Learning Collision Avoidance of Ship Manoeuvring based on Gated Recurrent Unit

Wenli Sun\(^1\)\(^*\) and Xu Gao\(^2\)

\(^1\) Navigation College, Dalian Maritime University, Science Hall, room 611, 1 Linghai Road, Dalian, 116026, China
\(^2\) National Engineering Research Center of Maritime Navigation System, Dalian Maritime University, Science Hall, room 517, 1 Linghai Road, Dalian, 116026, China
*Email: itslab@dlmu.edu.cn

Abstract. Unmanned surface vehicle (USV) has progressed quickly in recent decades, with widespread research and practical applications in academic and industry circles. Collision avoidance is a fundamental capability of USVs. It is extremely challenging to develop an ideal and sophisticated collision avoidance algorithm for USV in complex environments and practical offshore situations. However, a supervised learning method provides the USV a way of learning the process of avoidance and imitating a human pilot maneuvering to navigate in the real environment. This study analyzes the relative relationships and features of the own and target ships in the avoidance process firstly. And then a Gated Recurrent Unit (GRU) recurrent neural network model is constructed. Maneuvering commands during avoidance by human pilots are utilized as tags for training. Finally, the validity of the method is proven by performing navigation experiments. In particular, we also compare the effectiveness of GRU with Long Short-Term Memory (LSTM) network which is also a kind of recurrent neural network. The experimental results indicate that the proposed GRU model is better than the LSTM model, and the USV can autonomously navigate during the collision avoidance process by using the well-trained GRU model, reaching a level similar to that of human pilots.

1. Introduction
In recent decades, there have been tremendous advances in unmanned systems, whether on land, in the air, at sea or underwater. This is inseparable from the breakthrough development of artificial intelligence technology [1]. Unmanned surface vehicles (USVs) are autonomous robots at sea, with the advantages of high speed, small size, high flexibility and low cost. Therefore, it is suitable for completing tasks such as waterway surveys and marine environmental inspections [2]. A reliable and safe navigation system is a fundamental capability of USV. Therefore, it is crucial to achieve autonomous collision avoidance [3].

When a manned ship avoids a collision, the human pilot firstly maintains a regular lookout by sight, hears, and all available means appropriate to the circumstances, and makes a sufficient assessment of the situation and the risk of collision, subsequently taking avoidance action until the situation is clear [4]. Analogous to the manned ship, the collision avoidance process of USV can be divided into three sub-processes: the pre-avoidance phase, the action of avoidance, and the post-avoidance phase. In the pre-avoidance phase, various types of sensors, such as Radar, LIDAR, white-light/infrared cameras, are employed to identify and detect obstacles, determine if the ship is in danger, and trigger the avoidance action [5]. During avoidance, the ship is controlled to steer away from the planned route for
avoiding possible collisions [2]. When sailing past the obstacle clearly, ships are controlled back to the planned route [6]. Among the three sub-processes, the action of avoidance is the core step to autonomous collision avoidance of USV [7]. This study focuses on this central question and therefore makes assumptions about the pre and post phases, i.e., the observing equipment will be able to provide accurate information, determine the threat condition, and return to the planned route by trajectory tracking.

The USV takes avoidance actions in accordance with a number of information such as maps, obstacles, own ship and so on. Traditionally, rule-based approaches existed, with a simple ”if-then” rule for avoidance [8]. Lyu et al. devised the Artificial Potential Field (APF) method for ship collision avoidance, where the APF generates repelling potential fields around obstacles and attractive potential fields at the target, and the sum of these potential fields determines the combined force of the ship motion, giving the direction [9]. Many scholars have transformed the collision avoidance problem into the optimization problem with good achievements. Johansen et al. selected course and speed as inputs and denoted the distance between the trajectory and the obstacle by a cost function. Then, they used optimization to minimize the cost function and thus find the collision-free solution [10]. Zhuang et al. used the velocity obstacle method, where the obstacle was first viewed as a circle. Then, a velocity vector relationship between the ship and the obstacle is constructed. Then, the vector relationship is used to define the optimization goal of collision avoidance. Finally, the optimization method is used to find the optimal solution [11]. However, none of these methods considered the avoidance habits of professional pilots, and a great deal of simplification and constraint was applied.

