

Maritime Anomaly Detection using Density-based Clustering and Recurrent Neural Network

Liangbin Zhao and Guoyou Shi

(Navigation College, Dalian Maritime University, Dalian, China)

(E-mail: vszlb@126.com)

Maritime anomaly detection can improve the situational awareness of vessel traffic supervisors and reduce maritime accidents. In order to better detect anomalous behaviour of a vessel in real time, a method that consists of a Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm and a recurrent neural network is presented. In the method presented, the parameters of the DBSCAN algorithm were determined through statistical analysis, and the results of clustering were taken as the traffic patterns to train a recurrent neural network composed of Long Short-Term Memory (LSTM) units. The neural network was applied as a vessel trajectory predictor to conduct real-time maritime anomaly detection. Based on data from the Chinese Zhoushan Islands, experiments verified the applicability of the proposed method. The results show that the proposed method can detect anomalous behaviours of a vessel regarding speed, course and route quickly.

KEYWORDS

1. Vessel Trajectories.
2. Anomaly detection.
3. DBSCAN.
4. Recurrent neural network.

Submitted: 14 July 2018. Accepted: 27 December 2018. First published online: 8 February 2019.

1. INTRODUCTION. Maritime safety and security are achieved via Maritime Situational Awareness (MSA), which is supported by surveillance and tracking systems such as the Automatic Identification System (AIS) (Sidibé and Shu, 2017). Surveillance operators are required to search for and predict conflict situations such as risks of collision, abnormal vessels and suspicious activities emerging from a large number of vessels within vast sea areas. Early detection of such situations allows time in which to take appropriate action, possibly before potential problems can occur (Wiersma, 2010; Riveiro, 2011). To offer support for operators of maritime surveillance systems in detecting these situations and choosing timely responses, many studies have been made on the topic of maritime anomaly detection.

As summarised in Riveiro et al. (2018) based on the data processing strategy used, maritime anomaly detection methods can be divided into three main categories: data-driven approaches, signature-based approaches and hybrid approaches. Data-driven approaches appear easier to apply on a larger scale to gain efficient classification performance and detect different types of anomalies. Better AIS terminal networks and greater data storage

and collection capacity have made it possible to access notably large volumes of data related to sea traffic, including ship location information. The majority of studies in this area fall under the data-driven category.

In studies of maritime anomaly detection, data-driven approaches usually consist of two steps, that is, normalcy extraction and detection. In the first step, a model representing normal vessel movement behaviour is taught from massive vessel historical movement data. In the second step, new vessel position data are assessed by the learned model, which considers any mismatch as anomalous behaviour.

As reviewed by Riveiro et al. (2018), three main methods are used to extract the normalcy condition for maritime anomaly detection: parametric methods, nonparametric methods and clustering methods. Gaussian Mixture Models (GMMs) are one of the most popular parametric methods. In work by Laxhammar (2008), the Gaussian mixture model and a greedy version of the expectation-maximisation algorithm were used to extract normalcy. For the nonparametric model, Kernel Density Estimation (KDE) is the most commonly used method. Ristic et al. (2008) used an adaptive KDE to derive a normal traffic model and used particle filters to predict the positions of vessels based on the derived density. However, clustering has recently become an increasingly popular method due to its favourable performance and the fact that it is easy to implement. K-medoids is a version of the k-means algorithm, which is based on classification. Zhen et al. (2017a) used the k-medoids algorithm to cluster ship trajectories and a Bayesian network to detect abnormal vessels. Li et al. (2017) proposed a multi-step trajectory clustering method for robust AIS trajectory clustering based on Dynamic Time Warping (DTW) distance, Principal Component Analysis (PCA) and the k-medoids algorithm. The main drawback of the k-medoids algorithm is that the number of clusters is difficult to determine. Moreover, not all of the ship trajectories might be components of the traffic pattern on water because a ship has freedom of movement in most water areas, and classification-based clustering cannot recognise those trajectories that do not present the characteristics of the traffic pattern. Consequently, density-based clustering methods that can recognise noise data appear to be the most appropriate approach to pattern recognition due to their convenient properties compared with classification-based clustering. Density-based methods do not require specification of the number of clusters; they have the ability to derive arbitrarily shaped clusters and incorporate by-products of classification of the noise points (Riveiro et al., 2018). Zhen et al. (2017b) utilised AIS data and the DBSCAN algorithm to obtain clusters of encountered vessels. Pallotta et al. (2013) proposed an approach based on an incremental variant of the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm (Ester et al., 1996) and the turning points to learn maritime traffic routes from vessel AIS data. These approaches selected the trajectories that deviated significantly from the patterns of the abnormal trajectories. Yan et al. (2016) proposed an unsupervised method that applied density-based clustering for the identification of stops and moves in tracking points via the extraction of stationary areas of interest from the stops and detection of the main traffic routes from the moves. Liu et al. (2014) improved the DBSCAN algorithm for extraction of normal ship trajectory patterns by considering the non-spatial attributes of the tracking point (speed and direction).

However, the DBSCAN clustering algorithm is sensitive to the input parameter values, the determination of which requires complex and time-consuming computation or expert domain knowledge, and studies are seldom aware of this problem. Yan et al. (2016) adopted a heuristic approach based on k-nearest neighbour distances to determine the parameters of

the DBSCAN algorithm. However, this approach requires many human interventions. Pan et al. (2014) used entropy theory to determine the parameters of the DBSCAN algorithm, but the calculation in this approach is highly complex. Moreover, most of the methods available in the literature are designed for offline anomaly detection in trajectories where detection is performed after the entire vessel trajectory has been observed, and this is a serious limitation in surveillance applications. Therefore, additional effort should be devoted to real-time maritime anomaly detection.

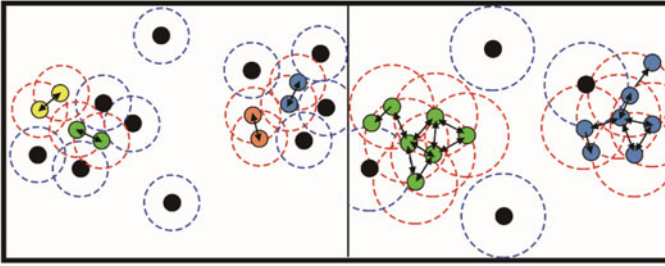
Neural networks are powerful learning models that can achieve state-of-the-art results in a wide range of supervised and unsupervised machine learning tasks. Rhodes et al. (2007; 2005) presented a neural network classifier known as fuzzy Adaptive Resonance Theory Map (ARTMAP) (Carpenter et al., 1992) to evaluate the behaviours of vessels. Bomberger et al. (2006) developed a method based on associative learning and a neural network to predict future vessel behaviours and detect abnormal vessels. In their work, vessel locations in the region are discretised with grids, but the prediction accuracy in the application is not very high. Xu et al. (2011) used a Back Propagation (BP) neural network to predict vessel trajectories. Daranda (2016) built a three-layer BP neural network model to learn the clustered turning points and used it to predict marine traffic. Recurrent Neural Networks (RNNs) are connectionist models with the ability to selectively pass information across sequence steps while processing sequential data one element at a time. Thus, these methods can model inputs and outputs consisting of sequences of elements that are not independent. Further, recurrent neural networks can simultaneously model the sequential and time dependencies at multiple scales (Lipton et al., 2015). Compared with traditional neural networks, RNNs are more capable of processing time series data. Vessel trajectory data consists of tracking points with time stamps, a type of typical time series data. Consequently, RNNs are well suited to the tasks of maritime anomaly detection.

The main contribution of this paper is to improve the clustering performance for extracting the normalcy condition of vessel trajectories, and a method for the problem of input parameter selection in the DBSCAN algorithm is proposed. In the process of applying the DBSCAN algorithm, inverse Gaussian fitting is used to statistically determine the appropriate parameters. In addition, to improve the performance in availability and timeliness for detecting the anomalous behaviour of vessels, an RNN is applied in the field of maritime anomaly detection, which treats the vessel trajectory data from the result of clustering as the training data. Based on large amounts of real vessel data from AIS, we conducted an experiment for verification.

The remainder of the paper is structured as follows. Section 2 proposes the method for determining the parameters of the DBSCAN algorithm. Section 3 introduces the recurrent neural network for maritime anomaly detection. Section 4 shows the results of the verification experiment and Section 5 presents the conclusions.

