



Risk Perception Oriented Autonomous Ship Navigation

Zhang, Ruolan

(Degree)

博士 (工学)

(Date of Degree)

2020-09-25

(Date of Publication)

2021-09-01

(Resource Type)

doctoral thesis

(Report Number)

甲第7892号

(URL)

<https://hdl.handle.net/20.500.14094/D1007892>

※ 当コンテンツは神戸大学の学術成果です。無断複製・不正使用等を禁じます。著作権法で認められている範囲内で、適切にご利用ください。



博 士 論 文

Risk Perception Oriented Autonomous Ship Navigation

(リスク認知指向の自律船舶ナビゲーションに関する研究)

2020 年 7 月

神戸大学大学院海事科学研究科

張 若 瀾

ZHANG RUOLAN

Risk Perception Oriented Autonomous Ship Navigation

(リスク認知指向の自律船舶ナビゲーションに関する研究)

Summary

Autonomous ship navigation will play an important role in the future of marine transportation systems. This dissertation proposal a human-in-the-loop decision-making conversion model and a human-out-of-the-loop risk perception model, provides the whole deep learning-based Generative Adversarial Network (GAN) model with risk scenario image dataset generation, presents an image-based situation awareness and a real Automatic Identification System (AIS) environment description, construct an autonomous collision avoidance decision system using deep reinforcement learning. Also approaches target detection with real navigation images/videos, and end-to-end deep reinforcement learning training with a real AIS navigation environment.

As shown in Fig.1.1, this study can be divided into two parts. The upper section is **data processing**, including AIS data and image / video data processing, using the data obtained by the sensors to identify obstacles and perceive the risk around the ship navigation environment. The lower section is **representation learning**, which is the autonomous decision-making process. Using the obtained data, it is abstracted into a suitable mathematical model and described as the environmental characteristics that the agent (own ship) can perceive. Approaches various deep reinforcement learning algorithms as the core to make risk perception and collision avoidance decisions: Use AIS data, combined with other data such as video, images, and radar, to approximately describe the environment of ship navigation, and use different evaluate methods to optimize decision performance. Our research results showed that the decision neural network trained with AIS data has good robustness and a high ability to collision avoidance or many other tasks. Furthermore, instructive guidance for processing visible camera, radar, VHF, and other training data types has been illustrated. It has important implications for the realization of fully autonomous unmanned navigation.

AIS data: using historical AIS data to extract geographic location information (Time, LAT, LON), encounter ships navigation state information (MMSI, SOG, COG, Heading, Vessel type, etc.), in the order of time flow, presented on the electronic chart, which can approximately describe the encounter state of the ship

navigation. Take the image (frame) of the electronic chart as the input of the decision neural network and train the neural network. In future research, the author will use the Gym-Gazebo package to describe the ship's navigation states more accurately and achieve better decision-making results.

Image / Video: The cost of using cameras to obtain ship navigation data is relatively low. Deep learning has made great achievements in the fields of image recognition and target detection. Image-based environmental perception will definitely play an important role in the future research of unmanned ships. In this study, image data from a real ship bridge perspective was used to label the target ship and obstacles, train a deep neural network, identify and perceive the ship environment. The results obtained were higher than traditional methods.

Training Processing and Vision Perception: This study uses images (frames) obtained by AIS data as input and trains a deep reinforcement learning network (Deep-Q Network) to achieve end-to-end decision output. Deep reinforcement learning requires strong computing power support. One of the goals of this research includes the training of end-to-end decision networks for autonomous ship navigation. More optimized input is necessary, so direct vision perception can help make faster decisions.

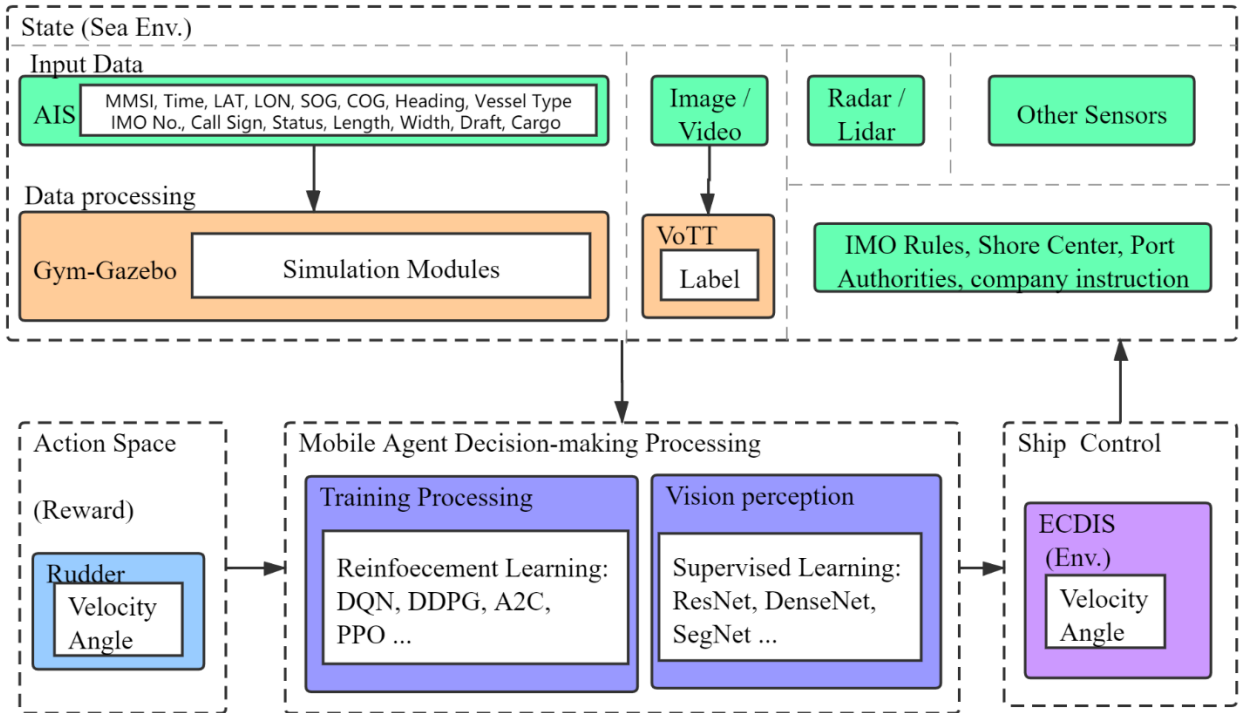


Fig.1.1 Doctoral research map

Contents

Summary	3
Chapter 1 Introduction	8
1.1 Background	8
1.1.1 Seafarer and navigation aids interaction	8
1.1.2 Different way to achieve collision avoidance	9
1.1.3 Modern approaches for risk perception	10
1.2 Objectives	11
1.3 Methodology	11
1.4 Dissertation Structure	12
1.5 References	17
Chapter 2 Human-in-the-loop decision and risk perception	19
2.1 Literature review and motivation	19
2.2 Autonomous decision-making level	21
2.2.1 The levels of autonomous decision-making	22
2.2.2 Markov Decision Process for target perception	23
2.2.3 Target perception and decision-making	25
2.3 Adversarial decision cycles	26
2.3.1 Unmanned ship decision cycles	26
2.3.2 Autonomous decision process training	27
2.4 Summary	28
2.5 References	28
Chapter 3 Theoretical foundations of reinforcement learning	30
3.1 Literature review and motivation	30
3.2 Mathematical Description of the Navigation State	31
3.3 Markov decision process	33
3.3.1 State Space	34
3.3.2 Action Set	35
3.3.3 Reward Function	36
3.3.4 Environment Formulation	36
3.4 Decision-making training and analysis	37
3.4.1 Simplified Decision Algorithm	37
3.4.2 Training example and its results	39
3.4.3 Steering action data analysis	40

3.5	Summary	40
3.6	References	41
Chapter 4	Extend dataset using generative adversarial network	42
4.1	Literature review and motivation	42
4.2	Generative adversarial network and its application	44
4.2.1	The principle of GAN	44
4.2.2	GAN application examples	45
4.2.3	Global optimality of discrete actions	46
4.3	GAN method for generating training data sets	47
4.3.1	Conventional acquisition of target ship data set method	47
4.3.2	Example of the GAN generate lifeboat image data	47
4.4	GAN with policy gradient for optimize discrete actions	54
4.5	Summary	56
4.6	References	56
Chapter 5	Approach autonomous decision in a real AIS environment	58
5.1	AIS data in collision avoidance.....	58
5.1.1	AIS Collision Avoidance Decision System.....	58
5.1.2	Rule and Neural Based Decision-Making System.....	59
5.1.3	Deep Reinforcement Learning for Collision Avoidance.....	60
5.2	AIS data processing.....	61
5.2.1	Traditional application of AIS.....	61
5.2.2	Segment Ship Domain with AIS Data	62
5.2.3	Frame Motion and Moving States Processing	63
5.3	Training decision neural network.....	67
5.3.1	Reinforcement Learning Architecture.....	67
5.3.2	Training Environment	67
5.3.3	Training Results	68
5.4	Discussion	70
5.5	Summary	70
5.6	References	71
Chapter 6	Ocean target detection for autonomous ship navigation	73
6.1	Introduction	73
6.2	Literature review	75
6.2.1	Ship visual perception tasks and motivations	75
6.2.2	Deep learning-based target detection framework	77
6.2.3	Current status of visual perception	78

6.3	Ocean target classification using a deep residual network.....	79
6.3.1	Vision based target detection process	79
6.3.2	Introduction to a deep residual network.....	80
6.3.3.	Comparison of ocean target classification tasks	81
6.4	Ocean tiny target detection experiment.....	83
6.4.1	Data set structure design and detection strategy	83
6.4.2	Bounding box strategy and IoU for long-distance tiny target.....	85
6.4.3	Model training using modified YOLO v3 framework	89
6.4.4	Target recognition model performance evaluation criterion.....	89
6.4.5	Training result verification and testing	90
6.5	Consideration	94
6.5.1	Ocean target detection in different scenarios	94
6.5.2	Improve end-to-end risk perception efficiency	97
6.6	Summary	98
6.7	References	99
Chapter 7	Conclusion	103
	Acknowledgement.....	105
	List of publication	106
	List of figures	108
	List of tables	109

Chapter 1

Introduction

1.1 Background

1.1.1 Seafarer and navigation aids interaction

Equipment on modern ships requires internal and external information that helps the Officer of the Watch (OOW) determine the current situation and make the appropriate decisions, enabling ships to reach their destinations safely and economically. The total shipborne equipment can be divided into two categories: equipment such as GPS, compass, and logs, which can sense the state of navigation of the ship; other devices such as radar, Automatic Identification System data (AIS), Very High Frequency communication (VHF), and Camera, which sense the surrounding environment and target the ship's state. All kinds of equipment have their own characteristics, and the OOW must consider several factors to engage in decision-making according to the advantages, disadvantages, and credibility of different equipment. On the one hand, ship navigation data is diversified, and rule-based decision-making presents new challenges to the OOW. On the other hand, the rise in computing capacity and the growing amount of data become conducive to the development of autonomous navigation systems. Combined basic COLREG rules and neural networks are the future developing tendency.

Humans can exhibit the most intuitive feelings concerning data such as images and videos. Computers receive a significant amount of pixel information, and it can be difficult for computers to form their own perceptions. The current focus of convolutional neural networks is to identify a target and perform image classification, and it can be difficult to determine the motion pattern of the target ship. Menze and Geiger proposed a novel model and dataset for 3D scene flow estimation, which can be applied for autonomous driving [1]. In the field of radar

target detection, methods such as those proposed by Nayyar et al. include a System-on-a-Chip (SOC) that performs the initial steps of signal processing for object detection on digital beat signals generated by multiple receive radar channels [2]. However, as an analog signal input, the radar data can only roughly determine the motion state of the target ship. It is easy to lose the target when there is an obstruction in the surrounding environment.

1.1.2 Different way to achieve collision avoidance

The seafarers using their experience and their excellent seamanship can make appropriate decisions to avoid collisions. Ozoga and Montewka using a case study to present a Multi-ARPA system on the basis of an algorithm, designating direct hazards for the own ship in a heavily trafficked basins situation for numerous ships [3]. Perera and Zhao et al. has discussed the application of deep learning in autonomous ship navigation, which is a general framework to support the navigation need to require some of the various maritime technologies, such as situation awareness, collision avoidance, COLREGs etc. all of the ocean autonomy levels need to be considered [4] [5]. Perera and Soares also from the internal environment and Integrated Bridge Systems (IBS) discussed the collision risk and quantification in ship navigation, illustrate that it is an important part of e-navigation [6].

And there have other examples, Xiang and Zhang proposed an underwater robotic vehicle intelligent risk analysis and critical decision [7]. Two-layer fault treatment system including risk analysis subsystem and intelligent decision subsystem proposed to enhance the onboard safety. Hou et al. presents an agent-based communication called Vehicular fog computing, the authors conceive the idea of utilizing vehicles as the infrastructures for communication and computation, utilizes a collaborative multitude of end-user clients or near-user edge devices to carry out communication and computation [8]. However, IBS also requires human participation, seafarer interaction. At the current navigation stage, it can cover a higher level of automation, and even continue to achieve shore-based remote

control through some methods, such as optimization collision avoidance algorithms, but it cannot achieve fully autonomous unmanned navigation.

1.1.3 Modern approaches for risk perception

Recently, deep reinforcement learning in artificial intelligence has defeated human champions in strategy games and Go competition, making related algorithms applications widely used in other fields. Especially in the field of autonomous transportation, situation awareness, risk perception and so on. Autonomous driving requires a sequence of decision-making controls. For example, Jaradat et al. solved the problem of mobile robot path planning in an unknown dynamic environment [9]. This chapter approach has a high Hit rate and that the robot succeeded to reach its target in a collision-free path. However, Q-table stores limited state, which is feasible for simple environments, but for unknown dynamic environments, searching will take a lot of time and it is difficult to satisfy the requirements of collision avoidance operations. Duguleana and Mogan has the same problem issue [10]. Li et al. explores unmanned free-flight traffic that may operate in dense airspace [11]. The authors using deep reinforcement learning to compute corrections to account for dense airspace.

Arel et al. and Chinchali et al. demonstrate multi-agent reinforcement-based control over longest-queue-first governed isolated single-intersection traffic signal control and multi-objective optimal control [12] [13]. Illustrate data-driven strategies enable safe interaction, model identification, and dimensionality reduction. Wang et al. Cheng and Zhang using deep reinforcement learning obstacle avoidance decision-making (DRLOAD) algorithm solve the unmanned ships collision avoidance in unknown environments[14] [15]. The authors use the combination of Long Short-Term Memory (LSTM) and deep reinforcement learning to deal with the collision avoidance of dynamic unknown environments, but this simulated unknown environment is still not a real navigation environment. This chapter presents a real automatic identification data environment, combined with a rule-based decision and a neural-based decision, and construct an

autonomous collision avoidance decision system. The authors present a discussion on the various training requirements and different training precisions for text data and frame data.

1.2 Objectives

The objective of this study is that detect the ship's navigation environment by different shipborne sensors, collect various navigation data sets (images / video, AIS, radar, etc.), use the collected data to describe the ship state during navigation, establish dynamic equations and mathematical models. A series of autonomous decision-making processes such as end-to-end risk perception and collision avoidance is implemented using deep reinforcement learning methods.

1.3 Methodology

These are the step conducting this work:

- Literature review and motivation.

Introduce the current situation of research on environment perception and autonomous decision-making based on deep learning and reinforcement learning, and introduce the motivation of this work.

- Data collection.

Collect actual raw AIS, images / video data, fake image data from generative model. Using different data from sensors to describe the ship navigation state (for example: collision, fire disaster, etc.)

- Experiment.

Use the adversarial generation network to create fake image data that is difficult to obtain in reality navigation situation; labeling the image data obtained from the perspective of the ship bridge, use the deep learning method to train a real-time detection model, perceive the ship's navigation environment; use QGIS software to visualize the AIS data, and then use the screen frame as the input, train our deep reinforcement learning network using discrete actions as output, achieve intelligent agent autonomous decision-making.

1.4 Dissertation Structure

This dissertation detail has 7 chapters and is arranged as explained below. The whole structure of each chapter shown as Fig.1.2.

Chapter 1 – Introduction

This chapter presents the whole research background, objective, methodology, the summary of each chapter, the overall structure of the dissertation. The significance of this study is introduced in terms of the interaction between seafarers and navigation aids, the method of ship collision avoidance, and how to realize risk perception. At this stage, the development direction of ships is automation and unmanned navigation. Currently, many challenges are still being faced. However, the accumulation of technology in various aspects is ready for achieve autonomous ship navigation.

Chapter 2 – Human-in-the-loop decision and risk perception.

A risk perception and decision framework are proposed. The aim of this chapter is to construct an unmanned ship swarms monitoring model to improve autonomous decision-making efficiency and perception performance of unmanned ship navigation, figure out the relationship between on-board decision-making (agent-based) and shore side monitoring (human-in-the-loop). The process of ship data detection, tracking, analysis and loss, and the application of decision-making algorithm, discuss the different risk responses of specific unmanned ship types under various latent hazard environments, particularly in terms of precise conversion timing in switching over to remote control and full manual monitoring, to ensure safe navigation when the capability of autonomous risk response inadequate. This framework makes it easier to train data and the adjustment for machine learning based risk prediction. It can be concluded that the automation level can be increased and the workload of shore-based seafarers can be reduced easily.