Artificial intelligence approaches with supervised learning can take advantage of the enormous data of human pilot for learning avoidance maneuver actions. A human pilot already intuitively predicts many aspects when maneuvering a ship for collision avoidance, such as the encounter situation, the collision risk, and the rules [12]. If the USV neglects these behavior habits, it may still cause collision damage. Consequently, collision avoidance actions by USV require a human-like approach. Identification and exploitation of the time-series features in which human pilots perform avoidance actions is the key to supervised learning of collision avoidance actions.

Among deep learning methods, Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) networks are widely used to recognize changing patterns in time-series data [13,14]. Gated Recurrent Unit (GRU) is a variant of LSTM, which was proposed by Cho et al. in 2014 [15]. GRU is a simplification of LSTM, but kept the same effect [16]. GRU has been widely used for complex time series prediction by imposing a gated mechanism to solve the RNN vanishing gradient problem, especially the decrease of correlation with increasing time steps in the time series. In this study, a GRU-based deep learning approach is adopted, which firstly learns the data from the human pilot maneuvering for collision avoidance actions in a supervised manner, and then applies the trained GRU network to predict the desired course based on the avoidance factors.

The main contributions of this study are as follows.

(1) A supervised learning model is constructed to recognize and learn the relative motion relationship between the own ship and the target ship, and to predict the course orders in real-time. Simulation results show that the model attained an avoidance capability analogous to that of a human pilot.

(2) GRU is applied to the feature extraction of the avoidance time-series data. Compared with the LSTM model, GRU can distinguish the relationship between the relative avoidance factors of the own ship and the target ship in the avoidance process better.

The other sections are organized as follows: Section 2 describes the problem and clarifies the key factors of the collision avoidance process. Section 3 provides a detailed explanation of the principle of GRU and its usage. Section 4 gives the training effects of GRU and LSTM, and compares the experimental effects of manned and unmanned ship collision avoidance based on GRU and LSTM models. Section 5 concludes the paper.
2. Analysis of Avoidance Factors

The primary issue in ship collision avoidance research is safe avoidance action, the key is how to take appropriate avoidance actions. It is very significant to understand the factors in the avoidance process, for proper avoidance measures to be taken. The avoidance factors are represented in two modes: absolute motion and relative motion. Absolute motion is the motion of a moving point with respect to fixed coordinates, and is the only motion that exists objectively. Relative motion is the motion of a moving point with respect to dynamic coordinates, and is an analytical method for solving complex motion problems in engineering. On manned ships, human pilots routinely configure the marine radar to a relative motion display mode because it is so intuitive and identical to what the human pilot would actually see if he looked out from the bridge. This study aims to mimic the avoidance manoeuvring of human pilot, and hence focuses on the avoidance factors in the relative motion patterns between two ships.

The avoidance factors include: relative speed and course, relative distance and relative bearing, distance to closest point of approach (DCPA), and time to closest point of approach (TCPA) [17-20]. Among them, the ship speed and course are the parameters reflecting the movement state, the relative distance and bearing are the parameters reflecting the variation of the position relationship. DCPA is a measure of whether two ships will cause a collision, and the TCPA is a basis for determining the potential collision risk of two ships [21]. This is explained in detail below.

2.1. Relative Speed and Relative Course

Relative motion of a ship is the movement of another ship or target as observed on a moving ship. As depicted in Figure 1, \( \vec{V}_o \) and \( \vec{V}_t \) is the velocity vector of the motion of the own ship (O) and the target ship (T), respectively. Following the vector triangle, \( \vec{V}_r \) is the relative velocity vector of the target ship relative to the own ship, i.e., Equation (1).

\[
\vec{V}_r = \vec{V}_t - \vec{V}_o
\]  

![Figure 1](image-url) 