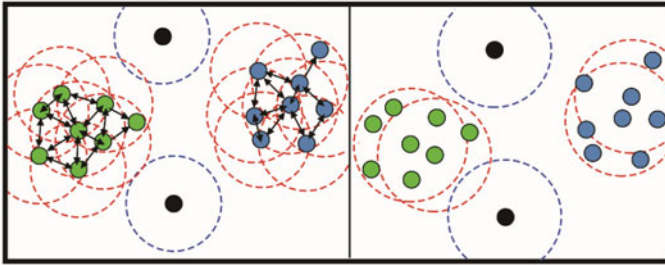
2. DETERMINATION OF DBSCAN PARAMETERS. The DBSCAN algorithm is a classic density-based clustering algorithm that has two input parameters $minPts$ and ϵ . Consider a set of points in a space to be clustered. For the purpose of DBSCAN clustering, the points are classified as core points, density-connected points and outliers. A point p is a core point if at least $minPts$ points are located within distance ϵ . A point q is directly density-reachable from point p if q is within distance ϵ of p and p is a core point. Density reachability is the transitive closure of direct density reachability, and this

(a) $minPts: 2 \ \epsilon: \odot$ Number of outliers: 10 $minPts: 3 \ \epsilon: \odot$ Number of outliers: 4



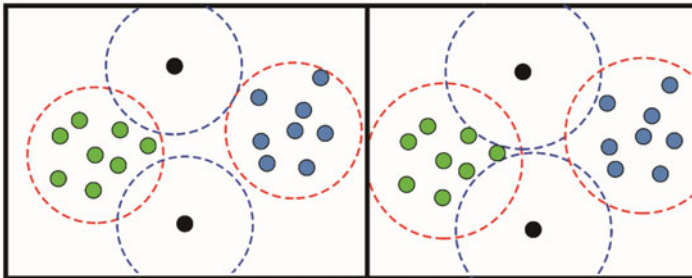
Clustering results with $minPt$ of 2 and 3. Clusters significantly absorb outliers around them. Black dot represents the noise data. Colour dot represents the data in a cluster.

(b) $minPts: 4 \ \epsilon: \odot$ Number of outliers: 2 $minPts: 5 \ \epsilon: \odot$ Number of outliers: 2



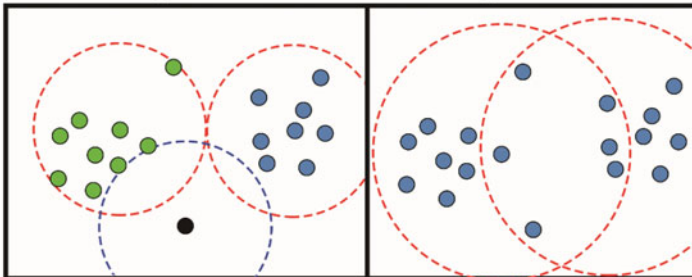
Clustering results with $minPt$ of 4 and 5. Clusters become stable (ideal clustering performance).

(c) $minPts: 6 \ \epsilon: \odot$ Number of outliers: 2 $minPts: 7 \ \epsilon: \odot$ Number of outliers: 2



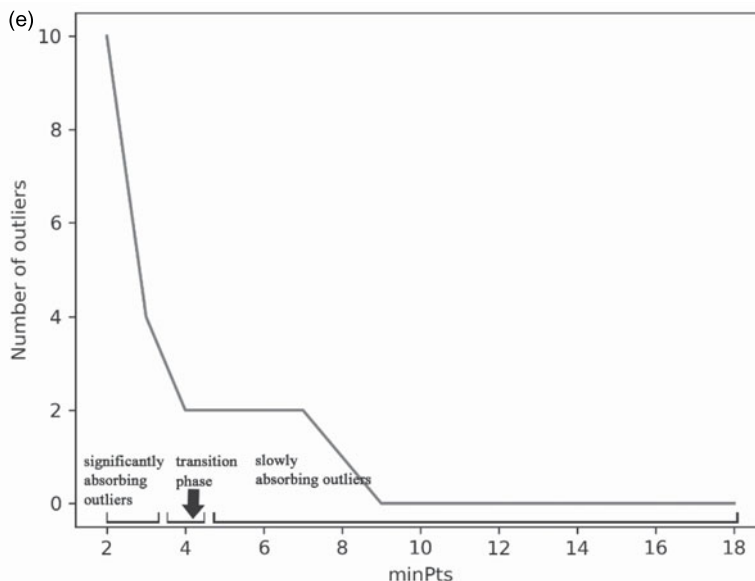
Clustering results with $minPt$ of 6 and 7. Outliers around the clusters become sparser.

(d) $minPts: 8 \ \epsilon: \odot$ Number of outliers: 1 $minPts: 9 \ \epsilon: \odot$ Number of outliers: 0



Clustering results with $minPt$ of 8 and 9.

Figure 1. Continued.



Line graph of the number of outliers.

Figure 1. The process of determining the parameter combination of DBSCAN algorithm. (a)-(d) Show the clustering results with different parameter combinations (ascending order). (e) Is the line graph for identifying the transition phase and optimal parameters.

relationship is asymmetric. This asymmetric density reachability is density connectivity. Based on the parameters, the points that are density-connected with each other are grouped into a cluster. All points that are not density-connected with any other point are outliers. In the DBSCAN algorithm, the combination of parameters plays a highly important role in the process. However, determination of the appropriate parameters is difficult because their practical meanings are abstract, and many possible combinations of parameters exist. In the first step of our proposed method, we use the statistical relationship to calculate the parameter of ε given a fixed $minPts$, which could sharply reduce the number of possible combinations and transform the problem of two parameter determinations into the problem of parameter combination selection. The second step consists of the selection of a parameter combination based on the clustering results.

2.1. *Statistical relationship between parameters.* The core-distance of an object p in the method presented here is the distance to the $minPts$ closest point, which is a concept in the OPTICS (Ordering points to identify the clustering structure) algorithm (Ankerst et al., 1999), written as $Dist(minPts)$. In the ideal result of density-based clustering, all data in the high-density area should be grouped such that the density of the cluster can reach its maximum. From a local perspective, for an individual data point p , in the case of clustering with the centre data by itself given the fixed $minPts$, the core distance of p $Dist(minPts)$ can exactly meet the conditions of a core point and cluster formation. Therefore, we believe that for a fixed $minPts$, the value of the core distance in this dataset that can enable the most point data to become core points is the appropriate parameter ε . Therefore, we use the inverse Gaussian distribution to find the mode value of the core distances and treat it as the appropriate parameter ε in this dataset when $minPts$ is given. The formulae for the

probability density function of the inverse Gaussian distribution and its mode value are given as follows:

$$P(x) = \left[\frac{\lambda}{2\pi x^3} \right]^{\frac{1}{2}} e^{-\frac{\lambda(x-\mu)^2}{2\mu^2 x}} \quad (1)$$

$$Mode = \mu \left[\left(1 + \frac{9\mu^2}{4\lambda^2} \right)^{\frac{1}{2}} - \frac{3\mu}{2\lambda} \right] \quad (2)$$

where λ and μ can be obtained by Maximum Likelihood Estimate (MLE).

Because the parameter *minPts* ranges from one to the size of the dataset, based on the statistical relationship above, we can obtain all of the parameter combinations.

2.2. Selection of parameter combination. The clustering process is conducted with all of the parameter combinations and the number of outliers is taken as the clustering performance to select the optimum parameter combination.

A parameter combination (*minPts* and its ϵ) corresponds to a certain degree of strength that can aggregate the elements into the clusters. A higher value of *minPts* indicates a greater strength. In multiple experiments, the change in the clustering result during the process of increases in the value of *minPts* can be viewed as a process in which each cluster absorbs the data around it. When *minPts* is equal to one, all data are classified as a cluster, and no outliers remain, which is apparently not appropriate. Thus, the process of increasing *minPts* is started at two, and at the beginning, many outliers occur around the clusters. With the increasing value of *minPts*, before the ideal clustering results are found, outliers are significantly absorbed into the clusters, which contributes to the sharp drop in the number of outliers. However, the outliers around the clusters subsequently become sparser. Hence, the reduction rate of the number of outliers decreases, which means that the number of outliers becomes stable.

Thus, we believe that the optimal parameter of *minPts* occurs at the transition phase from the stage of significant absorption of outliers to the stage of slowly absorbing outliers. In our method, we use a line graph of the number of outliers (in ascending order of *minPts*) to find the transition phase, as illustrated in Figure 1. Figures 1(a)-(d) show the clustering results with different parameter combinations (ascending order). The dotted-line circle represents the area of the corresponding ϵ of *minPts*. The number of outliers is plotted on the line graph, as illustrated in Figure 1(e). From Figure 1(e), *minPts* = 4 is identified in the transition phase. As shown in Figure 1(b), the clustering result with the corresponding parameter combination (*minPts* = 4) has exactly the ideal clustering performance.

3. ANOMALOUS VESSEL BEHAVIOUR DETECTION USING RECURRENT NEURAL NETWORK.

3.1. Method of maritime anomaly detection based on RNN. The artificial neural network is one of the most effective data-driven approaches for the supervised learning task. It infers a function that maps an input to an output based on example input-output pairs derived from existing data. The characteristics can be used for anomaly detection, not only because it can predict the trajectories that conform to historical patterns, but also for the trajectories that did not conform to historical patterns it may sensitively become inefficient in prediction. While the latter is the property that other trajectory prediction methods that do not rely on historical data, such as the Kalman filter (Perera and Soares, 2010; Stateczny

and Kazimierski, 2011), do not have. Compared to those methods, a neural network can store the trajectory information in the customary route, which is extracted from historical data; only the regular ship movement can be predicted accurately. In the prediction that does not rely on historical data, the goal is to predict the vessel trajectory as accurately as possible no matter whether the movement is abnormal or not. This is the reason the neural network is applied as the predictor in this research, which has different predictive effects for different degrees of abnormality.