Chapter 3 –Theoretical foundations and applications of deep reinforcement learning

This chapter provides the theoretical foundations of deep reinforcement learning (DRL), the autonomous decision-making architecture of unmanned vessel navigation has been formulated. The main purpose of this chapter is to explain the principles of Markov Decision Process (MDP) and DRL and their applications. Show the advancement of mathematical methods in the ship transportation field with relevance to collision avoidance scenario applications. Using this method (reinforcement learning mathematical formalized by MDP), the appropriate action order based on a seafarer's experience not necessary. The model scores each step's reward by its action behavior and learns how to avoid obstacles by itself. By deploying decision timing, state, reward, and digitizing the seafarer's decision, a reinforcement learning algorithm based on MDP has been established. In the model training, under a single factor influence, the vessel tends to change course with the best appropriate action behavior, which is almost consistent with decision-making behavior based on actual experience at sea.

Chapter 4 – extend training images sets using generative adversarial nets

Throughout the doctoral phase, the data used in the research included real ocean navigation image / video data and AIS data. This chapter mainly introduces another data acquisition method: Generative Adversarial Networks (GANs) generation method. This part solves two crucial problems in the unmanned ship navigation: insufficient training data for neural networks and convergence of optimal actions under discrete conditions. To achieve smart collision avoidance of unmanned ships in various sea environments, first, this study proposes a collision avoidance decision model based on a deep reinforcement learning method. Then, it utilizes GANs to generate enough realistic image training sets to train the decision model. According to generative network learning, the conditional probability distribution of ship maneuvers is learnt (action units). Subsequently, the decision system can select a reasonable action to avoid the obstacles due to the discrete responses of

the generated model to different actions and achieve the effect of intelligent collision avoidance. The experimental results showed that the generated target ship image set can be used as the training set of decision neural networks.

Chapter 5 – Autonomous decision approach in a real AIS environment

This chapter applies the decision neural network training. The authors proposed a real automatic identification system (AIS) environment, combined with a rule-based decision and a neural-based decision, and construct an autonomous collision avoidance decision system. Rule based decision-making has several applications in the field of adaptive systems, expert systems, and decision support systems, including automatic ship navigation. However, with the growing amount of data and the expansion of some complex application scenarios, including automation and unmanned situation requirements, it has become progressively challenging to satisfy autonomous decision-making development requirements. In this chapter, the authors examine the performance of decision neural networks under different input types of AIS data. A discussion on the various training requirements and different training precisions for text data and frame data has been presented.

Chapter 6 – Ocean target detection for perception-oriented ship navigation

In this chapter, a proprietary ocean dataset is proposed that uses deep learning methods to achieve automatic ship classification and target recognition, especially the risk perception of tiny targets. In recent years, the application of convolutional backbone-based network in various fields has gradually increased. However, most neural network algorithms and frameworks still use ImageNet, COCO, and other public datasets to train and verify their accuracy, efficiency, and other performance. Ocean targets have their own characteristics. For unmanned ship navigation, long-distance targets, tiny targets, and large maneuverability gaps between different targets should be considered. These factors have important evaluation value for target detection and risk perception. Therefore, it is necessary to construct proprietary dataset to design and train our ocean target detection model. In this

study, the author use transfer learning to train and compare various neural network models for ocean target classification, analyze the deep residual network and YOLO v3 framework, implement real-time ocean target detection. Considering the results of the analysis, the reliable risk perception capability, the performance, and efficiency of the modified model by improving the actual collected ocean target dataset can be verified. The proposed dataset and approach can be used for ship autonomous navigation or assistant navigation.

Chapter 7 – Conclusion

This last chapter concludes with an explanation of the autonomous decision-making level, unmanned ship navigation, visual-based target classification, detection and deep reinforcement learning for risk perception oriented ship navigation. How to use the collected data to describe the ship state during navigation, establish dynamic equations and mathematical models. A series of autonomous decision-making processes such as end-to-end risk perception and collision avoidance are implemented using earlier mentioned methods.

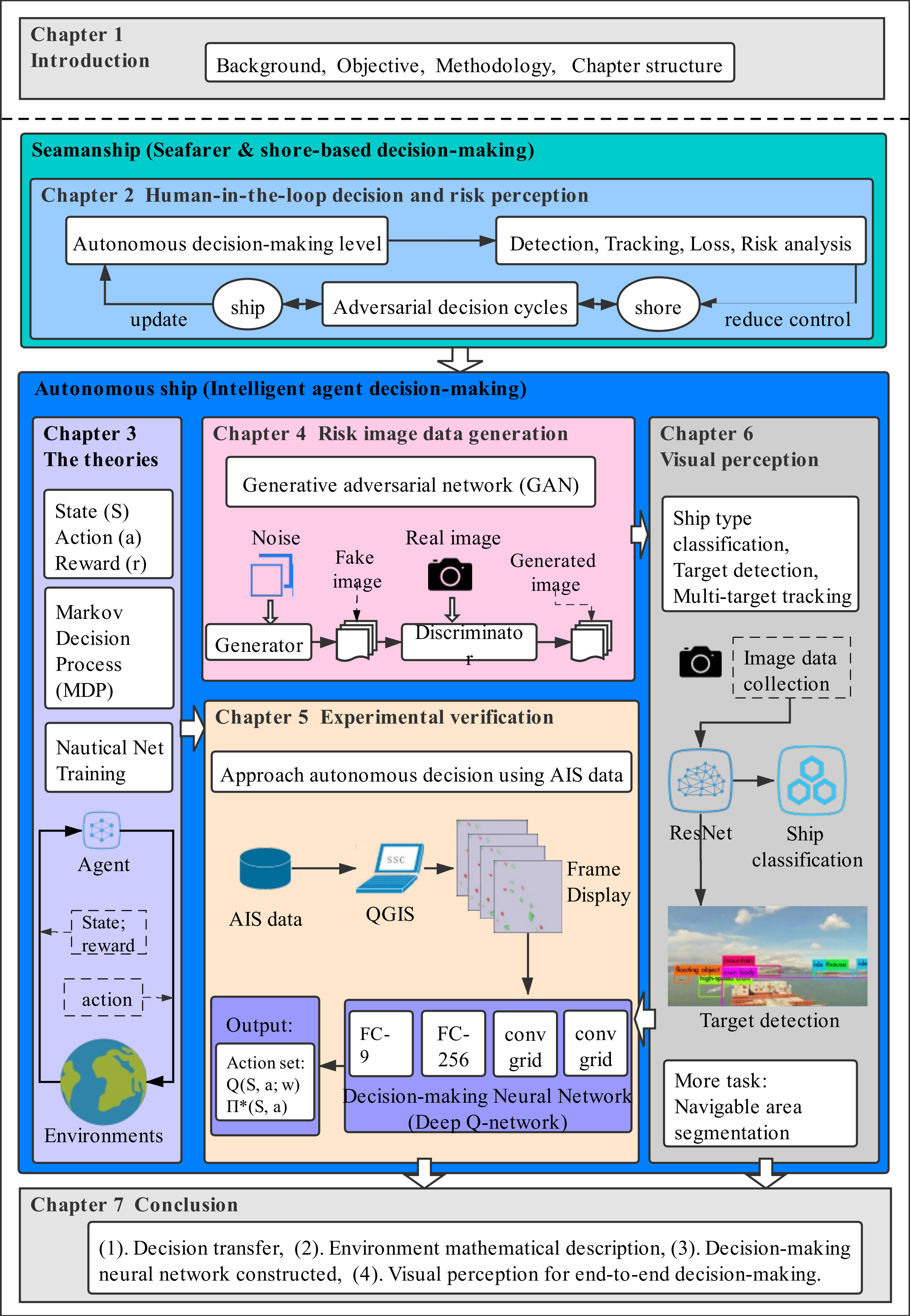


Fig.1.2 Dissertation structure

1.5 References

- [1] Menze M, Geiger A. (2015): Object scene flow for autonomous vehicles[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 3061-3070.
- [2] Nayyar J S, Ginsburg B P, Bose S, et al. (2016): Distributed Radar Signal Processing in a Radar System: U.S. Patent Application 14/633,647[P].
- [3] Ożoga, B., & Montewka, J. (2018). Towards a decision support system for maritime navigation on heavily trafficked basins. *Ocean Engineering*, 159, 88-97.
- [4] Perera, L. P. (2018). Autonomous Ship Navigation under Deep Learning and the challenges in COLREGs. In *ASME 2018 37th International Conference on Ocean, Offshore and Arctic Engineering*. American Society of Mechanical Engineers Digital Collection.
- [5] Zhao, L., & Roh, M. I. (2019). COLREGs-compliant multiship collision avoidance based on deep reinforcement learning. *Ocean Engineering*, 191, 106436.
- [6] Perera, L. P., & Soares, C. G. (2015). Collision risk detection and quantification in ship navigation with integrated bridge systems. *Ocean Engineering*, 109, 344-354.
- [7] Xiang, X., Yu, C., & Zhang, Q. (2017). On intelligent risk analysis and critical decision of underwater robotic vehicle. *Ocean Engineering*, 140, 453-465.
- [8] Hou, X., Li, Y., Chen, M., Wu, D., Jin, D., & Chen, S. (2016). Vehicular fog computing: A viewpoint of vehicles as the infrastructures. *IEEE Transactions on Vehicular Technology*, 65(6), 3860-3873.
- [9] Jaradat, M. A. K., Al-Rousan, M., & Quadan, L. (2011). Reinforcement based mobile robot navigation in dynamic environment. *Robotics and Computer-Integrated Manufacturing*, 27(1), 135-149.
- [10] Duguleana, M., & Mogan, G. (2016). Neural networks based reinforcement learning for mobile robots obstacle avoidance. *Expert Systems with Applications*, 62, 104-115.
- [11] Li, S., Egorov, M., & Kochenderfer, M. (2019). Optimizing Collision Avoidance in Dense Airspace using Deep Reinforcement Learning. *arXiv preprint arXiv:1912.10146*.

- [12] Arel, I., Liu, C., Urbanik, T., & Kohls, A. G. (2010). Reinforcement learning-based multi-agent system for network traffic signal control. *IET Intelligent Transport Systems*, 4(2), 128-135.
- [13] Chinchali, S. P., Livingston, S. C., Chen, M., & Pavone, M. (2019). Multi-objective optimal control for proactive decision making with temporal logic models. *The International Journal of Robotics Research*, 38(12-13), 1490-1512.
- [14] Wang, C., Zhang, X., Cong, L., Li, J., & Zhang, J. (2019). Research on intelligent collision avoidance decision-making of unmanned ship in unknown environments. *Evolving Systems*, 10(4), 649-658.
- [15] Cheng, Y., & Zhang, W. (2018). Concise deep reinforcement learning obstacle avoidance for underactuated unmanned marine vessels. *Neurocomputing*, 272, 63-73.

Chapter 2

Human-in-the-loop decision-making and risk perception

2.1 Literature review and motivation

The study of autopilot aircraft and vehicles has entered a period of vigorous development in the aviation and land transportation industries. Artificial Intelligence (AI) has begun to play a major role in these industries [1]. The research trend has been moving toward the conception of robotic vehicles capable of making spontaneous and effective decisions in demanding or uncertain situations. Considering the depression of the current shipping industry, optimizing and adjusting the industrial structure is crucial, in cases where the over-head of seafarer's expenditure and crew company management account for a large part of the payment. Unlike Unmanned Aerial Vehicles (UAVs) and unmanned vehicles, unmanned ship swarms may have a higher superiority for certain ship types such as regular container ships for certain voyages [2].

The real demanding circumstance of Vehicle to Everything (V2X) scenarios demands the need of handling fully automatic driving, remote management personnel in the state of surveillance, as well as any latent hazard requiring the car (where there is the possibility of the driver being involved) to take their own effective action at the critical instant. Undoubtedly, the current technical and legal aspects are still facing enormous changes. Many difficulties must be overcome for including fully automatic navigation in unmanned ship swarms, remote multi-agent monitoring, and efficient transportation of goods. On the other hand, in specific transport areas, under remote monitoring and perception, multi-agent transportation is technically advanced [3]. Therefore, Autonomous Ship Swarms Transportation is a very promising field of research for unmanned ships [4].

The study objective of this chapter is unmanned container ship swarms. Container ships, especially regular container ships, have characteristics such as standardized operations schedule, high level of automation in the process of loading and unloading of containers, and easy remote monitoring of goods condition. Upon removal of the bridge, an unmanned container ship can expand its packing capacity, improve enterprise business income, and reduce seafarer costs, in order to facilitate the shipping of large-scale company cargo dispatch, resulting in significantly improved efficiency [5]. Yang and Wang designed a fully automated loading and unloading platform specifically for an unmanned container ship. When in a specific terminal wharf, both sides of the ship can load and unload simultaneously, thus improving ship loading and unloading efficiency, increasing wharf benefits and reducing the stagnation time of ships in the anchorage, it also allows the optimization of the port traffic flow [6]. Nonetheless, market factors determine the research prospects of unmanned container ships. The container ship traffic is a huge business, even considering that a tiny accident may cause immeasurable disaster and loss. Therefore, it is necessary to construct the Autonomous Ship Swarms monitoring model before proceeding with the technical details. The ship itself has a limited ability to resist risks, especially under demanding or uncertain environment. Therefore, the decision-making cycle by itself may not have adequate capacity to avoid hazards. However, frequently invoking decision-making resources from the expert system of shore station may cause remote monitoring capacity insufficient, and shore-based monitoring seafarers may lead to more human errors or other latent failures. Therefore, it is necessary to maintain the resource balance between the decision-making cycle and the remote monitoring of unmanned ships, thereby optimizing the structure of the decision model.

Autonomous risk prediction and autonomous decision-making are genuine and significant parts of the unmanned ship swarms monitoring model [7]. They are effective ways to diminish the workload of shore-based seafarers, reduce human

error and improve commercial interests, by maximizing the risk prediction ability and risk avoidance decision-making of unmanned ships.

Even if the relationship between decision and maneuvering automation has already been proposed, if high automation is selected for the action part, then designers should resist the temptation of high automation levels of decision-making [8]. Even the real “noisiness” world always has some kind of unexpected situation emerged [9]. The purpose of this chapter is to build an automatic decision cycle model to improve the decision-making efficiency and safety performance of unmanned ships. According to the previous assumptions, with unmanned container ship response to various hazards, calculated training data and parameter adjustment, the performance of automatic decision-making can be improved. The situation of remote human intervention will then become less significant. When a new uncertainty situation appears, original data tracking and analysis and handling of data loss are required. In this case, the insufficient ability of automatically respond to facing hazards will cause the model to shift to manual monitoring and remote control to ensure safe navigation.

2.2 Autonomous decision-making level

Because the term “automation” has been used in many different ways, the British Dictionary defines automation as:

- The use of methods for controlling industrial processes automatically, esp. by electronically controlled systems, often reducing manpower.
- The extent to which a process is so controlled.

Autonomous decision-making should have a new implication, which the authors defined as automaticity between the different decision-making cycles. Although researchers have already described the conflict between high automation levels and the automation of decision-making [8], their opinions are focused mainly on

high levels of automation, and do not considered the decision-making aspect mistake. Moreover, "error-trapping" shows that lower link communication automation can allow more action errors. When high automation is selected for maneuvering, researchers should resist the temptation for high automation levels of decision-making. Therefore, high levels should be executed only for low-risk situation awareness; for all other situations, the level of automation decision should not exceed the level of the computer suggesting a preferred alternative to controller.

On the other hand, with the improvement of machine learning algorithms, more unlabeled data can be used, and more reliable automatic decision-making does not require human intervention, considering mainly the concept of "human-centered automation" re-understanding [10]. As there are two distinct centers (ship swarms and shore expert station) for the autonomous decision-making of unmanned container ship groups, and more decisions can be made by the cycle itself before the "human-centered" is involved [11], the environment for automatic decision-making has become easier.

2.2.1 The levels of autonomous decision-making

The basis for automatic decision-making must be based on a good automaticity classification (see Table 2.1). From low to high, automated carry forward also shows the development of ship automation decision-making process, which proves that the direction towards automation is inevitable. It can be seen from the table that the lowest level, level 1, is completely comprised of manual operations; the second level of the decision-making system can provide all the decision options, but at this level, the system doesn't make its own decisions (data learning and training process); The third level can optimize the selection and reduce the possible decisions output (perception process); the fourth level can provide an optimal decision-making program, but still cannot take action (optimization process); The fifth level of decision-making is a conversion point, in which the system usually needs to be agreed with the seafarer to take action, but this level of decision-making is able to provide a decision-making operation by itself, so any level of

automaticity higher than this can be considered as completely autonomous decision-making. Note that the decision of the fifth level according to this table is important for this chapter, as it is also a turning point for the ships own decision-making cycle and shore-based decision-making. For decisions taken at a level higher than this, such as the sixth level, the system can take action on its own, and only part of the uncertain data is sent to the shore-based seafarer for record; for seventh level and above decision-making, the system can recognize the timing of conversion by itself, and take action, without human involvement.

