effect of the speed difference between the test set and the training set on the prediction accuracy compared with the traditional long short-term memory prediction method. Finally, comparison experiments are carried out to validate the precision and effectiveness of the proposed approach. The experimental results show that the root-mean-square error coefficient of the prediction error is decreased on average by roughly 78% for various modes and speeds when compared with the conventional non-incremental long short-term memory prediction approach. Additionally, the proposed prediction technology and the traditional approach have virtually the same algorithm times, which may fulfill the real engineering requirements.

Keywords: maritime autonomous surface ship; cubature Kalman filter; incremental motion prediction; long short-term memory



Citation: Zhou, Z.; Xu, H.; Feng, H.; Li, W. A Non-Equal Time Interval Incremental Motion Prediction Method for Maritime Autonomous Surface Ships. *Sensors* **2023**, *23*, 2852. <https://doi.org/10.3390/s23052852>

Academic Editor: Raquel Caballero-Aguila

Received: 2 February 2023

Revised: 27 February 2023

Accepted: 27 February 2023

Published: 6 March 2023



Copyright: © 2023 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) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In the global economy, marine transportation is becoming increasingly significant. More than two thirds of all freights in international trade are transported by sea. Therefore, the ability of marine transportation to link the globe and support technical advancement in maritime equipment is extremely important. In recent years, progressive intelligence based on digitalization and aiming for autonomy has emerged as a new trend and hot spot in the development of the shipbuilding industry, driven by the development of cutting-edge theories and technologies such as the Internet of Things, big data, and artificial intelligence (AI). The world's major shipbuilding and shipping nations have expanded their investments in the creation and use of autonomous ships. Autonomous ships, also known as maritime autonomous surface ships (MASS), have been developed and are increasingly

being used [1]. The desire to avoid human error—which contributes significantly to the bulk of maritime catastrophes—led to the creation of MASSs. Additionally, crewed ships have also been linked to high operating costs. Therefore, the requirement to avoid the financial expenditures and human mistakes linked to crewed ships serves as the main driving force behind developments in autonomous shipping [2].

The autonomous degree of MASS has been divided into L1–L4 in the 100th Maritime Safety Committee (MSC) by the International Maritime Organization (IMO) [3]. The mode of the perception module in a ship's autonomous navigation system changes from a shared view of mariners and machines into a total machine perception as the degree of autonomy develops, which results in an increase in the number and variety of sensors carried by MASS. Therefore, research into multi-sensor fusion technology in autonomous navigation systems is crucial to raising the level of ship autonomy.

The perception data of a ship's autonomous navigation system are mostly composed of information about the ship's motion state (such as its position, heading, speed, attitude, etc.), as well as information about the surrounding environment (such as the status of other ships, pontoons, and other obstacles that may threaten the navigation safety of the ships). Nevertheless, the accuracy of the former's perception serves as the primary guarantee of the latter's perception correctness. For instance, if the perception of a ship's own motion status is erroneous, even though the relative azimuth information of an obstruction perceived by the shipborne sensor has adequate precision, the absolute coordinates of the obstacle will be guessed incorrectly.

The perception of a ship's own motion status is mostly based on the fusion information provided by shipboard positioning systems and inertial navigation systems (INSs), such as the global positioning system (GPS), global navigation satellite system (GLONASS), Galileo positioning system, BeiDou, a compass, and a gyroscope. In the past decade, researchers have carried out various research works on developing efficient multi-sensor fusion algorithms to estimate the motion status of ships. For instance, a fusion technique based on the Kalman filter (KF) and particle filter (PF) was developed [4]. This method accurately estimated the status of a ship by fusing its position and attitude data. To further improve the accuracy of ship motion attitude estimates, Ref. [5] proposed a novel transfer alignment method for a gimballed inertial navigation system (GINS) and strap-down inertial navigation system (SINS) based on an iterative computation approach. By using this technique, the alignment complexity is reduced. Additionally, a dynamic positioning ship state estimation technique combining the unscented Kalman filter (UKF) and PF was also presented in [6]. In this approach, the UKF optimizes each particle state update while the PF serves as the general framework. Then, using the particles' importance distribution, the low-frequency condition of a ship's motion was determined. Ref. [7] focused on the least-squares linear fusion filter design for discrete-time stochastic signals from a multi-sensor. A covariance-based approach was used to derive easily implementable recursive filtering algorithms under centralized, distributed, and sequential fusion architectures. The year after, using the linear minimum variance optimality criterion, the local least-squares linear filter obtained with each sensor was improved as a distributed fusion filter with a matrix-weighted linear combination while taking into account the autocorrelation and cross-correlation of multi-sensor measurements [8]. Furthermore, considering the random packet loss in the transmission process of sensor measurement signals, a recursive filtering algorithm was designed by using a method based on covariance and two compensation strategies based on measurement prediction [9]. In this algorithm, an alternative method based on direct estimation measurement noise and innovative technology was used to compensate for the packet loss system. To make the navigation system more reliable, Ref. [10] proposed an improved RISS-GPS ship navigation approach. Modern magnetometer and azimuth calibration technology served as the foundation for this technique.

The research on the fusion technology of a ship's own motion state has gradually matured and is widely used. The following are its main steps: The mathematical model of

a ship's motion is first discretized, and then the measurements from each sensor are used as the input to perform the fusion estimation of the step with an equal time interval based on the framework of the Kalman filter and its enhanced method. Nevertheless, sampling frequency varies widely since there are so many different types of sensors on board. The sampling rate of ambient sensors, such as cameras and lidar, is typically between 10 and 30 Hz, whereas the sampling rate of GPS is typically between 1 and 5 Hz. As a result, the latest estimate of a ship's state of motion at the time ambient sensors take samples is often tens or even hundreds of milliseconds prior to the sampling time. The higher the speed of a ship, the more significant the difference in motion state will be. As a consequence, matching the sample from an external environment detecting sensor with the outcomes of a ship motion state estimate in the time dimension is the key technology in fusion technology.

Currently, ship motion status prediction makes considerable use of the long short-term memory (LSTM) model. It may learn the features of ship motion through historical observations, and then predict the motion state at a specific future time. However, this method is often used to predict a ship's trajectory with strong regularity and is rarely used to predict a ship's irregular motion directly, such as roll and pitch motion. The method, however, does not account for the modification of the motion law brought on by varying speeds. This paper proposes a non-equal time interval incremental prediction (NETIIP) method. First, a collection of cruise data is trained offline using the LSTM architecture. The time deviation and state increment are used as inputs, and the increment of actual state at the moment to be estimated is taken as the output of the network. Then, using the cubature Kalman filter (CKF) estimator, the position, attitude, speed, and other characteristics of the ships are projected at regular intervals. The estimation result of the ship motion state at this moment is finally derived based on the network estimation from offline training after receiving the time stamp of the environmental measurement sensor. The advantages of this approach are as follows:

- (a) NETIIP uses the CKF estimates as the input for the LSTM prediction network rather than the sensor's original data. On the one hand, it can effectively avoid the problem of reduced prediction accuracy caused by sensor measurement error signals as the object of network learning. On the other hand, the amplification of sensor measurement noise caused by first-order state difference can be suppressed.
- (b) NETIIP adopts a semi-supervised learning mode, which not only learns the changes in a ship's position and attitude but also incorporates the changes in the ship's speed into the learning features of the network, which can minimize the impact of the poor learning performance caused by ship speed differences. It merely needs to learn annotated datasets of changes in ship movement at any speed. It is feasible to foresee the ship's motion status at various speeds.
- (c) NETIIP employs the technique of learning the properties of state increments rather than directly learning the features of motion states. As opposed to the non-incremental LSTM prediction (NI-LSTM) method, it avoids the shortage of the poor network learning rate caused by the difference between the state characteristics of various speeds or sailing modes and the state characteristics of the training set.

The rest of this paper is organized as follows. The related works on the key technologies for matching the information in an asynchronous system are introduced in Section 2. In Section 3, the NETIIP process and algorithm modules are provided. Section 4 contrasts the NETIIP approach and the NI-LSTM method using contrast model experiments, and the NETIIP method is proved to be effective and feasible. Some conclusions are given in Section 5.

2. Related Works

2.1. Establishing Fusion Model

Establishing an asynchronous fusion model is one of the key technologies for solving the problem of information matching in asynchronous systems. Some scholars have conducted related research on such techniques. A cubature Kalman interactive multi-modal fusion algorithm was one of these methods [11]. After initially generating the

models under different frequencies based on the observed values of different sampling frequencies, this algorithm used the higher frequency fusion period, and the interactive multiple model (IMM) approach was used to weight the predicted values of each model. Furthermore, Ref. [12] proposed two distributed asynchronous fusion algorithms, batch generalization covariance intersection (B-GCI) and sequential generalization covariance intersection (S-GCI), for the asynchronous sensor fusion problem for an arbitrary number of sensors with different sampling rates within the framework of the random finite set (RFS) theory. S-GCI was utilized to lessen the B-heavy GCI's computational load, while B-GCI was used to avoid the challenging computations of cross-covariance among the local posteriors of sensors. To provide an asynchronous track fusion technique with information feedback, an asynchronous track fusion model with an irregular time interval of the observation data and in conjunction with the track quality with multiple model (TQMM) was developed in [13]. The asynchronous multi-sensor fusion system's performance was improved by the method's weight allocation, which made use of the TQMM. The unique issue of each sub-navigation sensor in the multi-sensor integrated navigation system with data sampling rates that were different and several times the rational rate was addressed by reconstructing the state equation and measurement equation of the system and basing it on the matrix operator with scale and wavelet properties [14]. Moreover, Ref. [15] studied the multi-target tracking problem based on an asynchronous network of sensors with different sampling rates. This method carried out an arithmetic averaging approach, recursively, in a timely manner according to the network-wide sampling time sequence in order to fuse the filter estimates obtained at different sensors conditioned on asynchronous measurements.

2.2. Time Alignment

Time alignment is another strategy for overcoming asynchronous fusion. Local estimation results for varied period sizes are brought into agreement with the uniform time by employing timestamp alignment. To ensure that the time alignment parameters converged well, a convex combination of the instantaneous cost and the squared difference between current and past estimates was proposed, and arithmetic average fusion was used to merge the intensities from various nodes [16]. Additionally, Ref. [17] proposed batch time-aligned asynchronous fusion using unbiased converted measurements. The connection between measurement error covariance and the actual measurement was used to demonstrate that this approach is theoretically suboptimal. Ref. [18] believed that applying the time alignment would result in additional estimation error. In order to address this issue, a technique for asynchronous track-to-track association without time alignment was presented. In this algorithm, the deviation in the time sequence was considered as the deviation in each local estimated track.