An RNN is a class of artificial neural network in which connections between nodes form a directed graph along a sequence. RNNs can use their internal state (memory) to process sequences of inputs. This process allows the model to learn temporal behaviours in a time sequence. The basic principle of the method presented here is that an RNN trained by trajectory data from the results of the DBSCAN algorithm is used as a vessel trajectory predictor. According to the previous tracking points and the traffic pattern, the network can predict the normal position of the next tracking point. It is hypothesised that anomalous vessel behaviours regarding vessel speed, course and route can cause abnormal performance in vessel position. Here, anomalous vessel behaviours are identified through the detection of abnormal position, and the deviation value between the predictive position and actual position is the indicator that shows the normality degree of the tracking point, defined by:

$$deviation_value = GeoDistance (Lat_{pre}, Long_{pre}, Lat_{act}, Long_{act}) \quad (3)$$

where *GeoDistance* is the function for calculating the geographical distance between two geographical coordinates. Lat_{pre} and $Long_{pre}$ are the geographical coordinates of the predictive position. Lat_{act} and $Long_{act}$ are the geographical coordinates of the actual position.

To supply a warning, the thresholds of the deviation value can be determined based on the user's practical requirements and the performance of training. The method here sets high and low levels of thresholds. New vessel position data generates a warning to attract additional attention from supervisors when the deviation value exceeds the high level or when the deviation value has exceeded the low level for several consecutive tracking points as these situations might be the anomalous vessel behaviours. In contrast, new vessel position data are treated as normal vessel behaviour when the prediction of the network is accurate.

The trajectories can be described by a set $T = \{t_i | t_i, i = 1, \dots, 2, n\}$, where t is the trajectory in the experiment and n is the number of trajectories. Trajectory data t consists of AIS tracking points, defined by:

$$t = \{p_j | p_j = (x_j, y_j, v_j, c_j), j = 1, 2, \dots, m\} \quad (4)$$

where j is the time instance index, m is the total number of tracking points in the trajectory and p_j is the state vector of the ship at time index j . For the state vector, x_j and y_j are the coordinates of the tracking point and v_j and c_j are the speed and course of the ship at time index j . In addition, the values were normalised before training of the RNN. Feature scaling is used to bring all values into the range $[0, 1]$.

Trajectory data were processed into input and output datasets by a sliding window for the neural network. The processing method is shown in Algorithm 1.

3.2. *RNN and long short-term memory.* Compared to the traditional structure of a neural network, namely feed-forward neural network, the training efficiency of an RNN is higher in processing time series data. When applying a feed-forward neural network in the prediction of time series data, a fixed time step of information vectors of track points is concatenated into a larger vector by windowing. The information flow from the input layer to the hidden layer is shown in Figure 2. The hidden node value h is calculated as Equation (5).

$$h = \tanh (W_{xh} \cdot \text{concat} (x_1, x_2, \dots, x_n) + b) \tag{5}$$

With a single training example, the weight of feed-forward neural network W_{xh} can be trained by information at all the time steps only once. While in the RNN, the parameters are shared by all time steps, which may increase the training efficiency sharply. The basic structure of our RNN is shown in Figure 3.

As shown in Figure 3, the training of each moment is independent. At time t nodes with recurrent modules receive input from the current data point x_t and from the hidden node value h_{t-1} in the previous state. At time t , nodes with recurrent modules receive input from

Algorithm 1. Creation of a dataset for the neural network.

Require: Vessel trajectory set T , $window_length$ (the number of tracking points used in prediction)

- 1: $Input_dataset = \emptyset; Output_dataset = \emptyset;$
 - 2: **for each** t in T **do**
 - 3: **for** i in $\text{range}(1, m - window_length)$ **do**
 - 4: $Input_sub = \{P_j | i \leq j \leq i + window_length - 1\}$
 - 5: $Output_sub = \{x_j, y_j | j = i + window_length\}$
 - 6: **Add** $Input_sub$ into $Input_dataset$
 - 7: **Add** $Output_sub$ into $Output_dataset$
 - 8: **end for**
 - 9: **end for**
 - 10: **return** $Input_dataset; Output_dataset$
-

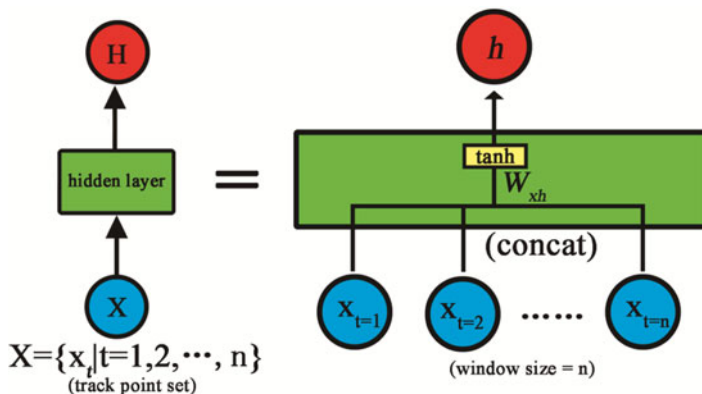


Figure 2. Information flow in a feed-forward neural network.

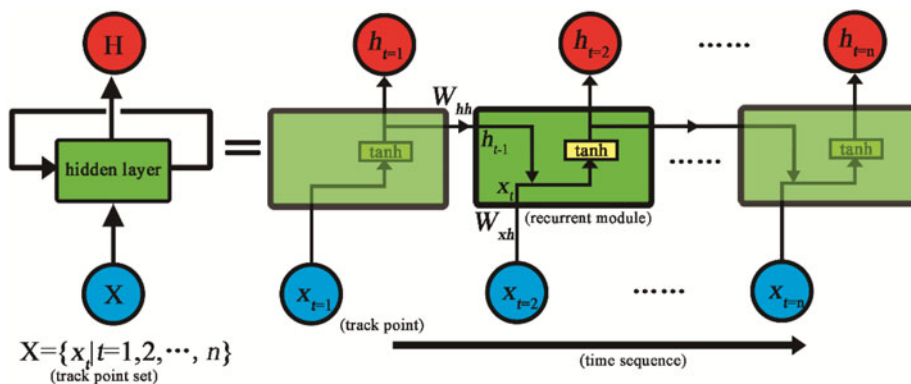


Figure 3. Information flow in a recurrent neural network.

the current data point x_t and from the hidden node value h_{t-1} in the previous state. The hidden node value h_t at time t is calculated as Equation (6). That is to say, RNNs share weights across different time steps of the sequence. Consequently, if there are n time steps in the input data, with a single training example, the weights of RNN W_{xh} , W_{hh} can be trained n times.

$$h_t = \tanh(W_{xh} \cdot x_t + W_{hh} \cdot h_{t-1} + b) \quad (6)$$

In this paper, prediction of the vessel trajectory is performed by a fully connected RNN. A single hidden layer is used, and the number of hidden dynamic features of the hidden layer is determined based on the input data. In updating the weights, the Adam algorithm is applied (Kingma and Ba, 2014); this is a method for efficient stochastic optimisation that requires only first-order gradients and has low memory requirements. The method computes the individual adaptive learning rates for different parameters from estimates of the first and second moments of the gradients. Additionally, the model uses the Rectified linear unit (Relu) as the activation function (Nair and Hinton, 2010).

One of the most successful RNN architectures for sequence learning was introduced to solve the problem of vanishing gradients, namely, the Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997). In this method, the traditional nodes in the hidden layer are replaced with the LSTM unit, which is shown in Figure 4.

The key to the LSTM unit is the cell state. The cell state c_t at time t is shown as the horizontal line running through the top of the LSTM unit in Figure 4. The LSTM unit has the ability to remove or add information to the cell state through the gates, which are marked with f_t , i_t , and o_t in Figure 4. Gates are a way to optionally let information through and are composed of a sigmoid neural net layer (marked with σ) and a point-wise multiplication operation (marked with a multiplication symbol). f_t indicates the forget gate, which decides what information should be discarded from the cell state. The process looks at h_{t-1} and x_t , and outputs a number between 0 and 1 for each number in the cell state C_{t-1} . The number 1 represents “completely retain this”, whereas the number 0 represents “completely discard this”. Additionally, i_t indicates the input gate, which decides what new information from h_{t-1} and x_t can be stored in the cell state, and o_t denotes the output gate, which decides how much information in the cell state can be output. The equations of these

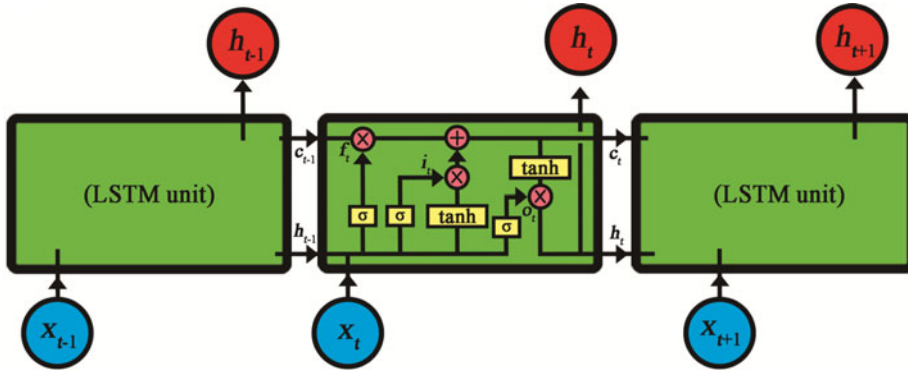


Figure 4. Information flow in a recurrent neural network with LSTM unit.