Table 2.1 Automaticity selection by ship decision & shore-based monitoring

Automaticity Levels High ↑ ↓ Low	9 • Full decision making and take actions autonomously,
	8 • Informs the seafarers if interrogated, or
	7 • Conversion timing permitted automatically, take maneuvering, or
	6 • Autonomous decision making, data presented to the seafarers, and
	5 • Executes the optimal action if the seafarers approve, or
	4 • Present an optimal solution automatically but no action,
	3 • Optimization options, compression all results, or
	2 • The system offers a complete set of decision/action alternatives,
	1 • Seafarers take all decisions and Maneuvering

There is a problem that must be noted: when an unmanned container ship encounters a totally new situation, automaticity can transit from a high level to a lower level; thus, automaticity can change dynamically according to the ship navigation. The following discussion of the different target ships for the object of the decision-making will use this table 2.1 as standard to illustrate.

2.2.2 Markov Decision Process for target perception

In prior research [11], the authors used the Robot Operating System as a tool, extended the Markov Decision-making (MDM) and supported the decision-making methodologies based on Markov Decision Process (MDP) The aim of the MDP is to provide an action set of decision-making for the on-board cycle. When an action is executed, marine areas whose state changes according to a known

probability distribution, are converted to another state, and the probability distribution is related to the actions performed. As shown in Fig.2.1, three different target ships can be considered as three continuously changing MDP by detecting, tracking, analyzing and losing. According to the data obtained from the sensors of the ship (unmanned container ship), in order to observe the navigational status of the target ships, the table 2.2 was obtained. MDP3 is a large-scale ship, and, as such, the course of navigation is always within the scope of the own ships control, and the target ship can be completely in accordance with the navigation rules and decision-making procedures for collision avoidance or other actions; MDP1 indicates the course of the sailboat 1. Although the tracking process is blocked by a large obstruction, the ship itself can be based on the heading course and speed before disappearance, in order to speculate the state of sailboat 1, and for the reappear to be verified; MDP2 illustrate the blockage of the sailboat by a large obstruction, for example, due to fishing work or other unpredictable factors, which led to the target disappearance from the control scope (loss state). In this situation, wherein the system cannot be based on past data training or experience to get more information, a shift to remote control to acquire the necessary expert support is the best choice.

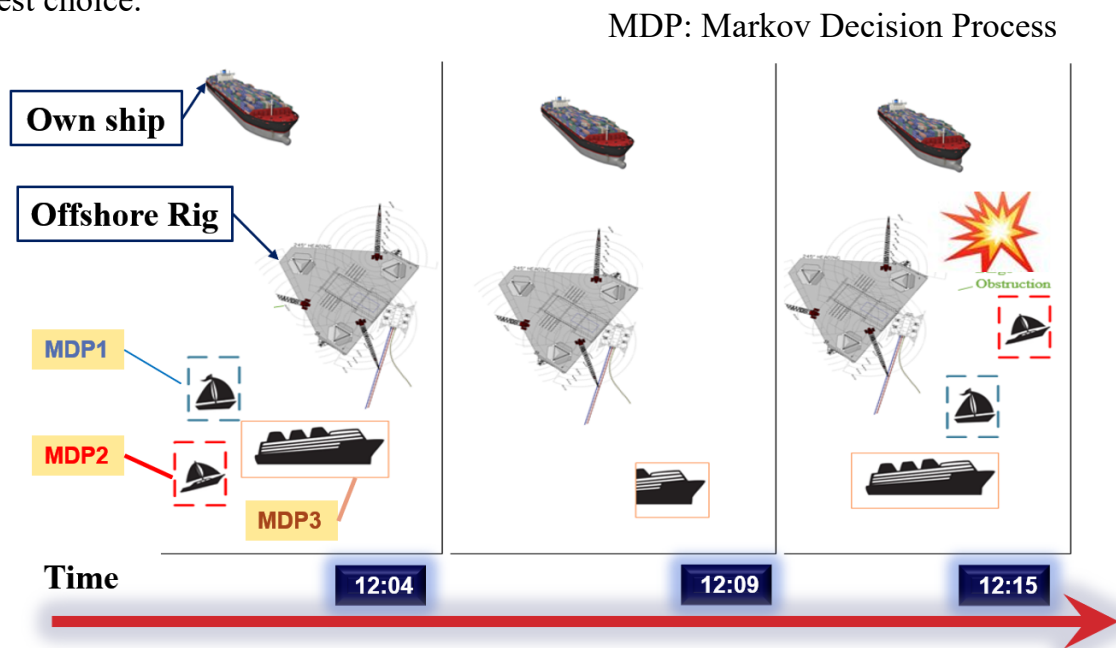


Fig.2.1 Target detection, tracking, loss process from the bridge view

Table 2.2 Multiple Target Tracking in a Markov Decision Process (MDP)

MDPs	Targets name	Process 1	Process 2	Process 3	Process 4	...	Process (n)
MDP 1	Sailboat 1	Detected	Tracking	Loss	Tracking	...	Unknown
MDP 2	Sailboat 2	Detected	Tracking	Loss	Loss	...	Tracking
MDP 3	Ro-ro ship	Detected	Tracking	Tracking	Tracking	...	Tracking

2.2.3 Target perception and decision-making

According to the previous scenario, the combination of the Table 2.1 and Fig.2.1 originates Fig.2.2 Because of the size, shape and speed different of the target ships, divergent detectors may also obtain different data reliability. The level of predictability varies, as the automaticity level is constantly unstable. According to the Table 2.1, it can be concluded that there are nine different automaticity levels, in different states, in which target ships have their own decision level. It can also be concluded that Markov Decision Processes can produce decision-making cycles of different automaticity levels decision-making under the closed ship itself, the fifth level can be seen as a timing conversion standard, lower than fifth level leads to the conversion to shore-based remote control, higher than fifth level belongs to the on-board decision cycle.

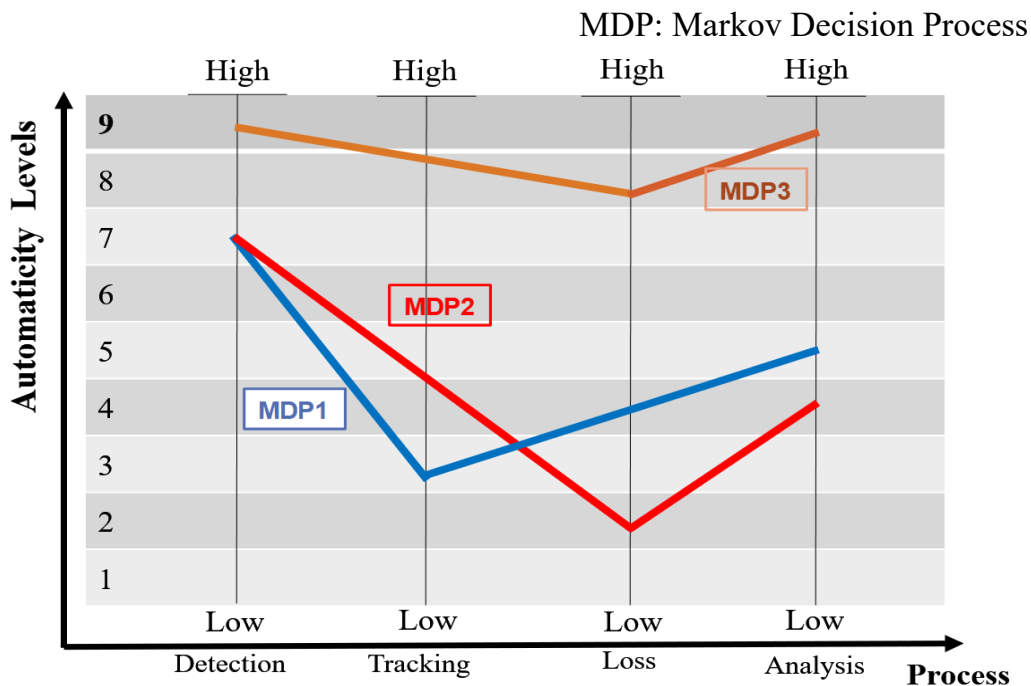


Fig.2.2 Automaticity level assessment for each process stage

2.3 Adversarial decision cycles

The mainstream technology of auto-driving prediction and decision-making is becoming clearer nowadays that machine learning is based on both deep learning and reinforcement learning. However, machine learning needs large data to achieve high performance and high reliability. It means that developers need to install an automatic driving equipment in a large number of automatic ships, so that the ships in the current operation can produce the required amount of data to enrich the decision-making cycle in order for the amount of data to lead to an improvement in the decision-making efficiency.

2.3.1 Unmanned ship decision cycles

Unmanned ship maneuvering cycle is generally divided into four main sections: sensory detection, tracking & perception, decision-making and optimal action. The model identifies the object from the environment, performs tracking and risk recognition, makes decisions and takes effective action, and the effect of the action is fed back into the environment to confirm the new position relationship. Consider a decision cycle as shown in Fig.2.3 The external environment also constitutes a part of the cycle, including the ships surrounding marine environment and the ships own hardware environment. The decision-making stage is the link to the shore-based remote control and ship swarms, so having decision-making as the end of the cycle (human decision center automation), it is possible to build the dependency relationship of the big decision-making cycle. It can be seen in Fig.2.4.

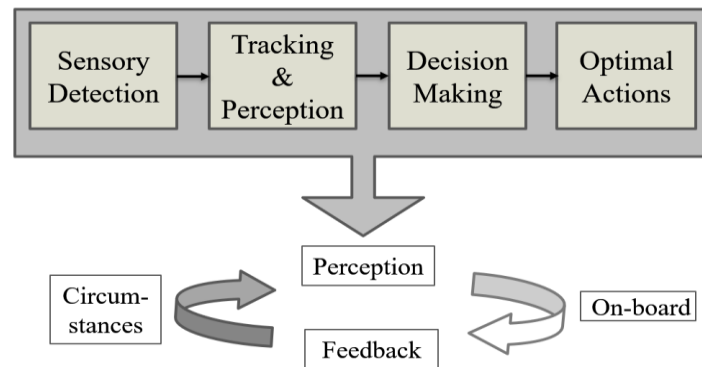


Fig.2.3 An example of unmanned ship maneuvering cycle

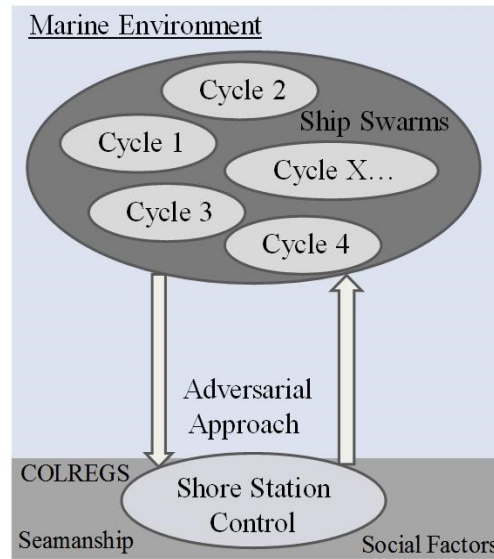


Fig.2.4 A flowchart of conversion timing optimization of decision-making

2.3.2 Autonomous decision process training

There are two main decision-making cycles of the unmanned ship: the first one is on-board decision cycle, countless monomer decision-making cycle constitutes the ship swarms; the second is a large group of ships and shore-based cycle. The relation-ship between the two main sections is similar to the Generative Adversarial Nets in deep unsupervised learning [12]. Ship swarms and shore station center belong to two main bodies of confrontation. At first, the decision-making cycle of a single ship is very small, and most of the decision-making cycle needs shore support to complete. However, as unmanned training data increases, due to the fact that the ships own decision-making capacity can be continuously enhanced, high quality decisions can finally be achieved, being as good as shore station seafarers would make. In this process, the workload of shore-based seafarers is gradually reduced, and the ship swarms automatic decision-making capacity is gradually increased.

2.4 Summary

This chapter presented an automatic decision-making solution in extreme cases where ships are rarely encountered in such complex situations when navigating in open waters. In order to improve the decision-making efficiency of the unmanned ship under remote control, it is needed to take into account this kind of situation in the beginning of the system design. The authors first proposed automaticity stratification, with the data accumulation of the on-board decision cycle, floating up and down real-time assessment data training of the situation encountered. Between the on-board decision and shore side monitoring of the adversarial decision cycle model it is demonstrated that the conversion timing continuously changes and always tends to on-board if the decision cycle can make decisions similar to those of the seafarer's (expert system). This structural involved human-in-the-loop, with the advantages of easier data train and parameter adjustment, easier improvement of the automation level. It can reduce the workload of shore-based crew significantly.

2.5 References

- [1] Prashanth, C. R., Sagar, T., Bhat, N., Naveen, D., Rupanagudi, S. R., & Kumar, R. A. (2013, August). Obstacle detection & elimination of shadows for an image processing based automated vehicle. In 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI) (pp. 367-372). IEEE.
- [2] Gudelj, A., & Krčum, M. (2012). The Container Transportation Problem: Model and Solution Methods. *TRANSPORT ENGINEERING*, 41.
- [3] Ren, W., Beard, R. W., & Atkins, E. M. (2007). Information consensus in multivehicle cooperative control. *IEEE Control systems magazine*, 27(2), 71-82.
- [4] Sarda, E. I., Qu, H., Bertaska, I. R., & von Ellenrieder, K. D. (2016). Station-keeping control of an unmanned surface vehicle exposed to current and wind disturbances. *Ocean Engineering*, 127, 305-324.

- [5] Dubrovsky, V. A. (2010). Multi-hulls: new options and scientific developments. *Ships and Offshore Structures*, 5(1): 81-92.
- [6] Yang, C., & Wang, N. (2011). Decision-making Method for Berth Allocation Disruption Management in Container Terminal. *Operations Research and Management Science*, 4: 14.
- [7] Kirsch, A. (2016). Human-aware Navigation in Domestic Environments Using Heuristic Decision-Making.
- [8] Parasuraman, R., Sheridan T. B. & Wickens C. D. (2000). A model for types and levels of human interaction with automation. *IEEE Transactions on systems, man, and cybernetics-Part A: Systems and Humans*, 30(3): 286-297.
- [9] Endsley, M. R. & Kiris, E. O. (1995). The out-of-the-loop performance problem and level of control in automation. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 37(2): 381-394.
- [10] Metzger, U. & Parasuraman R. (2005). Automation in future air traffic management: Effects of decision aid reliability on controller performance and mental workload. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 47(1): 35-49.
- [11] Zhang, R. & Furusho, M. (2016). Constructing a Decision-Support System for Safe Ship-Navigation Using a Bayesian Network. *International Conference on Human-Computer Interaction*. Springer International Publishing, 616-628.
- [12] Goodfellow, I., Pouget-Abadie, J., Mirza, M. (2014). Generative adversarial nets. *Advances in Neural Information Processing Systems* 2672-2680.

Chapter 3

Theoretical foundations of reinforcement learning

3.1 Literature review and motivation

In the past few decades, increasing attention has been paid to Markov decision process (MDP) algorithms, partly due to the success of self-driving car research using reinforcement learning methods. With the rapid development of automatic control, the Internet of Things (IoT), big data, state awareness, telecommunication, and other navigation technologies, the technical feasibility of smart ships has increased broadly [1]. In particular, the unmanned vessel yet to be christened, Yara Birkeland, is expected to begin its voyage in 2018 [2]. A large-scale merchant transportation of unmanned vessels has obvious advantages. For example, a fleet of unmanned container ships would maximize the capacity of storage space through the exclusion of the bridge and living area for the on-board seafarers; thereby maximizing the cargo volume and improving the transportation efficiency.

Moreover, many of the facilities on board serve the on-board seafarers, such as life-saving equipment, firefighting apparatus, pollution prevention, and living facilities. In the absence of seafarers, such equipment will not be required, reducing the weight of the ship and energy consumption, lowering the construction and operating costs, and increasing the ship's cargo capacity. Moreover, the main causes of maritime accidents are human-based factors, such as inadequate decision-making, operational negligence, deficient emergency response, and other seafarer factors. In an unmanned vessel, ship maneuvering is conducted primarily through automatic decision-making and remote monitoring by personnel working under better working conditions at a shoreside control station [3]. Thus, the impact of human-based factors is reduced.

However, the challenges are still many. One of the key problems of maritime navigation is the current trend towards human-centered decision-making systems. The process of seafarers maneuvering a ship at sea entails enacting the appropriate decision-making procedures at the appropriate time. Reinforcement learning has proven capable of developing learning models that are effective in planning [4]. Based on the complex maritime environment, it is observed that predetermination is the basic requirement to achieve safe navigation of unmanned vessels. This study consists of three sections. The first section clarifies the concept of MDP, especially focusing on how to use MDP to optimize the decision-making procedure under the situation of no seafarer on the bridge. In the second section, using MDP to formulate a mathematical model, the author classifies the different elements of the navigation behavior state. The third section concludes this study by presenting an algorithm for a very simple demonstration.

3.2 Mathematical Description of the Navigation State

Generally, in a vessel navigating at sea, the officer of the watch learns to recognize the state of the navigation environment through equipment and his look-out experience. The scenario in this study describes a case without any seafarers on the bridge; therefore, the vessel only recognizes the surrounding environment with devices such as radar, camera, and automatic identification system (AIS). Within the scope of this study, the author assume that the data obtained by all other devices has been integrated into a similar state framework (radar screen or electronic chart display). As shown in Fig.3.1, a cross encounter situation has been constructed between the vessel at the port side and the own ship. According to the International Regulations for Preventing Collisions at Sea (COLREGS), the own ship has no obligation to take an action to avoid the target vessel, but it should always pay attention to the changing state. Further, there is an obstacle (oil rig) in front of the starboard side of the own ship, and it needs to pass beyond a safe distance. This constitutes one of the most common navigational environments at sea.