As can be seen from the research mentioned above, three categories can be made for asynchronous multi-sensor fusion techniques. The first is creating an asynchronous fusion model and feeding it inputs that are sensor measurements at varied sampling rates. Theoretically, this method is the most accurate, but creating a workable model for asynchronous fusion is difficult. Inaccurate or imprecise models will result in a reduction in fusion accuracy. The second method is distributed timing alignment. The time stamp alignment method is used to align the results of each local estimation after sensors with various sampling frequencies are estimated locally. This method is simpler to implement than the first, but the key to ensuring the correctness of the fusion using this method is to figure out how to eliminate the excess mistake. The final approach disregards the timing problem and considers it as a measurement error for direct fusion. This method performs better with measuring systems that change slowly, but it has the lowest accuracy and the least amount of theoretical support when the thing being measured changes fast and often.

The research shows that it is difficult to create an asynchronous sampling model based on the self-positioning sensor group and the ambient sensing sensor group, which are not independent of one another, for the navigation systems of MASS. In other words, the information gathered through environmental sensing is reliant on a ship's own attitude

and position. Therefore, the timing alignment method is more suitable for this paper. Using the method of prediction to achieve time alignment can not only consider the change law of the system with time but also has no need to establish an accurate model, which is easy to realize in a practical engineering application.

3. Materials and Methods

3.1. Problem Statement

The sample frequency of each sensor is not exactly the same in a heterogeneous multi-sensor system. For instance, the positioning sensor's sample frequency in a MASS intelligent navigation system is typically 1 to 5 Hz (such as GPS), whereas a visible light camera and other sensors that are frequently used to collect data from the environment have a sampling frequency that is typically greater than 20 Hz. Figure 1 depicts the sampling diagram for a kind of heterogeneous multi-sensor.

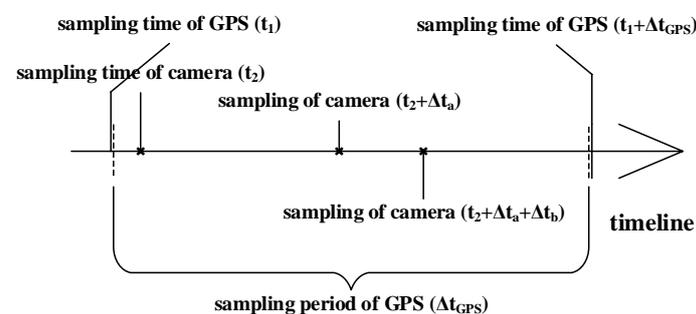


Figure 1. Multi-sensor system sampling diagram for MASS.

In Figure 1, Δt_a and Δt_b represent the length of the time difference between the continuous sampling times for camera.

Because the estimating technique often requires much less time than a positioning sensor's sampling period, the equal time interval estimation strategy is frequently employed to estimate a ship's own motion state. Nevertheless, the complexity of the external environment for a sensor of the external environment is positively connected to the time of the recognition and localization algorithm of the external target. The length of time required for detection and localization algorithms varies depending on how complicated the environment is (as illustrated in Figure 1, $\Delta t_a \neq \Delta t_b$). In other words, the sample interval used by an external environment sensing sensor can be thought of as not being equal.

In a ship's autonomous navigation system, the sensing module must simultaneously superimpose the information gathered by the external environment sensing sensor and the state vector of the sensor's measurement coordinates origin in order to determine the type of the external target obstacle and its absolute position coordinates in real-time. The origin state of sensor measurement coordinates is related to the position and attitude of the ship, etc. If the offset and rotation of the sensor installation position are not taken into account, and the measurement coordinate system of the sensor coincides with the coordinate system of ship movement, the relationship can be described as follows:

$$\boldsymbol{\eta}_{absolute} = \boldsymbol{\eta}_{ship} + \boldsymbol{\eta}_{relative} \quad (1)$$

where $\boldsymbol{\eta}_{absolute}$ refers to the motion state of the external target obstacle in the geodetic coordinate system (absolute state), $\boldsymbol{\eta}_{ship}$ is the present motion state of the ship, and $\boldsymbol{\eta}_{relative}$ is the motion state of the target measured with the sensor (relative state). However, because of the difference in frequency, Formula (1) is described as (taking the estimation at time t_2 as an example):

$$\boldsymbol{\eta}_{absolute}(t_2) = \boldsymbol{\eta}_{ship}(t_2) + \boldsymbol{\eta}_{relative}(t_2) \neq \boldsymbol{\eta}_{ship}(t_1) + \boldsymbol{\eta}_{relative}(t_2) \quad (2)$$

where $\boldsymbol{\eta}(t)$ means the motion state at time t .

Therefore, one need for guaranteeing the sensing accuracy of a ship's autonomous navigation sensing module is to anticipate the ship's motion state at uneven time intervals according to the value $\eta_{ship}(t_1)$ at equal time intervals.

3.2. Design of the Prediction Process

To ensure the real-time performance of the heterogeneous multi-sensor system and reduce the computation required for the prediction process, a marginal computing framework is employed, as shown in the flowchart in Figure 2.

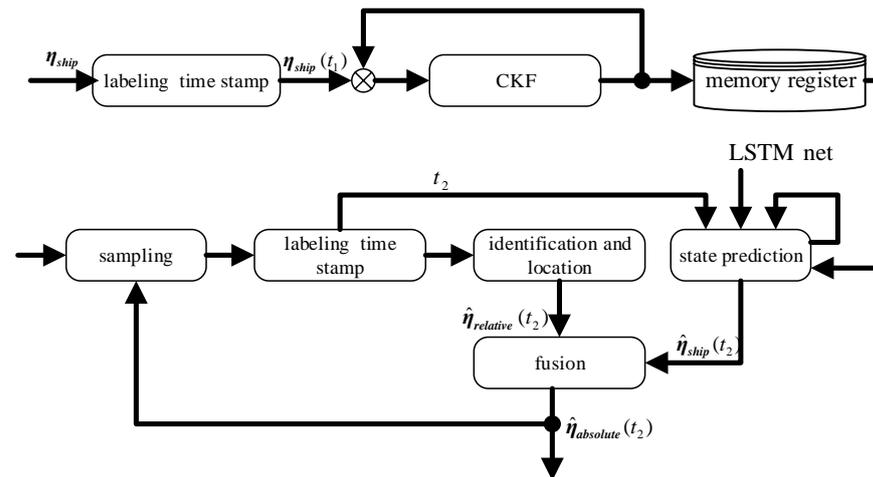


Figure 2. The algorithm flowchart of NETIIP.

In Figure 2, η represents the measurement value of the sensor, while $\hat{\eta}$ represents the estimated value of the state.

The NETIIP algorithm is mostly composed of two components. The first section is the equal time interval estimation of a ship's own motion state, in which the CKF estimator is used to carry out a real-time estimation of the ship's 6-DOF motion state in accordance with the position and navigation sensor data sampled at the same period. The results $\hat{\eta}_{ship}(t_1)$ are stored in the memory register.

The second component is a state predictor built using the LSTM framework. First, according to the time stamp sampled at irregular intervals, the most recent historical moment dataset is taken from the memory register. The ship motion state at time t_2 is then predicted using the LSTM network based on offline training, and the network is simultaneously updated. In contrast to the conventional NI-LSTM network, the network will additionally use the timestamp of non-equal interval sampling as an input variable to realize non-equal interval state prediction. In the subsections that follow, we go into further detail on the designs of the two pieces.

3.3. Ship Motion Mathematical Model

The kinematic model and dynamic model are two categories for the mathematical representation of ship motion [19]. The dynamic model is primarily used to examine the link between a ship's propulsion and external environmental factors on the ship's motion, whereas the kinematic model treats the ship as a particle and is primarily used to study the change in the ship's position and heading over time. In contrast to the kinematic model, the dynamic model is built using the ship's precise hydrodynamic characteristics, thrust force, and environmental force. Accurately obtaining these variables for the application is very challenging. As a result, we estimate the ship's 6-DOF motion state using the ship kinematic model.

Firstly, as shown in Figure 3, the geodetic and shipboard coordinate systems are constructed.

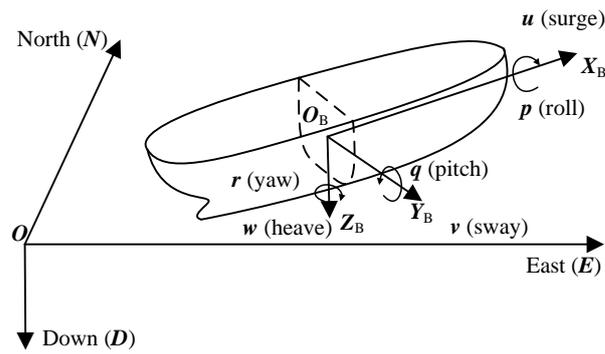


Figure 3. Coordinate system diagram.

All coordinate systems in Figure 3 follow the right-handed criterion, where $O - NED$ is the geodetic coordinate system, also known as the navigation coordinate system, while $O_B - X_B Y_B Z_B$ is the shipboard coordinate system, with the center of gravity of the ship as the origin, and the positive direction of $X_B Y_B Z_B$ points to the bow, starboard, and bottom of the ship, respectively. The ship moves in three axes of surge, sway, and heave as well as in three axes of roll, pitch, and yaw.

As a result, the 6-DOF motion status of the ship can be expressed as follows:

$$\boldsymbol{\eta}_{ship} = [x \ y \ z \ \theta \ \varphi \ \psi]^T \quad (3)$$

$$\boldsymbol{v}_{ship} = [u \ v \ w \ p \ q \ r]^T \quad (4)$$

where $\boldsymbol{\Omega} = [x \ y \ z]$ indicates the coordinate of O_B in the $O - NED$ and $\boldsymbol{\Theta} = [\theta \ \varphi \ \psi]$ represents the ship's roll angle, pitch angle, and heading at the current moment, respectively, and \boldsymbol{v}_{ship} represents the speed of the ship's 6-DOF motion in $O_B - X_B Y_B Z_B$.

In Figure 3, according to the spatial coordinate transformation relation, the first derivative $\dot{\boldsymbol{\eta}}_{ship}$ of the $\boldsymbol{\eta}_{ship}$ has the following relation to \boldsymbol{v}_{ship} :

$$\dot{\boldsymbol{\eta}}_{ship} = \begin{bmatrix} \boldsymbol{R}_{\Theta} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \boldsymbol{I}_{3 \times 3} \end{bmatrix} \boldsymbol{v}_{ship} = \boldsymbol{J}_{\Theta} \boldsymbol{v}_{ship} \quad (5)$$

where $\boldsymbol{I}_{3 \times 3}$ is a third-order identity matrix, $\mathbf{0}_{3 \times 3}$ is a third-order zero matrix, and \boldsymbol{R}_{Θ} is the rotation matrix, which can be expressed as

$$\boldsymbol{R}_{\Theta} = \boldsymbol{R}_{Z_B} \boldsymbol{R}_{Y_B} \boldsymbol{R}_{X_B} = \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\varphi & 0 & s\varphi \\ 0 & 1 & 0 \\ -s\varphi & 0 & c\varphi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\theta & -s\theta \\ 0 & s\theta & c\theta \end{bmatrix} \quad (6)$$

with c and s standing for $\cos()$ and $\sin()$ trigonometric operators, respectively.