**Figure 1.** Relative velocity vector relationship between the own ship and the target ship.

2.2. Relative Range and Rate of Variation in Relative Distance

The positions of own ship and the target ship are defined in the X-Y coordinate system, in meters. The position of own ship is \( (o_x, o_y) \), and the target ship is \( (t_x, t_y) \). The relative range is as illustrated in Equation (2).
range = \sqrt{(o_x - t_x)^2 + (o_y - t_y)^2} \quad (2)

The rate of variation in relative distance \( ror \) is the derivative of the relative distance \( range \) to time \( t \), as given in Equation (3).

\[ ror = \frac{d(range)}{dt} \quad (3) \]

2.3. Relative Bearing and Rate of Variation in Relative Bearing

As indicated in Figure 2, the relative bearing \( \beta \) of the two ships indicates the difference in course between the own ship and the target ship, as shown in Equation (4).

\[ \beta = |t_C - o_C|^{360^\circ} \quad (4) \]

where \(|\cdot|^{360^\circ}\) represents modulus operation by 360 of relative bearing. \( o_C \) and \( t_C \) means the course of the target and the own ship.

The rate of variation in relative bearing \( rob \) is the derivative of the relative bearing \( \beta \) to time \( t \), as given in in Equation (5).

\[ rob = \frac{d\beta}{dt} \quad (5) \]

2.4. DCPA and TCPA

In this study, we used the calculation of DCPA and TCPA following the method proposed by Benjamin et al. [22]. Firstly, the position-updated triangles are constructed for the own ship and the target ship, respectively, as shown in Figure 3. Then, the distance is converted to a function with time \( t \) as the invariant. Finally, the time \( t \) is calculated at which the minimum distance occurs by making the first-order derivative of the function zero.

The update of own ship and target position can be expressed as Equations (6)-(9), where \( o_S \) and \( t_S \) are the speed of the own ship and the target ship.

\[ o_x^{\text{new}} = \sin(o_C) \cdot o_S \cdot t + o_x^{\text{old}} \quad (6) \]
\[ o_y^{\text{new}} = \cos(o_C) \cdot o_S \cdot t + o_y^{\text{old}} \quad (7) \]
\[ t_x^{\text{new}} = \sin(t_C) \cdot t_S \cdot t + t_x^{\text{old}} \quad (8) \]
\[ t_y^{\text{new}} = \cos(t_C) \cdot t_S \cdot t + t_y^{\text{old}} \quad (9) \]

The distance between two ships can be expressed as equation (10), and it can convert to a function of \( t \) as Equation (11). For details about the meaning of \( k_2, k_1, k_0 \), refer to [22].

\[ \text{dist}^2 = (o_x^{\text{new}} - t_x^{\text{new}})^2 + (o_y^{\text{new}} - t_y^{\text{new}})^2 \quad (10) \]
\[ \text{dist}(t)^2 = k_2 t^2 + k_1 t + k_0 \quad (11) \]

For Equation (11), the first-order derivative of \( t \) is calculated to obtain Equation (12).
Let \(2k_2 t + k_1 = 0\), the time \(t\) of the minimum distance is obtained, as shown in Equation (13). This is the value of TCPA. And then put it into Equation (11) to obtain the DCPA.

\[
t = -\frac{k_1}{2k_2}
\]

3. GRU-based Time Series Prediction Model for Collision Avoidance Process

3.1. Application of GRU

Because collision avoidance is a time-series process, it contains complex temporal information, and there is not direct linear relationship between the features and the avoidance action. GRU is very beneficial for finding the patterns and features of a human pilot maneuvering a boat to avoid collision. The trained GRU can reveal the correlation between the maneuvering actions and the information about the own ship and the target ship at different times during the avoidance process.