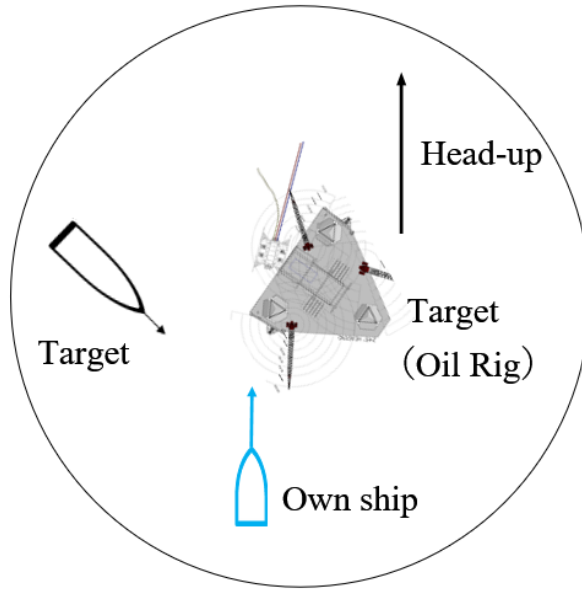


Fig.3.1 Example shoes the state that own ship encounters an obstacle (target)

This state changes, based on the surrounding environment and the change of the navigational state of the own ship. Each state change of the model requires a decision that weighs the next safe action behavior and the risk probability of exercising the action behavior. Therefore, the own ship must comprehensively predict and decide for the action change of each state to determine the optimal maneuver policy. There are two main ways to change the action behavior of the own ship: change the course by steering (turn to port side 'P' or turn to starboard side 'S') or change speed (accelerate 'A' or decelerate 'D'). Due to the different amplitude of each action change, it can be regarded as a discrete vector set, as well as the entire control tensor consisting of the free combination of the two action sets: $T_s = \{A, D, P, S, AP, AS, DP, DS\}$.

In practical applications, ships can take more operations than this set. In order to simplify the state description, the author only choose these combinations, and stipulate that the actions of ship operation are discrete actions. Discrete actions facilitate gradient calculation and make the function quickly converge.

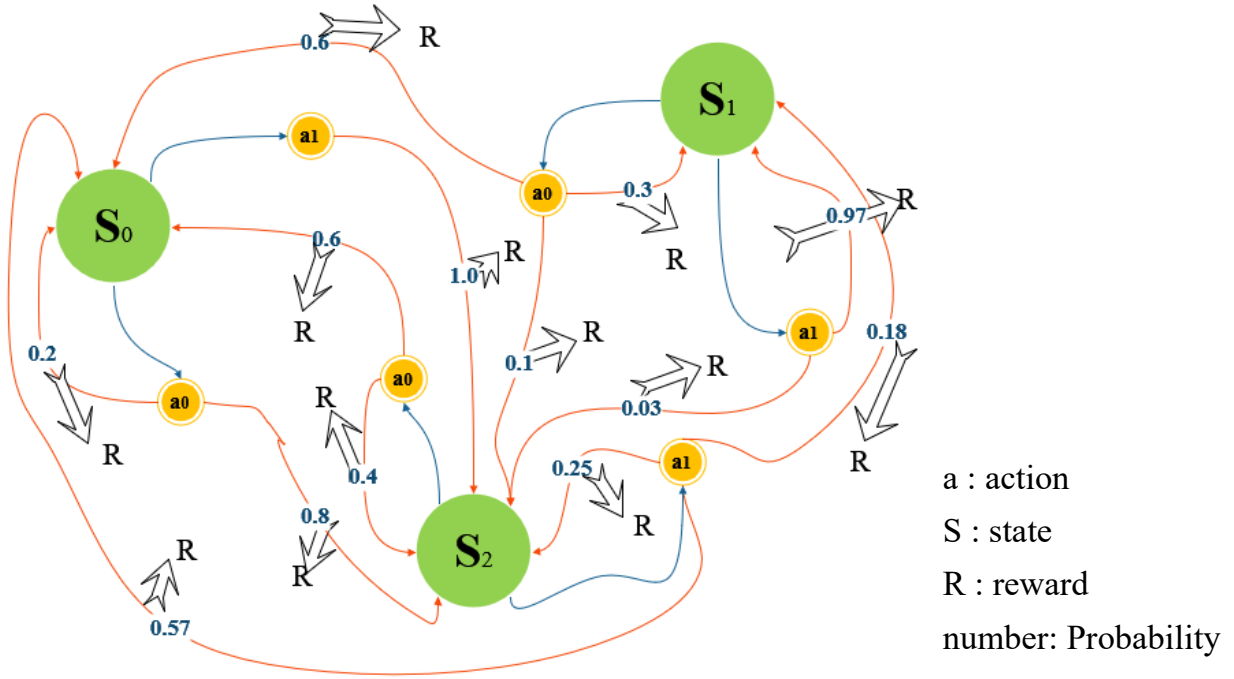


Fig.3.2 Example of the own ship encounters an obstacle state

3.3 Markov decision process

MDP are usually advantageous for approaching a wide range of optimization problems solved through dynamic programming and reinforcement learning. It is a class of stochastic sequential decision processes, in which the reward and transition functions depend only on the current state of the model and the current action [5]. With the mathematical framework of the navigation states constructed, it can approach a simple model of completely autonomous decision-making based on Markov processes, as shown in Fig. 2. The green and orange circles represent the different states and actions, respectively. MDP show that there may be more than one result per action in different states. For example, after the state S_1 passes the action a_0 , it may return to the previous state S_0 or change to the state S_2 . Moreover, it is also possible to keep the current state S_1 with nothing to change. The number above the vector arrow represents the probability that the previous state will shift to another state after the action. R, corresponding to the vector arrow, is the reward need to observe.

3.3.1 State Space

The decision state space, S , is the state of the unmanned navigation environment, which is the combination of a vector set of the state, $\mathbf{P}_{(x)}$, and a vector set of the dynamic change of external context, $Q_{(y)}$. $\mathbf{P}_{(x)} = P_1, P_2, \dots, P_m$ describes the set of actions, from act 1 to act m ; while $\mathbf{Q}_{(y)} = Q_1, Q_2, \dots, Q_n$ is the set of vectors whose model takes a specific decision procedure to change the navigation state. Therefore, the author obtain the formula $\mathbf{S} = (P_{(x)}, Q_{(y)}, \rho)$, where ρ is the factorial of the set, $i!$, that describes the total summation of the most recent situation. The decision state space can be described as

$$\bar{\theta} = \left\{ (P_{(x)}, Q_{(y)}, \rho) \left| \begin{array}{l} \sum_{m=0}^{i-1} x_i \leq x \\ \sum_{n=0}^i y_i \leq y \\ x_i, y_i \geq 0 \\ r = 0, 1, 2, \dots, i \\ m = 0, 1, 2, \dots, i-1 \\ n = 0, 1, 2, \dots, i \end{array} \right. \right\}, \quad (1)$$

where $y > 0$ is the model within a time period (which may be a very short period of time) for the amount of operations permitted.

$$0 \leq y \leq \lim_{x \rightarrow \infty} \sum P_{(x)} \quad (2)$$

The following four equations can be obtained from $P_{(x)}$, $Q_{(y)}$, and r :

$$\theta_1 = \left\{ \left((P_{(x)}, Q_{(y)}, 0) \left| \begin{array}{l} \sum_{m=0}^{i-1} x_m = 0 \\ \sum_{n=0}^i y_n = 0 \end{array} \right. \right) \right\} \in \bar{\theta} \quad (3)$$

θ_1 describes the model with no change in the unmanned vessel's maneuvers and no change in the dynamic state of the external context for all the possible states. Here, $r = 0$.

$$\theta_2 = \left\{ \left((P_{(x)}, Q_{(y)}, 0) \left| \begin{array}{l} \sum_{m=0}^{i-1} x_m > 0 \\ \sum_{n=0}^i y_n = 0 \end{array} \right. \right) \right\} \in \bar{\theta} \quad (4)$$

θ_2 describes a single change in the unmanned vessel's maneuvers, but no change in the external context. This situation rarely occurs. Thus, the index weight will be relatively small.

$$\theta_3 = \left\{ \left(P_{(x)}, Q_{(y)}, i \right) \left| \begin{array}{l} \sum_{m=0}^{i-1} x_m = 0 \\ \sum_{n=0}^i y_n > 0 \end{array} \right. \right\} \in \bar{\theta} \quad (5)$$

θ_3 describes a single external context change. For instance, it may have detected a new target that could have an impact on the own ship's safety. The model needs to continue to follow up the situation, so $r = i$.

$$\theta_4 = \left\{ \left(P_{(x)}, Q_{(y)}, i \right) \left| \begin{array}{l} \sum_{m=0}^{i-1} x_m > 0 \\ \sum_{n=0}^i y_n > 0 \end{array} \right. \right\} \in \bar{\theta} \quad (6)$$

θ_4 describes the situation where a change occurs in both the unmanned vessel's maneuvers and the external context. This situation commonly occurs in complex waters, and the model needs a continuous decision-making process. Thus, the index weight will be relatively large.

Subsequently, $\bar{\theta} = \theta_1 \cup \theta_2 \cup \theta_3 \cup \theta_4$. When the unmanned boat's performance limit y is a constant, θ represents all the possible operational decision states.

$$\mathbf{S} = \sum_{m=0}^{i-1} x_i \quad (7)$$

3.3.2 Action Set

For an unmanned vessel in the actual navigation process, from point A to point B, the system's navigation environment variables may occur in the entire navigation time interval $[a, b]$. In a safe navigation condition, the unmanned vessel does not have to make any maneuvers to change the model's navigational state; the actual decision-making time occurs only when the unmanned vessel encounters

something affecting the safety of the navigation. Thus, the number of valid environment variables is the number of the MDP model's decision moments.

For the embedded MDP, it follows from the previous discussion that the effective decision time can only be generated in the θ_2, θ_4 decision state space, and the set of ship operations that can be selected belongs to set T_s .

$$C_s = \begin{cases} \{0\}, & s \in \theta_1 \\ \{A, D, P, S, AP, AS, DP, DS\}, & s \in \theta_2 \\ \{0\}, & s \in \theta_3 \\ \{A, D, P, S, AP, AS, DP, DS\}, & s \in \theta_4 \end{cases}, (8)$$

3.3.3 Reward Function

It is assumed that the operating time interval of the unmanned vessel decision model is a time series that follows an exponential distribution. For the state S transferring to state S' by action A, the mathematical expression is:

$$r(s'|s, a) = \begin{cases} 0, & s \in \theta_1 \cup \theta_2 \\ \int_a^b r_{\delta\omega} td(1 - e^{-p_{\delta\omega}}), & s \in \theta_3 \cup \theta_4 \end{cases}, (9)$$

where $r_{\delta\omega}$ is the reward generated by the traffic event occurrence of the action ω^{th} , when the model is under the decision operation δ^{th} , and $p_{\delta\omega}$ is the probability that the unmanned vessel in the δ^{th} decision-making operation needs to act upon the ω^{th} operation.

The MDP decision model based on a continuous timeline is now complete.

3.3.4 Environment Formulation

Maritime long-distance transportation vessels have great inertia, resulting in difficult control. In the case of deceleration, the ship's power system may cause damage, while frequent steering using the rudder could reduce the service life of the steering gear and propeller. Therefore, to optimize maneuvering, this study

introduces the Markov process, with safe navigation constraints, as an unmanned vessel decision model. It integrates all the navigation data in a continuous decision-making timeline for the action set T . In an avoidance scenario, the local reward function not only needs to obtain the maximum value, but also needs to satisfy the global process to get the maximum reward value. Thus, it needs to achieve the purpose of optimizing navigation with less manipulation and obtain a larger reward function value. It can continue to simplify the MDP model, through the Python computer programming language, to build a simple marine collision avoidance environment for simulation and authentication. $\{A, D, P, S, AP, AS, DP, DS\}, s \in \theta_4$

3.4 Decision-making training and analysis

3.4.1 Simplified Decision Algorithm

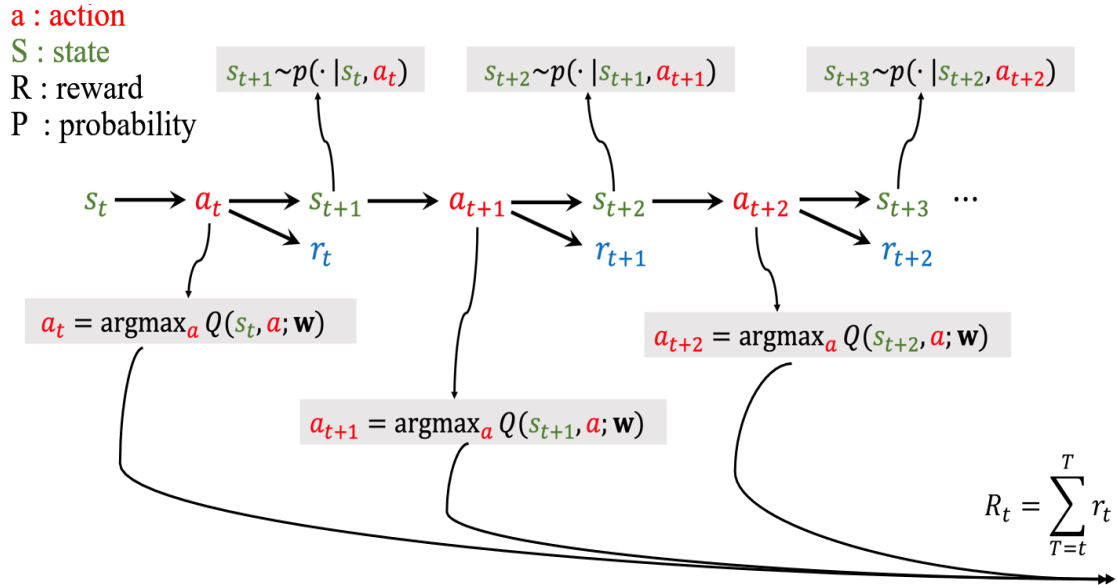


Fig.3.3 Simplified single-target, single-task Markov trajectory process

The process shown in Fig.3.3 is a navigation situation usually encountered in a seaborne ship. The own ship belongs to the original state from S_0 . According to the changes in the surrounding environment, the model tries to bring up a series of

simple action behavior a_0 . The model keeps changing the status of this ship until a complete collision avoidance has been achieved and gives the state S_n . The purpose of the decision-making algorithm is to try to give manipulation in the process. It does not directly tell the model manipulation of right or wrong. Thus, the model only scores a reward value by random action and makes the final global reward R_t approach an optimal value. The Markov decision process algorithm are as follows:

```

Initialize state  $S$ ,
 $MP(S, a) \leftarrow MP(S, a) + \alpha[R + \gamma \max_{a'} MP(S', a') - MP(S, a)]$ ,
 $S \leftarrow S'$ ,
until  $S$  is under the safe situation.
 $MP(S, a)$  for all possible values of the composition in  $S\_A\_table$ ,
Pick a random action  $A$  from the original state and use the policy derived from
 $S\_A\_table$ ,
 $S\_A\_table$  is State_Action_table,
Take action  $A$ , observer,  $S'$ .

```

In the above algorithm, α is the learning efficiency, γ is the decay rate, $MP(S', a')$ is the true value, and $MP(S, a)$ is the estimated value. In the model training of this study, the author assume that the encountered scenario at sea is: according to the navigation experience and collision avoidance rules (COLREGS), the appropriate action behavior of an experienced officer of the watch is that if the “starboard five” is held for 10 s, it can safely avoid obstacles at sea. Therefore, to simplify the flow of demonstration algorithm, it can make a single-threaded simulation, where every second there is an action behavior taken. The action behavior can be the same as before the manipulation of this state, and only “turn to port five” or “starboard five” types of steering action can be chosen. After setting the reward value gained by different manipulative behaviors, the whole training process cannot interfere with human factors. If the model is trained, it can perform the avoidance and steering

operation around 10 actions, indicating that this method can be applied to the unmanned vessel collision avoidance operations in a real offshore environment.

3.4.2 Training example and its results

After 20 iterations of model training, the author obtains the training results, as shown in Fig.3.4 In the first training process, the model did not know which one was suitable and could avoid the manipulation of obstacles. Therefore, it took 72 steps to achieve the aim of obstacle avoidance, which is not suitable for the requirement of collision avoidance in a practical situation at sea. However, with the increase in training times, especially after training 5 times, the model quickly learned that if "starboard five" is the most effective action to avoid the obstacles in front of it, the number of subsequent rudder steps is obviously cut back. Finally, the model can be stabilized in about 10 steps, to complete the collision avoidance manipulation.