When the discrete time is short enough, the 6-DOF velocity of the ship can be regarded as constant. Therefore, the ship's motion can be described discretely as:

$$\boldsymbol{\eta}_{ship}(t_k) = \boldsymbol{I}_{6 \times 6} \times \boldsymbol{\eta}_{ship}(t_{k-1}) + \boldsymbol{T} \times \dot{\boldsymbol{\eta}}_{ship}(t_{k-1}) \quad (7)$$

where $\boldsymbol{T} = \text{diag}_{6 \times 6}(\Delta t_{\text{GPS}})$ represents the discrete time matrix.

Considering that the increment $\boldsymbol{\eta}_{ship}$ is related to \boldsymbol{v}_{ship} , it is necessary to estimate the state of $\boldsymbol{\eta}_{ship}$ and \boldsymbol{v}_{ship} at equal time intervals. Therefore, the state $\hat{\boldsymbol{X}}_{ship}$ to be estimated is

$$\hat{\boldsymbol{X}}_{ship} = [\boldsymbol{\eta}_{ship}^T \ \boldsymbol{v}_{ship}^T]^T \quad (8)$$

According to Equations (5)–(8), ship motion can be described as the following state–space equation:

$$\hat{\mathbf{X}}_{ship}(t_k) = \begin{bmatrix} \mathbf{I}_{6 \times 6} & \mathbf{J}_{\Theta}(t_{k-1}) \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_{6 \times 6} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{I}_{6 \times 6} & \mathbf{T} \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_{6 \times 6} \end{bmatrix} \begin{bmatrix} \mathbf{I}_{6 \times 6} & \mathbf{J}_{\Theta}(t_{k-1}) \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_{6 \times 6} \end{bmatrix} \hat{\mathbf{X}}_{ship}(t_{k-1}) \quad (9)$$

where $\mathbf{J}_{\Theta}(t_{k-1})$ and $\hat{\mathbf{X}}_{ship}(t_{k-1})$ are all related to $\Theta(t_{k-1})$. Therefore, the ship motion is nonlinear.

However, considering that the common shipborne sensors cannot observe all the states in the $\hat{\mathbf{X}}_{ship}$, we only use the sensor group (e.g., GPS and INS) to observe $\boldsymbol{\eta}_{ship}(t_k)$:

$$\mathbf{Z}_{ship}^1(t_k) = [x^{\text{GPS}}(t_k) \quad y^{\text{GPS}}(t_k) \quad z^{\text{GPS}}(t_k) \quad \theta^{\text{INS}}(t_k) \quad \varphi^{\text{INS}}(t_k) \quad \psi^{\text{INS}}(t_k)]^T \quad (10)$$

where \cdot^{GPS} and \cdot^{INS} indicate that the state is observed using GPS and INS, respectively.

Due to the limited accuracy of GPS for the ship's current elevation measurement, which may reduce the accuracy of estimation, z^{GPS} are set to 0. Moreover, the time difference method is used to observe the velocity state $v_{ship}(t_k)$:

$$\mathbf{Z}_{ship}^2(t_k) = \mathbf{J}_{\Theta}(t_k)^{-1} \left(\frac{\mathbf{Z}_{ship}^1(t_k) - \mathbf{Z}_{ship}^1(t_{k-1})}{\Delta t_{\text{GPS}}} \right) \quad (11)$$

The observation corresponding to Equation (9) of state can be expressed as:

$$\mathbf{Z}_{ship}(t_k) = [\mathbf{Z}_{ship}^1(t_k)^T \quad \mathbf{Z}_{ship}^2(t_k)^T]^T \quad (12)$$

Equations (9)–(12) demonstrate that the equations for state and observation are both nonlinear; hence, the nonlinear estimating method must be used to estimate the motion state of the ships over equal time intervals.

3.4. Equal Time Interval Estimation Method for Discrete Nonlinear Systems

Conventional linear system estimating methods such as the Kalman filter cannot be used to estimate ship motion in nonlinear systems in real-time. As a result, nonlinear estimation techniques, such as the extended Kalman filter (EKF) [20], the UKF [21], and the CKF [22], are frequently utilized for the estimation of ship motion. As nonlinear filtering techniques, both the EKF and UKF can accomplish real-time state estimation in nonlinear ship motion systems. Meanwhile, the UKF offers the advantages of quicker convergence, higher filtering precision, and easier implementation [23]. However, when estimating in nonlinear systems with high dimensions (dimensions greater than three), the CKF performs more accurately than the UKF [24]. Next, we will introduce the calculation flow of the two popular nonlinear system point estimate methods, the CKF and UKF, and the flow charts are shown in Figure 4.

3.4.1. CKF Estimation Method

The CKF is based on the third-order spherical–radial cubature criterion and uses a set of cubature points to approximate the mean and covariance of states for nonlinear systems with additional Gaussian noise, which is theoretically the closest approximation algorithm to Bayesian filtering. The CKF mainly consists of two parts—time update and measurement update—and the algorithm flow is as follows.

(I) Time Update

Step 1: Perform Cholesky decomposition for the error covariance $\mathbf{P}_{t_{k-1}}$ at time t_{k-1} :

$$\mathbf{P}_{t_{k-1}} = \mathbf{S}_{t_{k-1}} \mathbf{S}_{t_{k-1}}^T \quad (13)$$

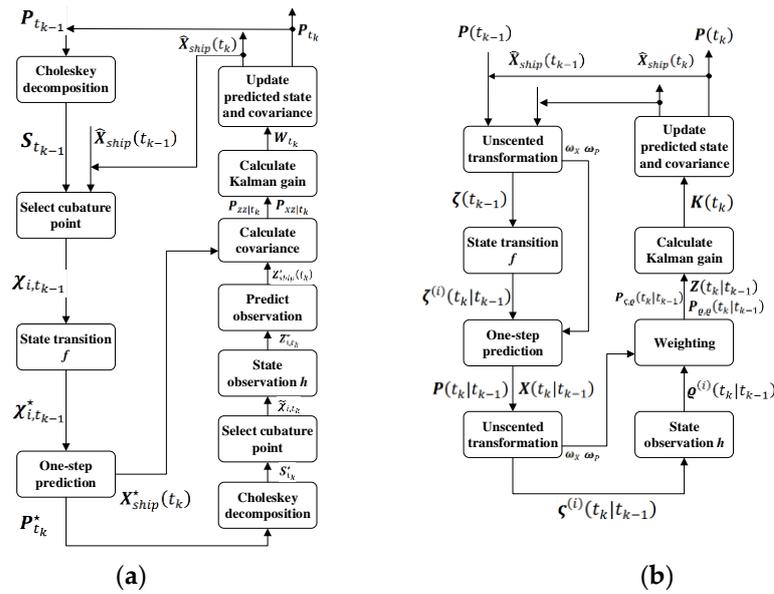


Figure 4. Algorithm flow charts: (a) CKF and (b) UKF.

Step 2: Select cubature point ($i = 1, 2, \dots, 2n$):

$$\chi_{i,t_{k-1}} = S_{t_{k-1}} \zeta_i + \hat{X}_{ship}(t_{k-1}) \quad (14)$$

$$\zeta_i = \sqrt{n} [I_n - I_n]_i \quad (15)$$

where $n = 12$ represents the dimension of $\hat{X}_{ship}(t_{k-1})$, I_n is the identity matrix of order n , and $[I_n - I_n]_i$ is the i th column vector of the matrix $[I_n - I_n]$.

Step 3: Propagate the cubature point through Equation (9):

$$\chi_{i,t_{k-1}}^* = f(\chi_{i,t_{k-1}}) \quad (16)$$

Step 4: Estimate the predicted value of state at time t_k :

$$X_{ship}^*(t_k) = \frac{1}{2n} \sum_{i=1}^{2n} \chi_{i,t_{k-1}}^* \quad (17)$$

Step 5: Estimate the predicted value of the error covariance at time t_k :

$$P_{t_k}^* = \frac{1}{2n} \sum_{i=1}^{2n} \chi_{i,t_{k-1}}^* \chi_{i,t_{k-1}}^{*T} - X_{ship}^*(t_k) X_{ship}^*(t_k)^T + Q \quad (18)$$

where Q represents the process noise matrix of the system, which is an adjustable parameter.

(II) Measurement Update

Step 1: Perform Cholesky decomposition for the predicted value of the error covariance $P_{t_k}^*$ at time t_k :

$$P_{t_k}^* = S_{t_k}^* S_{t_k}^{*T} \quad (19)$$

Step 2: Select cubature point:

$$\tilde{\chi}_{i,t_k} = S_{t_k}^* \zeta_i + \chi_{i,t_{k-1}}^* \quad (20)$$

Step 3: Propagate the cubature point through Equations (10)–(12):

$$\mathbf{Z}_{i,t_k}^* = h(\tilde{\chi}_{i,t_k}) \quad (21)$$

Step 4: Estimate the predicted value of measurement at time t_k :

$$\mathbf{Z}_{ship}^*(t_k) = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{Z}_{i,t_k}^* \quad (22)$$

Step 5: Estimate the autocorrelation covariance matrix:

$$\mathbf{P}_{zz|t_k} = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{Z}_{i,t_k}^* \mathbf{Z}_{i,t_k}^{*T} - \mathbf{Z}_{ship}^*(t_k) \mathbf{Z}_{ship}^*(t_k)^T + \mathbf{R} \quad (23)$$

where \mathbf{R} represents the measurement noise matrix of the system, which is also an adjustable parameter.

Step 6: Estimate the cross-correlation covariance matrix:

$$\mathbf{P}_{xz|t_k} = \frac{1}{2n} \sum_{i=1}^{2n} \tilde{\chi}_{i,t_k} \mathbf{Z}_{i,t_k}^{*T} - \mathbf{X}_{ship}^*(t_k) \mathbf{Z}_{ship}^*(t_k)^T \quad (24)$$

Step 7: Estimate the Kalman gain:

$$\mathbf{W}_{t_k} = \mathbf{P}_{xz|t_k} \mathbf{P}_{zz|t_k}^{-1} \quad (25)$$

Step 8: Estimate the state at time t_k :

$$\hat{\mathbf{X}}_{ship}(t_k) = \mathbf{X}_{ship}^*(t_k) + \mathbf{W}_{t_k} (\mathbf{Z}_{ship}(t_k) - \mathbf{Z}_{ship}^*(t_k)) \quad (26)$$

Step 9: Update the error covariance matrix at time t_k :

$$\mathbf{P}_{t_k} = \mathbf{P}_{t_k}^* - \mathbf{W}_{t_k} \mathbf{P}_{zz|t_k} \mathbf{W}_{t_k}^T \quad (27)$$

3.4.2. UKF Estimation Method

In contrast to the CKF estimation, the UKF estimation method depicts the Gaussian distribution of a nonlinear function via unscented transformation (UT). The algorithm flow is as follows : $\mathbf{P}(t_{k-1})$.

