Ocean Engineering 311 (2024) 118880

Contents lists available at ScienceDirect

Ocean Engineering

journal homepage: www.elsevier.com/locate/oceaneng

Research paper

Advancing ship trajectory prediction: Integrating deep learning with enhanced reference trajectory correction techniques

Xueyin Li [a], Chunshan Liu [a,*], Jianghui Li [b], Lou Zhao [a], Zhongping Du [c]

[a] School of Communication Engineering, Hangzhou Dianzi University, Hangzhou, Zhejiang, China
[b] College of Ocean and Earth Sciences, Xiamen University, Xiamen, Fujian, China
[c] Loongship Technology, Beijing, China

ARTICLE INFO

Keywords:
Ship trajectory prediction
Deep learning
AIS

ABSTRACT

Ship trajectory prediction is crucial for maritime trade and navigational safety. In this paper, we present a deep learning (DL) based trajectory prediction framework that can exploit the navigation pattern of a reference trajectory, a historical trajectory that resembles the target one, to enhance the prediction accuracy. A Differential Long Short-Term Memory (DLSTM) model is first proposed for trajectory prediction, which takes solely the past motion characteristics of the target trajectory as the input. Building on DLSTM, an enhanced DLSTM with reference trajectory correction (Ref-DLSTM) is proposed to integrate the features of both the target and the reference trajectory for better prediction accuracy. The DLSTM can be applied when a reference trajectory is not available, while the Ref-DLSTM is applied when a reference trajectory is present. To reduce the complexity of reference trajectory identification, a grid-based search algorithm is proposed to restrain the search in a local area. The efficacy of the proposed framework is evaluated using AIS datasets from the US Coast Guard and the Danish Maritime Authority. Numerical results demonstrate notable improvement over the state-of-the-art trajectory prediction methodologies, showcasing reductions in geographical prediction errors by 19.1% and 33.0% for DLSTM and 34.0% and 35.8% for Ref-DLSTM, respectively.

1. Introduction

The maritime industry holds a pivotal role in global trade facilitation, providing impetus to maritime transportation and fostering economic exchanges among nations through the movement of goods and passengers (Yu et al., 2021; Xiao et al., 2019). Large-trading vessels have propelled the growth of maritime trade while concurrently introducing complexities to the maritime traffic environment. In economic and trade globalization, there arises a heightened demand for improved efficiency and safety within maritime transportation. Given the multifaceted nature of maritime traffic environments, ship navigation is influenced by various factors such as weather conditions, sea states, and vessel presence (Álvarez et al., 2021; Kanazawa et al., 2022). Consequently, ship trajectory prediction emerges as a critical domain for ensuring maritime traffic safety and optimizing shipping operations (Pan et al., 2021; Zhang et al., 2022). For instance, ship trajectory prediction can facilitate early detection of potential safety hazards and mitigate ship collisions (Fang et al., 2018; Zhao et al., 2018), thus contributing to the overall safety and efficiency of maritime transportation. Additionally, accurate trajectory prediction plays a crucial role in reducing vessel energy consumption (Chen et al., 2024b) and improving the efficiency of port operations (Chen et al., 2024a).

In recent years, the Automatic Identification System (AIS) has emerged as a critical asset in ensuring maritime transportation safety, presenting novel opportunities for leveraging big data analytics within the maritime domain Emmens et al. (2021). The widespread adoption of AIS has revolutionized the acquisition of vessel data through communication technology. AIS captures a plethora of vital information, including voyage information, and static and dynamic data of vessels. Voyage information encompasses essential parameters such as Maritime Mobile Service Identity (MMSI), destination, and voyage duration. AIS static data comprises crucial vessel attributes, such as dimensions, draft, and vessel type. AIS dynamic data includes dynamic vessel parameters, such as position, heading, and speed. These datasets offer a rich source of information for research endeavors in ship trajectory prediction. However, the vast amount of data presents formidable challenges for conventional prediction methodologies. The emergence of deep learning (DL) has significantly addressed these challenges. DL

* Corresponding author.
E-mail addresses: 221080060@hdu.edu.cn (X. Li), chunshan.liu@hdu.edu.cn (C. Liu), jli@xmu.edu.cn (J. Li), lou.zhao@hdu.edu.cn (L. Zhao), duzp@loongship.com (Z. Du).

https://doi.org/10.1016/j.oceaneng.2024.118880
Received 11 April 2024; Received in revised form 23 July 2024; Accepted 1 August 2024
Available online 12 August 2024

techniques not only unravel intricate spatiotemporal relationships but also furnish more precise and practical solutions for vessel trajectory prediction (Feng et al., 2022). Consequently, ship trajectory prediction methodologies grounded in DL frameworks have steadily acquired prominence in recent years, becoming the predominant approach in this area.

Most existing DL-based ship trajectory prediction techniques can be categorized based on the inclusion of data beyond the historical information of vessels to be predicted. The first category (Tang et al., 2022; Zhang et al., 2021; Xiao et al., 2022; Forti et al., 2020; Guo et al., 2023; Nguyen and Fablet, 2021; Wu et al., 2023) focuses on the internal factors (the historical information of the target vessel to be predicted), predicting solely by analyzing the historical movement patterns of the vessel. These approaches, however, are limited by the singular data source, which may lead to lower prediction accuracy and an overlook of detailed motion traits. Specifically, when trajectories exhibit sharp turns or when vessel behavior deviates from navigational inertia, predicting future trajectories becomes significantly more challenging.

To address this issue, the second category (Huang et al., 2022; Zhang et al., 2023; Mehri et al., 2021; Murray and Perera, 2021, 2020; Gao et al., 2021) attempts to explore extra trajectory information by considering external factors like environmental conditions and historical trajectories of the vessels other than the *target one*, i.e., the one to be predicted. Although existing efforts of this kind have somewhat alleviated the aforementioned problem, the information they provide tends to be vague and lacks specificity. Therefore, there is still room for improvement in prediction accuracy.

This paper presents a new solution to vessel trajectory prediction based on DL. The core idea is to integrate the features from the target trajectory, extracted before the time of prediction, and the features from a *reference trajectory*, a historical trajectory that exhibits similar behaviors to the target one, for more accurate trajectory predictions.

To account for the possibility that a reference trajectory is not always available because of the lack of historical data, our proposed framework introduces two carefully tailored DL models. The first one is designed to operate solely based on the features extracted from the target trajectory, aka, the *internal* features, where a differential Long Short Term Memory (DLSTM) prediction model is developed to learn both the instantaneous and average properties of speed and course to improve the continuity of the predicted vessel motion.

Building on DLSTM, an enhanced DLSTM with reference trajectory correction (Ref-DLSTM) is developed to integrate the internal features with the external features obtained from a reference trajectory for better prediction accuracies. To streamline the identification of suitable reference trajectories, a Grid-based Reference trajectory Identification (GRI) algorithm is introduced. The proposed GRI partitions maritime areas into uniformly sized grids, assigning each target trajectory to a specific grid for a local reference search. This localized approach avoids the need for an exhaustive global search, allowing for the efficient identification of the most analogous reference trajectory within the same grid as the target trajectory. As a result, GRI not only improves the efficiency of finding highly similar reference trajectories in adjacent maritime zones, but also bolsters computational efficiency and stability, significantly reducing the overall complexity of the prediction process. The effectiveness of the overall trajectory prediction framework is evaluated using real AIS datasets.

The remainder of the paper is organized as follows. Section 2 presents a brief review of existing vessel trajectory prediction approaches. Section 3 explains the proposed methodology. Section 4 presents the numerical results obtained based on real AIS datasets, while Section 5 draws conclusions and discusses future research directions.

2. Related work

Ship trajectory prediction has attracted significant interest from both academia and industry. Prior research has focused on improving the precision and reliability of ship trajectory forecasts. As highlighted earlier, DL-based models have played a predominant role in advancing this field. Consequently, the models examined in this section are exclusively founded on the DL. This discussion focuses on two primary classes: models that leverage internal data, specifically the historical trajectory of the vessel under prediction, and models that integrate external factors, including environmental conditions and the historical trajectories of other maritime traffic.

2.1. Prediction models based on internal information

The development of trajectory prediction algorithms based on DL has advanced rapidly, with key models including LSTM (Tang et al., 2022; Zhang et al., 2021; Xiao et al., 2022; Forti et al., 2020; Wu et al., 2023), GAN (Generative Adversarial Networks) (Guo et al., 2023), and transformer (Nguyen and Fablet, 2021). Among these, LSTM and its variants have emerged as particularly influential in shaping the landscape of ship trajectory prediction.

The capacity of LSTM for integrating sequence prediction techniques has been explored by Tang et al. (2022), who put force into a methodology tailored for ship trajectory forecasting. Building upon LSTM, the Bidirectional LSTM (BiLSTM) variant has been adopted to enhance the prediction accuracies (Zhang et al., 2021). This approach is further refined by the integration of segments of past and future trajectory data as the input feature, enabling a more effective capture of the contextual information within the trajectory data (Xiao et al., 2022).

In tackling more intricate maritime scenarios, the Sequence to Sequence (Seq2Seq) model, composed of an encoder and a decoder, has been deployed. Forti et al. (2020) leveraged a Seq2Seq architecture based on LSTM to capture long-term dependencies within AIS data sequences, thereby amplifying predictive accuracy. Furthermore, the ConvLSTM-based Seq2Seq model has been proposed to extract spatial and temporal characteristics embedded in ship trajectories concurrently (Wu et al., 2023).

In parallel, innovative approaches such as GAN have been explored by Guo et al. (2023) to discern ship movement patterns and establish a bidirectional mapping between modal latent vectors and anticipated trajectories. Additionally, Nguyen and Fablet (2021) introduced a novel representation of AIS data, coupled with a tailored loss function, to mitigate challenges associated with data heterogeneity and multimodality.

Despite these advancements, the focus has predominantly been on internal dynamics, relying on historical movement patterns for predictions and thus limiting the breadth of input information. This has prompted a shift towards integrating external factors into predictive models, broadening the scope and enhancing the robustness of trajectory predictions.