gates are shown as follows:

$$f_t = \text{sigmoid}(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \tag{7}$$

$$i_t = \text{sigmoid}(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \tag{8}$$

$$o_t = \text{sigmoid}(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \tag{9}$$

Finally, the cell state c_t at time t is calculated based on the gates (f_t and i_t), input x_t , previous cell state c_{t-1} and previous hidden node value h_{t-1} . The hidden node value h_t at time t is calculated based on cell state c_t and output gate o_t . The equations of c_t and h_t are shown as follows:

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \tag{10}$$

$$h_t = o_t \odot \tanh(c_t) \tag{11}$$

4. EXPERIMENTAL CASE STUDIES.

4.1. *Experimental setup.* AIS data was collected from the Zhoushan Islands in January 2015. The research area is located outside of Beilun-Zhoushan port, which is one of the most important ports in China. This water area is situated near the middle portion of the China coastline. Many vessels that use the north-south transportation routes sail through this area. In addition, this area is close to the entrance of the Shrimp main gate waterway, which is the main waterway of Beilun-Zhoushan port, and thus, it is the main area where in-and-out of port ship encounters occur, as shown in Figure 5. All of the Class-A AIS messages from tankers and cargo ships were selected as experimental data from a total of 3,228 trajectories of 1,932 vessels. The data were pre-processed with respect to space and time (Zhao et al., 2018).

4.2. *Results of vessel trajectory clustering.* As described in Section 2.2, in the process of determining the parameter combination, the line graph of the number of outliers and the transition phase is shown in Figure 6. It is worth mentioning that there are slight increases in the noise levels in Figure 6. This is because in the process of increasing the value of $minPt$, when the merger of clusters occurs, the data around the edge of the original cluster might

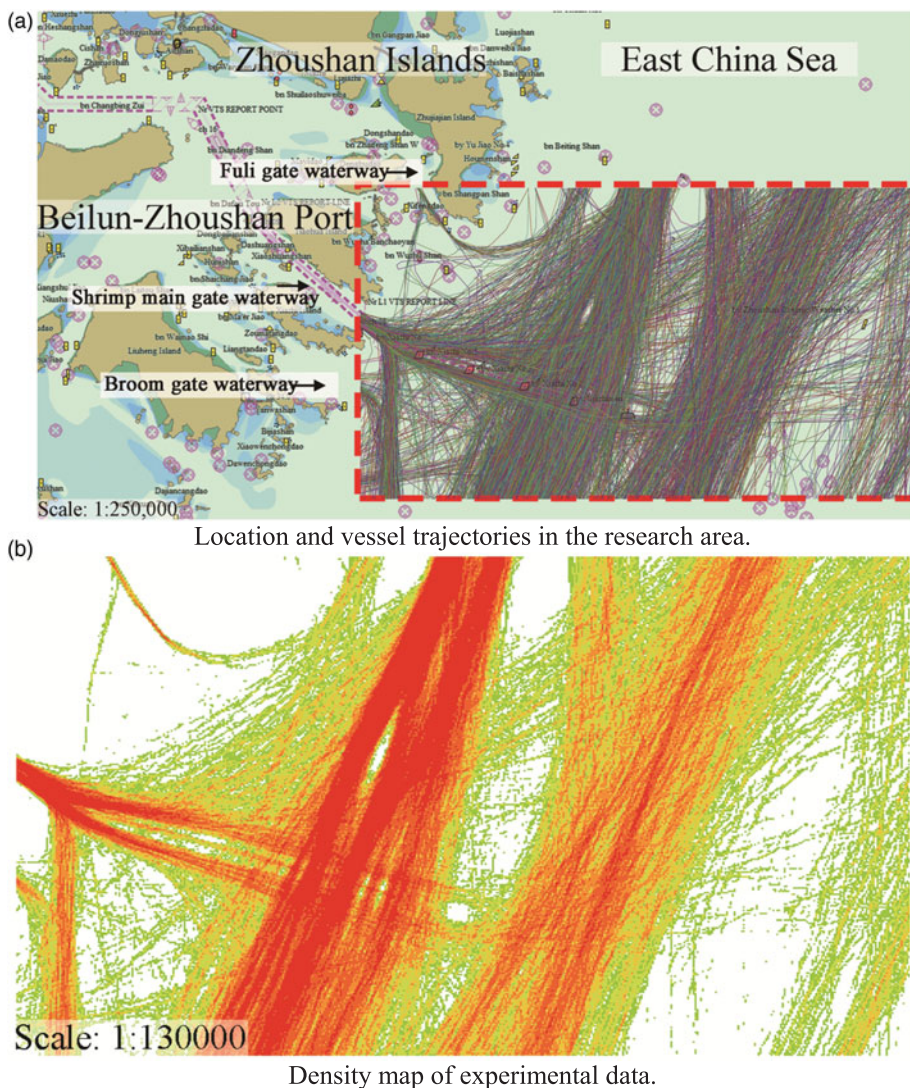


Figure 5. Experimental data source.

be identified as noise data. However, it does not affect the overall trend. After checking the clustering results with the parameter combinations in the transition phase, the parameter combination ($minPt = 10$, $\varepsilon = 0.861$) was determined. The results of the clustering for normalcy extraction are shown in Figure 7 and Table 1.

From Figure 7 and Table 1, it is shown that the vessel trajectories that belong to the different customary routes are grouped into different clusters. From observation, the vessel trajectories in the same cluster are almost consistent in terms of shape. Compared to the results of classification-based clustering (Figure 8), the noise trajectories are removed, and the meaning of the cluster is clearer. The results show that the DBSCAN algorithm with the proposed method for determining the parameters is able to recognise the traffic patterns on

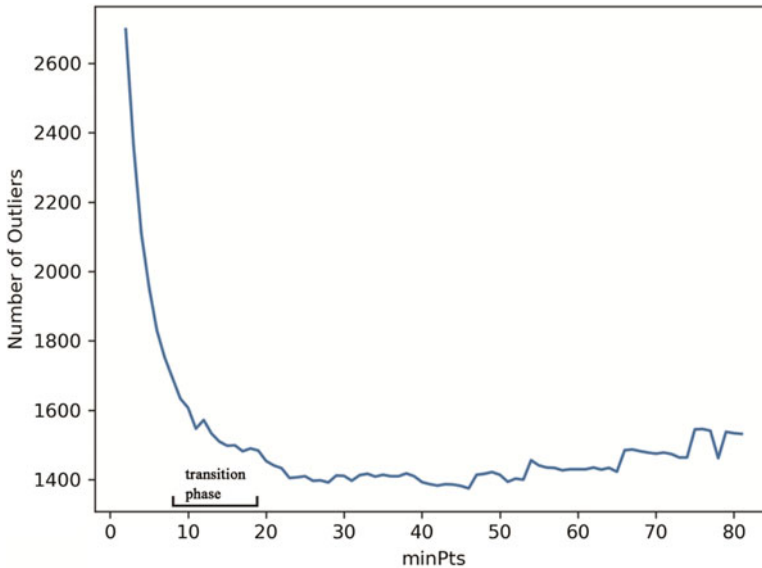


Figure 6. Line graph of the number of outliers in the experiment.

the water. Although not all of the customary routes in this experiment area were detected, the clusters consisting of vessel trajectories in the results can still be used as normalcy data for maritime anomaly detection.

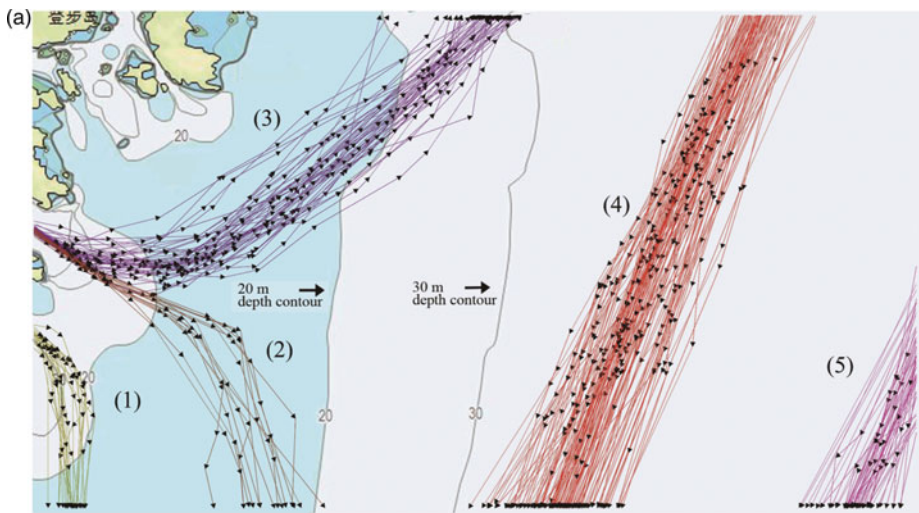
4.3. *Experiment on maritime anomaly detection using recurrent neural networks.* As described in Section 3, the vessel trajectories in the results of the DBSCAN algorithm were used to train the recurrent neural network with the LSTM unit introduced in Section 3.2. The parameter used in the process of obtaining the training dataset, namely *window_length*, is determined as five, which means that in our method, the prediction of the tracking point is made based on five previous tracking points. The time interval of the tracking points in the data set was processed into two minutes. In other words, in this experiment, the method assesses whether the behaviour of the vessel is anomalous by comparing the position that was predicted two minutes ago with the actual current position. The deviation value between the predictive position and the actual position is the indicator used to evaluate results from the neural network.