Step 1: The UT is used to compute $2n + 1$ Sigma points $\zeta(t_{k-1})$, the state weight matrix $\omega_X(t_{k-1})$, and variance weight matrix $\omega_P(t_{k-1})$:

$$\zeta^{(1)}(t_{k-1}) = \hat{\mathbf{X}}_{ship}(t_{k-1}) \quad (28)$$

$$\zeta^{(i)}(t_{k-1}) = \hat{\mathbf{X}}_{ship}(t_{k-1}) + \sqrt{(n + \lambda) \mathbf{P}^{(i-1)}(t_{k-1})} \quad (i = 2, 3, \dots, n + 1) \quad (29)$$

$$\zeta^{(i)}(t_{k-1}) = \hat{\mathbf{X}}_{ship}(t_{k-1}) - \sqrt{(n + \lambda) \mathbf{P}^{(i-n-1)}(t_{k-1})} \quad (i = n + 2, n + 3, \dots, 2n + 1) \quad (30)$$

$$\omega_X^{(1)}(t_{k-1}) = \frac{\lambda}{n + \lambda} \quad (31)$$

$$\omega_P^{(1)}(t_{k-1}) = \frac{\lambda}{n + \lambda} + (1 - \alpha^2 + \beta) \quad (32)$$

$$\omega_X^{(i)}(t_{k-1}) = \omega_P^{(i)}(t_{k-1}) = \frac{\lambda}{2(n + \lambda)} \quad (i = 2, 3, \dots, 2n + 1) \quad (33)$$

where $\cdot^{(i)}$ represents the i th column of the matrix. $n = 12$ represents the dimension of $\hat{\mathbf{X}}_{ship}$. $\lambda = \alpha^2(n + \kappa) - n$ is the proportional parameter. α , β , and κ are adjustable parameters, and the values set in this paper are as follows: $\alpha = 0.99999$, $\beta = 0.3$, and $\kappa = 3$.

Step 2: One step predicts the state of Sigma points through Equation (9):

$$\zeta^{(i)}(t_k|t_{k-1}) = f(\zeta^{(i)}(t_{k-1})) \quad (i = 1, 2, 3, \dots, 2n + 1) \quad (34)$$

Step 3: The Sigma points are weighted to obtain a one-step prediction of the state $\mathbf{X}(t_k|t_{k-1})$ and variance matrix $\mathbf{P}(t_k|t_{k-1})$:

$$\mathbf{X}(t_k|t_{k-1}) = \sum_{i=1}^{2n+1} [\omega_X^{(i)}(t_{k-1})\zeta^{(i)}(t_k|t_{k-1})] \quad (35)$$

$$\mathbf{P}(t_k|t_{k-1}) = \sum_{i=1}^{2n+1} \{ \omega_P^{(i)}(t_{k-1})[\mathbf{X}(t_k|t_{k-1}) - \zeta^{(i)}(t_k|t_{k-1})][\mathbf{X}(t_k|t_{k-1}) - \zeta^{(i)}(t_k|t_{k-1})]^T \} + \mathbf{Q} \quad (36)$$

Step 4: The first step is repeated to obtain a new set of Sigma points $\zeta(t_k|t_{k-1})$ by performing UT on the one-step prediction of the state $\mathbf{X}(t_k|t_{k-1})$ and variance matrix $\mathbf{P}(t_k|t_{k-1})$.

Step 5: The observed value $\varrho(t_k|t_{k-1})$ of the Sigma point is obtained through Equations (10)–(12):

$$\varrho^{(i)}(t_k|t_{k-1}) = h(\zeta^{(i)}(t_k|t_{k-1})) \quad (i = 1, 2, 3, \dots, 2n + 1) \quad (37)$$

Step 6: By weighting the observed values of the new Sigma point set, the mean value of the observation results, the auto-covariance matrix, and the cross-covariance matrix are obtained:

$$\mathbf{Z}(t_k|t_{k-1}) = \sum_{i=1}^{2n+1} [\omega_X^{(i)}(t_{k-1})\varrho^{(i)}(t_k|t_{k-1})] \quad (38)$$

$$\mathbf{P}_{\varrho, \varrho}(t_k|t_{k-1}) = \sum_{i=1}^{2n+1} \{ \omega_P^{(i)}(t_{k-1})[\mathbf{Z}(t_k|t_{k-1}) - \varrho^{(i)}(t_k|t_{k-1})][\mathbf{Z}(t_k|t_{k-1}) - \varrho^{(i)}(t_k|t_{k-1})]^T \} + \mathbf{R} \quad (39)$$

$$\mathbf{P}_{\zeta, \varrho}(t_k|t_{k-1}) = \sum_{i=1}^{2n+1} \{ \omega_P^{(i)}(t_{k-1})[\mathbf{Z}(t_k|t_{k-1}) - \zeta^{(i)}(t_k|t_{k-1})][\mathbf{Z}(t_k|t_{k-1}) - \varrho^{(i)}(t_k|t_{k-1})]^T \} \quad (40)$$

Step 7: The Kalman gain is estimated:

$$\mathbf{K}(t_k) = \mathbf{P}_{\zeta, \varrho}(t_k|t_{k-1})\mathbf{P}_{\varrho, \varrho}(t_k|t_{k-1})^{-1} \quad (41)$$

Step 8: The ship motion state $\hat{\mathbf{X}}_{ship}(t_k)$ and covariance $\mathbf{P}(t_k)$ are updated:

$$\hat{\mathbf{X}}_{ship}(t_k) = \mathbf{X}(t_k|t_{k-1}) + \mathbf{K}(t_k)[\mathbf{Z}_{ship}(t_k) - \mathbf{Z}(t_k|t_{k-1})] \quad (42)$$

$$\mathbf{P}(t_k) = \mathbf{P}(t_k|t_{k-1}) - \mathbf{K}(t_k)\mathbf{P}_{\varrho, \varrho}(t_k|t_{k-1})\mathbf{K}(t_k)^T \quad (43)$$

3.5. Incremental LSTM Prediction Method

The LSTM neural network is a unique type of recurrent neural network (RNN), which may successfully combat the issue of gradient disappearance or explosion that typical RNN networks experience as training time and network layer rise. Therefore, the LSTM network is often used for timing prediction.

The LSTM network is often only used to forecast a ship's track information in a ship track prediction application, and the projected step size is typically the equal time interval [25]. Its mapping connection may be expressed as follows:

$$\mathbf{\Omega}(t + 1) = f(\{\mathbf{\Omega}(t - \kappa + 1), \dots, \mathbf{\Omega}(t - 1), \mathbf{\Omega}(t)\}) \quad (44)$$

where κ represents the length of the historical sequence in the LSTM network.

Figure 5 depicts the network structure of the LSTM. Three gate structures make up the LSTM network. They are as follows:

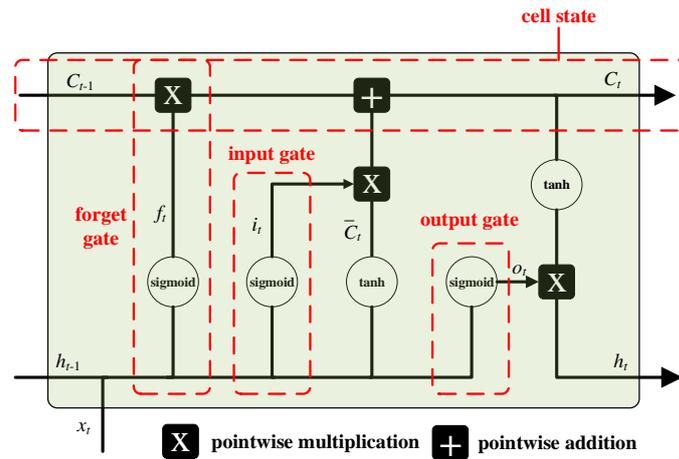


Figure 5. LSTM network structure.

(1) Forget gate: This is used to decide whether to keep or discard information. The information from the previous hidden state h_{t-1} and the input information ε_t are simultaneously processed via a sigmoid function to calculate the output value. The closer the result f_t is to 0, the more it should be discarded. The forgetting factor can be calculated using the following formula:

$$f_t = \sigma(W_f \cdot [h_{t-1}, \varepsilon_t] + b_f) \quad (45)$$

where W_f is the weight matrix of the forgetting gate, b_f is the offset term of the forgetting gate, and σ is the sigmoid function, which is specifically expressed as:

$$\sigma(net) = \frac{1}{1 + e^{-net}} \quad (46)$$

(2) Input gate: This is used to update cell status. The information of the hidden state of the previous layer and the information of the current input are calculated via the sigmoid function to determine the type of updated information. Meanwhile, the information of the hidden state of the previous layer and the information of the current input are calculated via the \tanh function to create a new candidate value vector. Finally, the output of the sigmoid function is multiplied by the output of the \tanh function. The calculation formula is

$$i_t = \sigma(W_i \cdot [h_{t-1}, \varepsilon_t] + b_i) \quad (47)$$

$$\bar{C}_t = \tanh(W_c \cdot [h_{t-1}, \varepsilon_t] + b_c) \quad (48)$$

where W_i and W_c are the weight matrices of the input gate, b_i and b_c are the offset terms, and \tanh represents the \tanh function, which is specifically expressed as:

$$\tanh(net) = \frac{e^{net} - e^{-net}}{e^{net} + e^{-net}} \quad (49)$$

After passing through the forget gate and the input gate, the old cell state can be updated as

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \bar{C}_t \quad (50)$$

(3) Output gate: This is used to determine the output value according to the cell state. First, the sigmoid function is used to determine the part of the cell state that needs to be output, then the cell state is calculated via the \tanh function, and, finally, the output of network is multiplied. The calculation formula is

$$o_t = \sigma(W_o \cdot [h_{t-1}, \varepsilon_t] + b_o) \quad (51)$$

$$h_t = o_t \cdot \tanh(C_t) \quad (52)$$

where W_o is the weight matrix of the output gate, and b_o is the offset term of the output gate. In Equations (47)–(52), parameters W_f , W_i , W_c , and W_o and b_f , b_i , b_c , and b_o are all parameters that can be adjusted.

Compared with the ship trajectory prediction method, the traditional NI-LSTM prediction relationship of the ship's six degrees of freedom motion can be expressed as

$$\eta_{ship}(t+1) = f(\{\eta_{ship}(t-\tau+1), \dots, \eta_{ship}(t-1), \eta_{ship}(t)\}) \quad (53)$$

Some researchers utilize LSTM to predict ship motion attitude in addition to ship track prediction. Since ship attitude changes more irregularly and with a narrower change range than ship track, it is challenging to guarantee the accuracy of attitude prediction using solely LSTM. Researchers have made improvements to the LSTM network, including the addition of a self-attention mechanism for multi-scale prediction [26] and the decomposition of ship attitude data by superimposing the LSTM network with other prediction networks [27].