2.2. Prediction models incorporating external features

This subsection explores advanced ship trajectory prediction models that use both interval features and external features. Huang et al. (2022) developed an environment-aware ship trajectory prediction model that employs a Convolutional Neural Network (CNN) to extract navigational intention from ship density maps and incorporates this information into the prediction model. This enables the model to capture navigational intent from ship density maps, significantly enhancing both short-term and long-term prediction accuracy. A seq2seq-based prediction model is proposed by Zhang et al. (2023) that encodes position, course, speed, and sailing distance separately. Additionally, it inputs a combined vector of vessel type and departure time into a

type-oriented decoder to achieve accurate predictions. In Mehri et al. (2021), a hybrid approach combining curve model (Best and Norton, 1997) and deep neural networks is introduced, and this method uses coastline data to verify the logical consistency of compressed trajectory methods, then constructs a context-aware LSTM network based on ship type and other contextual variables.

Furthermore, leveraging the historical trajectories of other vessels has proven beneficial for enhancing predictive accuracy (Murray and Perera, 2021; Xu et al., 2022; Murray and Perera, 2020; Gao et al., 2021). Common strategies involve clustering historical data and tailoring specific models for each cluster. For instance, Murray and Perera (2021) employed a variational recurrent auto-encoder alongside hierarchical clustering for trajectory segmentation, while Xu et al. (2022) utilized DBSCAN for clustering and directed Hausdorff distance for trajectory classification based on similarity. Additionally, the use of Gaussian mixture models and K-Nearest Neighbor (KNN) algorithms for trajectory clustering and classification was observed in Murray and Perera (2020).

As another relevant work, Gao et al. (2021) proposed an innovative method for leveraging historical trajectories by identifying a reference trajectory for each target trajectory and segmenting the prediction process into two distinct phases: support points and destination points. The initial phase employs dual LSTM models for differential prediction of support points, whereas the subsequent phase utilizes points from the reference trajectory to predict the destination points, thus introducing a structured approach to trajectory forecasting.

These methodologies collectively aim to dissect the intricate dynamics governing ship movements to forecast future trajectories with higher precision. Our research builds upon the concept of employing reference trajectories but distinguishes itself by implementing a differential-based LSTM framework as the core predictive mechanism. This approach not only forecasts ship information across identical time intervals, but also incorporates the corresponding reference trajectory for each prediction, facilitating substantial corrections and improvements in the prediction of future trajectories.

3. Methodology

In this section, we describe our proposed methodology for vessel trajectory prediction. Fig. 1 presents a flowchart of the proposed framework, which has three key components: the DLSTM model, the Ref-DLSTM model, and the Grid-based Reference Identification (GRI) module. Both the DLSTM and Ref-DLSTM models are trained using historical AIS data that has undergone preprocessing steps such as denoising and interpolation. In the real-time prediction phase, the proposed system initially employs the GRI module to determine whether there exists a historical reference trajectory that satisfies similarity check. If such a reference trajectory is found, the system invokes the Ref-DLSTM model, leveraging the corrective power of the reference trajectory for trajectory predictions. Otherwise, the system resorts to the DLSTM model, relying solely on the past trajectory information of the target vessel to predict its future trajectory.

The proposed framework utilizes a recursive forecasting approach, repeatedly invoking the DLSTM or Ref-DLSTM model to accomplish predictions over multiple time steps. Without loss of generality, this section considers predicting the vessel positions at time instances $\{t+\tau : \tau = 1, 2, 3, \dots, n\}$, where t is the time instance that the prediction is made and n is the prediction horizon. In the remaining part of this section, we explain the DLSTM model, the Ref-DLSTM model, and the GRI module.

3.1. Differential LSTM (DLSTM)

Denote a trajectory point at time instance t as

$$x_t = \left(lon_t, lat_t, SOG_t, COG_t, \overline{SOG}_t, \overline{COG}_t\right), \tag{1}$$

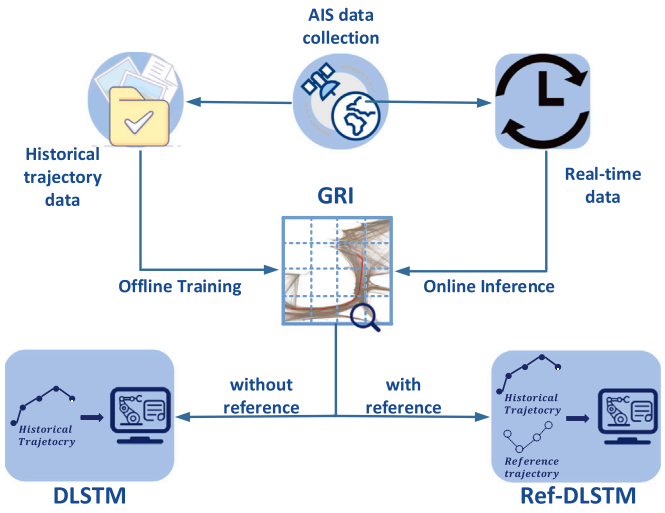

Fig. 1. A flowchart of the proposed trajectory prediction framework: (1) The Grid-based Reference Identification (GRI) module is used to examine the existence of a reference trajectory relative to the target vessel; (2) Ref-DLSTM is used to produce trajectory predictions if a reference trajectory exists; (3) DLSTM is used in the absence of a reference trajectory.

where lon_t, lat_t, SOG_t, COG_t represent the longitude, latitude, SOG, COG at time instance t, respectively, and $\overline{SOG}_t$ and $\overline{COG}_t$ are the average SOG and COG from time $t-1$ to t. Denote further

$$\tilde{x}_t = \left(\Delta lon_t, \Delta lat_t, SOG_t, \Delta COG_t, \overline{SOG}_t, \Delta\overline{COG}_t\right) \tag{2}$$

as the collections of time-differential features, where, e.g., Δlon_t and Δlat_t can be obtained as

$$\begin{cases} \Delta lon_t = lon_t - lon_{t-1}, \\ \Delta lat_t = lat_t - lat_{t-1}. \end{cases} \tag{3}$$

Other features can be obtained similarly.

To predict the vessel position for time instance $t+\tau$, the DLSTM model takes a series of historical trajectory points, stored in $\tilde{X}_{t+\tau-1}$, as the input feature, and outputs the predicted value $\hat{y}_{t+\tau}$. Denote the predicted output trajectory as

$$\hat{Y} = \{\hat{y}_{t+1}, \hat{y}_{t+2}, \dots, \hat{y}_{t+n}\} \tag{4}$$

and

$$Y = \{y_{t+1}, y_{t+2}, \dots, y_{t+n}\} \tag{5}$$

as the corresponding true data (the label data used during model training).

Fig. 2 illustrates the model architecture of the DLSTM. As shown in the figure, the model comprises three sub-networks: the position prediction network (*POS-Net*), the SOG prediction network (*SOG-Net*), and the COG prediction Network (*COG-Net*). The three sub-networks all take $\tilde{X}_{t+\tau-1}$ as the input feature, and predict the position information $pos_{t+\tau} = (\Delta lon_{t+\tau}, \Delta lat_{t+\tau})$, SOG information $sog_{t+\tau} = (SOG_{t+\tau}, \overline{SOG}_{t+\tau})$, COG information $cog_{t+\tau} = (\Delta COG_{t+\tau}, \Delta\overline{COG}_{t+\tau})$ of the vessel at the time $t+\tau$, respectively. The structures of the three sub-networks are identical, each employing LSTM as the core architecture. As illustrated in Fig. 2, the input sequence $\tilde{X}_{t+\tau-1}$, after feature extraction through an LSTM cell, is flattened and passed through a fully connected layer to produce the output.

In executing multi-step predictions, when $\tau = 1$, the input feature sequence is entirely extracted from the trajectory of the target vessel at and before time instance t, i.e., $\tilde{X}_{t+\tau-1} = \{\tilde{x}_{t-m}, \dots, \tilde{x}_{t-1}, \tilde{x}_t\}$, where m is the length of input feature; when $\tau > 1$, we adopt a recursive prediction method, using the predicted value $\hat{y}_{t+\tau-1}$ at time $t+\tau-1$ to update

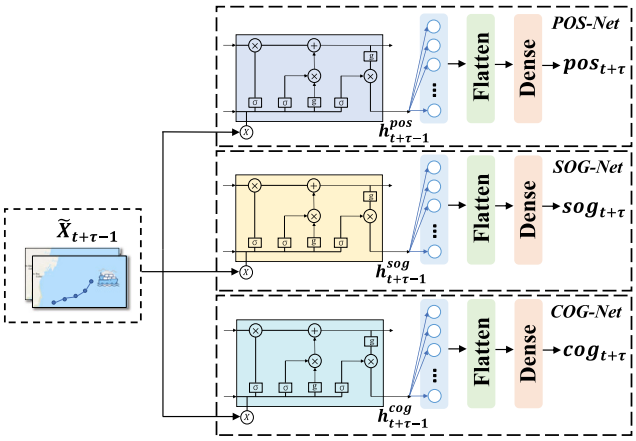

Fig. 2. Architecture of the DLSTM model, consisting of three sub-networks: POS-Net, SOG-Net, and COG-Net.