4.3.1. *Prediction of vessel position.* First, a comparison experiment regarding the prediction performances of a feed-forward neural network (three-layer BP neural network) and recurrent neural network was conducted. The deviation values of these two neural networks after different training epochs are shown in Figure 9. Figures 9(a) and 9(b) are the comparison results of prediction error based on the training dataset and testing dataset (data other than the vessel trajectories contained in the training set), respectively. In this experiment, the testing dataset consists of the vessel trajectories that account for 15% in each cluster.

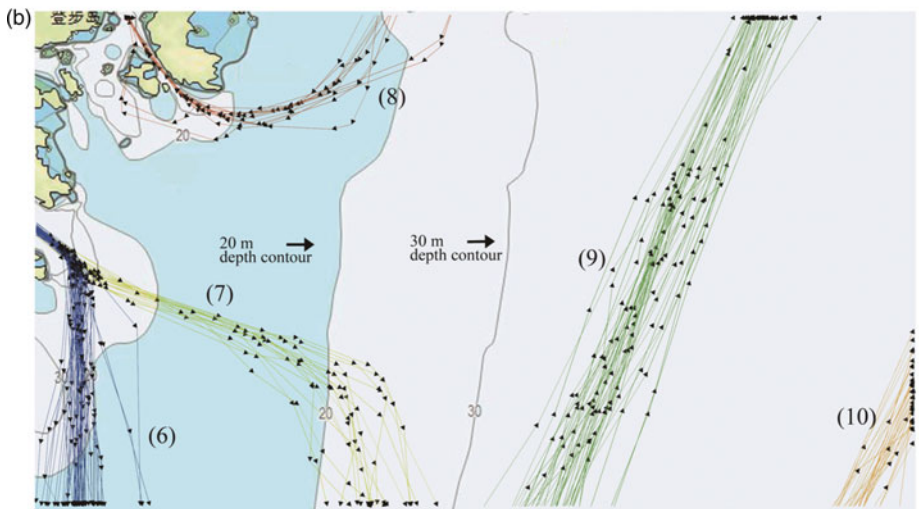
From Figure 9, it can be seen that after the same training epoch the average prediction error of the RNN is 12.8% (training dataset: 20%) below the error of the feed-forward neural network. It is shown that the recurrent neural network used here has a better performance regarding training efficiency and prediction accuracy compared to the feed-forward neural network.

In the verification experiment, the testing dataset was used to validate the ability to predict the normal vessel position and the ability to detect anomalous behaviours of the vessel.

As shown in Figure 9(b), after the prediction, the prediction error (the mean of all of the deviation values using the testing dataset) is 60 m, which can satisfy the accuracy needs of traffic surveillance. Three detailed results of prediction are shown in Figure 10. As shown in Figure 10(a), these three vessel trajectories are the normal trajectories that follow the traffic pattern in this water area, which are selected from the results of clustering. These vessel trajectories are all located in the customary route, and the movement of the ship is

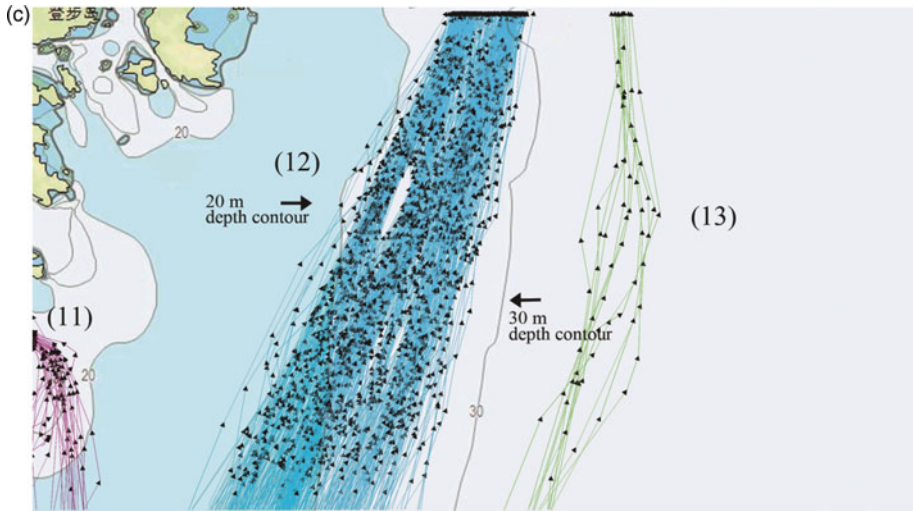


Part 1 of the clustering result.

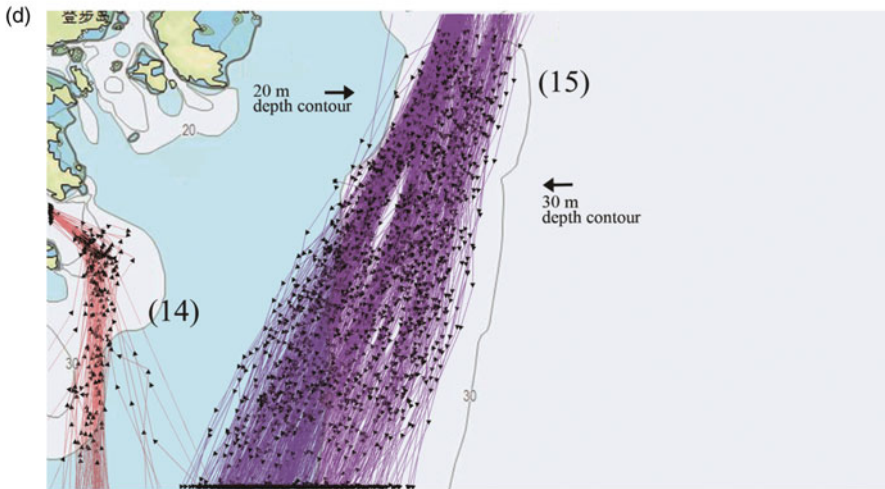


Part 2 of the clustering result.

Figure 7. Continued



Part 3 of the clustering result.



Part 4 of the clustering result

Figure 7. Results of proposed Density-based clustering.

steady (no sudden change in course and speed). Figures 10(b)-10(d) show that the position can be predicted with acceptable precision by the neural network that has learned normalcy. Consequently, the method has been proven to have the ability to predict the position where the vessel is expected to be according to the traffic pattern on the water.

4.3.2. *Detection of anomalous vessel behaviours.* The vessel tracking points that do not follow the traffic pattern are treated as anomalous vessel behaviours. In marine traffic, the situations that might cause such mismatches include abnormal changes in course, abnormal changes in speed and sailing on an abnormal route. In this experiment, three types of abnormal vessel trajectories were selected to verify the detection capability. In the process of detection, two thresholds of deviation value of position were set to supply

Table 1. Description of clusters.

Class	Number of trajectories	Descriptions
(outlier)	1505	Not shown in Figure 7 .
(1)	23	After leaving the port from Broom gate waterway, directly heading to the water area in the south
(2)	15	Leaving the port from Shrimp main gate waterway, turning south in the areas where the depth is relatively shallow
(3)	50	After leaving the port from Shrimp main gate waterway, heading to the water area in the north
(4)	115	Through water area where the depth is in the range of 30 m to 40 m (heading southwest)
(5)	56	Through water area where the depth is larger than 40 m (heading southwest)
(6)	62	After leaving port from Shrimp main gate waterway, directly heading to the water area in the south
(7)	18	Leaving the port from Shrimp main gate waterway, turning south in the areas where the depth is relatively deep
(8)	13	Sailing to the Fuli gate waterway for entry to the port
(9)	41	Through water area where the depth is in the range of 30 m to 40 m (heading northeast)
(10)	31	Through water area where the depth is larger than 40 m (heading northeast)
(11)	32	Sailing to Broom gate waterway for entering the port
(12)	506	Through water area where the depth is in the range of 20 m to 30 m (heading northeast)
(13)	12	Through water area where the depth is in the range of 30 m to 40 m (courses have changed from northeast to north)
(14)	88	Sailing to the Shrimp main gate from the area in the south waterway for entry to the port
(15)	661	Through water area where the depth is in the range of 20 m to 30 m (heading southwest)

the forewarning signal, namely, 75 m (low level) and 250 m (high level). The detailed results are shown in [Figures 11–13](#).

As observed from [Figure 11\(a\)](#), the trajectory is generated for a vessel that sails from the area in the south to the Shrimp main gate waterway for entry to the port. However, unlike most other vessels, which take a steady course, the course that is shown in this trajectory has changed several times within a short period (marked with a black block). The detailed predictive results in this period are shown in [Figure 11\(b\)](#). In [Figure 11\(b\)](#), the deviation value of the tracking point that has a sharp change in course is significantly larger than that of the others. Based on the thresholds of deviation, those tracking points (anomalous vessel behaviours) have been detected and are marked with a symbol. The detailed detection results in this period are shown in [Table 2](#). The deviation between the actual position and predictive position that is caused by the sharp change in course exceeds the threshold value, which is identified as anomalous vessel behaviour.