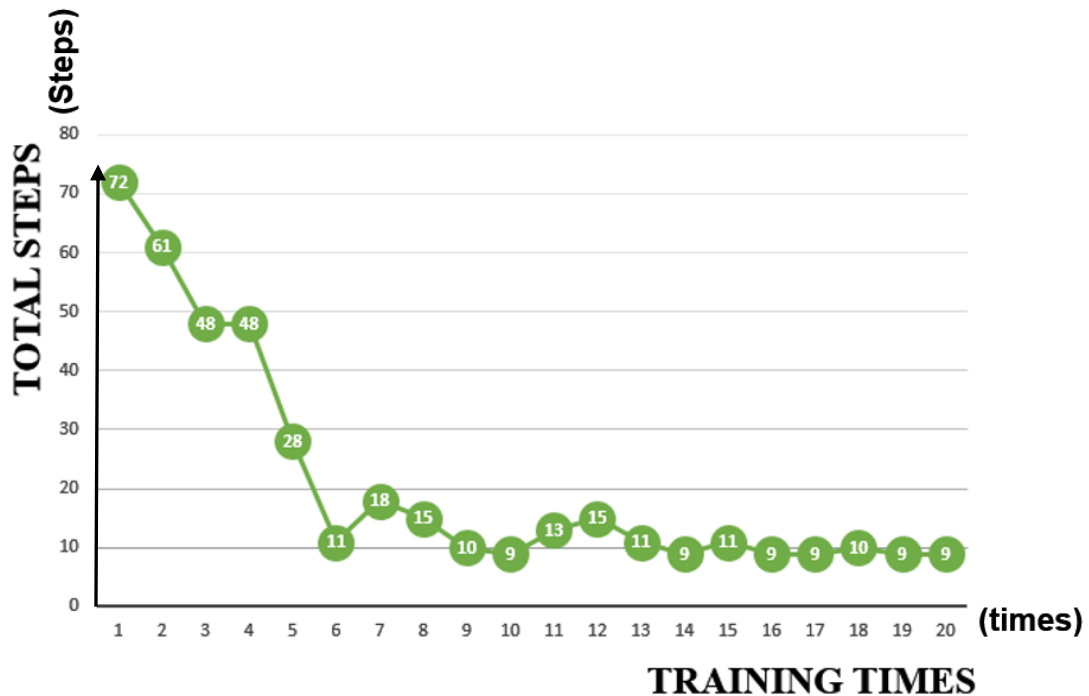


Fig.3.4 Training model results (approach 10 steps to avoid obstacles)

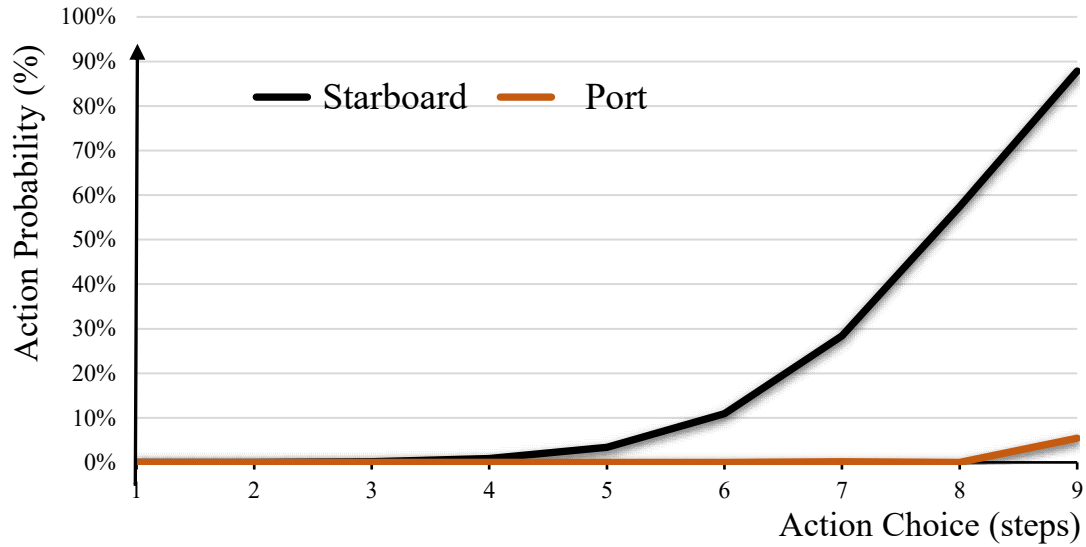


Fig.3.5 Embodiment decision-making from the perspective of action selection

3.4.3 Steering action data analysis

Moreover, as the training times increase, the author can also obtain results as shown in Fig.3.5. The abscissa axis in the figure is the number of steps performed in one state, and the ordinate axis is the average of the probabilities for decision-making. With the change of state, the decision made by the algorithm at the beginning does not know whether turn to port or starboard can be obtained a desirable outcome, but statistics of the results of 20 times training can be observed. Generally, after five steps, the "starboard five" action has significantly improved its probability compared to the "port five" decision, making it quicker to learn how to avoid the obstacles in front of it. Therefore, it also proves that the vessel is controlled to navigate safely, and the Markov processes-based autonomous decision-making model is fully applicable to the unmanned vessel navigation at sea.

3.5 Summary

This chapter presents the theoretical foundations of reinforcement learning method and formulated a Markov processes-based autonomous decision-making model to

help an unmanned vessel achieve collision avoidance. Further, using a case study constructed a simple common encounter environment, through the Python programming language, for a "starboard five" action, while keeping a 10 s rudder order to complete the obstacle avoidance action behavior that has training, which is known by officers of a watch's experience. No human-based factors control the model to avoid the obstacles. The model explores on its own, and according to each step, gets the appropriate reward for choosing the best appropriate solution to achieve the collision avoidance goal.

3.6 References

- [1] Littman, M. L. (1994). Markov games as a framework for multi-agent reinforcement learning. In *Machine learning proceedings 1994* (pp. 157-163). Morgan Kaufmann.
- [2] Ang, J. H., Goh, C., & Li, Y. (2016, July). Smart design for ships in a smart product through-life and industry 4.0 environment. In *2016 IEEE Congress on Evolutionary Computation (CEC)* (pp. 5301-5308). IEEE.
- [3] Maritime, K. (2017). Autonomous ship project, key facts about YARA Birkeland. URL <https://www.kongsberg.com/ks/web/nokbg0240.nsf/AllWeb/4B8113B707A50A4FC125811D00407045>.
- [4] Zarsky, T. (2016). The trouble with algorithmic decisions: An analytic road map to examine efficiency and fairness in automated and opaque decision making. *Science, Technology, & Human Values*, 41(1), 118-132.
- [5] Silver, D., van Hasselt, H., Hessel, M., Schaul, T., Guez, A., Harley, T., ... & Degris, T. (2017, August). The predictron: End-to-end learning and planning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70* (pp. 3191-3199). JMLR. org.
- [6] Puterman, M. L. (1990). Markov decision processes. *Handbooks in operations research and management science*, 2, 331-434.

Chapter 4

Extend dataset using generative adversarial network

4.1 Literature review and motivation

Unmanned ocean transportation is sure to revolutionize maritime unmanned navigation in the future. In October 2016, the Norwegian Maritime Administration and the Norwegian Coastal Authority established the world's first autonomous ship test zone in the Trondheim fiord as well as the Norwegian Forum for Autonomous Ships. It marked the promotion of unmanned ship research to the national level in Norway. In early December 2018, the "Suomenlinna II" polar passenger ferry successfully crossed the test area near the port of Helsinki, under the unmanned state, and passed the remote sea trial. The intelligent decision-making module is the "brain" of unmanned ships. It involves various technologies such as route optimization, risk warning, smart decision-making, and energy efficiency management. It can make most decisions based on the external navigation environment information, ship internal information, and shore-based support information. For example, it can take excellent navigation decisions and send control commands to the execution unit to make appropriate decisions [1].

Taking ship collision avoidance as an example, the intelligent decision-making module obtains the actual navigation situation around the ship according to the targets acquired by the radar, AIS, ship-borne infrared camera, visible light camera, and other sensors and its fusion information, and conducts a risk information analysis for the surrounding targets [2]. If there is a dangerous target, the collision avoidance decision is made through the intelligent collision avoidance technology combined with the current position, direction it is heading toward, and speed. The instructions formed by the decision, such as changing the course and changing the

speed, are sent to the rudder control system. In the process of collision avoidance of ships, the information transmitted by multiple sensors and equipment is continuously integrated, making sure the collision avoidance scheme is adjusted in time [3]. The core of the current research is how the decision-making module can satisfy the optimal navigational operations in all types of extreme offshore environments.

Therefore, in the risk assessment and early warning research of unmanned ship navigation, it is necessary to focus on the unmanned ship in complex navigation conditions (such as ports, straits, canals, and other intensive waters), ship collision avoidance and hydrometeorology, geographical environment, traffic situation, and other issues. This research is based on ship sensor data acquisition and training optimization of decision neural networks [4]. An intelligent risk warning model and method suitable for unmanned ships under complex navigation conditions is formed to approach real-time warning of ships [5].

In the intelligent decision-making research, an intelligent fusion correlation analysis is carried out on static and dynamic targets and navigation conditions around unmanned ships. Intelligent theories, such as deep learning, knowledge base, and situation calculation, are applied. Research on ship navigation intelligent decision theory based on ship navigation system information and shore-based support information, break through the key technologies of ship autonomous meteorological navigation technology. Technologies such as ship collision avoidance, reef avoidance, anti-shelf integration, and smart processing of navigation information support autonomous decision-making of ship navigation [6].

To achieve intelligent collision avoidance function of unmanned ships in various environments, a collision avoidance decision module based on deep reinforcement learning is proposed to make autonomous decisions under various conditions [7]. In the Cyber confrontation game, the DeepMind team collects enough data for training; however, in the real navigation environment, it is difficult to obtain data

in a rich and varied nautical environment. In particular, various types of encounter ships have different points of observation in different situations, and it is difficult to predict their future path of navigation [8]. In the process of calculating the global solution optimal solution, the decision model is difficult to differentiate due to the discrete action as a result, the global optimal solution cannot converge. Therefore, this study proposes a generative adversarial networks (GANs) model to solve the problem of neural network training data, and the combination of GAN and deep reinforcement learning to solve the convergence problem of optimal action under discrete action unit conditions.

4.2 Generative adversarial network and its application

4.2.1 The principle of GAN

GAN is a new method proposed by Goodfellow et al. to train generated models [9]. The method of GAN includes the generation and discrimination of two “adversarial” models. The generated model (G) is used to capture the data distribution, and the discriminant model (D) is used to estimate the probability that a sample is derived from real data rather than the generated data. Both the generator and discriminator are common convolutional networks as well as fully connected networks. The generator generates a sample from the stochastic vector, and the discriminator discriminates between the generated sample and the real training set sample.

This optimization process can be attributed to a two-player minimax game problem. Both purposes can be achieved through a backpropagation method. A well-trained generation network can transform any noise vector into a sample similar to the training set. This noise can be seen as the encoding of the sample in a low dimensional space. The generator generates meaningful data based on the stochastic vectors. In contrast, the discriminator learns how to determine real and generated data and then passes the learning experience to the generator, thereby,

enabling the generator to generate more workable data based on the stochastic vectors. Such a trained generator can have many uses; one of them being environmental generation in automatic navigation.

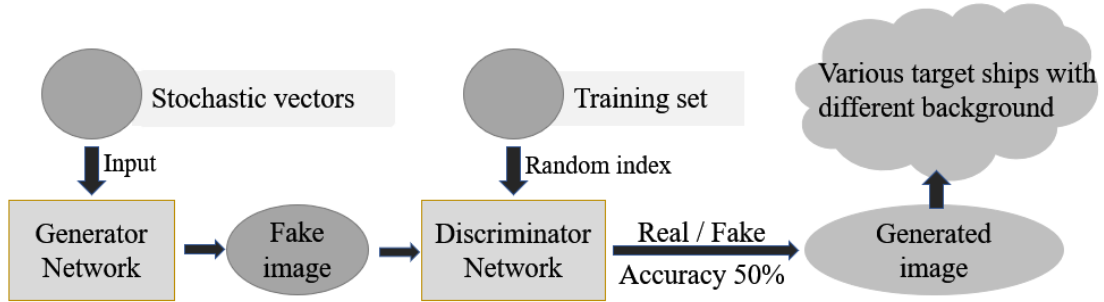


Fig.4.1 Applying GAN generate various target ships with different background

The specific process to obtain various target ships is shown in Fig.4.1 First, a few stochastic vectors are fed as input in the generator network, and fake data are subsequently generated by the generator. The aforementioned fake data can correspond to a few ship states pictures or navigation data such as AIS data of a nearby encounter of the given ship or the path planning data after the ship route is updated. The author input the fake data to the discriminator, and the discriminator determines whether the input data are real data or fake data generated by the generator. The similarity between the generated data and the real data gradually increases, then the discriminating ability required by the discriminator also increases accordingly. Furthermore, the generator and the discriminator share a mutually competitive and mutually adversarial equation. The generated data are considered to sufficiently mirror real data, and therefore, the fake data input by the generator appear sufficiently realistic. The approximate accuracy of the discriminator in this case is 50%. This corresponds to the target ship image data that are required in a critical sea environment.

4.2.2 GAN application examples

In the previous study, the authors proposed using the GAN to build an executable method for maritime navigation route re-planning. The unmanned ship can

independently generate new routes based on the environmental information around the ship before any possible danger occurs and can even interrupt the remote support. With the total generative time less than one second, the trained model helps the ship avoid obstacle or any latent disaster. In addition, GAN is easy to embed into the framework of reinforcement learning. For example, when using deep reinforcement learning to solve collision avoidance problems, GAN can be used to learn the conditional probability distribution of an action. The agent (own ship) can select a reasonable route based on the response of the generated model to different actions.

4.2.3 Global optimality of discrete actions

The mathematical equivalent of deep reinforcement learning can be considered as Markov decision processes in discrete time defined by five factors $(S, 'a', P, 'r', \gamma)$ with a neural network instead of Q-value [10]. Here, S is the finite state space (state set) in which the unmanned ship is located; a is the behavior decision space of the unmanned ship; i.e., the set of all actions or reactions in a space in any state, for example, the left rudder, right rudder, acceleration, deceleration, heeling, and stopping, etc. $P_a(S) = P(S'|S, a)$, where P is a conditional probability indicating that the unmanned ship reaches the next state under state s and action ' a '. The probability of the state S' , $r_a(S|S')$ is a reward function, which represents the excitation obtained by the unmanned ship from the state S to the next state S' in the case of action a . $\gamma \in (0,1)$ is reward attenuation factor, the reward at the next instant time t is attenuated using this factor.

In actual navigation practices, completing a collision avoidance process may require different operational coordination methods. These operations are incoherent and discrete. Further, the way different people respond to the same event may be different. Generally, both the reward ' r ' and the attenuation function ' γ ' are different. Thus, a method to converge an optimal global action group is

required. This study will discuss the possibility of combining GAN and deep reinforcement learning to solve this problem.

4.3 GAN method for generating training data sets

4.3.1 Conventional acquisition of target ship data set method

The acquisition of related target ship data is usually carried out at the position of the ship's bridge. Here, it can observe the state of the target ship's navigation, and then, photograph the target ship that will be encountered. The author carried out a seven-day summer research voyage on the university training ship "FUKAE MARU". A data set of a total of 4,000 images of valid target ships were obtained. However, this is not enough for the training of neural recognition networks. The target recognition and classification require a large number of data sets for both training as well as target identification, if the positional posture of the target ship is to be perceived. Therefore, a new approach to get training data is needed.

4.3.2 Example of the GAN generate lifeboat image data

This section mainly demonstrates the use of the demo provided by Big-GAN, as shown in Fig.4.2, which is the process of generating a lifeboat using GAN. In the images obtained at different times in the generation process, the sea surface appears in the image generated in Fig.4.2 a, and there are a group of fuzzy things in the middle that cannot identify the object. In Fig.4.2 b, the orange upper body and black rubber are common in the lifeboat. The hull, shown in Fig.4.2 c, is an almost completed generated image, and our eye and training model can roughly identify this as a lifeboat. The author can input different stochastic vectors and combine the real data input by the discriminator to get a large number of high-quality image data sets. As shown in Fig.4.3, it got three different types of lifeboats. More importantly, the background of the lifeboat could also be changed to provide a large amount of training data for our unsupervised decision model. It was observed to be much richer than the target ship data collected from the real seas.