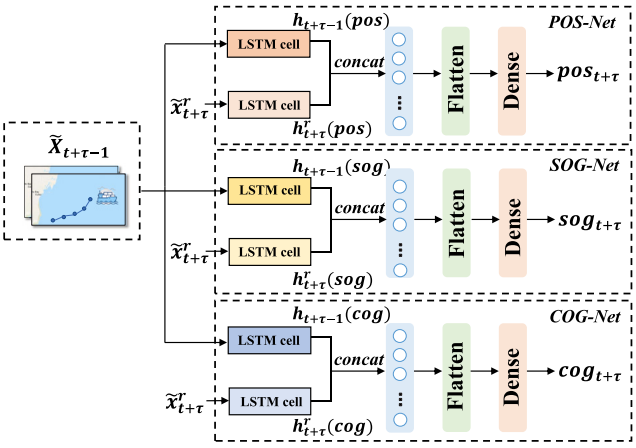

Fig. 3. Architecture of the Ref-DLSTM model, consisting of three sub-networks: POS-Net, SOG-Net, and COG-Net. Each network is composed of two parallel LSTMs, one receiving input $\tilde{X}_{t+\tau-1}$ and the other receiving input $\tilde{x}^r_{t+\tau}$.

the input feature sequence $\tilde{X}_{t+\tau-1}$. This process can be represented as follows:

$$\tilde{X}_{t+\tau-1} = \begin{cases} \{\tilde{x}_{t-m}, \ldots, \tilde{x}_{t-1}, \tilde{x}_t\}, & \tau = 1 \\ \{\tilde{x}_{t-m+(\tau-1)}, \ldots, \tilde{x}_t, \ldots, \hat{y}_{t+\tau-1}\}, & \tau > 1 \end{cases} \tag{6}$$

where

$$\hat{y}_{t+\tau} = concat\left(pos_{t+\tau}, sog_{t+\tau}, cog_{t+\tau}\right), \tag{7}$$

and

$$pos_{t+\tau} = g_{pos}\left(\tilde{X}_{t+\tau-1}\right), \tag{8}$$

$$sog_{t+\tau} = g_{sog}\left(\tilde{X}_{t+\tau-1}\right), \tag{9}$$

$$cog_{t+\tau} = g_{cog}\left(\tilde{X}_{t+\tau-1}\right). \tag{10}$$

Here, $g_{pos}(\cdot)$, $g_{sog}(\cdot)$ and $g_{cog}(\cdot)$ respectively denote the mapping relationships from $\tilde{X}_{t+\tau-1}$ to $pos_{t+\tau}$, $sog_{t+\tau}$, $cog_{t+\tau}$ for the POS-Net, SOG-Net, and COG-Net sub-networks. $concat(\cdot)$ denotes concatenation operations and hence $\hat{y}_{t+\tau}$ includes the concatenated vector of prediction results from the three sub-networks.

The function of the three sub-networks within the DLSTM model is to independently map the input features $\tilde{X}_{t+\tau-1}$ to the predicted values at time $t+\tau$. Thus, during the training of the DLSTM model, the three sub-networks adopt an independent single-step training method. That is, each network takes $\tilde{X}_{t+\tau-1}$ ($\tau = 1$) as input and separately learns the mapping relationships between $\tilde{X}_{t+\tau-1}$ and the vessel position ($\Delta lon_{t+\tau}, \Delta lat_{t+\tau}$), SOG ($SOG_{t+\tau}, \overline{SOG}_{t+\tau}$), and COG ($\Delta COG_{t+\tau}, \Delta\overline{COG}_{t+\tau}$) at time $t+\tau$, respectively. The training of the model adopts a supervised learning approach, with the loss function set to Mean Squared Error (MSE), which is formulated as:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(z_i - \hat{z}_i\right)^2, \tag{11}$$

where N is the number of samples, z_i and $\hat{z}_i$ denote the label and the predicted values of the ith sample. For POS-Net, $z = (\Delta lon, \Delta lat)$; SOG-Net, $z = (SOG, \overline{SOG})$; and for COG-Net, $z = (\Delta COG, \Delta\overline{COG})$. After the training, the parameters of the three networks are saved for use in real-time prediction tasks.

We pause to note that, for the input feature, we have chosen the change in position at equal time intervals ($\Delta lon, \Delta lat$) because this form of prediction can alleviate sudden jumps in position when making the predictions, hence making the predicted trajectory smoother. The change of COG can provide useful characteristics of the dynamic behavior of the vessel, which can be used as additional information to predict changes in vessel position. We opt to adopt the SOG directly as part of the input feature since there is a more direct relationship between the positional changes and the SOG. Additionally, to account for the changes in navigation states in the time interval that the input features were extracted, we incorporate both instantaneous and average attributes of SOG and COG.

3.2. Differential LSTM with reference trajectory correction (ref-DLSTM)

In this subsection, we explain the Ref-DLSTM model constructed based on DLSTM. As mentioned in Section 1, similar vessels navigating within the same maritime area often exhibit similar trajectory characteristics, thus the historical trajectories of vessels can serve as reference trajectories to provide additional useful information when making the predictions. Following this observation, Ref-DLSTM integrates the features of reference trajectories on the base of the DLSTM for more accurate trajectory predictions.

Fig. 3 illustrates the architecture of Ref-DLSTM. Similar to DLSTM, Ref-DLSTM also consists of three sub-networks: POS-Net, SOG-Net, and COG-Net. Their functions are identical to those in DLSTM. The difference lies in that each sub-network in Ref-DLSTM includes two parallel LSTM units at the input stage. The first unit takes $\tilde{X}_{t+\tau-1}$ as input, analyzing the historical trajectory characteristics of the target vessel, while the second unit takes $\tilde{x}^r_{t+\tau}$, extracted from the reference trajectory, as the input feature. The outputs of the two LSTM units are then concatenated and fed to the later parts of the model. Taking POS-Net as an example, the feature extracted by the first LSTM unit is denoted as $h_{t+\tau-1}(pos)$, and the one extracted from $\tilde{x}^r_{t+\tau}$ by the second LSTM unit is $h^r_{t+\tau}(pos)$. The Ref-DLSTM model concatenates $h_{t+\tau-1}(pos)$ and $h^r_{t+\tau}(pos)$, the result of which is flattened and fed to a fully connected layer to produce the prediction output. As previously mentioned, the approach to multi-step prediction in the Ref-DLSTM model is the same as that in DLSTM. Its training also employs single-step independent training for the three sub-networks, with the supervised learning loss function also being MSE.

Compared to DLSTM, the key additional information in Ref-DLSTM is $\tilde{x}^r_{t+\tau}$. We now explain the method of extracting $x^r_{t+\tau}$ from the reference trajectory. Suppose that the reference trajectory identified by the GRI module in Section 3.3 is denoted as H_{p^*} (refer to Section 3.3 for a detailed explanation of the method for reference trajectory identification), which also consists of a set of trajectory points as defined by (1). To obtain the reference feature $\tilde{x}^r_{t+\tau}$ from H_{p^*}, it is necessary to use the anchor point, which is the point on H_{p^*} that is closest to the current position of the target vessel:

$$x^r_t = argmin\{dis\left(u, x_t\right), u \in H_{p^*}\}, \tag{12}$$

where $dis(\cdot)$ represents the great-circle distance between two trajectory points on the Earth's surface. Based on the information of x_t^r, a set of interpolated points is obtained from trajectory H_{p^*}, with the interpolation time interval matching the time interval of each prediction step. After the interpolation, $n+1$ interpolated points, i.e., $r(t) = \{x_t^r, x_{t+1}^r, \ldots, x_{t+n}^r\}$, are used to calculate the differential information $\tilde{r}(t) = \{\tilde{x}_{t+1}^r, \tilde{x}_{t+2}^r, \ldots, \tilde{x}_{t+n}^r\}$, following similar operations as (3). We note that $\tilde{r}(t)$ only needs to be computed once for each prediction task, i.e., prediction n steps at time instance t. When making predictions for different values of $\tau(\tau = 1, \ldots, n)$, the reference feature $\tilde{x}_{t+\tau}^r$ is taken directly from $\tilde{r}(t)$.

3.3. Grid-based reference trajectory identification (GRI)

In this subsection, we explain identifying H_{p^*}, based on our proposed GRI algorithm. Denote the trajectory of the target vessel up to time instance t as X_t. The most direct method to find the reference trajectory is to compare X_t with each trajectory stored in the historical trajectory database, using similarity measures such as Dynamic Time Warping (DTW) (Li et al., 2017; Yuan et al., 2017; Li et al., 2020; Zhao and Shi, 2019) or Hausdorff Distance (HD) (Wang et al., 2021) to calculate the degree of match. However, this approach encounters two issues: 1) the trajectories in the database may significantly differ in length from the target trajectory X_t. This mismatch may lead to unstable similarity measures that do not accurately reflect the real similarity between trajectories; 2) there may exist many historical trajectories. An exhaustive search for matching may incur a significant computational burden.

To address these issues, the GRI algorithm is proposed, which leverages grid-based attribution information and the corresponding navigational statistical data of the historical trajectories in order to rapidly filter and reduce the number of candidate trajectories to be compared. The key steps of the proposed GRI algorithm are listed as follows.

Step-1: Maritime area gridification.

Define the maritime area of interest using the maximum and minimum longitude (LON_{max}, LON_{min}) and latitude (LAT_{max}, LAT_{min}), and set the grid dimensions to α and β for longitude and latitude, respectively. Suppose that the entire maritime area is divided into I grids, each with its unique identification number, and the area of the ith grid is denoted as $\mathcal{D}_i$.

Step-2: Grid attribution collection of historical trajectories.

We first denote a historical trajectory $H_p \in \mathcal{H}^0$, where $\mathcal{H}^0$ is the entire set of historical trajectories. The grid association indications of H_p are represented by a vector $\mathbf{a}_p \in \{0,1\}^{I\times 1}$, where $a_p(i) = 1$ indicates that the trajectory H_p has an association relationship with grid $\mathcal{D}_i$, and $a_p(i) = 0$ indicates there is no association. The association between trajectory H_p and grid $\mathcal{D}_i$ is determined by the number of points that trajectory H_p falls within $\mathcal{D}_i$:

$$a_p(i) = \begin{cases} 1, \left|H_p \in \mathcal{D}_i\right| \geq \gamma_h, \\ 0, \left|H_p \in \mathcal{D}_i\right| < \gamma_h, \end{cases} \tag{13}$$

where γ_h is the minimum number of trajectory points required to have an association and $\left|H_p \in \mathcal{D}_i\right|$ denotes the number of trajectory points of H_p that fall within the ith grid.

In addition to the associated vector, the algorithm also records the SOG and COG statistics for each trajectory associated with a grid. Specifically, it calculates the SOG vector, $\mathbf{v}_p = \{0, SOG_{ave}^i\}^{I\times 1}$, and the COG vector, $\mathbf{u}_p = \{0, COG_{ave}^i\}^{I\times 1}$, where SOG_{ave}^i and COG_{ave}^i denote the average SOG and COG, respectively, for the trajectory segment $h_i \in H_p$ within the ith grid. These vectors are populated with 0 for grids without a trajectory association.

The above process in this step is executed for all the historical trajectories $H_p \in \mathcal{H}^0$.

Step-3: Grid-based metadata organization.