From [Figure 12\(a\)](#), it can be seen that the trajectory is generated for a vessel that sails from the port area to the water area in the north. However, in its movement process, the vessel slowed down and stopped for an hour (marked with a black block) near the entrance to the waterway. This anomalous vessel behaviour seldom occurs in this water area and

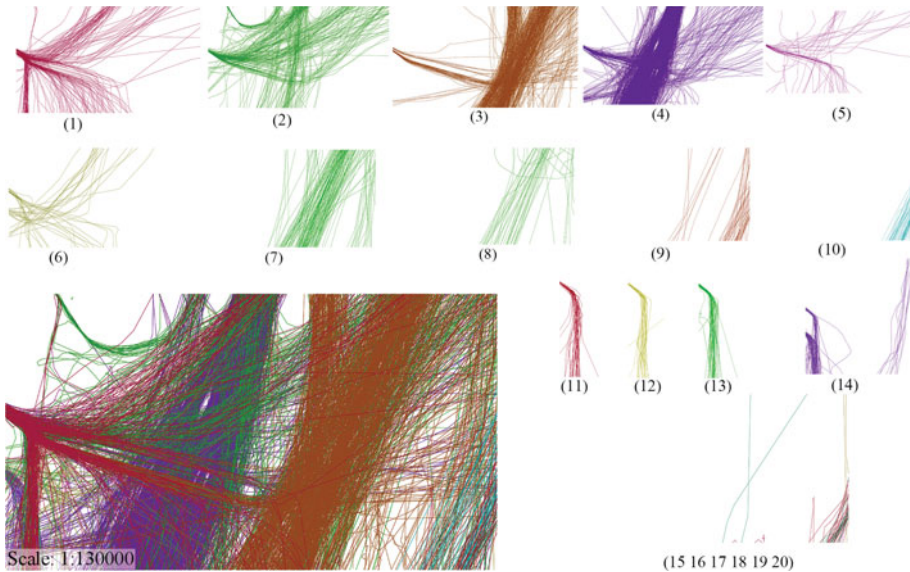


Figure 8. Results of classification-based clustering.

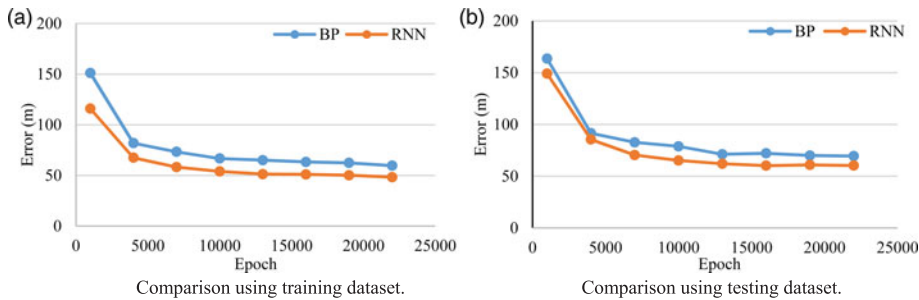
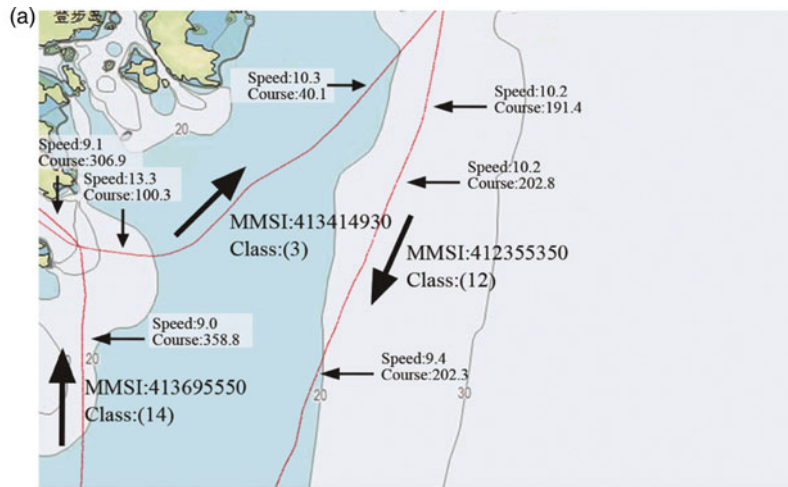


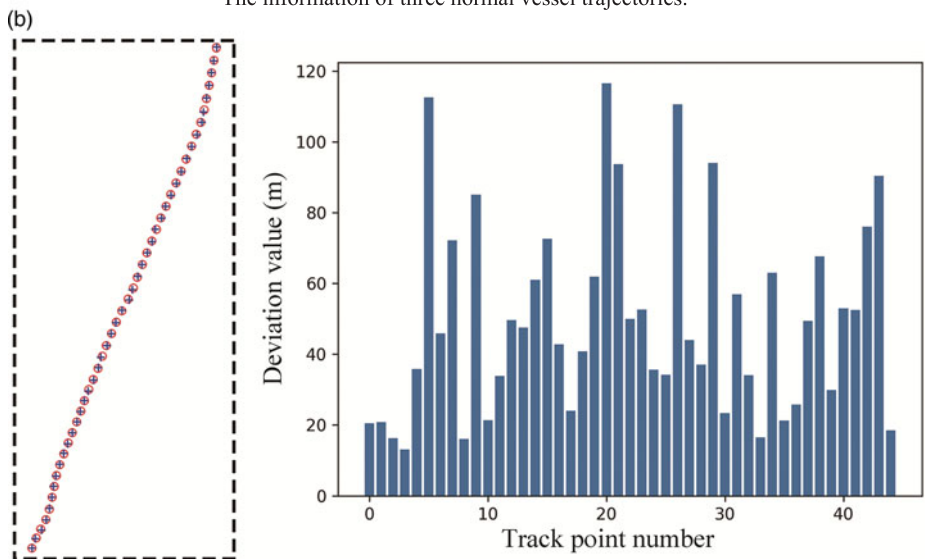
Figure 9. Comparison results of prediction error between BP neural network and recurrent neural network.

might influence the traffic flow through the waterway. The detailed predictive results are shown in Figure 12(b). From Figure 12(b), in the area marked with ‘(1)’ and ‘(3)’, the sudden increase in deviation value was caused by the abnormal change in speed and course. Moreover, in the area where the vessel abnormally stopped (marked with ‘(2)’), the tracking points that have larger deviation values were detected for the warning state. The detailed detection results are shown in Table 3. The deviation between the actual and predictive positions that is caused by the abnormal change in speed exceeds the threshold value, which is identified as anomalous vessel behaviour.

In Figure 13(a), the trajectory is generated for a vessel that sails from the Fuli gate waterway to the water area in the south. The first half of the vessel trajectory is a route that is not popular in this water area. The second half of the vessel trajectory follows the existing traffic pattern shown by the cluster (class 6) in the DBSCAN results. The detailed



The information of three normal vessel trajectories.

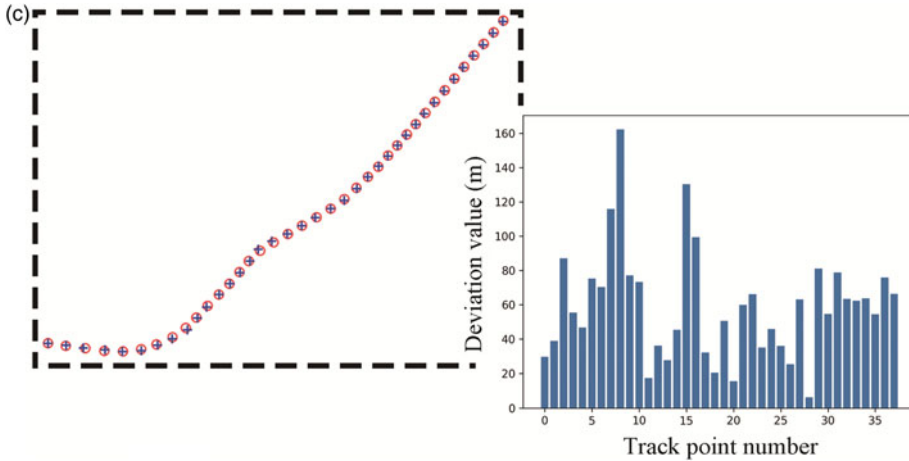


The position in the prediction result and the deviation value between actual position and actual position (MMSI: 413414930). Circle represents the actual position of the vessel, '+' represents the predictive position of the vessel. The bar chart shows all the deviation value of tracking points.

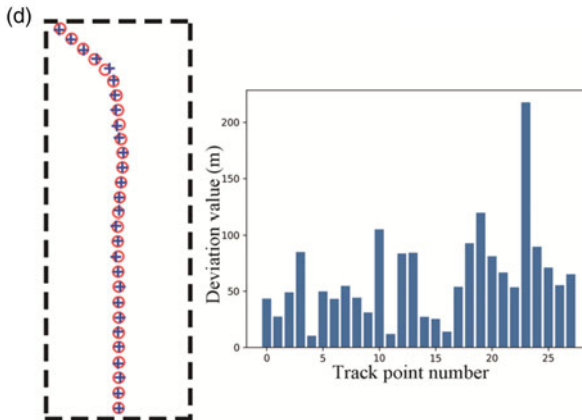
Figure 10. Continued.

predictive results of the first half (marked with a black block) are shown in Figure 13(b). Due to the abnormal route, the deviation values of the tracking point in this region are all relatively large. Those tracking points that exceed the low level consecutively have been detected by our method (marked with a block in Figure 13(b)).