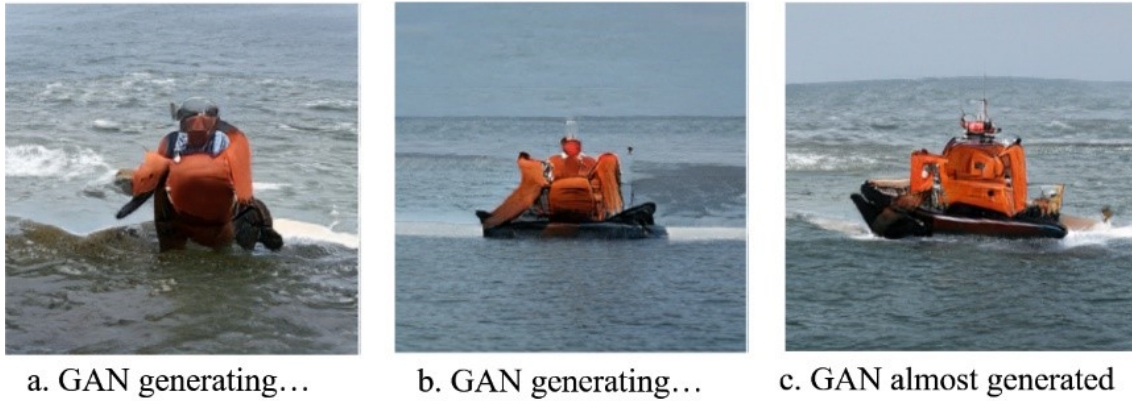


Fig.4.2 Target ship image data-generating process



Fig.4.3 Example of generated lifeboats with different back-grounds

The most important part of this research is to obtain a data set of the target ship with sufficiently high quality and quantity. As shown in Fig.4.4(Truncation: 0.14; Noise_seed: 0), a portion of the entire large-scale lifeboat target image data set is shown. These images were not taken by the camera and were generated entirely from the GAN model. Using different truncation and noise_seed, our model could generate various encounter situations at sea as shown in Fig.4.5 (Truncation: 0.28; Noise_seed: 0) and Fig.4.6 (Truncation: 0.56; Noise_seed: 0), as well as the various forms possible for the target ship at the time of the encounter; including various types of accidents such as collision, stranding, fire and loss of goods. The author obtained the training data set for lifeboats, ocean liner data sets as shown in Fig.4.7 (Truncation: 0.10; Noise_seed: 4) and Fig.4.8 (Truncation: 0.10; Noise_seed: 8), as well as data sets for various other types of marine moving targets.

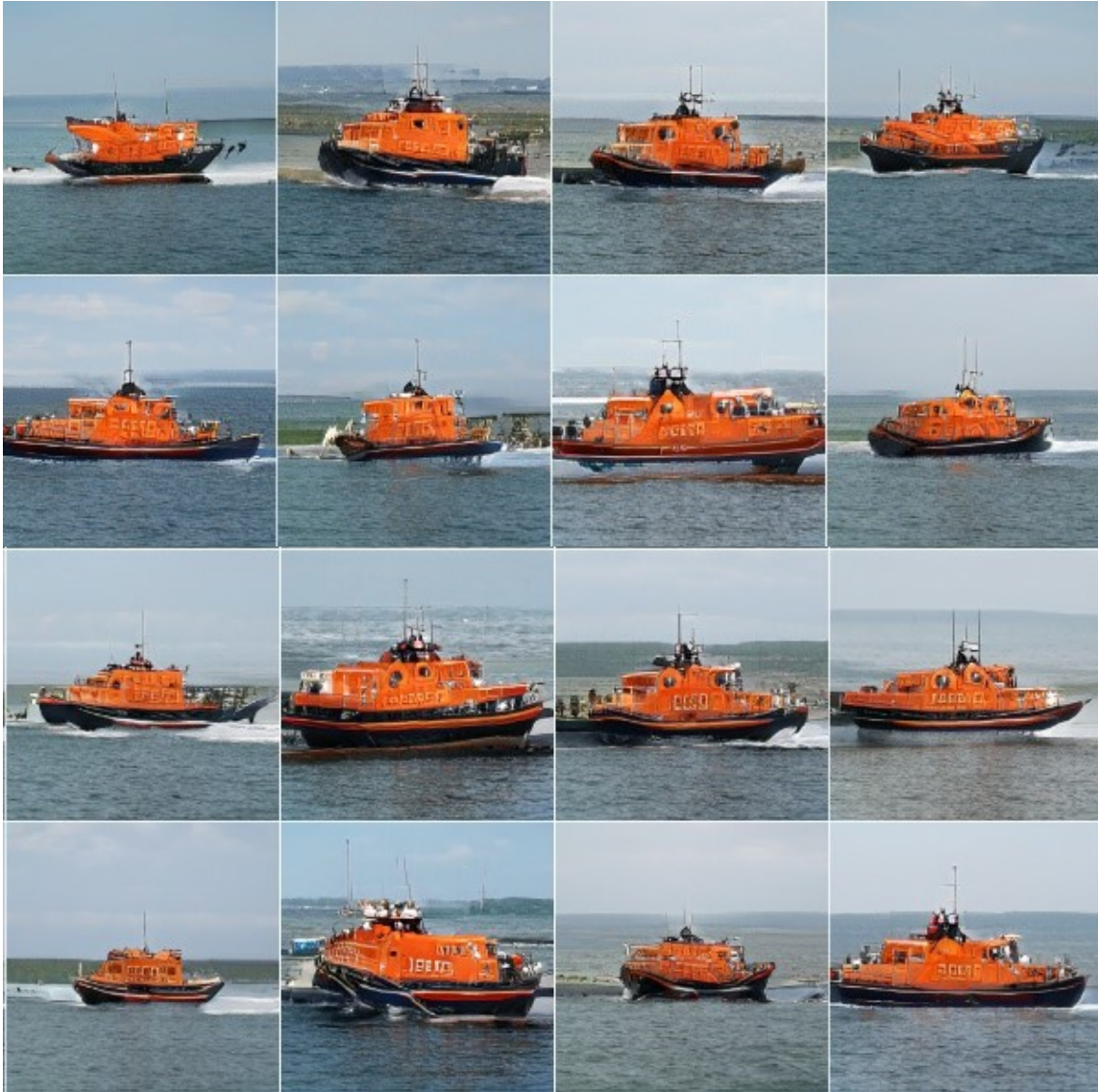


Fig.4.4 Large-scale data generated with different backgrounds. (Lifeboat)

According to different parameter settings (Truncation, Noise_seed), it can get different fake images. The real lifeboat image is used as the input of the discriminator, the author can get the image of the lifeboat as the main target of the images. If it is a fishing boat or other ocean floating objects, the author can also get other similar target images. This pair of target detection tasks can greatly expand the labeled samples and improve the detection accuracy.



Fig.4.5 Large-scale data generated with different ship status. (Lifeboat)

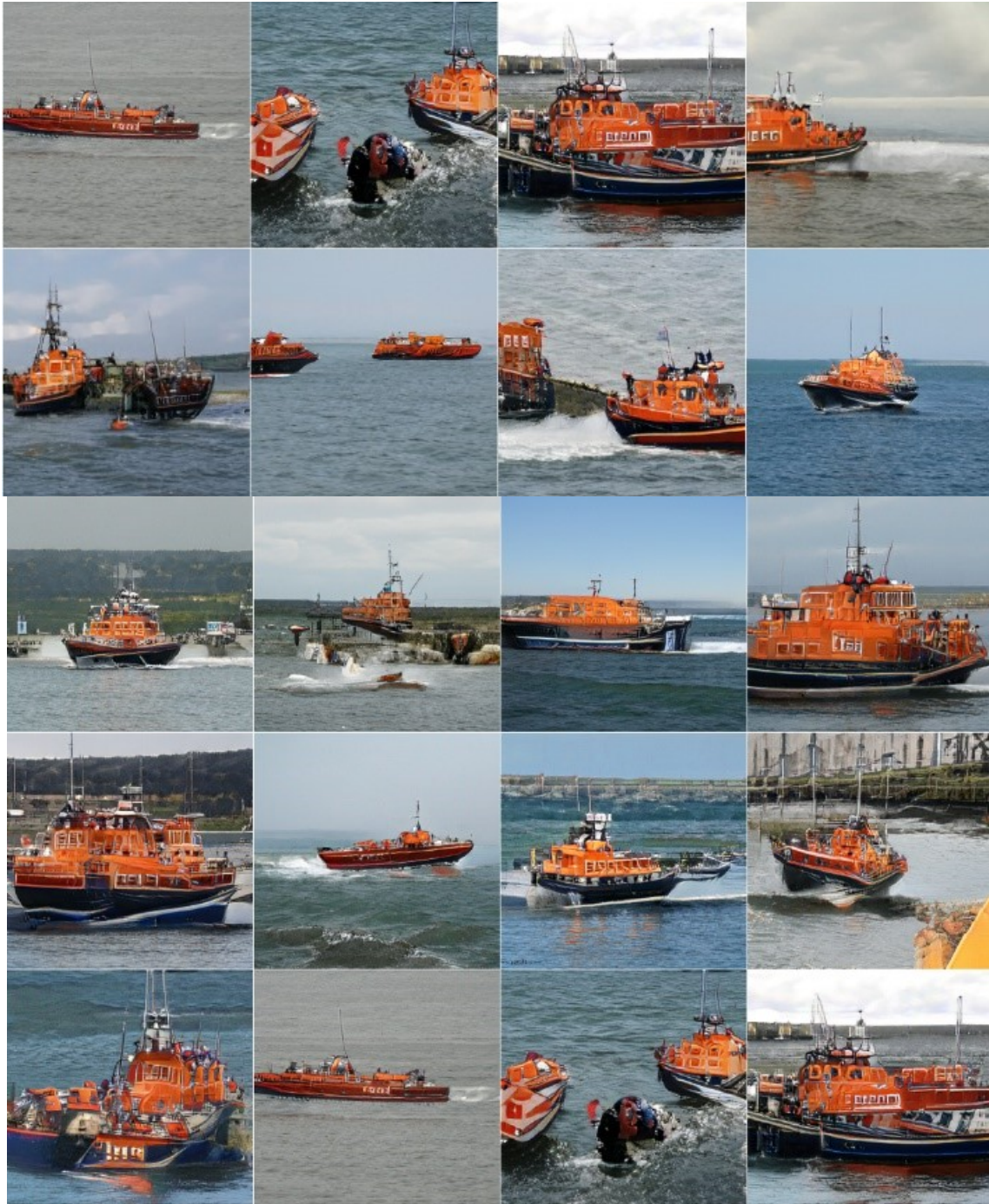


Fig.4.6 Lifeboat large-scale dataset (Lifeboat)

As shown in Fig.4.4– Fig.4.6, the larger the truncation, the greater the diversity of the generated samples. In fact, truncation controls the truncation distance of the hidden variable distribution (generally Gaussian), which is the sampling range. Therefore, it is not difficult to understand its role in diversity. As shown in Figures 7–8, the influence of the value of noise_seed on the generated result is the initial condition of each sample generation, and the final result will be different, which can be used to improve the generation diversity. When training image recognition models of convolutional neural networks and decision models, such as deep reinforcement learning, the quality of the input data considerably affects the effect of the training results. The target image data set generated by GAN has the same image size and image density, which can easily solve the problem of inconsistent input data during the training process. In addition, the GAN model solves many of the scene data that are difficult to obtain in a real navigation environment, making it possible to use large-scale data entry for deep reinforcement learning.



Fig.4.7 Ocean liner Large-scale generated dataset (1)



Fig.4.8 Ocean liner Large-scale generated dataset (2)

4.4 GAN with policy gradient for optimize discrete actions

Although the number of variants of GAN and their versatility is increasing, their adversarial-thinking has not changed. In other words, a discriminator that can identify the real data and generate the data is added in the generation process, so that the generator G and the discriminator D can compete with each other. The role of D is to try to distinguish the real data and the generated data to improve the generated data that can confuse D. When D can no longer separate the true and false data, it is considered that G has reached a stable state.

The numerous advantages are summarized as follows:

- It can generate better samples;
- No need to make inferences about hidden variables during training;
- The model only uses backpropagation without the need for a Markov chain;
- G's parameter update does not come directly from the data sample, however, uses backpropagation from D;
- In theory, as long as the differentiable function can be used to construct D and G, it can be combined with deep neural network to make a deep-generation model.

Part of the agent decision module to complete a collision avoidance evaluation may require different action coordination. These discrete actions are difficult to complete the mathematical differential operation; thus, it is necessary to find a way to converge a global optimal action group. The last of the above advantages is precisely its limitation. In discrete data, data are not continuous like image processing and can be differentiated. Therefore, GAN cannot be realized for discrete data.

As shown in Fig.4.9, when using the deep reinforcement learning model to solve the optimal decision problem, GAN and deep reinforcement learning are combined to select a reasonable global optimal action combination. In Fig4.9a, an adversarial idea is portrayed, where real data from the sea environment plus generated data of G are required to train D. However, from the content described in the related work

section, the discrete output of G is obtained, which makes it difficult for D to return a gradient to update G, and therefore, a few changes need to be made. As shown in Fig4.9b, the value returned by the policy network is G. The existing dot is called current state. The generated next dot operation is called action, because D needs to be a complete sequence score. Thus, the Monte Carlo tree search (MCTS) is used to complete the various possibilities of each action. D rewards these complete sequences, passes information back to G, and updates G by enhanced learning. This is done to use the reinforcement learning method to train a generation network that can generate the global action set.

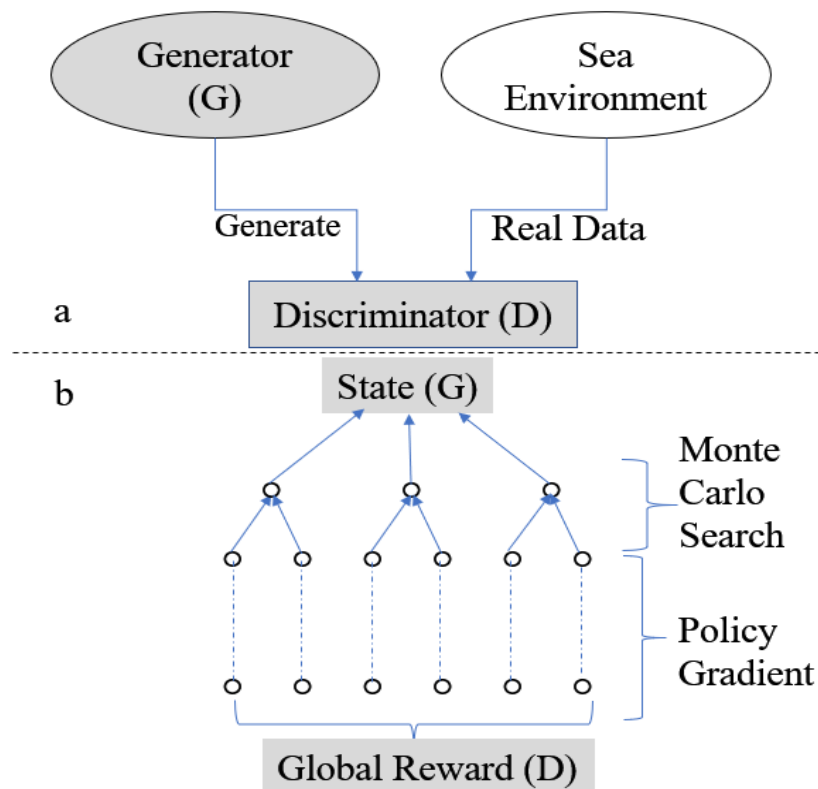


Fig.4.9 Policy gradient convergence of discrete data

4.5 Summary

This study uses the GAN method to implement a large number of generations of decision model training data sets. In the generated data set, according to the settings of the two parameters of truncation and noise_seed, different target ship image data can be obtained. Apart from the different positions of the target ship and state data, the encounter situation of different backgrounds and scenes and image data of various target ships under dangerous conditions are obtained. The target ship image data set generated by the generative adversarial model is useful to train the ship target recognition neural network under different environmental backgrounds, however, for the ship's motion situation prediction, collision avoidance decision, etc., it provided discontinuous data. At the time of processing, GAN did not satisfy this demand. Therefore, this study combines GAN with the idea of policy gradient in deep reinforcement learning, and a method for solving the convergence problem of discrete global action set is creatively proposed.

4.6 References

- [1] Finn, A. & Scheding, S. (2010). Developments and challenges for autonomous unmanned vehicles. *Intelligent Systems Reference Library*, 3, 128-154.
- [2] Trucco, P., Cagno, E., Ruggeri, F. & Grande, O. (2008). A Bayesian Belief Network modelling of organisational factors in risk analysis: A case study in maritime transportation. *Reliability Engineering & System Safety*, 93(6), 845-856.
- [3] Wang, X., Yadav, V., & Balakrishnan, S. N. (2007). Cooperative UAV formation flying with obstacle/collision avoidance. *IEEE Transactions on control systems technology*, 15(4), 672-679.
- [4] Mazurowski, M. A., Habas, P. A., Zurada, J. M., Lo, J. Y., Baker, J. A. & Tourassi, G. D. (2008). Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance. *Neural networks*, 21(2-3), 427-436.

- [5] Scheffer, M., Carpenter, S. R., Lenton, T. M., Bascompte, J., Brock, W., Dakos, V., ... & Pascual, M. (2012). Anticipating critical transitions. *science*, 338(6105), 344-348.
- [6] Capraro, G. T., Farina, A., Griffiths, H. & Wicks, M. C. (2006). Knowledge-based radar signal and data processing: a tutorial review. *IEEE Signal Processing Magazine*, 23(1), 18-29.
- [7] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G. ... & Petersen, S. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540), 529.
- [8] Sarukkai, R. R. (2000). Link prediction and path analysis using Markov chains1. *Computer Networks*, 33(1-6), 377-386.
- [9] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S. ... & Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems* (pp. 2672-2680).
- [10] Zhang, R. L. & Furusho, M. (2017). Conversion timing of seafarer's decision-making for unmanned ship navigation. *TransNav: International Journal on Marine Navigation and Safety of Sea Transportation*, 11.