Upon obtaining the association indicators and the corresponding statistics for all the trajectories, the collected information, including $\mathbf{a}_p$, $\mathbf{v}_p$ and $\mathbf{u}_p$, is gathered and re-arranged on a per-grid basis. Specifically, for the ith grid, the re-arranged information stored for grid i is a set of tuples, each of which contains the corresponding trajectory index, the SOG and COG statistics:

$$\mathcal{L}_i = \{\left(p, v_p(i), u_p(i)\right) \| a_p(i) = 1\}, \tag{14}$$

where p is the trajectory index. The collections of $\mathcal{L}_i$ for the I grids are stored as the metadata to accelerate the trajectory-matching process.

Step-4: Reference trajectory matching.

We now explain the method of leveraging the grid-based metadata for reference trajectory matching, which contains the following 4 sub-steps.

(1) Identify the grid association of the target trajectory. Suppose the length of the target trajectory segment X_t is L. X_t is deemed associated with grid i if the number of points that X_t falls in $\mathcal{D}_i$ satisfies

$$|X_t \in \mathcal{D}_i| > \varepsilon L, \tag{15}$$

where $\varepsilon \in (0,1)$ is a threshold. The metadata of the grids satisfying condition (15) is then unioned to form a candidate metadata set $\mathcal{L}^c$ to identify the candidate trajectory set:

$$\mathcal{L}^c = \cup_{i\in\mathcal{I}^c}\mathcal{L}_i, \tag{16}$$

where $\mathcal{I}^c = \{i \| X_t \in \mathcal{D}_i| > \varepsilon L\}$ denotes the set of grids that condition (15) holds for the target trajectory.

(2) Filter the candidate trajectory set based on the SOG and COG meta-information. Denote $\mathcal{H}^1$ as the set of candidate trajectories contained in the meta dataset $\mathcal{L}^c$. The current step is to filter $\mathcal{H}^1$ based on the average SOG sog_{ave} and average COG cog_{ave} of the target trajectory segment X_t and the corresponding meta-information contained in $\mathcal{L}^c$. Specifically, the filtered candidate trajectory set $\mathcal{H}^2 \subseteq \mathcal{H}^1$ can be identified as:

$$\mathcal{H}^2 = \{p | (p, v, u) \in \mathcal{L}^c, (18) \text{ holds for } (u, v)\}, \tag{17}$$

where

$$\begin{cases} \left|sog_{ave} - v\right| < \gamma_{SOG}, \\ \left|cog_{ave} - u\right| < \gamma_{COG}, \end{cases} \tag{18}$$

γ_{SOG} and γ_{COG} represent the threshold values. With the above filtering, it is expected that $\left|\mathcal{H}^2\right| \ll \left|\mathcal{H}^1\right|$. Hence, the computational load incurred by pair-wised computation of trajectory similarity can be reduced.

(3) Trim the candidate trajectories and refine the candidate trajectory set based on a mapping distance. This step begins with identifying a rectangular region $\mathcal{D}^t$ according to the target trajectory segment X_t. Specifically, let $\mathcal{D}^t$ be the area enclosed by the maximum and minimum latitude and longitude values of X_t(i.e., lon_{min}, lon_{max}, lat_{min}, lat_{max}), as illustrated by the red dashed rectangular in Fig. 4. If a candidate trajectory $H_p \in \mathcal{H}^2$ has non-zero points in $\mathcal{D}^t$, i.e., $\left|H_p \in \mathcal{D}^t\right| > 0$, then H_p is trimmed to contain only the trajectory points within $\mathcal{D}^t$, as also illustrated in Fig. 4. Denote further the trimmed trajectory as $H_p^{'}$, and let δ_p and δ_t be the geo-distance between the first and the last trajectory points of $H_p^{'}$ and X_t. The projection of δ_p and δ_t onto latitude and longitude direction, i.e., δ_p^{lon}, δ_p^{lat}, δ_t^{lon}, δ_t^{lat}, are used to further refined the candidate set. Specifically, the candidate trajectory is retained in the refined candidate set $\mathcal{H}^3 \subseteq \mathcal{H}^2$ if $\left|\delta_p^{lon} - \delta_t^{lon}\right| / \delta_t^{lon} < \gamma_{dis}$ and $\left|\delta_p^{lat} - \delta_t^{lat}\right| / \delta_t^{lat} < \gamma_{dis}$. Trajectories that do not meet the above conditions are excluded from further comparison.

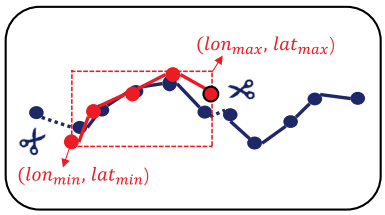

Fig. 4. An illustration of trajectory trimming in Step-4 3) of the GRI module.

(4) Calculate the trajectory similarities and identify the reference trajectory. Upon obtaining the refined candidate trajectory set $\mathcal{H}^3$, it is ready to evaluate the trajectory similarity between each trajectory $H_p \in \mathcal{H}^3$ and X_t to identify the reference trajectory. Specifically, the reference trajectory is the one that has the smallest Directed Hausdorff Distance (DHD) (Laxhammar, 2014):

$$p^* = \arg\min_{p\in\mathcal{H}^3} DHD\{\tilde{H}_p, X_t\}, \tag{19}$$

and the corresponding DHD must be below a pre-defined similarity threshold $DHD\{\tilde{H}_{p^*}, X_t\} < \gamma_{DHD}$. Here, $DHD\{\tilde{H}_{p^*}, X_t\}$ denotes the DHD between $\tilde{H}_{p^*}$ and X_t, and $\tilde{H}_{p^*}$ is an interpolated trajectory segment starting from the anchor point on H_{p^*} identified according to (12). It is noted that $\tilde{H}_{p^*}$ is interpolated with the same time granularity and has the same length as X_t. Additionally, it is possible that $DHD\{\tilde{H}_p, X_t\} > \gamma_{DHD}$. In this case, the GRI module will declare that there is no valid reference trajectory, hence the DLSTM model will be used to make trajectory predictions.

4. Numerical results

In this section, the performance of our proposed approach is evaluated using AIS data published by the US Coast Guard (the U.S. Coast Guard, 2024) and the Danish Maritime Authority (DMA) (DMA, 2024). The US AIS dataset is collected by the United States Coast Guard using ship-borne navigation safety devices. The collected AIS data allows for monitoring the positions and characteristics of vessels within US and international waters. The DMA AIS data is published by the Danish Maritime Authority, which is part of the Ministry of Industry, Business, and Financial Affairs in Denmark.

In the experiments, we examine cargo and tanker vessels navigating between longitudes $84°W$ and $72°W$, and latitudes $20°N$ to $42°N$ for the US data (Zone 17 and Zone 18), and cargo navigating between longitudes $4°E$ to $15°E$, and latitudes $53°N$ to $59°N$ for the DMA data. Our analysis spans datasets from Jan. 2013 to Oct. 2013 for the US and Jan. 2020 to June 2020 for Denmark as training data, with testing periods from Nov. 2013 to Dec. 2013 and July 2020 to Aug. 2020, respectively. Figs. 5 and 6 present the training and test datasets for the US and Denmark scenarios, respectively.

The raw AIS data is pre-processed before applying the proposed approach. Trajectories are extracted on a per-vessel basis and are then further processed by denoising and down-sampling. A description of pre-processing is provided in Appendix. When training and testing the prediction models, data samples are produced by extracting segments of trajectories from the training and the testing dataset. In our experiments, the length of the input feature is set to $m = 9$ and the prediction horizon is set to $n = 5$, corresponding to 2.5 h ahead trajectory predictions. For the US dataset, there are over 194,000 training samples and about 41,700 test samples, where the fraction of samples that a reference trajectory is identified is about 53.3%. For the DMA dataset, the numbers of training and test samples are about 31,000 and 9,000, where the fraction of samples that a reference trajectory can be identified is about 75.5%.

Table 1
The hyper-parameters of GRI.

	α	β	γ_h	δ	γ_{SOG}	γ_{COG}	γ_{dis}	γ_{DHD}
the US	1°	2°	4	0.3	7.5 knots	20°	0.1	5.8 km
DMA	1°	1°	4	0.3	5 knots	15°	0.1	8.5 km

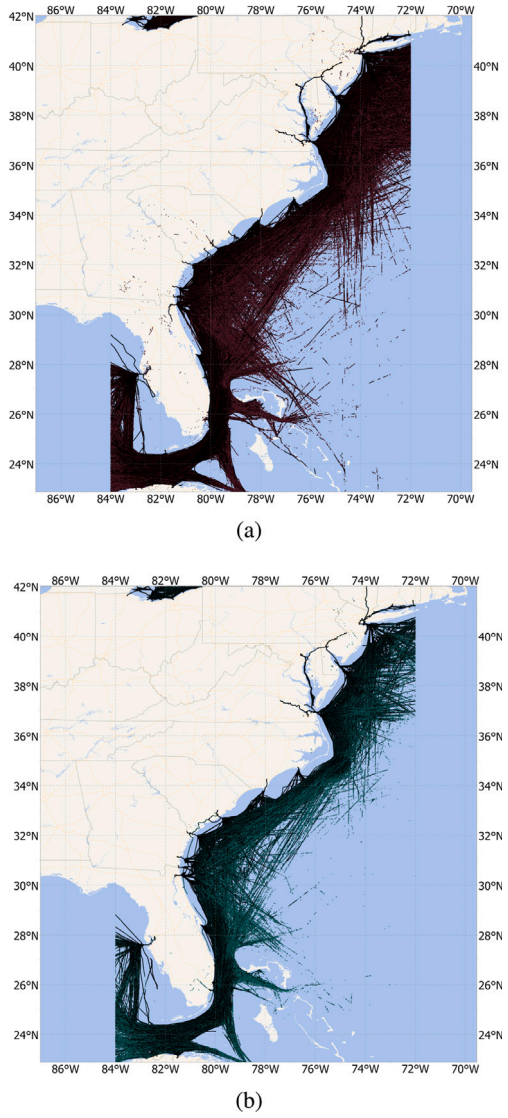

Fig. 5. An illustration of the raw AIS data from the US Guard: (a) Train dataset corresponding to the period between Jan. 2013 and Oct. 2013; (b) Test dataset obtained from Nov. 2013 to Dec 2013.