Although the aforementioned method can somewhat raise the LSTM network's prediction accuracy, it requires many more calculations than a single LSTM network, which is not fast enough for the application in this study. On the other hand, the prediction accuracy of timing sequences with non-equal time intervals will be lower because LSTM is typically utilized for timing sequence prediction with equal time intervals. The time difference and the increment of the ship's motion state is input into the network in this article using a single LSTM network structure. Considering that the speed of the ship also affects the change in motion state, the network also uses the ship's speed v_{ship} as an input. The mapping relationship can be expressed as follows:

$$\Gamma(t+1) - \Gamma(t) = f(\delta) \quad (54)$$

$$\delta = \bigcup_{i=1}^{\tau} \left\{ \Delta T_i, \Delta \Gamma_i, v_{ship}(t-i+1) \right\} \quad (55)$$

$$\Delta \Gamma_i = \Gamma(t) - \Gamma(t-i+1) \quad (56)$$

$$\Delta T_i = T_{predict} - T(t-i+1) \quad (57)$$

where $\Gamma = [\hat{x} \ \hat{y} \ \hat{\theta} \ \hat{\phi} \ \hat{\psi}]$ is the latest estimate result \hat{X}_{ship} of the CKF estimator for all elements except \hat{z} . $T_{predict}$ represents the moment to be predicted, and $T(t-i+1)$ ($i = 1, 2, 3, \dots, \tau$) represents the sequence of the latest τ estimated moments with the CKF estimator.

Since the observation of z^{GPS} is set to zero in the observation Equation (10), its estimation with the CKF estimator is inaccurate. In order to avoid the problem of the prediction accuracy being reduced due to the inaccurate quantity, we did not take it as the input for the prediction network.

4. Results and Discussion

The experimental verification work is based on the ship model of the unmanned ship platform "CHCNAV APACHE 6", which is equipped with a dual antenna GPS (installed at the fore and aft of the ship) for real-time measurement of the ship's position information and an attitude instrument (installed close to the center of gravity of the ship) for real-time measurement of the ship's attitude information. The experimental verification work was carried out in the Yanxi Lake experimental site, which is open water in a natural environment with an unknown random environment. Figure 6 displays the ship's model and experimental site.

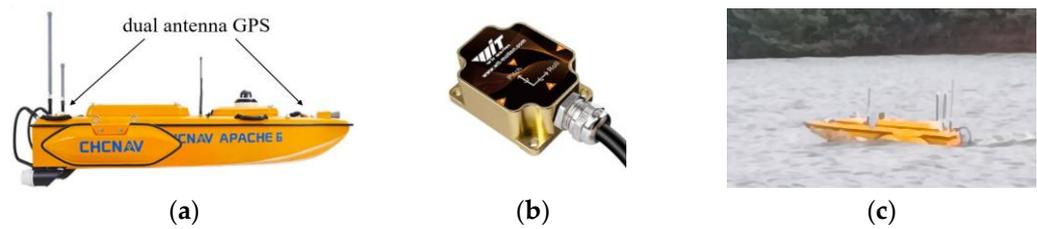


Figure 6. Experimental ship models and sensors: (a) experimental model, (b) attitude instrument, and (c) experimental site.

The navigation of an intelligent ship can be divided into two modes in the application: manual remote control and autonomous navigation. Different modes have different changing rules for the ship motion state. Furthermore, the characteristics of a ship's motion state that change with speed vary. To verify the effectiveness and reliability of the NETIIP method, six sets of experiments were carried out at low speed, medium speed, and high speed, respectively, under the manual remote mode (MRM) and autonomous navigation mode (ANM).

Figure 7 shows the results of the real-time estimation of the ship's position information (taking the north coordinates as an example) in the six trials by using the CKF estimator and UKF estimator. Herein, the longitude and latitude measured with the dual-antenna GPS are converted using the Universal Transverse Mercator (UTM) projection method. Meanwhile, the initial values of the error covariance P in Formulas (13), (29) and (30), the process noise matrix Q in Formulas (18) and (36), and the measurement noise matrix R in Formulas (23) and (39) are all set to the same value in order to compare the estimation accuracy of the two estimation techniques.

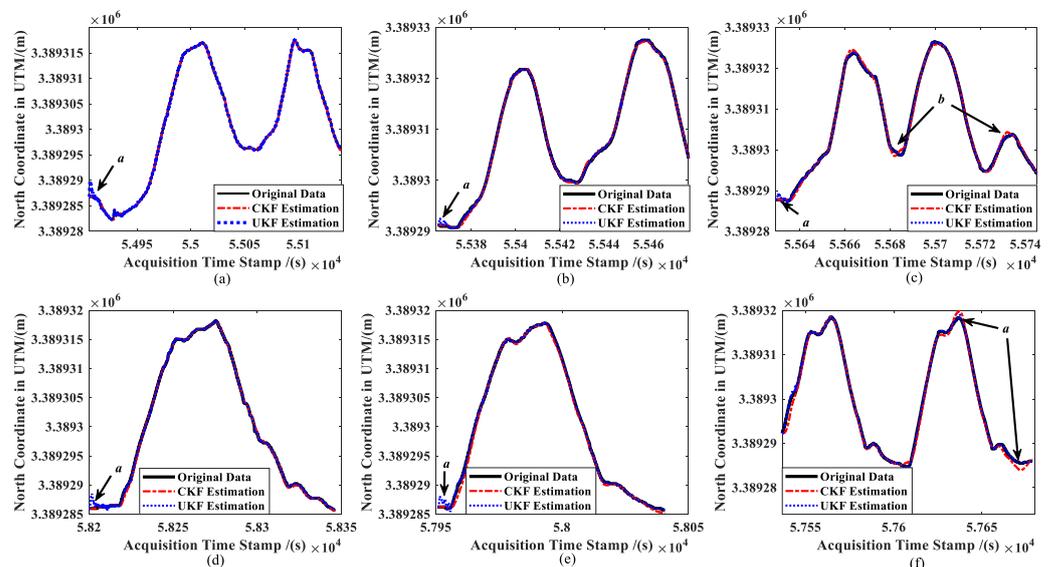


Figure 7. Comparison of ship position estimations: (a) slow speed of MRM, (b) medium speed of MRM, (c) high speed of MRM, (d) slow speed of ANM, (e) medium speed of ANM, and (f) high speed of ANM.

The CKF and UKF both have their own benefits and drawbacks for estimating ship positioning data in real-time. The initial convergence rate of the CKF is marginally quicker than the UKF's for the same set of algorithm parameters (a in Figure 7a–e). Nevertheless, the tracking effect of the CKF is worse than the UKF when a ship is turning with a high speed (b in Figure 7c and a in Figure 7f).

Figure 8 shows the results of the real-time estimation of the ship's attitude information (taking the pitch angle of ships as an example) in the six experiments by using the UKF estimator. Meanwhile, Figure 9 shows the CKF estimation results.

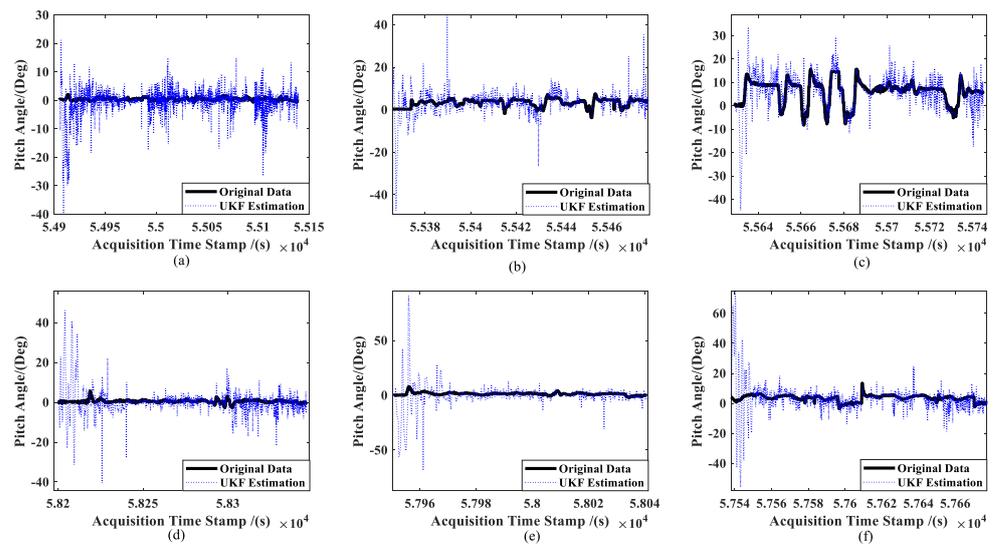


Figure 8. UKF estimation results for the ship's attitude: (a) slow speed of MRM, (b) medium speed of MRM, (c) high speed of MRM, (d) slow speed of ANM, (e) medium speed of ANM, and (f) high speed of ANM.

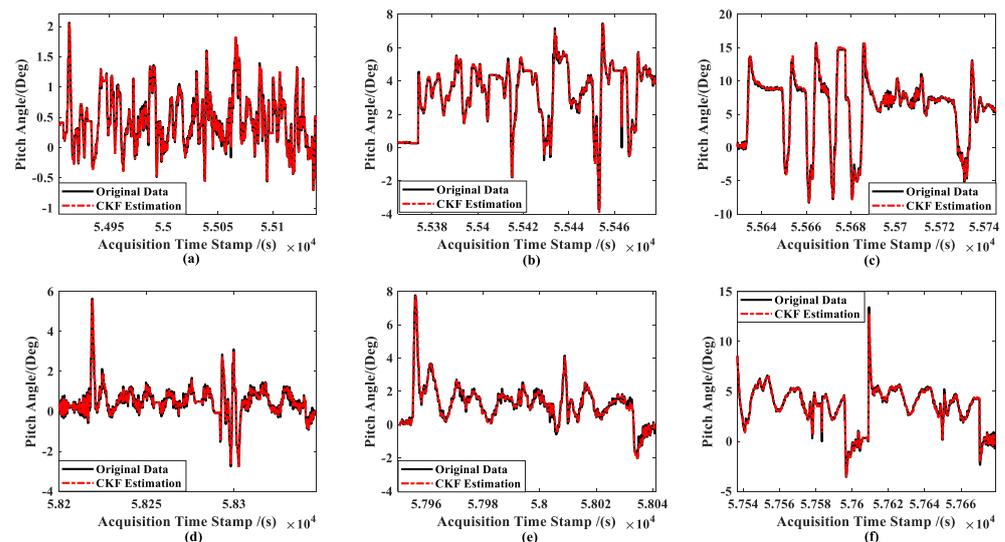


Figure 9. CKF estimation results for the ship's attitude: (a) slow speed of MRM, (b) medium speed of MRM, (c) high speed of MRM, (d) slow speed of ANM, (e) medium speed of ANM, and (f) high speed of ANM.