Figure 4 shows the GRU network built for the USV collision avoidance time series data. It is a 3-layer structure consisting of an input layer, a GRU layer, and an output layer. Firstly, the avoidance factors are extracted from the geometric relationship between the relative positions of two ships at each moment, as analyzed in Section 2, to constitute a time series. For the feature vectors \( \mathbf{x}_t = (x_{t1}, x_{t2}, x_{t3}, x_{t4}, x_{t5}, x_{t6}, x_{t7}, x_{t8}, x_{t9}, x_{t10}, x_{t11}) \) of moment \( t \), they correspond to the relative position, relative distance, TCPA, DCPA, relative speed, relative course, rate of variation in relative distance and bearing, desired speed, and desired course, respectively. Then, the different moments of series data are fed to the input layer separately. The trained GRU is able to learn the relationship between the information related to the own ship and the target ship at the first few moments of the avoidance process and the course instruction to take avoidance action at the next moment.

GRU is a variant of LSTM that is suitable for processing and predicting time series data due to its internal hidden layer nodes with feedback connections [23]. GRU can exploit the deep expression of temporal information to find the relationship between the current output and previous information [24]. GRU combines the forget gate and input gate in LSTM into a single update gate, and merges the cell state with the hidden layer state. There are two gates in GRU: the update gate and the reset gate. The update gate defines the effect of the relative motion relationship between the two ships in the previous moment on the current moment avoidance maneuver. The reset gate determines how to combine the updated information on the relative motion of the two ships with the previous effects. Equations (14)-(19) show the forward propagation process of the GRU, where \( U_z, U_r, U_c \) illustrate the weight matrix of each gate unit in the input and hidden layers, respectively, \( W_z, W_r, W_c \) are the weight matrices of the feedback connections of each gate unit in the hidden layer, respectively. \( V \) is denoted as the weight matrix from the hidden layer to the output layer. \( \mathbf{h}_t \) and \( \mathbf{\tilde{h}}_t \) are denoted as the state of hidden layer and the candidate hidden state, \( \mathbf{z}_t \) and \( \mathbf{r}_t \) are the updated gate and the reset gate, \( \text{sigmoid} \) and \( \tanh \) are the activation function of sigmoid and hyperbolic tangent. \( \mathbf{y}_t \) denotes the output. The MSE is used as the loss function (as shown in Equations (19)) and a stochastic gradient descent algorithm is used to minimize the loss function.

\[
\mathbf{h}_t = (1 - \mathbf{z}_t)\mathbf{h}_{t-1} + \mathbf{z}_t\mathbf{\tilde{h}}_t
\]

\[
\mathbf{z}_t = \sigma(\mathbf{x}_t U_z + \mathbf{h}_{t-1} W_z)
\]

\[
\mathbf{\tilde{h}}_t = \tanh(\mathbf{x}_t U_c + \mathbf{h}_{t-1} r W_c)
\]

\[
\mathbf{r}_t = \sigma(\mathbf{x}_t U_r + \mathbf{h}_{t-1} W_r)
\]

\[
\mathbf{y}_t = \sigma(\mathbf{h}_t V)
\]

\[
L = \frac{1}{2T} \sum_{t=1}^{T} (\mathbf{y}_t - \mathbf{\hat{y}}_t)^2
\]
3.2. Avoidance Action Dataset

The avoidance action dataset is produced as follows: first we deploy a target ship and then plan a set of straight routes with different initial relative orientations. We then have a human pilot remotely control the ship at different speeds (0.6m/s to 5m/s, with 0.2m/s intervals) and different initial relative orientations (0 degree to 30 degree, with 3 degree intervals) and perform avoidance actions, as shown in Figure 5.

![Desired Course](image)

Figure 4. Prediction process of GRU-based for USV collision avoidance time series data.

3.3. Model Training and Evaluation

Training of the model is performed on a computer configured with two NVIDIA TITAN RTX GPUs. The dataset has a total of 63,200 samples, which we divided into training, validation, and test sets in an 8:1:1 ratio. The learning rate, the batch-size and iterations are 0.001, 256, 1000, respectively.