For the proposed DLSTM and Ref-DLSTM models, each LSTM is a two-layer LSTM cell, with the number of hidden units set to 256, and the dimensions of the fully connected layers are $(2560, 2048, 1024, 512, 2)$ for Ref-DLSTM and $(2304, 2048, 1024, 512, 2)$ for DLSTM. The parameters of the GRI algorithm are presented in Table 1. The models are implemented under a Python 3.11 environment with PyTorch, empowered by an Nvidia GeForce RTX 3060 GPU. Each subnetwork is trained with a batch size of 128, a learning rate of 0.0001, and for 100 epochs. The optimizer is set to Adam. It is noted that the training data is normalized with max–min normalization before feeding into the model. The same normalizer is adopted to normalize the test samples.

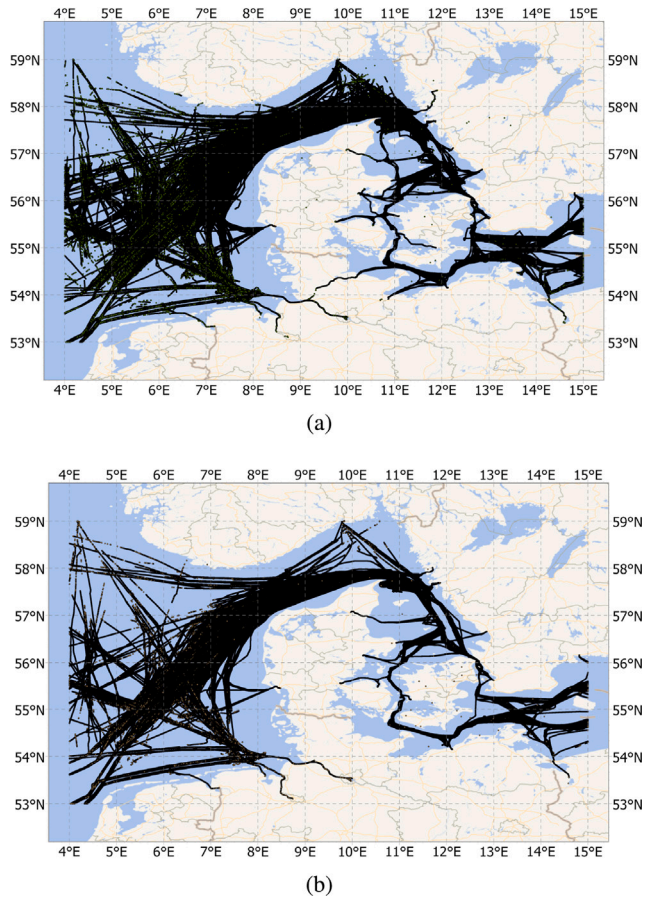

Fig. 6. An illustration of the raw AIS data from DMA: (a) Train dataset corresponding to the period between Jan. 2020 to June 2020; (b) Test dataset obtained from July 2020 to Aug. 2020.

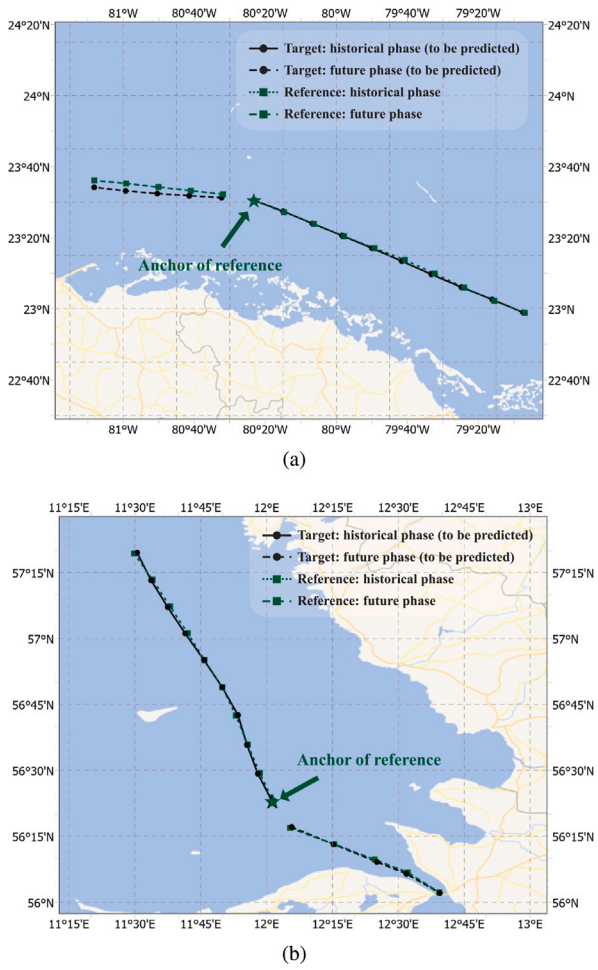

Fig. 7. Example 1 - GRI Results from (a) the US dataset and (b) the DMA dataset. This type is characterized by similarity in the direction and position of the target trajectory throughout the history and future phases.

4.1. Results for reference trajectory identification

To examine the efficacy of the GRI algorithm, we start by presenting three types of representative examples derived from the US and DMA datasets, as illustrated in Figs. 7–9. In these figures, each subplot presents both the historical (solid lines for the target and dotted lines for the reference) and future (dashed lines) trajectory segment relative to the current position of the target vessel.

The first type of reference trajectory, as depicted in Fig. 7, is characterized by its high fidelity towards the direction and position of the target vessel throughout both historical and future stages. This alignment provides crucial auxiliary data for trajectory prediction, enabling the Ref-DLSTM model to preserve the correct trajectory direction while preventing any unintended acceleration or deceleration in the predicted path. The precise positioning information of this type of reference trajectory ensures the accuracy of the predicted trajectory, particularly in maintaining the course and speed predictions for the target vessel.

Conversely, the second type of reference trajectory, shown in Fig. 8, maintains directional consistency with the target trajectory but displays variable positional discrepancies over time. Although this type may not be as effective in predicting future positions, its consistent directionality lends support to trajectory forecasts, especially in scenarios where the course of a vessel may abruptly change. The directional information from this type of reference trajectory helps in adjusting the prediction model to accommodate sudden deviations in the vessel path, underscoring its utility despite positional variances.

The third type of reference trajectory, illustrated in Fig. 9, exhibits a strong correlation with the target trajectory during the historical stage, aligning with the intended function of the GRI algorithm. However, it diverges significantly in the future stage, indicative of vessels entering a maritime zone with varying destinations, which leads to trajectory "divergence". Despite the high historical accuracy, this divergence cannot be rectified simply by tweaking the GRI parameters. In such instances, the prediction model must prioritize the historical data of the target trajectory, thereby diminishing the impact of this reference trajectory type on future trajectory predictions.

Table 2
The average number of candidate trajectories remaining at each sub-step in step-4 of the GRI.

	$\mathcal{H}^0$	$\mathcal{H}^1$	$\mathcal{H}^2$	$\mathcal{H}^3$
the US	9606	3433.42	826.98	37.46
Denmark	3254	1957.93	478.63	154.01

Besides the above examples, we further present results on the average number of trajectories within the candidate sets across different GRI steps to demonstrate the efficacy of the GRI algorithm in reducing the computational complexity incurred by trajectory similarity computation. As shown in Table 2, the sequential filtering steps of the GRI algorithm successively decrease the size of the candidate set, thereby efficiently reducing the computational load while still creating high-quality reference trajectories for analysis. Specifically, in the US dataset, the final candidate trajectory set comprised merely 0.4% of that of the original trajectory set. Similarly, for the DMA dataset, this

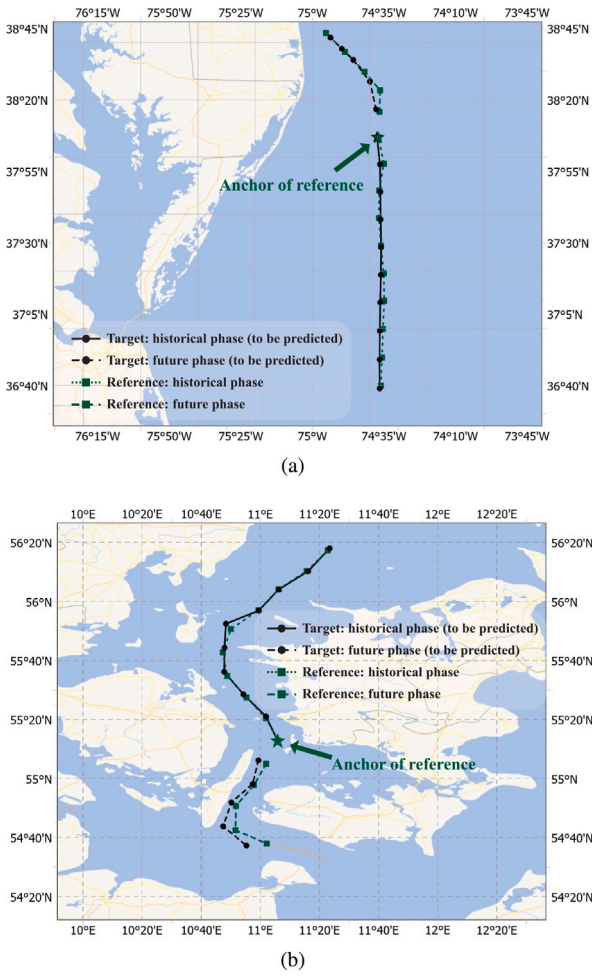

Fig. 8. Example 2 - GRI Results from (a) the US dataset and (b) the DMA dataset. This type is consistent with the direction of the target trajectory, but there is some variation in position over time.