It can be seen that the UKF estimations have a low accuracy when the attitude angle is close to zero and fluctuates often (see Figure 8a). However, the estimated results can fundamentally converge towards the original data when the attitude angle is far from zero and changes relatively regularly. Comparatively, the CKF produces more accurate results than the UKF for the parameters given, and the CKF estimation method's assessment of the ship's attitude angle generally converges around the observed value. We are unable to determine which approach performs better in terms of estimation as the parameter settings may not be optimal for them. However, it is clear from a comparison of Figures 8 and 9 that the UKF is more susceptible to changes in the features of the item to be estimated, making it more challenging to perform tasks such as parameter modification.

The results of the two approaches' real-time estimations of the ship forward speed u are shown in Figure 10. The figure shows that while the UKF estimation results are nearly non-convergent, the CKF estimation results are smoother. Meanwhile, Equations (58) and (59) are used to calculate the resultant velocities and original observations to accurately evaluate

the estimation accuracy for ship velocity with the two approaches, which is depicted in Figure 11.

$$v_{estimation}(t_k) = \sqrt{u(t_k)^2 + v(t_k)^2} \quad (58)$$

$$v_{observation}(t_k) = \sqrt{\left(\frac{x^{GPS}(t_{k+1}) - x^{GPS}(t_k)}{t_{k+1} - t_k}\right)^2 + \left(\frac{y^{GPS}(t_{k+1}) - y^{GPS}(t_k)}{t_{k+1} - t_k}\right)^2} \quad (59)$$

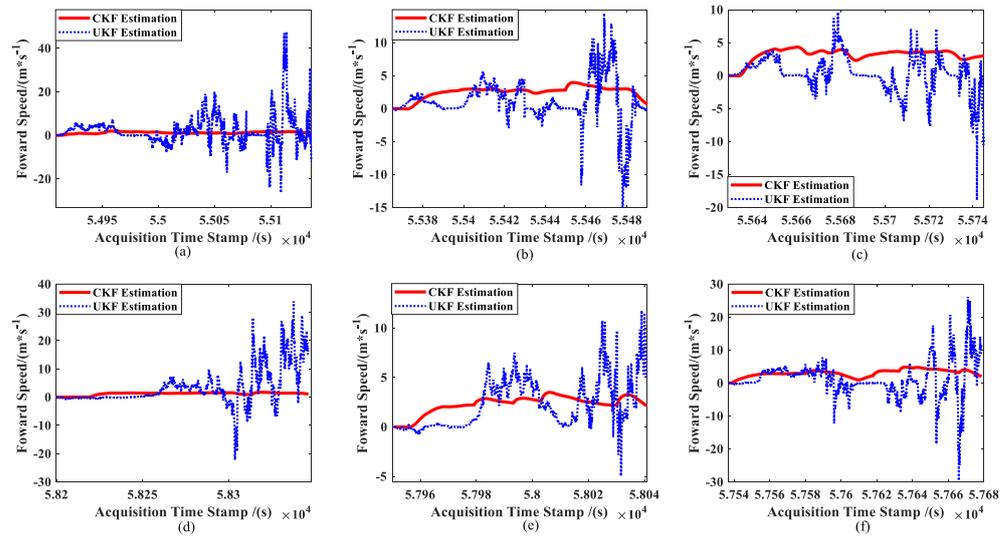


Figure 10. Comparison of ship forward speed estimations: (a) slow speed of MRM, (b) medium speed of MRM, (c) high speed of MRM, (d) slow speed of ANM, (e) medium speed of ANM, and (f) high speed of ANM.

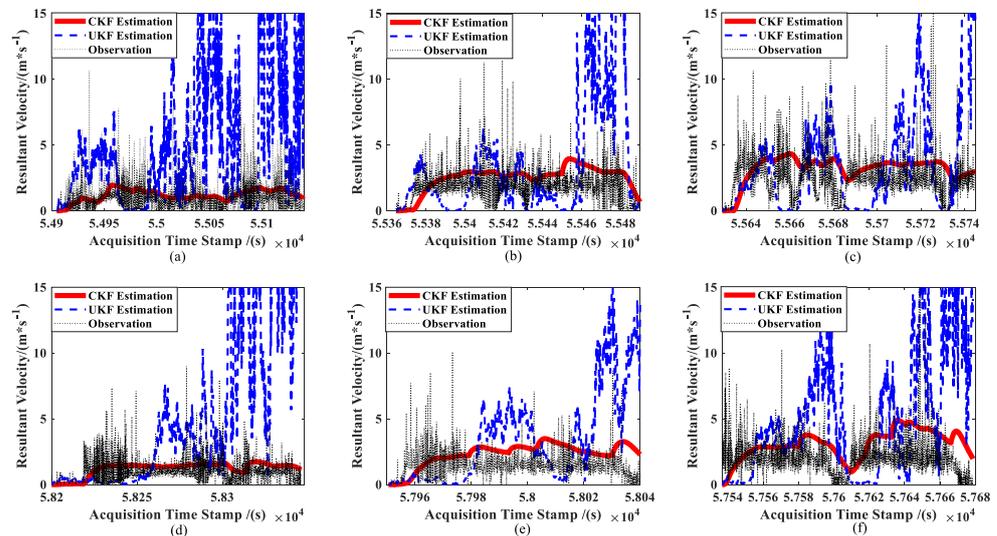


Figure 11. Comparison of ship resultant velocity estimations: (a) slow speed of MRM, (b) medium speed of MRM, (c) high speed of MRM, (d) slow speed of ANM, (e) medium speed of ANM, and (f) high speed of ANM.

Equation (58) is the method for calculating the ship's resultant velocity in the ship-board coordinate system, which is used for the vector addition of the estimated velocity. Meanwhile, Equation (59) is the method for calculating the resultant velocity in the geodetic coordinate system, which is appropriate for calculating the ship's resultant velocity under the sensor's original observation.

In contrast, Figures 10 and 11 show that the results of the CKF estimator for the first-order state (ship speed) essentially converge towards the observations, while the results of the UKF estimator practically diverge.

The root-mean-square error (RMSE) coefficient (Equation (60)) is used to calculate the estimation results of each variable for assessment in order to more effectively evaluate the estimation accuracy of the estimated state variables using the UKF and CK.

$$RMSE = \sqrt{\frac{1}{m} \sum_{i=1}^m (\hat{\chi}_i - \chi_i)^2} \quad (60)$$

where m represents the number of estimated cycles, $\hat{\chi}_i$ represents the variable estimated for the i th period, and χ_i represents the truth value. However, because the true value of the quantity to be estimated cannot be acquired in the experiment, we accept the sensor's observed value as the genuine value. The RMSE coefficient reflects the convergence and credibility of the estimated result. The smaller the value, the more the estimated result converges towards the observed value, and the higher its credibility. The RMSE coefficient of the estimated results is shown in Table 1.

Table 1. RMSE coefficients of different models and different estimators.

Estimator		North	East	Pitch (Rad)	Roll (Rad)	Head (Rad)
CKF	MRM_Slow	0.2081	0.1614	0.0013	0.0021	0.1911
	MRM_Medium	0.4058	0.4747	0.0053	0.0059	0.4508
	MRM_Fast	0.5097	0.7541	0.0110	0.0093	0.3328
	ANM_Slow	0.2378	0.2105	0.0031	0.0097	0.3885
	ANM_Medium	0.4869	0.4133	0.0035	0.0104	0.5389
	ANM_Fast	0.8318	0.5225	0.0059	0.0087	0.4275
UKF	MRM_Slow	0.2278	0.0883	4.0196	2.6769	0.6209
	MRM_Medium	0.1466	0.1160	2.3857	2.7318	1.1448
	MRM_Fast	0.1690	0.1322	1.8902	3.2087	0.9575
	ANM_Slow	0.1005	0.2516	4.1937	3.3752	1.5118
	ANM_Medium	0.2384	0.2239	3.8999	3.5797	1.4721
	ANM_Fast	0.1494	0.1753	2.9908	3.0315	1.0680

Combined with Figures 7–11 and the data in Table 1, under the same set of parameters, the UKF has a higher estimation accuracy than the CKF for ship position estimation. On the other hand, the UKF has better parameter robustness than the CKF. With the increase in ship speed, the estimation accuracy of the CKF decreases gradually; thus, for different ship speeds, the CKF must change various parameters to provide appropriate accuracy. However, the accuracy of the UKF estimation will not decrease with the increase in ship speed, that is, a set of parameters can adapt to the estimation of ship position at different speeds.

However, the CKF's accuracy is substantially greater than the UKF's for ship attitude. Even worse, the attitude estimated using the UKF does not steadily converge towards the observation (see Figure 8). Moreover, since attitude and ship velocity have a strong coupling connection, the UKF's estimation for ship velocity virtually exhibits a diverging trend (see Figure 11). Thus, the UKF is not appropriate for the estimation of the multidimensional autocorrelation state.

Ship speed is one of the inputs for the prediction network in this study (Formula (55)), which has high requirements for the speed-related estimate accuracy of the pre-state estimation module. Therefore, the CKF is chosen as the pre-state estimate module rather than the UKF.

In order to verify the prediction accuracy of the NETIIP, the CKF estimations of the ship in automatic tracking mode with a six-knots set speed are first collected as the training set. The above six groups of data are used as the test set for prediction and compared with the prediction results of the NI-LSTM algorithm. The ship position prediction results are shown in Figure 12, and the attitude prediction results are shown in Figure 13. Therein, the

prediction time is set to a random number with an upper limit of the estimation period of the CKF.

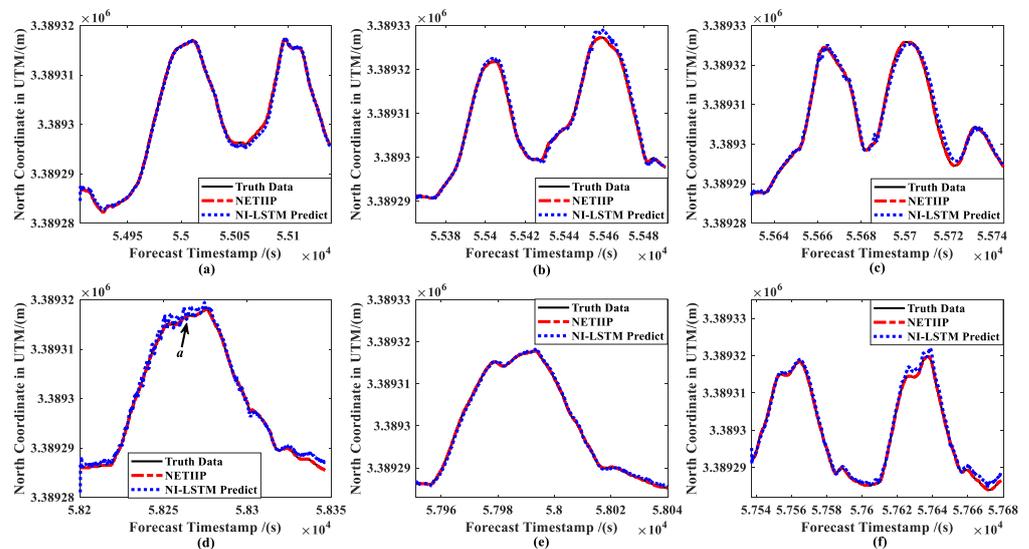


Figure 12. Comparison of ship position predictions: (a) slow speed of MRM, (b) medium speed of MRM, (c) high speed of MRM, (d) slow speed of ANM, (e) medium speed of ANM, and (f) high speed of ANM.