Finally, we evaluate the model on the test set by using Root Mean Square Error (RMSE) and R-Square, as illustrated in (20) and (21). RMSE gives the magnitude of the model prediction error. R-Square indicates the degree to whether the model is good or bad, with 1 indicating the best and 0 indicating the worst.
\[ RMSE = \sqrt{\frac{1}{T} \sum_{t=1}^{T} (y_t - \hat{y}_t)^2} \]  
\[ R^2 = 1 - \frac{\sum_{t=1}^{T} (y_t - \hat{y}_t)^2}{\sum_{t=1}^{T} (y_t - \bar{y}_t)^2} \]  

Figure 5. Navigational scenario for the avoidance action dataset.

Figure 6. Comparison of LSTM and GRU training effects.

4. Simulation Results

In this section, there are two aspects from which we describe the experimental results of this study. On the one hand, a comparison between LSTM and GRU is given. On the other hand, a simulation experiment of a head-on encounter scenario is devised, where USV uses GRU-based and LSTM-based model for real-time avoidance, comparing their navigation process with human pilot in maneuvering.

4.1. The Effect of Training Models

Figure 6 shows a comparison of the training effects of the two models, GRU and LSTM (the green line indicates the training effect of LSTM and the yellow line indicates the training effect of GRU), with GRU being slightly better than LSTM.

In addition, we evaluated the training effect of the model on the validation and test sets, and the results are presented in Table 1. The results show that the GRU model is effective.

| Neural Network | On Validation Set | On Test Set |
|----------------|-------------------|-------------|
|                | RMSE(°) | $R^2$ (%) | RMSE(°) | $R^2$ (%) |
| GRU            | 1.073   | 99.850   | 2.009   | 98.935   |
| LSTM           | 1.186   | 99.743   | 2.175   | 98.923   |

4.2. The Effect of Collision Avoidance

In the experiments, we utilize a trained GRU network and deploy it into the control system of USV. GRU collects previous relative position, relative distance, TCPA, DCPA, rate of variation in relative distance, rate of variation in relative bearing, speed, and course data to calculate the desired course at the next moment in real time and send it to the controller module, which in turn controls the USV navigation. At the end of the navigation experiment, we compare the effect of the navigation track on avoidance performed by the models and the human pilot, and analyze the effect.

This study focuses on the avoidance process and hence it is set up that both modes start the avoidance action at 20 meters away from the target vehicle. Figure 7 reflects the results in the head-on
encounter scenario. The red line and green dashed line represent the results of the trajectory of USV by means of GRU and LSTM, and the blue line represents the human pilot.

The experimental results show that the USV is close to the avoidance effect of the human pilot and can steadily avoid the target ship, with the same avoidance distance as the human pilot, both about 10m. This shows that the avoidance action of USV using the GRU model and LSTM model, which is in line with the human pilot's understanding of safe distances, is both safe and effective, and is a manifestation of good seamanship.

Figure 7. Comparison of voyage tracks for avoidance by USV and human pilots.

Figure 8 gives the variation of the desired course instructions given to the controller during the navigation of the unmanned ship and the human pilot. The red solid line is the desired course predicted using the GRU model, the green dashed line is the desired course predicted using the LSTM model, and the blue dotted line is the desired course issued by the human pilot while navigating the ship. It can be seen that the trend of the desired course is the same for all three models, which indicates that the supervised learning method based on the GRU model is reliable in learning the time-series characteristics of the human pilot maneuvering the ship for avoidance. In addition, the GRU model predicts a smoother course command curve, which is better than the LSTM model.

Figure 8. Variation of the desired course instruction.

5. Conclusions
This study addresses the problem of how to take avoidance actions during USV collision avoidance through a GRU-based deep learning method. According to the relative motion relationship between
two ships at the previous moment, the GRU model can predict the course instruction during avoidance process. Simulation show that the GRU model is superior to the LSTM model, and it can learn the avoidance effect of human pilot ship maneuvering. The avoidance distance and maneuvering instructions are very close to those of human pilot. In the future, this algorithm can be further extended to the avoidance process of USV in more complex scenarios such as restricted waters, emergency situations, and compliance with collision avoidance rules.