Chapter 5

Approach autonomous decision in a real AIS environment

5.1 AIS data in collision avoidance

5.1.1 AIS Collision Avoidance Decision System

The purpose of the AIS design was to establish lines of communication between two ships. Shipborne AIS is a global automatic identification system that can transmit large amounts of information such as cross obstacles, weather resistance, ship-to-shore, ship-to-ship information exchange, and close-range target displays. The two ships know the dynamic state of each other and can determine the purpose of the next navigation action before encountering that action. In addition to the related studies on traffic flow forecasting based on ship freight laws, Felski et al. presented results of AIS dynamic information with the highest integrity (position, heading, and speed) and indicated that AIS data, as a powerful complement to navigation aids, including a large amount of related information contained in the data, can play a more powerful role in the future of e-Navigation [1].

To explore the ship collision avoidance decision-making mechanism in a large range, the authors constructed a ship collision avoidance decision-making technical scheme in a real AIS environment. The characteristics of AIS data make it easy to exceed the limits of international navigation rules, and it is currently the easiest to implement a combination of rule-based and neural network-based data types. The authors also analyzed the superiority of AIS data for radar and ARPA data, including the accuracy of AIS data. Moreover, the authors identify the key problems that have long plagued the development of ship collision avoidance decision technology and propose the idea of applying AIS to ship collision avoidance decision systems. The mathematical model of the time collision risk and

the space collision risk of the collision avoidance system in the AIS environment, including its synthetic collision risk, are applied to various water bodies, including vast waters, narrow waterways, and busy waters to facilitate the determination of avoidance.

5.1.2 Rule and Neural Based Decision-Making System

Unmanned ships must have complete autonomous decision-making and intelligent obstacle avoidance capabilities to ensure that unmanned ships can independently perform midterm and long-term remote monitoring, detection, and information gathering tasks. Autonomous decision-making and collision avoidance technology is a key step in realizing highly intelligent and fully autonomous ship navigation. Many research institutes have performed extensive and in-depth research on intelligent ship collision avoidance systems. Figure 1 presents a self-determined collision avoidance system for unmanned ships, which combines rules based and neural based networks. Input data can include all existing sensor data and non-sensor data. The rule-based decision section primarily combines some of the research results of other studies, and most of the major decision-making components are concentrated in the neural-based decision section. This chapter presents a discussion on the regularization of AIS data before it is input as a collision avoidance decision network. The explanations of other sensor data and non-sensor data are concentrated in the discussion section below.

As can be observed from Fig.5.1, the AIS data must be cleaned in the rule-based section. Then two types of data representations are obtained. The first representation denotes text type data, which is used as the input to the neural-based decision network, the other representation denotes the relative relationship between the ship and the target ship on the image, and the image denotes the input training decision on a neural network.

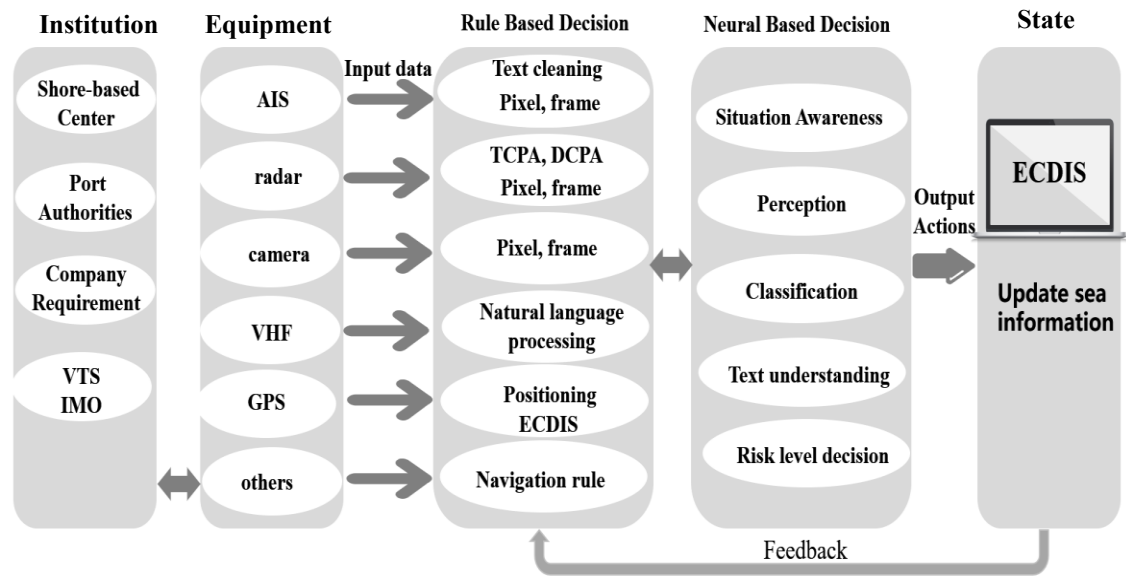


Fig.5.1 Framework of combined rule and neural based network

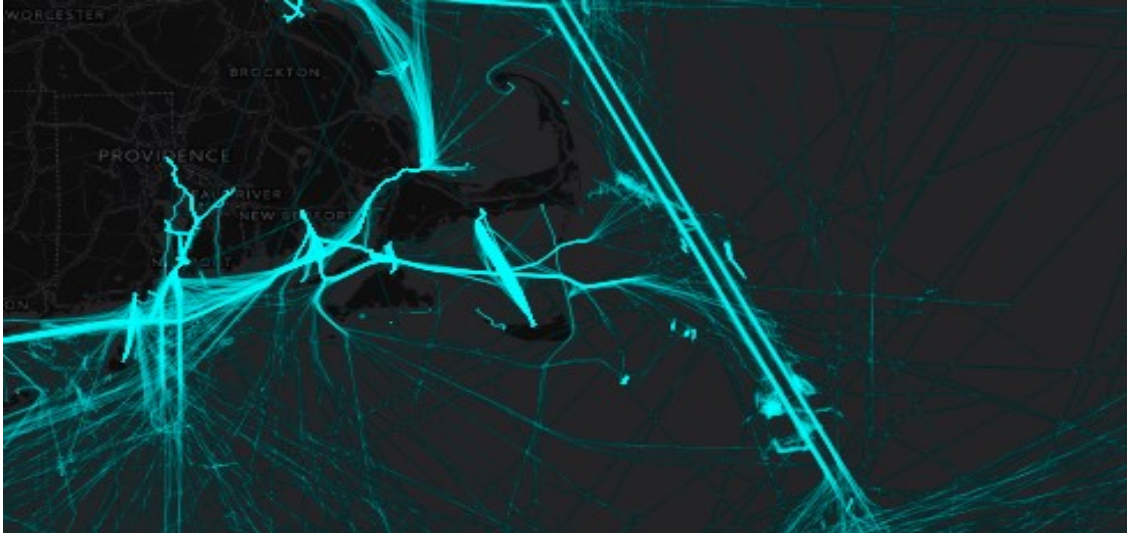
5.1.3 Deep Reinforcement Learning for Collision Avoidance

Deep reinforcement learning examines the manner in which situations can be mapped to actions so as to maximize a numerical reward signal. The agent is not told which actions must be taken, but instead the agent discovers which action yields the most reward by attempting different actions in the environment. In the unmanned ship collision avoidance case, trial-and-error search and delayed reward are the most important features.

Deep Q-Network (DQN) is an important algorithm for deep reinforcement learning. It combines neural networks and Q-learning. The core of Q learning involves transforming the value corresponding to each state and action into a Q table and subsequently determining the same state in the Q table using the search method and obtaining the most valuable action as an optimal action. The core of the DQN algorithm involves replacing the Q table with a multi-layered neural network. This avoids inefficient behavior wherein optimal actions are searched for in the Q table. The collision avoidance problem is essentially a product of the agent's action and the state's environment interaction, and so it is suitable for deep reinforcement learning.

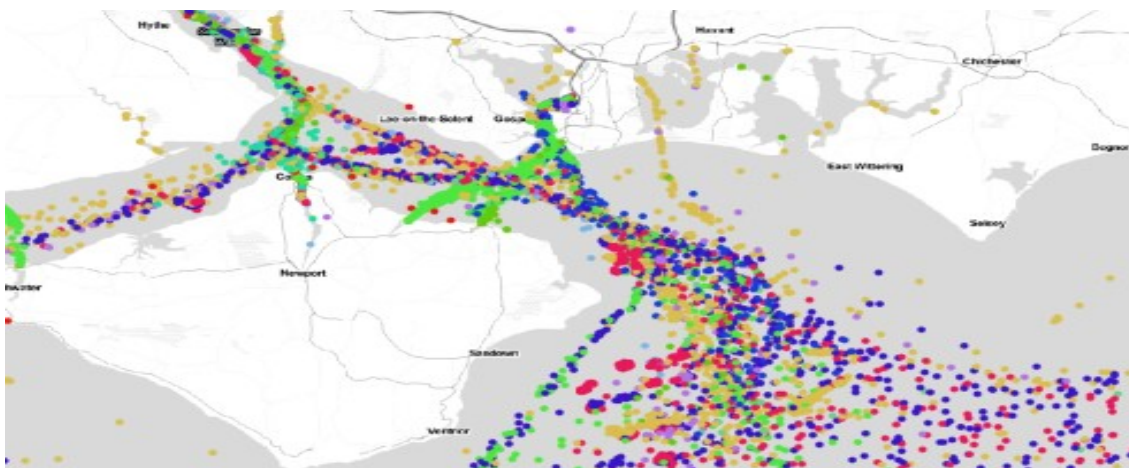
5.2 AIS data processing

5.2.1 Traditional application of AIS



Note: The brightness of light in the legend represents the size of traffic flow. The brighter areas represent greater traffic flow, the more congested the route.

Fig.5.2 Traffic flow planning and prediction (a)



Note: This image from the US Coast Guard website. The dots in different colors just represent different types of ships on the route.

Fig.5.2 Ship type and density analysis (b)

Fig.5.2 AIS traditional applications for VTS ship management

Data from: <https://www.xyht.com/spatial-itgis/using-shipping-data-gis/>

Automatic identification system creates historical data, it generally used to shore to ship communication services, manager traffic flow prediction, improved cargo delivery efficiency are traditional applications of AIS data. As shown in Fig.5.2 can be clearly seen, the figure (a) shows the channels are relatively busy traffic flow in a larger area. As shown in figure (b), through long-term data collection, the location of ship types, accidents density, etc. can be determined. Traditional AIS applications do not require very complex data structures, and many applications only require the positioning and corresponding time of the ship. Even so, it is a very difficult process in the AIS data cleansing and preprocessing phases. Zhao et al. presents the dimensions of AIS data quality for error detection of AIS tracks, including physical integrity, spatial logical integrity, and time accuracy [2]. AIS traffic data mining research will present more applications in the future.

5.2.2 Segment Ship Domain with AIS Data

The concept of the ship domain was first mentioned by Fujii and Tanaka in the article Traffic capacity, as shown in Fig5.3 [3]. Inspired by Goodwin in determining the dimensions of a domain based on radar simulator performance, the authors designed the following program [4]. Demarcate an area around the own ship (red mark). If there are obstacles or other target ships entering the own ship domain, the own ship must adopt the corresponding actions. This domain must include a larger range in the heading direction. To facilitate neural network training, the ship domain of this study is a circle with a radius of 2 nautical miles centered on the own ship. The circle is divided into 36 parts, each of which has an angle of 10° . As shown in the figure, the target ship T1 enters the ship field and the target ship T2 is nearing to make an entrance. The assumption made by our training model is that if the target ship enters the ship domain, the own ship must take action. If the target ship approaches the ship at 1.5 nautical miles, it means a collision will occur and this training will end. The authors determined that the position, direction, and speed of the ship, including the target ship, are derived from real AIS data.

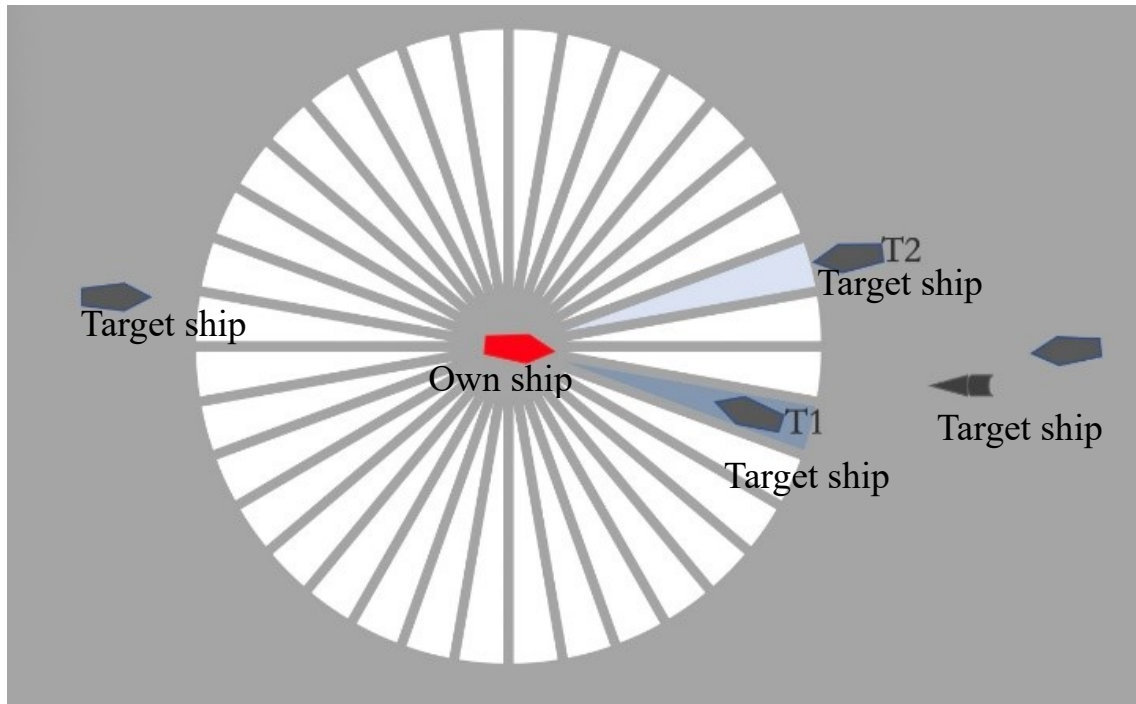


Fig.5.3 Mosaic AIS data with divided ship domain

By analyzing the AIS data from the Chengshantou vessel transportation service center between July and September 2014, the authors draw the following conclusions. The advantage of this training method is that the model is simple, and it is theoretically easy to learn an effective collision avoidance operation in a limited setting. However, the biggest disadvantage of this method is that the amount of input data is large, and there are many invalid calculations. Each batch of training needs to retrieve all the AIS data from the database for this time period. The navigation dynamics of the target ship around the ship must then be obtained, wasting considerable computational resources. Moreover, the ship can only learn how to avoid collision with the surrounding target ship (Local optimization) and cannot grasp the navigation of the entire voyage. The training result cannot or is difficult to achieve the global optimal state.

5.2.3 Frame Motion and Moving States Processing

According to rule-based data processing, the author first uses QGIS to present the data on the map. As shown in Fig.5.4, not only the position of the ship but also the direction of the ship and the subsequent movement situation are required. So how

do it determine the next movement of each ship? Frame motion is a limitation that must overcome. The authors record the position of the ship for each frame and then determine the relative motion relationship based on the difference in the next frame. This method has become a possibility of using currently available GPU computing resources. As shown in Fig.5.5, the image data obtained from QGIS is centered on the own AGENT, while the input image data of (RGB image size 120*120*3) is cropped, and resize it. Then the data is preprocessed. The authors only can choose the selected image, the previous frame is subtracted from the previous frame, and the change in matrix difference after each operation is collected. Linking the frames separated by a certain time, it can determine the general movement of the ship.