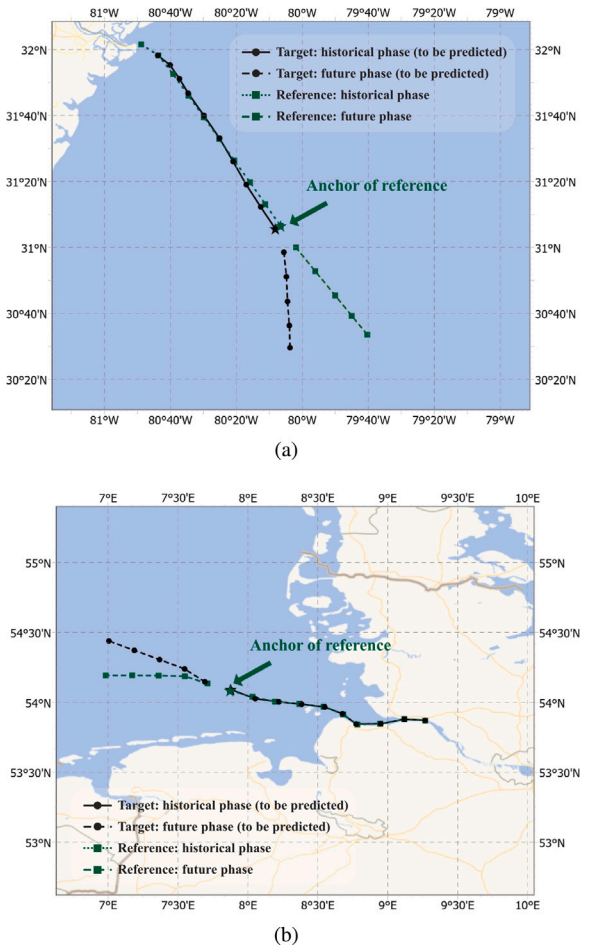

Fig. 9. Example 3 - GRI Results from (a) the US dataset and (b) the DMA dataset. This type has a strong correlation with the target trajectory in the historical phase, but there is a clear divergence in the future phase.

proportion stood at 4.7%, underscoring the significant computational savings afforded by the GRI algorithm.

We pause to note that the DMA dataset exhibits a less pronounced reduction in candidate set size at each GRI step compared to the US dataset. This difference can be attributed to the more consistent pattern of maritime traffic in Danish waters, where vessels predominantly adhere to established shipping lanes, resulting in a less substantial decrease in candidate trajectories at each filtering step. This contrast highlights the adaptability and efficiency of GRI in diverse maritime settings, affirming its utility in streamlining trajectory-matching process.

4.2. Results for trajectory prediction: quantitative analysis

In this subsection, we present a quantitative analysis of the accuracy of the proposed trajectory prediction framework, compared to the following established baselines:

- Simple-LSTM: Similar to the DLSTM methodology, this model employs a recursive strategy for multi-step forecasting across future n time steps. In contrast, it utilizes unprocessed trajectory sequences X_t at time t (outlined in (1)) as its input, rather than combining differentiated and unprocessed feature data. The architecture of the Simple LSTM includes two LSTM layers and four fully connected layers. The LSTM hidden layers have 256 units, while the dimensions of the fully connected layers are $(2560, 2048, 1024, 512, 2)$ respectively.
- Reference-only: The Reference-only baseline directly adopts information extracted from the reference $r(t)$ as the predictions. Since this baseline relies on the existence of a valid reference trajectory, it can only be applied to the instances that the Ref-DLSTM model is adopted.
- METO-S2S (Zhang et al., 2023): The METO-S2S model is one state-of-the-art prediction model that comprises dual encoders and one decoder. The first encoder integrates a five-layer bidirectional LSTM network to encode the historical trajectory data. The second encoder is designed to encode metadata, such as departure time and vessel type. The decoding phase is facilitated by a two-layer LSTM, which subsequently feeds into a fully connected layer to generate the prediction. The model processes input features of the following form:

  $$x_t = \left(lon_t, lat_t, SOG_t, COG_t, d_t\right),$$

  where d_t represents the cumulative distance sailed by the vessel until time t. More detailed information about this model is provided in Zhang et al. (2023).

We pause to note that among the above models, the transition from SimpleLSTM to DLSTM and then to Ref-DLSTM sequentially incorporates two major improvements we proposed: modifying the input to a differential form and integrating reference trajectories for feature

Table 3
MAE of the test samples with valid reference trajectories obtained from the US dataset.

	MAE_τ					Average MAE
	$\tau=1$	$\tau=2$	$\tau=3$	$\tau=4$	$\tau=5$	
Simple LSTM	13.77	16.24	20.92	27.33	35.11	22.67
Reference-only	2.29	3.94	5.78	7.74	9.81	5.91
METO-S2S	3.81	5.17	7.44	10.16	13.36	7.99
DLSTM	1.14	3.01	5.50	8.43	11.76	5.97
Ref-DLSTM	**1.25**	**3.06**	**5.10**	**7.30**	**9.62**	**5.27**

Table 4
MAE of the test samples with valid reference trajectories obtained from the DMA dataset.

	MAE_τ					Average MAE
	$\tau=1$	$\tau=2$	$\tau=3$	$\tau=4$	$\tau=5$	
Simple LSTM	5.53	7.69	10.92	14.68	18.58	11.48
Reference-only	2.03	3.37	4.75	6.26	7.89	4.86
METO-S2S	3.30	4.50	6.65	9.13	12.00	7.10
DLSTM	1.73	4.29	7.58	11.45	15.67	8.14
Ref-DLSTM	**1.32**	**2.83**	**4.45**	**6.20**	**8.02**	**4.56**

fusion. Therefore, this progression inherently serves the purpose of an ablation study.

In the analysis, we use the Mean Absolute Error (MAE) of the predicted positions to quantify the discrepancy between the predicted and the actual vessel trajectories. The MAE is calculated using the following formula:

$$\mathrm{MAE}_\tau = \frac{1}{N}\sum_{i=1}^{N} dis\left(P_{t+\tau}, \hat{P}_{t+\tau}\right), \tag{20}$$

where $dis(P_{t+\tau}, \hat{P}_{t+\tau})$ represents the great-circle distance between the predicted position $\hat{P}_{t+\tau}$ and the actual position $P_{t+\tau}$, and N is the number of test samples.

We note that the proposed method incorporates two prediction modes: the Ref-DLSTM is employed when a reference trajectory is available, and the DLSTM is used otherwise. Therefore, to evaluate the predictive efficacy of the model, we categorize the samples into two groups based on the availability of a valid reference trajectory as determined by the GRI module, creating subsets with and without a reference trajectory. Within the datasets from the US and DMA, valid reference trajectories were identified for 53.3% and 75.5% of the test samples, respectively. It is crucial to acknowledge that this bifurcation serves primarily to facilitate a nuanced comparison of our method across different scenarios. In real-world applications, including during testing, the selection between Ref-DLSTM and DLSTM modes is seamlessly managed by the GRI module based on the data at hand.

Tables 3 and 4 present a comparative analysis of the prediction accuracy of Ref-DLSTM against various baselines for cases where a reference trajectory is available. The evaluation metrics include the MAE at each prediction step, i.e., $\mathrm{MAE}_\tau, \tau = 1, 2, \ldots, n$, and the average MAE across all $n = 5$ steps, calculated as follows:

$$\mathrm{MAE} = \frac{1}{n}\sum_{\tau=1}^{n} \mathrm{MAE}_\tau. \tag{21}$$

Additionally, to demonstrate the enhancement because of the integration of a reference trajectory in the Ref-DLSTM model, a direct comparison with the DLSTM model is also provided in the aforementioned tables.[1]

The results presented in Tables 3 and 4 indicate that the Simple-LSTM model underperformed in terms of both individual-step prediction error (MAE_τ) and overall average prediction error (MAE), primarily because of its simplistic approach to input data, which hindered its capability to learn the dynamic and nonlinear aspects of vessel movements. Upon modifying the input to differential features as for DLSTM, the average MAE was reduced by 73.7% for the US dataset and 29.1% for the DMA dataset, respectively. The improvement comes in two-fold. On one hand, the differential input format enhances the model's sensitivity to trajectory positions and motion characteristics, enabling more accurate learning of information changes. On the other hand, since the prediction is made in a differential manner, where future position information is adjusted based on historical data, this approach ensures the continuity of the trajectory. The experimental results from both datasets also confirm the robustness of the DLSTM model.

Furthermore, long-term trajectories might exhibit motion characteristics in the future that differ greatly from historical patterns, leading to larger cumulative errors for DLSTM if the prediction horizon is large. To address this issue, reference trajectories are introduced for feature fusion, resulting in the Ref-DLSTM model. Experimental results demonstrate that the incorporation of reference trajectories can improve the prediction accuracy, reducing the MAE by 13.4% and 18.2% for the US dataset and 45.9% and 48.8% for the DMA dataset, respectively, achieved at $\tau = 4$ and $\tau = 5$ compared to the DLSTM model.

For the METO-S2S model, Tables 3 and 4 also demonstrate a more subdued error amplification across successive prediction steps, a direct outcome of its architecture which facilitates simultaneous multi-step trajectory forecasting. The design of METO-S2S inherently balances the prediction accuracy across all steps during the back-propagation process by adjusting the loss contribution of each step. However, this methodology can result in elevated errors for the initial trajectory points. In contrast, the DLSTM and Ref-DLSTM models, which predict each trajectory point individually, effectively counteract this early-stage prediction inaccuracy. In particular, Ref-DLSTM builds upon the DLSTM framework by incorporating reference trajectory data to further mitigate error propagation in long-range predictions. As a result, Ref-DLSTM achieves a significantly lower overall prediction error, with the average MAE showing a reduction of 34.0% and 35.8% in the US and DMA datasets, respectively, when compared to the METO-S2S model.

In the evaluation of test samples without a valid reference trajectory, where Reference-Only and Ref-DLSTM models are inapplicable, the DLSTM model performance is benchmarked against Simple-LSTM and METO-S2S baselines, as detailed in Tables 5 and 6. The DLSTM model demonstrates a substantial reduction in average MAE, outstripping Simple-LSTM by 63.2% and METO-S2S by 19.1% in the US dataset, and eclipsing them by 42.0% and 33.0%, respectively, in the DMA dataset. We will note that a comparative analysis with results from Tables 3 and 4 reveals higher prediction errors in the absence of a reference trajectory, highlighting the complex motion patterns of these trajectories which present significant learning challenges. This contrast accentuates the efficacy of the Ref-DLSTM's integration of reference trajectories in enhancing prediction accuracy.

Table 5
MAE of the test samples without valid reference trajectories obtained from the US dataset.