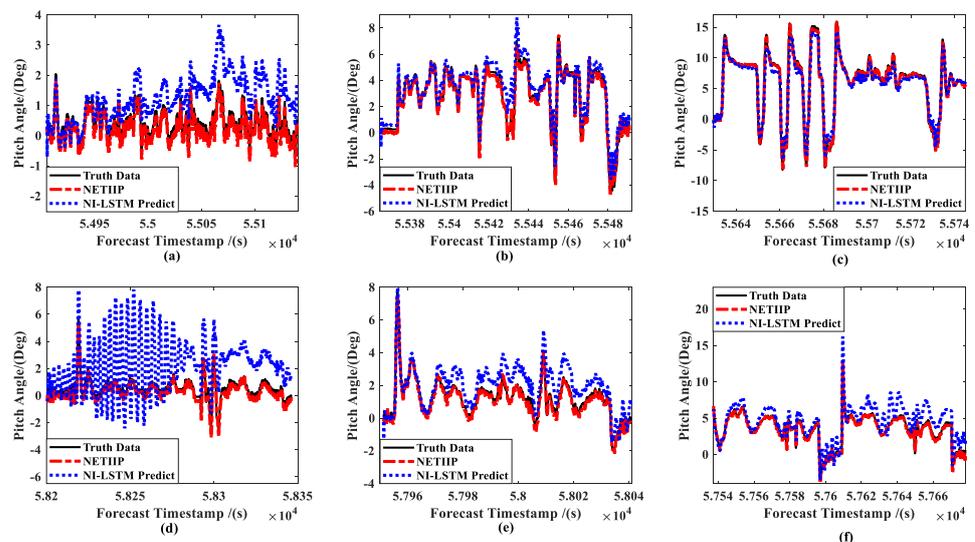


Figure 13. Comparison of ship attitude predictions: (a) slow speed of MRM, (b) medium speed of MRM, (c) high speed of MRM, (d) slow speed of ANM, (e) medium speed of ANM, and (f) high speed of ANM.

Comparing the prediction results of the two methods, it can be seen that the prediction result of NETIIP is closer to the truth data than NI-LSTM. On the other hand, with the change in ship speed, the accuracy of the NI-LSTM prediction also changes greatly. The prediction of the position (a in Figure 12d) and attitude (Figure 13a,d) of the low-speed sailing mode almost diverges. In contrast, the NETIIP technique nearly converges towards the truth data and is less impacted by speed.

To better explore the degree of influence of the ship speed, the absolute value of the prediction error–velocity distribution is plotted, as shown in Figures 14 and 15.

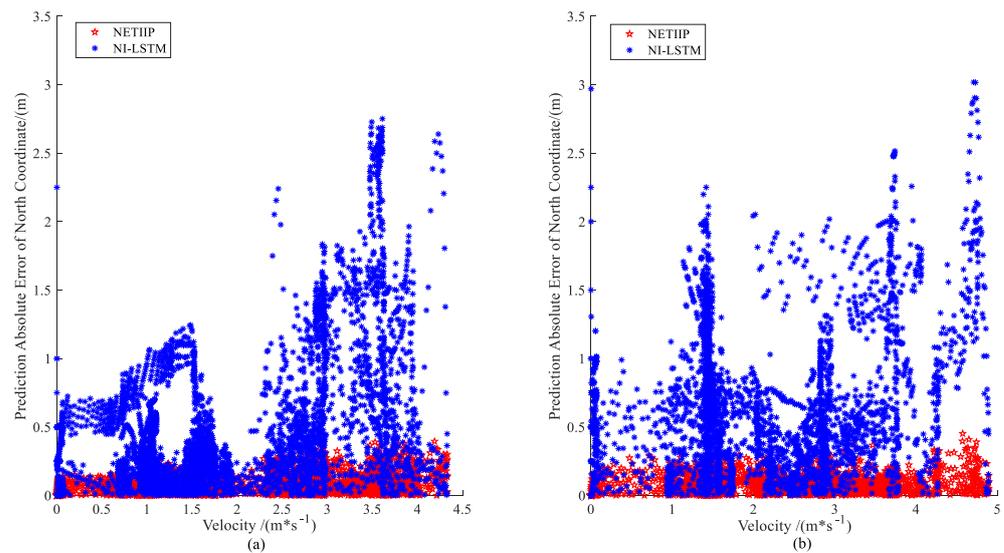


Figure 14. Prediction error–velocity distribution of ship position: (a) MRM and (b) ANM.

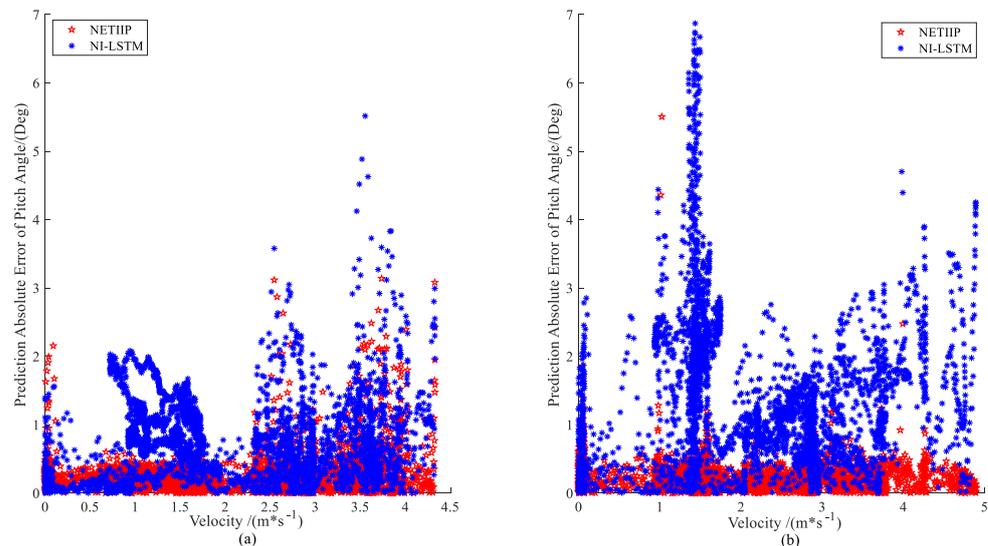


Figure 15. Prediction error–velocity distribution of ship attitude: (a) MRM and (b) ANM.

The prediction error of NI-LSTM tends to grow with the increase in ship speed in both the manual mode and automatic tracking mode, as can be seen from the distribution charts in Figures 14 and 15, while the error of the NETIIP prediction method scarcely changes with ship speed. Since the training set chooses data from the automatic tracking mode, the prediction errors of the two approaches for predicting ship attitude are comparable for the test set in MRM, generally speaking, the NETIIP algorithm is slightly superior to the NI-LSTM algorithm. In ANM, NETIIP maintains a high prediction accuracy, and its prediction error is essentially kept within 1° for the ship attitude. However, the prediction accuracy of NI-LSTM is poor, and its prediction error varies with the change in ship speed. The prediction results of the NI-LSTM algorithm almost exhibit a divergent trend throughout the range of 1–3 m/s, as observed when combined with Figures 13d and 14b.

The RMSE coefficients under each mode are calculated in Table 2.

Table 2. RMSE coefficients of different models and different predictions.

		North	East	Pitch (Rad)	Roll (Rad)	Head (Rad)
NETIIP	MRM_Slow	0.0823	0.0489	0.0039	0.0030	0.2336
	MRM_Medium	0.1079	0.0973	0.0055	0.0054	0.2599
	MRM_Fast	0.1333	0.1349	0.0104	0.0063	0.1927
	ANM_Slow	0.0835	0.0552	0.0050	0.0070	0.1741
	ANM_Medium	0.0994	0.0938	0.0046	0.0070	0.2214
	ANM_Fast	0.1197	0.1132	0.0060	0.0056	0.1753
NI-LSTM	MRM_Slow	0.4396	0.3611	0.0167	0.0193	0.8163
	MRM_Medium	0.7770	0.7772	0.0130	0.0190	1.0182
	MRM_Fast	1.0562	2.4098	0.0187	0.0257	1.6340
	ANM_Slow	0.8328	1.4724	0.0440	0.0218	1.3956
	ANM_Medium	0.3952	0.7633	0.0162	0.0138	0.5420
	ANM_Fast	1.0255	0.9913	0.0253	0.0177	0.7663

According to Equation (61) and the data in Table 2, the decrease ratios of the RMSE coefficient of the two prediction methods were calculated. The improving effect of the suggested prediction method on prediction accuracy is more noticeable the higher the value.

$$RMSE_{reduction} = \frac{RMSE_{NI-LSTM} - RMSE_{NETIIP}}{RMSE_{NETIIP}} \times 100\% \quad (61)$$

The decrease ratios of the RMSE coefficients of all the prediction state variables of each mode at the three speeds in Table 3 were averaged, as shown in Table 4, which represents the degree of improvement of the proposed prediction methods at different speeds.

Table 3. Decrease ratio of RMSE coefficient of each mode (%).

	North	East	Pitch	Roll	Head
MRM_Slow	81.28	86.50	76.65	84.45	71.38
MRM_Medium	86.10	87.48	57.70	71.58	74.47
MRM_Fast	87.38	94.40	44.38	75.50	88.21
ANM_Slow	89.97	96.25	88.63	67.89	87.53
ANM_Medium	74.80	87.71	71.60	49.28	59.15
ANM_Fast	88.33	88.60	76.30	68.36	77.12

Table 4. Average decrease ratio of RMSE coefficient (%).

Speed Mode	Slow	Medium	Fast	Average
Ratio (%)	83.25	71.99	78.858	78.03

The maximum ship velocities in the CKF estimator under each mode and those in the training set were obtained in accordance with Figure 11 in order to more accurately reflect the relationship between the degree of prediction accuracy improvement of NETIIP in order to more accurately reflect the relationship between the degree of prediction accuracy improvement and the change in ship speed, as shown in Table 5.

Table 5. Maximum velocities for each mode (m/s).