References
[1] Zhou Z 2016 Machine Learning (Tsinghua University Press, Beijing)
[2] Liu Z X, Zhang Y M, Yu X and Chi Y 2016 Unmanned surface vehicles: an overview of developments and challenges Annual Reviews in Control 123
[3] Huang Y M, Chen L Y, Chen P F, Negenborn R R and van Gelder P H A J M 2020 Ship collision avoidance methods: state-of-the-art Safety Science 121 451-473
[4] Szlapczynski R, Krata P and Szlapczynska J 2018 Ship domain applied to determining distances for collision avoidance manoeuvres in give-way situations Ocean Eng 165 43-54
[5] Kuwata Y, Wolf M T, Zarzhitsky D and Huntsberger T L 2014 Safe maritime autonomous navigation with COLREGS using velocity obstacles IEEE J. Oceanic Eng 39(1) 110–9
[6] Chen C, Chen X Q, Ma F, Zeng X J and Wang J 2019 A knowledge-free path planning approach for smart ships based on reinforcement learning Ocean Eng 189 106299
[7] Woo J and Kim N 2020 Collision avoidance for an unmanned surface vehicle using deep reinforcement learning Ocean Eng 199 107001
[8] Naeem W, Irwin GW and Yang A L 2012 COLREGS-based collision avoidance strategies for unmanned surface vehicles Mechatronics 22(6) 669-678
[9] Lyu H and Yin Y 2018 COLREGS-constrained real-time path planning for autonomous ships using modified artificial potential fields J. Navigation 72(3) 588-608
[10] Johansen T A, Perez T and Cristofaro A 2016 Ship collision avoidance and COLREGS compliance using simulation-based control behavior selection with predictive hazard assessment IEEE Trans. Intell. Transp. Syst 17(12) 3407-22
[11] Zhuang J Y, Zhang L, Zhao S Q, Cao J, Wang B and Sun H B 2016 Radar-based collision avoidance for unmanned surface vehicles China Ocean Eng 30(6) 867-83
[12] Tam C, Bucknall R and Greig A 2009 Review of collision avoidance and path planning methods for ships in close range encounters J. Navigation 62(3) 455-76
[13] Cho K, Van Merrinboer B and Gulcehre C 2014 Learning phrase representations using RNN encoder-decoder for statistical machine translation (arXiv preprint) arXiv: 1406.1078
[14] Hochreiter S and Schmidhuber J 1997 Long Short-term Memory Neural Computation 9(8) 1735-80
[15] Cho, Kyunghyun; et al. 2014 Learning Phrase Representations Using RNN Encoder-decoder for Statistical Machine Translation (arXiv preprint) arXiv: 1406.1078
[16] Wang X, Jiang R, Li L, Lin Y, Zheng X and Wang F Y 2017 Capturing car-following behaviors by deep learning.” IEEE Trans. Intell. Transp. Syst 19 (3) 910–920
[17] Chen P, Huang Y, Mou J and van Gelder P H A J M 2019 Probabilistic risk analysis for ship-ship collision: state-of-the-art. Saf. Sci 117 108-122
[18] Li B, Pang F W 2013 An approach of vessel collision risk assessment based on the D-S evidence theory Ocean Eng 74 16-21
[19] Zhao Y X, Li W, Shi P 2016 A real-time collision avoidance learning system for Unmanned Surface Vessels Neurocomputing 182 255-266
[20] Gang L H, Wang Y H, Sun Y, Zhou L P and Zhang M G 2016 Estimation of vessel collision risk index based on support vector machine Adv. Mech. Eng 8 11
[21] Lisowski J 2002 Game control of moving objects IFAC Proc 35(1) 373–8
[22] Benjamin M R, Leonard J J, Curcio J A and Newman P M 2006 A method for protocol-based collision avoidance between autonomous marine surface craft J. Field Rob 23(5) 333-346
[23] Gang C, Shouhui W and Xiaobo X 2016 Review of spatiotemporal models for short-term traffic forecasting,” Proc. Int. Conf.on IEEE international conference on intelligent transportation engineering (Singapore) pp 8-12

[24] Dai G, Ma C and Xu X 2019 Short-term traffic flow prediction method for urban road sections based on space-time analysis and GRU IEEE Access 7 143025–35