	MAE_τ					Average MAE
	$\tau=1$	$\tau=2$	$\tau=3$	$\tau=4$	$\tau=5$	
SimpleLSTM	11.45	13.37	17.04	21.97	27.85	18.34
METO-S2S	3.85	5.26	7.74	10.74	14.04	8.33
DLSTM	**1.33**	**3.50**	**6.28**	**9.50**	**13.09**	**6.74**

Table 6
MAE of the test samples without valid reference trajectories obtained from the DMA dataset.

	MAE_τ					Average MAE
	$\tau=1$	$\tau=2$	$\tau=3$	$\tau=4$	$\tau=5$	
Simple LSTM	12.51	14.44	17.73	21.67	26.00	18.47
METO-S2S	14.17	11.92	14.28	17.69	21.83	15.98
DLSTM	**2.37**	**5.76**	**9.99**	**14.92**	**20.49**	**10.71**

[1] In this case, DLSTM is applied to the samples that a valid reference trajectory is found.

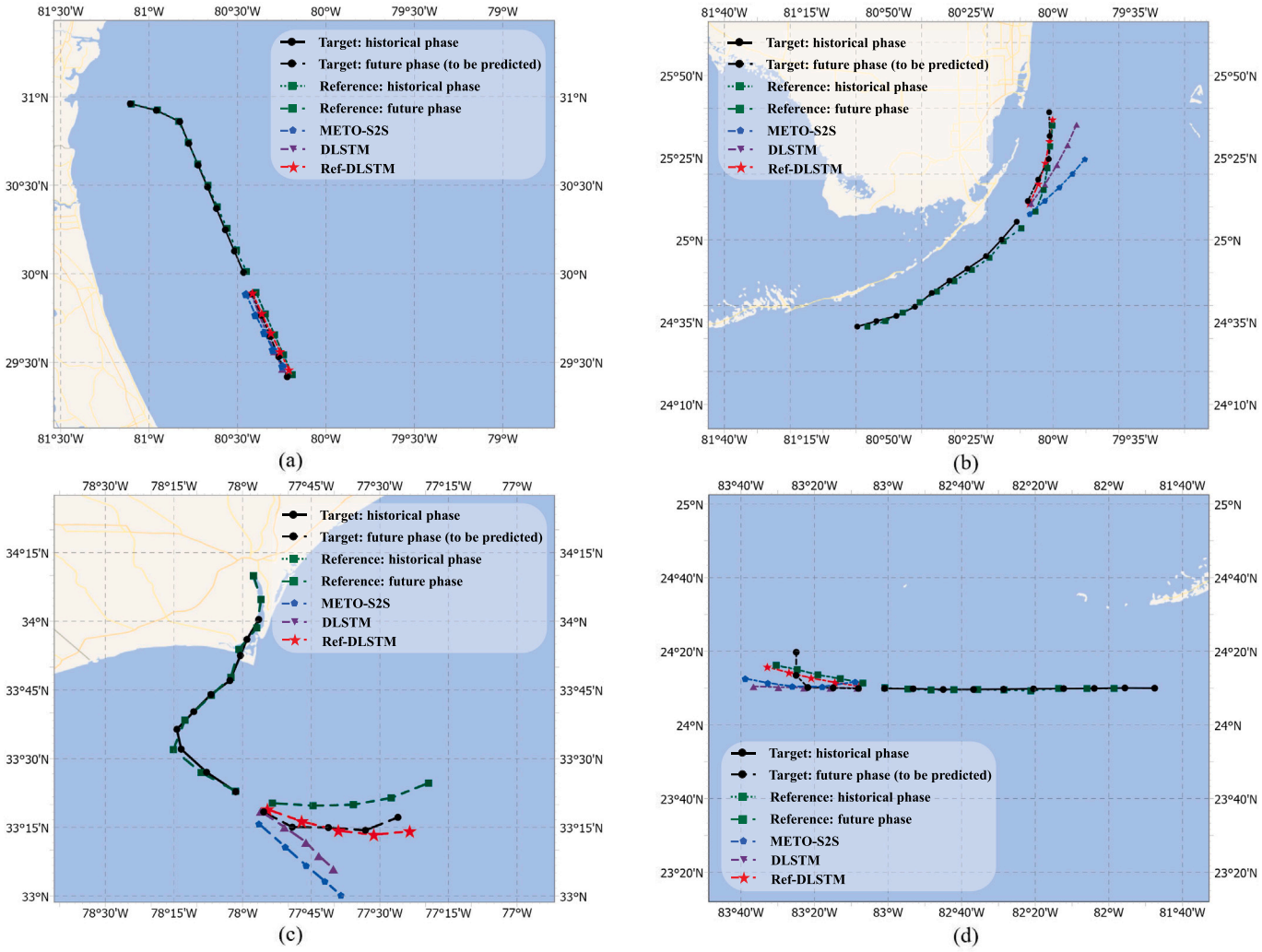

Fig. 10. Representative examples selected from the US test datasets with valid trajectories: (a) the target trajectory exhibits a smooth pattern, making the prediction relatively straightforward; (b) the target trajectory exhibits a smooth pattern in the historical phase but has a significant change of COG in the future phase; (c) the target trajectory exhibits significant changes in both SOG and COG in the historical and future phase; (d) the target trajectory exhibits a smooth pattern in the historical phase, but has much sharper changes in COG in the future phase.

4.3. Results for trajectory prediction: qualitative analysis

In Section 4.2, we analyzed the advantages of our proposed method over existing technologies through quantitative results. In this subsection, we will present several representative examples from the US dataset to demonstrate the predictive performance of the Ref-DLSTM and DLSTM models, as well as the state-of-the-art trajectory prediction method, METO-S2S. The predicted trajectories are shown in Figs. 10(a)–10(d), while Table 7 presents the prediction errors at each time step. In each of these figures, the black trajectory represents the historical and future stages of the target vessel; the green dashed trajectory is the reference trajectory found by the GRI module; the yellow trajectory represents the predicted trajectory by the METO-S2S model; the blue trajectory represents the trajectory predicted by DLSTM; and the red trajectory represents the trajectory predicted by Ref-DLSTM.

The example in Fig. 10(a) is a typical one in that both the Ref-DLSTM and the baselines perform well on trajectory predictions. This is because the target trajectory is a smooth curve with minor variations in SOG and COG, reflecting a relatively straightforward navigational pattern.

In contrast to the relatively straightforward scenario presented in Fig. 10(a), the trajectory in Fig. 10(b) poses a greater challenge because of significant COG changes in the future phase, diverging from the historical phase of near-linear motion. In this case, the Ref-DLSTM model outperforms the METO-S2S and DLSTM models in terms of prediction accuracy, benefiting from the inclusion of a reference trajectory that provides critical insights into forthcoming COG shifts. As can be seen

Table 7
Prediction errors for the example presented in Fig. 10.

		MAE_τ				
		$\tau = 1$	$\tau = 2$	$\tau = 3$	$\tau = 4$	$\tau = 5$
(a)	METO-S2S	3.39	3.19	4.06	5.50	7.16
	DLSTM	0.78	1.98	3.25	4.43	5.90
	Ref-DLSTM	0.45	1.14	2.17	3.24	4.42
(b)	METO-S2S	2.72	3.15	5.30	12.73	21.59
	DLSTM	2.13	4.40	5.27	10.36	15.51
	Ref-DLSTM	1.87	2.53	2.93	3.11	4.55
(c)	METO-S2S	3.76	7.61	16.67	25.76	37.82
	DLSTM	1.14	2.71	9.61	18.87	30.18
	Ref-DLSTM	1.63	3.96	3.63	3.39	6.90
(d)	METO-S2S	3.64	5.08	7.19	13.61	26.89
	DLSTM	0.67	1.34	1.78	10.11	26.02
	Ref-DLSTM	1.14	2.83	4.97	3.53	15.19

from Table 7, while all models achieve reasonable accuracy at $\tau = 1$, the DLSTM and METO-S2S models cannot anticipate the COG adjustment between $\tau = 2$ and $\tau = 3$, leading to a marked error increase from $\tau = 3$ forward. In contrast, the Ref-DLSTM model effectively leverages GRI-derived predictive priors to adjust its predictions, significantly mitigating errors associated with the abrupt COG changes.

The trajectory shown in Fig. 10(c) transitions from a narrow river channel to the open sea, significantly increasing the difficulty of prediction compared to those in Fig. 10(a) and Fig. 10(b). Analysis of the ground truth trajectory reveals notable changes in COG during the historical phase, as precise control of the vessel course is required within the narrow river. In the future phase, especially after $\tau = 2$, the trajectory undergoes a sharp turn. Additionally, after entering the open sea at the end of the historical phase, the vessel enters an accelerated motion mode. As a result, both the METO-S2S and DLSTM models fail to provide trajectory predictions with satisfactory quality (predicting both the sharp turn and the accelerated speed). Leveraging GRI-derived predictive priors for SOG and COG, the Ref-DLSTM model markedly reduces the prediction error by accurately adjusting for the sudden motion shifts. This demonstrates the capability of Ref-DLSTM to adapt to rapid changes in motion patterns and hence the value of integrating precise prior information to enhance prediction accuracy in intricate navigational contexts.

Fig. 10(d) presents a trajectory with a similar structure to that in Fig. 10(b), characterized by a simple historical phase followed by an abrupt turning in the future phase. However, the example in Fig. 10(d) is significantly more challenging since the COG change is much sharper. While the Ref-DLSTM model can partially redirect towards the accurate trajectory, its effectiveness is limited by the limited information from the reference trajectory. Nonetheless, compared to DLSTM and METO-S2S, the performance of Ref-DLSTM remains superior.

5. Conclusions and discussions

This paper has developed a deep learning-based trajectory prediction framework that utilizes navigation patterns from reference trajectories to improve prediction accuracy. Relying solely on the past motion characteristics of the target trajectory, a Differential Long Short-Term Memory (DLSTM) mode has been first designed for trajectory predictions in the absence of a reference trajectory. Based on DLSTM, an enhanced version, Ref-DLSTM, has been developed to incorporate features from both the target and the reference trajectories, offering refined prediction capabilities. To accelerate the identification of reference trajectories, a grid-based search algorithm has been developed to confine the search to a localized area. Evaluations using AIS datasets from the US and DMA highlight the superiority of the proposed framework over existing models, with significant reductions in geographical distance errors.