MRM			ANM			Training Set
Slow	Medium	Fast	Slow	Medium	Fast	—
1.9383	3.9522	4.3282	1.7456	3.4997	4.8914	3.2647

The accuracy of the prediction results of NETIIP is not significantly higher than the traditional NI-LSTM prediction method when the ship speed in the test set approaches

that in the training set, as can be seen by comparing Tables 4 and 5 and combining with the results of the prediction error distribution in Figures 14 and 15. This is because the prediction accuracy of the traditional method is sufficient. However, the conventional NI-LSTM prediction technique has a low learning rate for the state change rule, and even fails to learn the proper rule, when the test set's ship speed significantly differs from the training set's. Therefore, the prediction error increases with the increase in the speed difference. Compared with the traditional NI-LSTM method, the prediction accuracy of the NETIIP prediction method is less affected by the speed. Therefore, compared with the traditional NI-LSTM prediction method, the improvement rate of the prediction accuracy increases with the increase in the speed difference.

It is vital to determine whether the time consumption of this technique is less than the need in order to confirm that the real-time performance of the suggested method satisfies the actual engineering requirements. According to Figure 1, the time of each cycle of the algorithm should not be longer than the sampling period of the sensor GPS, which is set to 0.1 s in this experiment. The NETIIP and NI-LSTM prediction methods are performed using software named MATLAB on a Windows 10 system with an Intel® Core(TM) i5-3210m 2.50 Ghz processor. The average time consumption of the methods is calculated as shown in Table 4.

Because NETIIP uses a single LSTM network structure, which is the same as that of the NI-LSTM prediction technique, the algorithm time is comparable to that of NI-LSTM, and the average time is considerably less than the upper limit of the forecast time 0.1 s (see Table 6). Therefore, the real-time performance of the proposed prediction method meets the engineering needs.

Table 6. Average time consumption.

Predictor	MRM			ANM		
	Slow	Medium	Fast	Slow	Medium	Fast
NETIIP	0.02297	0.02318	0.0218	0.0244	0.0215	0.0234
NI-LSTM	0.0223	0.0225	0.0229	0.02338	0.0227	0.0227

5. Conclusions

This paper studied a non-equal time interval incremental prediction method for ship motion state to solve the problem of ship state estimation at different rates and sensor sampling times in intelligent ship navigation. First, as the method's time state estimation modules, the estimation results of the CKF and UKF were first compared and studied, and the justifications for selecting the CKF estimator were given. Then, the prediction results of the NETIIP method and the traditional NI-LSTM prediction method were compared, and the impact of a change in ship velocity on the two prediction methods' accuracy was examined. The comparative findings demonstrate that the suggested approach has good prediction accuracy for ship condition prediction under various modes and speeds when compared with the conventional NI-LSTM prediction methods. Meanwhile, the algorithm time is almost the same as that of the traditional prediction methods, and both can satisfy the requirements of actual engineering.

Nevertheless, the following are some drawbacks of the suggested approach in this paper: (a) In order to guarantee the algorithm's prediction accuracy, the CKF estimation algorithm's state estimation accuracy must first be verified. The CKF algorithm needs to adjust various sets of parameters for different speeds of the ships, which restricts how simple the prediction algorithm can be. (b) Different from the traditional NI-LSTM prediction method, the NETIIP algorithm needs to use time intervals as inputs for training and prediction, necessitating high-precision time stamps in the training set and test set. Inaccurate time stamps will lower the algorithm's forecast accuracy.

As a result, future studies should take into account a more straightforward and efficient state estimation method as the prediction method's equal time interval state estimation

module. Furthermore, in order to lessen the effects of inadequate timestamp accuracy, a novel predictive compensating approach must be created.

Author Contributions: Conceptualization, H.X. and H.F.; Methodology, Z.Z.; Software, Z.Z. and W.L.; Validation, Z.Z. and W.L.; Data curation, Z.Z.; Writing—original draft preparation, Z.Z.; Writing—review and editing, Z.Z. and H.F.; Supervision, H.X.; Project administration, H.F. and H.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Nature Science Foundations of China, grant numbers 51879210 and 51979210.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank the constructive suggestions from reviewers and the associate editor. Part of the experimental verification work was completed with the help of other members of the Key Laboratory of High-Performance Ship Technology (Wuhan University of Technology). The authors would like to thank CHC NAVIGATION COMPANY for providing us with the experimental ship platform and manual and automatic navigation control system, which provided a useful platform for the experimental verification of the algorithm.

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

References

1. Osaloni, O.S.; Ayeni, V.O. The Development of Maritime Autonomous Surface Ships: Regulatory Challenges and the Way Forward. *Beijing Law Rev.* **2022**, *13*, 544. [\[CrossRef\]](#)
2. Hannaford, E.; Hassel, E.V. Risks and Benefits of Crew Reduction and/or Removal with Increased Automation on the Ship Operator: A Licensed Deck Officer's Perspective. *Appl. Sci.* **2021**, *11*, 3569. [\[CrossRef\]](#)
3. Liu, C.G.; Chu, X.M.; Wu, W.X.; Li, S.L.; He, Z.B.; Zheng, M.; Zhou, H.M.; Li, Z.X. Human-machine cooperation research for navigation of maritime autonomous surface ships: A review and consideration. *Ocean Eng.* **2022**, *246*, 110555. [\[CrossRef\]](#)
4. Rigatos, G.G. Sensor fusion-based dynamic positioning of ships using Extended Kalman and Particle Filtering. *Robotica* **2013**, *31*, 389–403. [\[CrossRef\]](#)
5. Liu, X.; Xu, X.; Liu, Y.; Wang, L. A fast and high-accuracy transfer alignment method between M/S INS for ship based on iterative calculation. *Measurement* **2014**, *51*, 297–309. [\[CrossRef\]](#)
6. Cao, Y.S.; Sun, Q. Application research of state estimation algorithm for dynamic positioning ship on fusion of unscented Kalman filtering and particle filtering. *Ship Sci. Technol.* **2017**, *39*, 49–53.
7. Caballero-Águila, R.; Hermoso-Carazo, A.; Linares-Pérez, J. Centralized, distributed and sequential fusion estimation from uncertain outputs with correlation between sensor noises and signal. *Int. J. Gen. Syst.* **2019**, *48*, 713–737. [\[CrossRef\]](#)
8. Caballero-Águila, R.; Garcia-Garrido, I.; Linares-Perez, J. Distributed fusion filtering for multi-sensor systems with correlated random transition and measurement matrices. *Int. J. Comput. Math.* **2020**, *97*, 263–274. [\[CrossRef\]](#)
9. Caballero-Águila, R.; Hu, J.; Linares-Pérez, J. Two Compensation Strategies for Optimal Estimation in Sensor Networks with Random Matrices, Time-Correlated Noises, Deception Attacks and Packet Losses. *Sensors* **2022**, *22*, 8505. [\[CrossRef\]](#)
10. Zhang, C.; Cao, C.; Li, T.; Guo, C. An improved RISS-GPS ship navigation approach via azimuth updates and magnetometer-calibration technology. *Measurement* **2021**, *175*, 109101. [\[CrossRef\]](#)
11. Liu, Y.Q. Research on the Multi-sensor Asynchronous Fusion Methods of Dynamic Positioning System. Master's Thesis, Harbin Engineering University, Harbin, China, 2017.
12. Li, G.C.; Yi, W.; Li, S.Q.; Wang, B.L.; Kong, L.J. Asynchronous multi-rate multi-sensor fusion based on random finite set. *Signal Process.* **2019**, *160*, 113–126. [\[CrossRef\]](#)
13. Zhang, K.; Wang, Z.; Guo, L.; Peng, Y.; Zheng, Z. An asynchronous data fusion algorithm for target detection based on multi-sensor networks. *IEEE Access* **2020**, *8*, 59511–59523. [\[CrossRef\]](#)
14. Liu, L.; Lin, X.; Chen, X. An improved asynchronous fusion algorithm for multi-sensor integrated navigation system. *Chin. Space Sci. Technol.* **2021**, *41*, 77.
15. Yu, B.; Li, T.; Ge, S.; Gu, H. Robust CPHD Fusion for Distributed Multitarget Tracking Using Asynchronous Sensors. *IEEE Sens. J.* **2021**, *22*, 1030–1040. [\[CrossRef\]](#)
16. Li, G.C.; Battistelli, G.; Chisci, L.; Kong, L. Distributed multi-target tracking over an asynchronous multi-sensor network. In Proceedings of the IEEE Radar Conference (RadarConf20), Florence, Italy, 21 September 2020.
17. Zhang, D.; Duan, Z.; Hanebeck, U.D. Asynchronous Multi-Radar Tracking Fusion with Converted Measurements. In Proceedings of the 25th International Conference on Information Fusion, Linköping, Sweden, 4 July 2022.

18. Yi, X.; Han, J.Y.; Guan, X. An asynchronous track-to-track association algorithm without time alignment. *Procedia Eng.* **2015**, *99*, 1120–1125.
19. Giorgi, G.; Davidson, J.; Habib, G.; Bracco, G.; Mattiazzo, G.; Kalmár-Nagy, T. Nonlinear dynamic and kinematic model of a Spar-Buoy: Parametric resonance and yaw numerical instability. *J. Mar. Sci. Eng.* **2020**, *8*, 504. [[CrossRef](#)]
20. Song, C.; Zhang, X.; Zhang, G. Attitude prediction of ship coupled heave–pitch motions using nonlinear innovation via full-scale test data. *Ocean Eng.* **2022**, *264*, 112524. [[CrossRef](#)]
21. Cahyadi, M.N.; Asfihani, T.; Mardiyanto, R.; Erfianti, R. Performance of GPS and IMU sensor fusion using unscented Kalman filter for precise i-Boat navigation in infinite wide waters. *Geod. Geodyn.* **2022**, *95*, 1–10. [[CrossRef](#)]
22. Guo, P.; Li, J.; Chen, T.; Wu, Z. Heave Motion Estimation Based on Cubature Kalman Filter. In Proceedings of the International Conference on Cyber-Physical Social Intelligence (ICCSI), Beijing, China, 18 December 2021.
23. Chen, Z.; Bai, M.; Lei, J.; Huang, Y.; Wang, J.; Xia, X. Comparison of UKF and EKF Filter Algorithm in INS/BDS Tightly Mode. In Proceedings of the 30th Chinese Control and Decision Conference, Shenyang, China, 9 June 2018.
24. Sun, F.; Tang, L.J. Estimation precision comparison of Cubature Kalman filter and Unscented Kalman filter. *Control Decis.* **2013**, *28*, 303–308, 312.
25. Zhang, Z.; Ni, G.; Xu, Y. Ship trajectory prediction based on LSTM neural network. In Proceedings of the IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 12 June 2020.
26. Zhang, T.; Zheng, X.Q.; Liu, M.X. Multiscale attention-based LSTM for ship motion prediction. *Ocean Eng.* **2021**, *230*, 109066. [[CrossRef](#)]
27. Peng, X.; Zhang, B. Ship motion attitude prediction based on EMD-PSO-LSTM integrated model. *J. Chin. Inert. Technol.* **2019**, *27*, 421–426.

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.