This experiment shows that our method has the ability to detect the anomalous behaviours of a vessel in terms of speed, course and route within a short time.



The position in the prediction result and the deviation value between actual position and actual position (MMSI: 412355350).

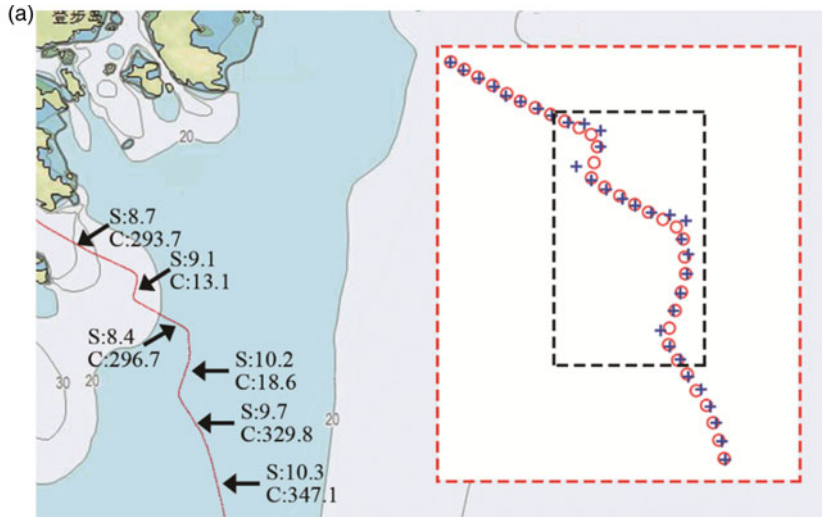


The position in the prediction result and the deviation value between actual position and actual position (MMSI: 413695550).

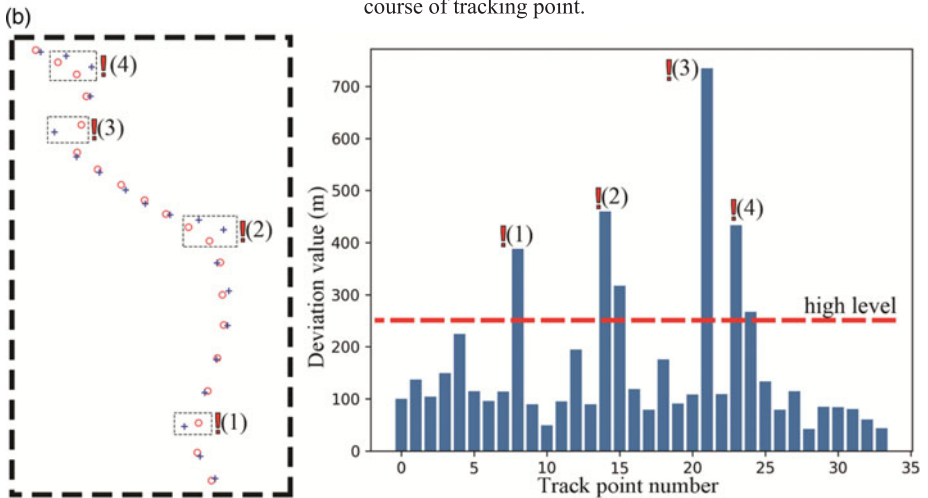
Figure 10. Three detailed results of prediction and deviation value between actual position and actual position.

5. CONCLUSION. As discussed in this paper, maritime anomaly detection in maritime surveillance can improve the situational awareness of vessel traffic supervisors and reduce casualties and losses of goods caused by maritime accidents. To detect anomalous vessel behaviours, a method consisting of pattern recognition by the DBSCAN algorithm and anomaly detection using a recurrent neural network is proposed.

In this method, the parameters of the DBSCAN algorithm are determined based on statistical analysis and subsequently use the vessel trajectories from the results of DBSCAN clustering as the training data to train the recurrent neural network built in the LSTM unit. Moreover, the network that has learned the normalcy of water traffic in this area



A Vessel trajectory with abnormal course. ‘S’ represents the speed of the tracking point, ‘C’ represents the course of tracking point.

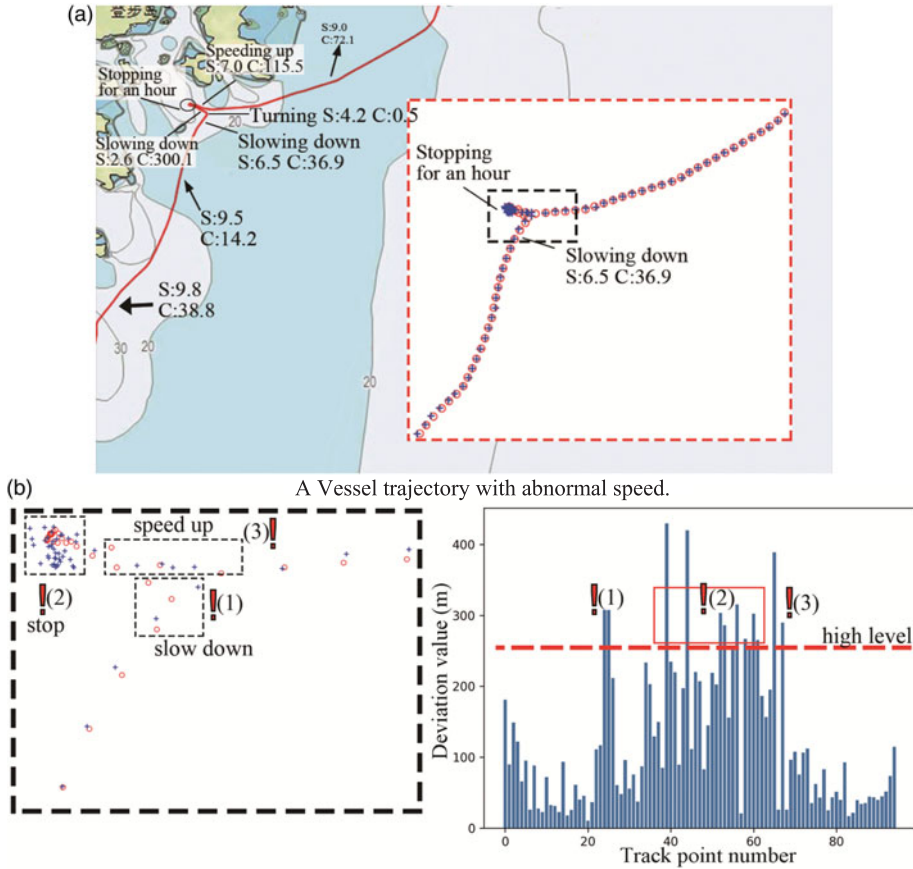


Detection of abnormal tracking points through the deviation value between actual position and predictive position. Circle represents the actual position of the vessel, ‘+’ represents the predictive position of the vessel. The exclamation mark represents the warning.

Figure 11. Detailed prediction result of vessel trajectory (abnormal course).

is applied as the vessel trajectory predictor to detect anomalous vessel behaviours in real time.

Based on a month of AIS trajectory data collected from the area of the Zhoushan Islands, a verification experiment was conducted. The experimental result shows that the method has the ability to detect anomalous behaviours of a vessel in terms of speed, course and route and can supply forewarning within a short time, which might aid the supervisor in decision-making.



Detection of abnormal tracking points through the deviation value between actual position and predictive position.

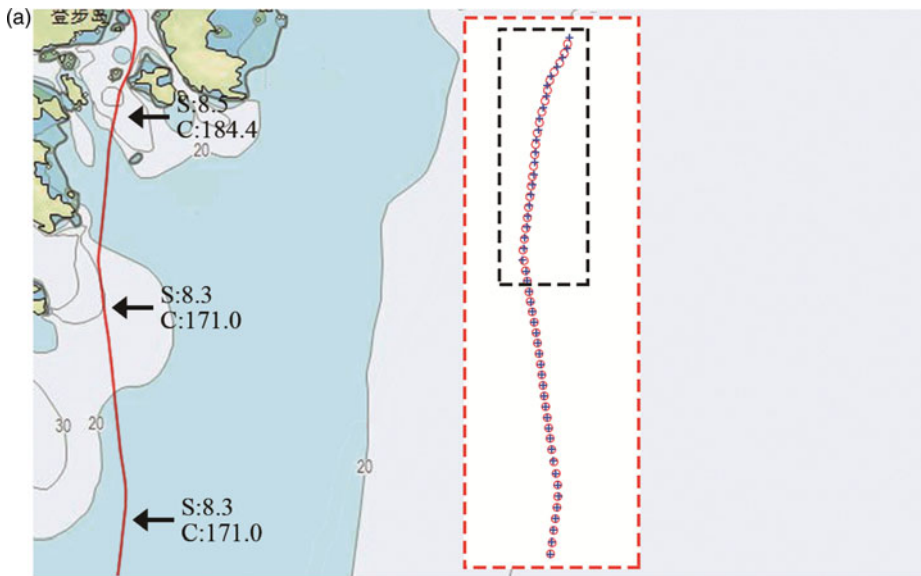
Figure 12. Detailed prediction and detection result of vessel trajectory (abnormal speed).