The current research has primarily capitalized on historical reference trajectories to refine trajectory prediction accuracy. However, future advancements could explore the integration of environmental factors, such as weather conditions and ocean currents, into the prediction models. Employing techniques like feature fusion and encoding could enable the incorporation of critical weather variables, including wind speed, direction, temperature, and precipitation, alongside the effects of ocean currents on vessel trajectories, potentially enhancing fuel efficiency and voyage timing predictions. Further, there is an opportunity to enhance the feature fusion technique utilized within the Ref-DLSTM model. The existing method, which merges LSTM-processed trajectories, might not adequately capture the intricate relationships present in the trajectory data. Investigating more sophisticated feature fusion approaches and leveraging cutting-edge methodologies such as multimodal deep learning and Graph Neural Networks could provide deeper insights and more effective amalgamate of varied data types, culminating in more precise and adaptable trajectory forecasting.

CRediT authorship contribution statement

Xueyin Li: Writing – original draft, Visualization, Software, Methodology, Formal analysis, Conceptualization. **Chunshan Liu:** Writing – review & editing, Writing – original draft, Visualization, Supervision, Methodology, Funding acquisition, Formal analysis, Conceptualization. **Jianghui Li:** Writing – review & editing, Visualization, Validation, Supervision, Methodology, Conceptualization. **Lou Zhao:** Writing – review & editing, Visualization, Methodology, Conceptualization. **Zhongping Du:** Writing – review & editing, Validation, Supervision, Methodology, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was jointly supported by the Zhejiang Provincial Natural Science Foundation of China under Grant No. LZ22F010001 and the Fundamental Research Funds for the Provincial Universities of Zhejiang under Grant No. GK229909299001-013.

Appendix

The raw AIS data encompasses a variety of vessel information and has a very fine temporal granularity. Additionally, it may also contain noise, errors, and anomalies, necessitating preprocessing to ensure data accuracy and reliability. To extract valid target trajectories, the raw AIS data is filtered and cleaned by the following steps:

1. Trajectory Extraction. Maritime Mobile Service Identity (MMSI) serves as the unique identifier for each vessel. Therefore, each trajectory is identified based on the MMSI embedded in the AIS messages by extracting and sorting the AIS data entries according to their timestamps.
2. Denoising. For each trajectory identified in the previous step, we remove trajectory points with abnormal status, such as those indicating anchoring or mooring, and points where the SOG exceeds 40 knots or falls below 2 knots. Additionally, trajectory points with sudden changes in position are eliminated. Specifically, we calculate the change in latitude and longitude between adjacent trajectory points; if the difference in latitude or longitude of a point compared to the two adjacent points exceeds 0.1°, it is considered an outlier and should be removed.
3. Down-sampling. Since this work considers trajectory prediction at equal time intervals, each trajectory point is expected to have the same time interval between them. This is achieved through linear interpolating the trajectory data after denoising, with the time interval between adjacent points set to 30 min. The interpolated trajectory points adopt the SOG and COG of the nearest original trajectory point as their instantaneous attributes, while the average SOG and COG are calculated for the adjacent interpolated points.

References

Álvarez, N.G., Adenso-Díaz, B., Calzada-Infante, L., 2021. Maritime traffic as a complex network: A systematic review. Netw. Spat. Econ. 21 (2), 387–417.

Best, R.A., Norton, J., 1997. A new model and efficient tracker for a target with curvilinear motion. IEEE Trans. Aerosp. Electron. Syst. 33 (3), 1030–1037.

Chen, X., Liu, S., Zhao, J., Wu, H., Xian, J., Montewka, J., 2024a. Autonomous port management based AGV path planning and optimization via an ensemble reinforcement learning framework. Ocean Coast. Manag. 251, 107087.

Chen, X., Lv, S., Shang, W.-l., Wu, H., Xian, J., Song, C., 2024b. Ship energy consumption analysis and carbon emission exploitation via spatial-temporal maritime data. Appl. Energy 360, 122886.

DMA, 2024. AIS data of Denmark. https://dma.dk/safety-at-sea/navigational-information/ais-data. (Accessed April 2024).
Emmens, T., Amrit, C., Abdi, A., Ghosh, M., 2021. The promises and perils of automatic identification system data. Expert Syst. Appl. 178, 114975.
Fang, Z., Yu, H., Ke, R., Shaw, S.-L., Peng, G., 2018. Automatic identification system-based approach for assessing the near-miss collision risk dynamics of ships in ports. IEEE Trans. Intell. Transp. Syst. 20 (2), 534–543.
Feng, H., Cao, G., Xu, H., Ge, S.S., 2022. IS-STGCNN: An improved social spatial-temporal graph convolutional neural network for ship trajectory prediction. Ocean Eng. 266, 112960.
Forti, N., Millefiori, L.M., Braca, P., Willett, P., 2020. Prediction oof vessel trajectories from AIS data via sequence-to-sequence recurrent neural networks. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, IEEE, pp. 8936–8940.
Gao, D., Zhu, Y., Zhang, J., He, Y., Yan, K., Yan, B., 2021. A novel MP-LSTM method for ship trajectory prediction based on AIS data. Ocean Eng. 228, 108956.
Guo, S., Zhang, H., Guo, Y., 2023. Toward multimodal vessel trajectory prediction by modeling the distribution of modes. Ocean Eng. 282, 115020.
Huang, Z., Wang, Z., Chen, H., Zhang, Z., Wang, J., Yuan, Z., Jin, Y., Wu, X., 2022. EA-VTP: Environment-aware long-term vessel trajectory prediction. In: 2022 International Joint Conference on Neural Networks. IJCNN, IEEE, pp. 1–7.
Kanazawa, M., Wang, T., Skulstad, R., Li, G., Zhang, H., 2022. Knowledge and data in cooperative modeling: Case studies on ship trajectory prediction. Ocean Eng. 266, 112998.
Laxhammar, R., 2014. Chapter 4 - Anomaly detection. In: Conformal Prediction for Reliable Machine Learning. Morgan Kaufmann, pp. 71–97.
Li, H., Liu, J., Liu, R.W., Xiong, N., Wu, K., Kim, T.-h., 2017. A dimensionality reduction-based multi-step clustering method for robust vessel trajectory analysis. Sensors 17 (8), 1792.
Li, H., Liu, J., Yang, Z., Liu, R.W., Wu, K., Wan, Y., 2020. Adaptively constrained dynamic time warping for time series classification and clustering. Inform. Sci. 534, 97–116.
Mehri, S., Alesheikh, A.A., Basiri, A., 2021. A contextual hybrid model for vessel movement prediction. IEEE Access 9, 45600–45613.
Murray, B., Perera, L.P., 2020. A dual linear autoencoder approach for vessel trajectory prediction using historical AIS data. Ocean Eng. 209, 107478.
Murray, B., Perera, L.P., 2021. An AIS-based deep learning framework for regional ship behavior prediction. Reliab. Eng. Syst. Saf. 215, 107819.
Nguyen, D., Fablet, R., 2021. Traisformer-A generative transformer for AIS trajectory prediction. arXiv e-prints, arXiv–2109.
Pan, N., Ding, Y., Fu, J., Wang, J., Zheng, H., 2021. Research on ship arrival law based on route matching and deep learning. J. Phys. Conf. Ser. 1952 (2), 022023.
Tang, H., Yin, Y., Shen, H., 2022. A model for vessel trajectory prediction based on long short-term memory neural network. J. Mar. Eng. Technol. 21 (3), 136–145.
the U.S. Coast Guard, 2024. AIS data of the US. https://marinecadastre.gov/data/. (Accessed April 2024).
Wang, L., Chen, P., Chen, L., Mou, J., 2021. Ship AIS trajectory clustering: An HDBSCAN-based approach. J. Mar. Sci. Eng. 9 (6), 566.
Wu, W., Chen, P., Chen, L., Mou, J., 2023. Ship trajectory prediction: An integrated approach using ConvLSTM-based sequence-to-sequence model. J. Mar. Sci. Eng. 11 (8), 1484.
Xiao, Z., Fu, X., Zhang, L., Goh, R.S.M., 2019. Traffic pattern mining and forecasting technologies in maritime traffic service networks: A comprehensive survey. IEEE Trans. Intell. Transp. Syst. 21 (5), 1796–1825.
Xiao, Y., Li, X., Yao, W., Chen, J., Hu, Y., 2022. Bidirectional data-driven trajectory prediction for intelligent maritime traffic. IEEE Trans. Intell. Transp. Syst. 24 (2), 1773–1785.
Xu, X., Liu, C., Li, J., Miao, Y., 2022. Trajectory clustering for SVR-based time of arrival estimation. Ocean Eng. 259, 111930.
Yu, H., Murray, A.T., Fang, Z., Liu, J., Peng, G., Solgi, M., Zhang, W., 2021. Ship path optimization that accounts for geographical traffic characteristics to increase maritime port safety. IEEE Trans. Intell. Transp. Syst. 23 (6), 5765–5776.
Yuan, G., Sun, P., Zhao, J., Li, D., Wang, C., 2017. A review of moving object trajectory clustering algorithms. Artif. Intell. Rev. 47, 123–144.
Zhang, X., Fu, X., Xiao, Z., Xu, H., Qin, Z., 2022. Vessel trajectory prediction in maritime transportation: Current approaches and beyond. IEEE Trans. Intell. Transp. Syst. 23 (11), 19980–19998.
Zhang, Y., Han, Z., Zhou, X., Li, B., Zhang, L., Zhen, E., Wang, S., Zhao, Z., Guo, Z., 2023. METO-S2S: A S2S based vessel trajectory prediction method with multiple-semantic encoder and type-oriented decoder. Ocean Eng. 277, 114248.
Zhang, S., Wang, L., Zhu, M., Chen, S., Zhang, H., Zeng, Z., 2021. A bi-directional lstm ship trajectory prediction method based on attention mechanism. In: 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference. IAEAC, IEEE, pp. 1987–1993.
Zhao, L., Shi, G., 2019. A trajectory clustering method based on douglas-peucker compression and density for marine traffic pattern recognition. Ocean Eng. 172, 456–467.
Zhao, M., Yao, X., Sun, J., Zhang, S., Bai, J., 2018. GIS-based simulation methodology for evaluating ship encounters probability to improve maritime traffic safety. IEEE Trans. Intell. Transp. Syst. 20 (1), 323–337.