Table 2. Detailed detection result of vessel trajectory (abnormal course).

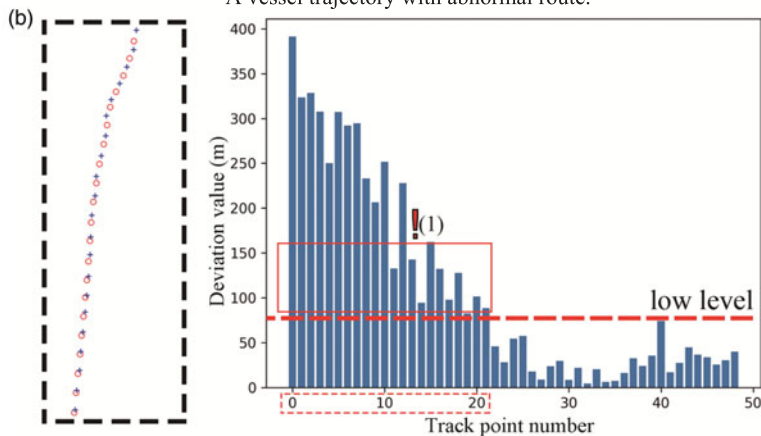
Track point number	Course (degree)	Deviation of vessel position(m)	Detection result	Track point number	Course (degree)	Deviation of vessel position(m)	Detection result
6	326.1	96.0		16	297.0	118.9	
7	337.2	113.9		17	298.8	79.2	
8	15.7	387.8	Warning (1)	18	298.8	175.5	
9	21.3	89.1		19	298.7	91.2	
10	22.6	49.4		20	343.7	108.6	
11	2.2	95.3		21	18.5	735.0	Warning (3)
12	352.3	194.9		22	6.6	109.1	
13	1.6	89.6		23	311.9	433.3	Warning (4)
14	306.9	459.9	Warning (2)	24	295.5	266.7	Warning (4)
15	296.3	317.1	Warning (2)	25	292.5	133.3	

Table 3. Detailed detection result of vessel trajectory (abnormal course).

Track point number	Speed (kn)	Deviation of vessel position (m)	Detection result	Track point number	Speed (kn)	Deviation of vessel position (m)	Detection result
20	9.1	10.5		30	0.7	55.7	
21	9.4	36.7		31–63	<1	exceed repeatedly	Warning (2)
22	9.4	110.9		64	3.7	195.2	
23	6.8	116.6		65	7.0	389.2	Warning (3)
24	4.0	307.5	Warning (1)	66	8.0	25.7	
25	4.7	307.5	Warning (1)	67	8.7	289.9	Warning (3)
26	4.7	211.8		68	8.7	26.0	
27	3.0	60.3		69	8.8	96.3	
28	2.3	47.8		70	8.8	107.8	
29	0.9	95.7					



A vessel trajectory with abnormal route.



Detection of abnormal tracking points through the deviation value between actual position and predictive position.

Figure 13. Detailed prediction and detection result of vessel trajectory (abnormal route).

However, the method has a high requirement on the quality of sample vessel trajectory data, which makes practical application more complicated. The tasks such as data collection, pre-processing extensive data and similarity measurement may directly affect the efficiency and effect of this method. For the traffic patterns of different vessel types or period, the particular normal model should be built based on a particular AIS track dataset. In this paper, the normal model has only used the AIS messages from tankers and cargo ships; for shorter vessels such as fishing vessels, the RNN used here may not have effective results. Future research will focus on improving the performance of the DBSCAN algorithm and long-term prediction.

FINANCIAL SUPPORT

This work was partly supported by “National Natural Science Foundation of China” (grant number: 51579025).

REFERENCES

- Ankerst, M., Breunig, M.M., Kriegel, H.P. and Sander, J. (1999). OPTICS: Ordering points to identify the clustering structure. *The 1999 ACM SIGMOD International Conference on Management of Data, Philadelphia, USA*, 49–60.
- Bomberger, N.A., Rhodes, B.J., Seibert, M. and Waxman, A.M. Associative learning of vessel motion patterns for maritime situation awareness. (2006). *The 2006 9th International Conference on Information Fusion, Florence, Italy*, 1–8.
- Carpenter, G. A., Grossberg, S., Markuzon, N., Reynolds, J. H., and Rosen, D. B. (1992). Fuzzy artmap: a neural network architecture for incremental supervised learning of analog multidimensional maps. *IEEE Transactions on Neural Networks*, 3(5), 698–713.
- Daranda, A. (2016). Neural Network Approach to Predict Marine Traffic. *Transactions in Baltic Journal of Modern Computing*, 4(3), 483.
- Ester, M., Kriegel, H.P., Sander, J. and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of the KDD, *Portland, Oregon*, 226–231.
- Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735–1780.
- Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Laxhammar, R. (2008). Anomaly detection for sea surveillance. *The 11th International Conference on Information Fusion, Cologne, Germany*, 55–62.
- Li, H., Liu, J., Liu, R. W., Xiong, N., Wu, K. and Kim, T. H. (2017). A Dimensionality Reduction-Based Multi-Step Clustering Method for Robust Vessel Trajectory Analysis. *Sensors*, 17(8), 1792.
- Liu, B., de Souza, E. N., Matwin, S. and Sydow, M. (2014). Knowledge-based clustering of ship trajectories using density-based approach. *IEEE International Conference on Big Data, Washington, USA*, 603–608.
- Lipton, Z. C., Berkowitz, J. and Elkan, C. (2015). A critical review of recurrent neural networks for sequence learning. *arXiv preprint arXiv:1506.00019*.
- Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted Boltzmann machines. *The 27th international conference on machine learning, Haifa, Israel*, 807–814.
- Pallotta, G., Vespe, M. and Bryan, K. (2013). Vessel pattern knowledge discovery from ais data: a framework for anomaly detection and route prediction. *Entropy*, 15(6), 2218–2245.
- Pan, J., Jiang, Q. and Shao, Z. (2014). Trajectory clustering by sampling and density. *Marine Technology Society Journal*, 48(6), 74–85.
- Perera, L. P. and Soares, C. G. (2010). Ocean vessel trajectory estimation and prediction based on extended kalman filter. *The Second International Conference on Adaptive and Self-Adaptive Systems and Applications, Lisbon, Portugal*, 14–20.
- Rhodes, B.J., Bomberger, N.A., Seibert, M. and Waxman, A.M. (2005). Maritime situation monitoring and awareness using learning mechanisms. *The 2005 Military Communications Conference, Atlantic City, USA*, 646–652.

- Rhodes, B.J., Bomberger, N.A. and Zandipour, M. (2007). Probabilistic associative learning of vessel motion patterns at multiple spatial scales for maritime situation awareness. *The 10th International Conference on Information Fusion, Quebec, Canada*, 1–8.
- Ristic, B., La Scala, B., Morelande, M. and Gordon, N. (2008). Statistical analysis of motion patterns in AIS data: Anomaly detection and motion prediction. *The 11th International Conference on Information Fusion, Cologne, Germany*, 1–7.
- Riveiro, M., Pallotta, G. and Vespe, M. (2018). Maritime anomaly detection: A review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, e1266.
- Riveiro, M. (2011). Visual analytics for maritime anomaly detection, Doctoral dissertation, Örebro universitet.
- Sidibé, A. and Shu, G. (2017). Study of automatic anomalous behaviour detection techniques for maritime vessels. *The Journal of Navigation*, **70**(4), 847–858.
- Stateczny, A. and Kazimierski, W. (2011). Multisensor tracking of marine targets - decentralized fusion of kalman and neural filters. *International Journal of Electronics & Telecommunications*, **57**(1), 65–70.
- Wiersma, J. W. F. (2010). *Assessing vessel traffic service operator situation awareness*. Doctoral dissertation, Delft University of Technology.
- Xu, T., Liu, X. and Yang, X. (2011). Ship Trajectory online prediction based on BP neural network algorithm. *IEEE International Conference on Information Technology, Computer Engineering and Management Sciences, Nanjing, China*, 103–106.
- Yan, W., Wen, R., Zhang, A. N. and Yang, D. (2016). Vessel movement analysis and pattern discovery using density-based clustering approach. *IEEE International Conference on Big Data, Washington, USA*, 3798–3806.
- Zhao, L., Shi, G. and Yang, J. (2018). Ship Trajectories Pre-processing Based on AIS Data. *The Journal of Navigation*, **71**(5), 1210–1230. doi:10.1017/S0373463318000188
- Zhen, R., Jin, Y., Hu, Q., Shao, Z. and Nikitas, N. (2017a). Maritime anomaly detection within coastal waters based on vessel trajectory clustering and naïve Bayes classifier. *The Journal of Navigation*, **70**(3), 648–670.
- Zhen, R., Riveiro, M. and Jin, Y. (2017b). A novel analytic framework of real-time multi-vessel collision risk assessment for maritime traffic surveillance. *Ocean Engineering*, **145**, 492–501.