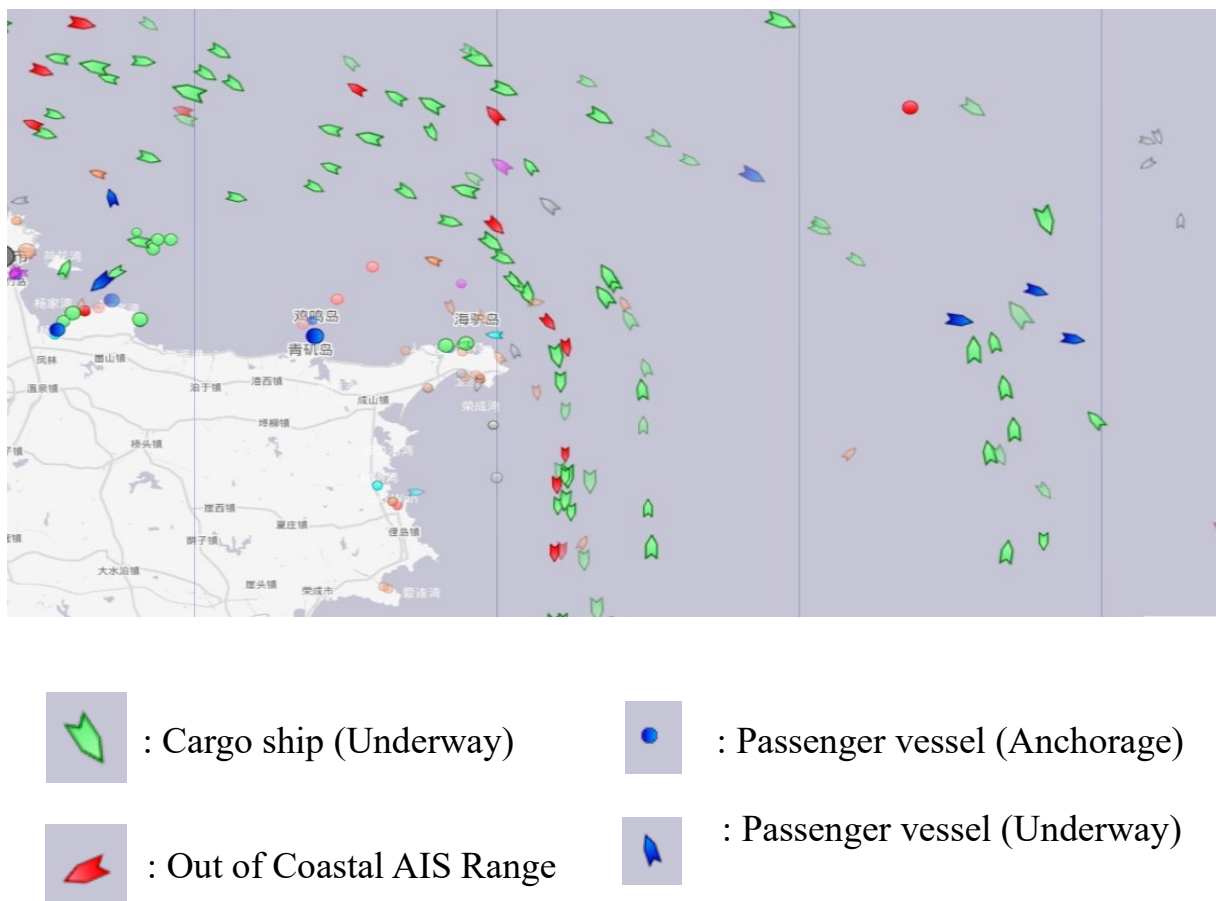


Fig.5.4 Example of the AIS data with heading, ship domain and ship types

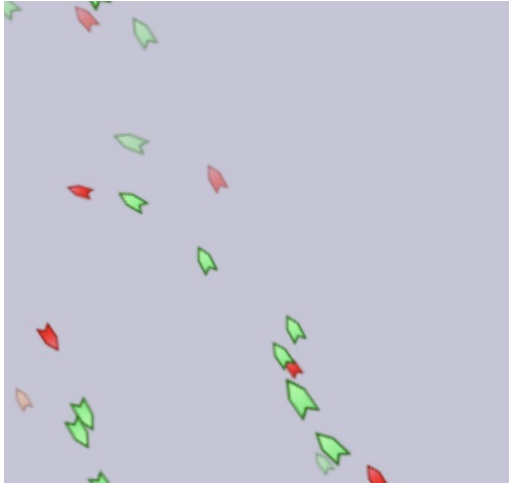


Fig.5.5 (a) Frame No.1

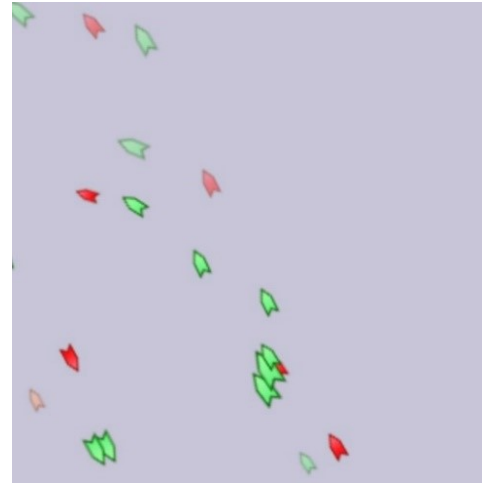


Fig.5.5 (b) Frame No.2

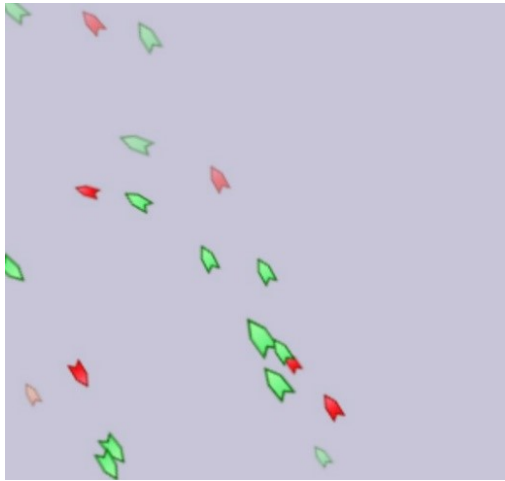


Fig.5.5 (c) Frame No.3

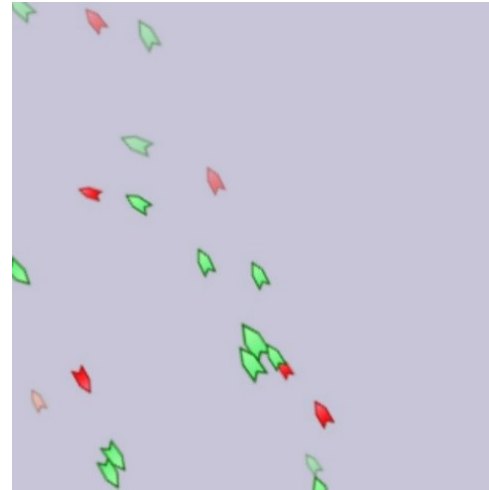


Fig.5.5 (d) Frame No.4



: Cargo ship (Underway)



: Passenger vessel (Anchorage)



: Out of Coastal AIS Range



: Passenger vessel (Underway)

Fig.5.5 Sequent of the frame motion and moving state at the same area

Fig.5.6 presents an example of a target ship image data set that has been simplified. The image data (RGB image size 80*80*3) that the authors collected can be used as a training sample, centered on the ship's agent. When the target ship appears 2 nautical miles away, it is considered safe. The human or any decision maker do not

perform any action or just perform a random action; when the target ship appears within 1.5 to 2 nautical miles, the decision-making model operate a series of actions of the ship for learning purposes. When the target ship is more than 2 nautical miles away from the ship's position, a positive reward is provided to the decision-making neural network. When a target ship appears within 1.5 nautical miles, it indicates a training failure and provides a negative reward to the neural network. This is the ship field method based on a rule-based decision and the training method based on a combination of the neural-based decision network.

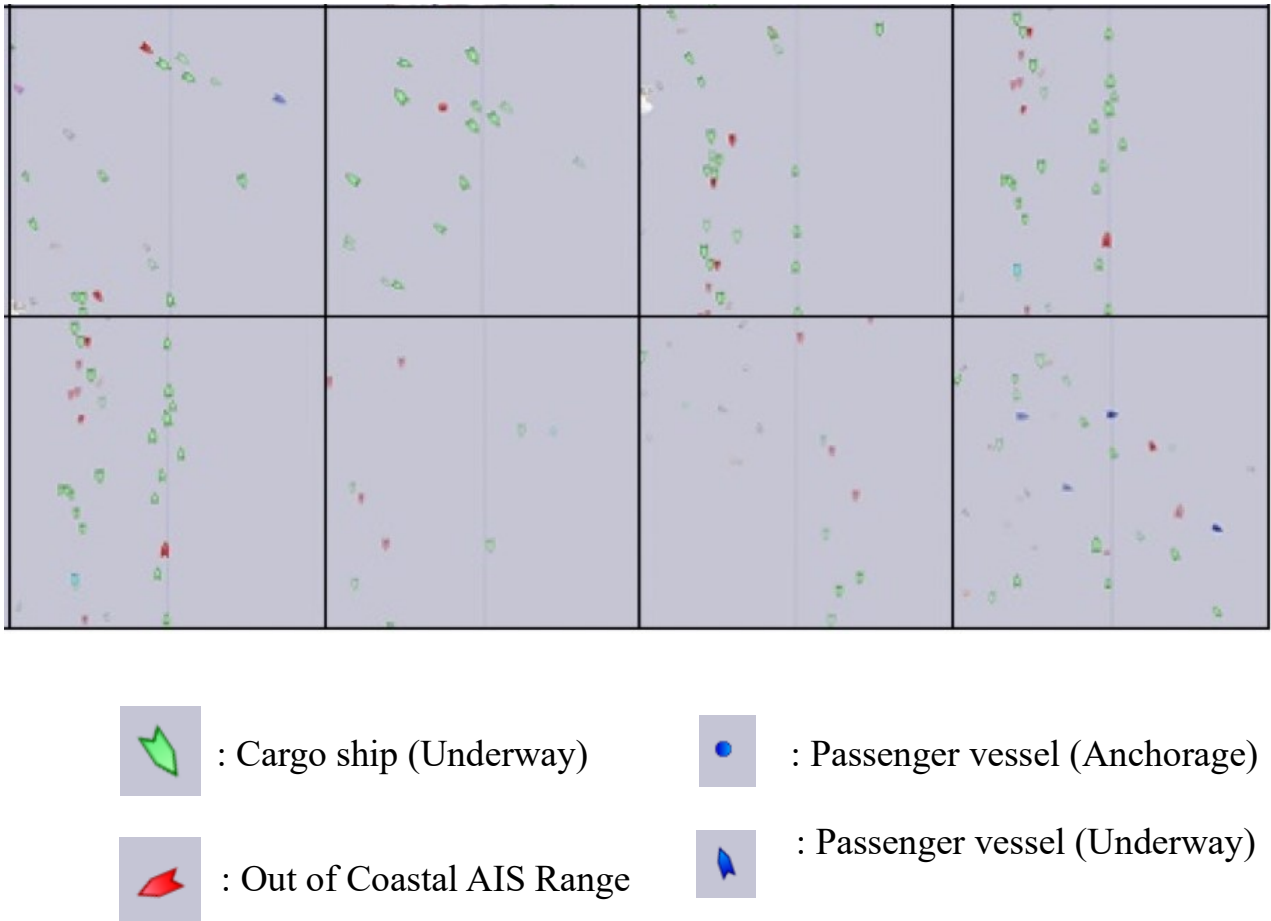


Fig.5.6 Training example of AIS image dataset

5.3 Training decision neural network

5.3.1 Reinforcement Learning Architecture

There are many special optimization methods and policy choices for reinforcement learning algorithms according to different application scenarios. Here, the authors present a discussion on the specific implementation method, a general algorithm called Deep Q-Network (DQN). The main practice of the DQN algorithm corresponds to experience replay wherein the data obtained by the system exploration environment is stored and the parameters are randomly sampled to update the parameters of the deep neural network. The motivation for experience replay is as follows: 1) A deep neural network as a supervised learning model requires data to satisfy independent and identical distribution, 2) However, the sample obtained by the Q Learning algorithm is related before and after. To ensure that the data is not correlated, the experience replay method performs storage-sampling.

Our agent operations (actions) include 7 types of actions, the action space is {forward 10, forward 15, forward 20, port 5, port 10, starboard 5, starboard 10}. In this chapter, the authors constructed a neural based decision network comprising 6 convolutional layers and 4 dense layers. The input data type includes QGIS processed frame motion state images (RGB image size 80*80*3). In our case study, the target network exhibits almost the same structure. The neural network is trained to accept 7 actions with a defined reward. Our reward function is based on the division of weights and angles of the own ship domain. The network uses a mini-batch size of 64, trained with a learning rate of 0.001, and update steps comprising 10,000 steps.

5.3.2 Training Environment

According to the conditions of our laboratory, the authors have been configured the corresponding training experimental environment as follows:

Training framework: PyTorch

GPU = GTX1080Ti & RTX2080

Input Data = frame-by-frame motion image dataset
Input Size = 80*80*3
Output Results = duration times

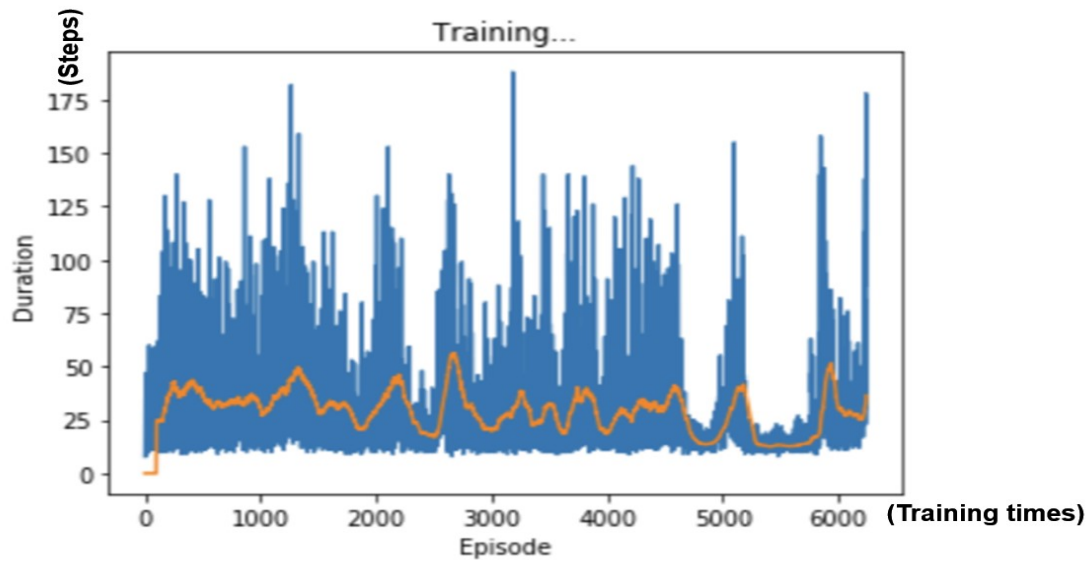
Hyperparameter setting as follows:

BATCH_SIZE = 128
GAMMA = 0.999
EPS_START = 0.9
EPS_END = 0.1
TARGET_UPDATE = 10
RANGE 1 = 6,000
RANGE 2 = 500,000

5.3.3 Training Results

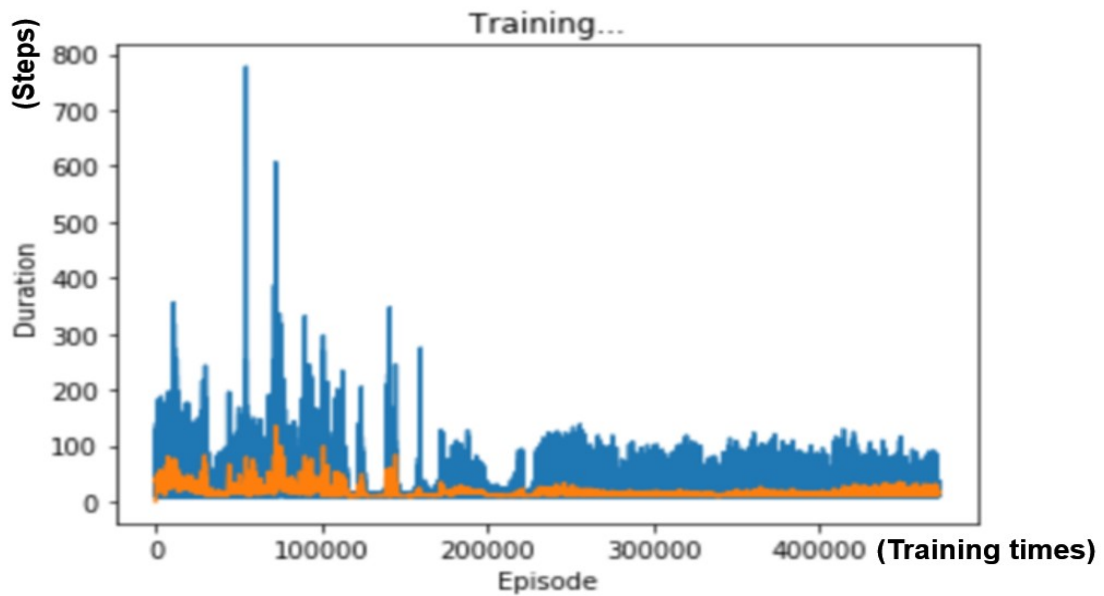
Our training goal is to reach the destination safely from a starting point. According to the set ship domain, the ship takes one of 9 actions that can be taken from a corresponding safe distance. The collision avoidance is completed within a reasonable and limited range of action (around 80 actions), the destination is reached, and the agent ends the training.

The authors trained twice using our configured environment and the DQN decision neural network, the first time training 6,000 times, and the second time training 50,000 times. As can be seen from Fig.5.7, with the progress of training, the action taken by our agent (own ship) to complete collision avoidance is relatively large and no robust. After randomizing the input image data, the training results obtained are shown in Fig.5.8. After about 25,000 training times, our agent can basically guarantee that the collision avoidance operation is completed within the appropriate number of actions. And finally, the agent reaches their destination.



- : The blue curve represents the number of steps required for each training.
- : The orange curve represents the trend of steps change.

Fig.5.7 Unstable collision avoidance actions (6,000 times)



- : The blue curve represents the number of steps required for each training.
- : The orange curve represents the trend of steps change.

Fig.5.8 Stable collision avoidance actions (500,000 times)

5.4 Discussion

The authors collected historical AIS data is the real nautical environment. Such data truly reflects the trajectory and movement state of target ships. The states of these target ships are dynamic and most of their movement complies with the rules of COLREGs. Therefore, the highlight of this chapter is the authors no longer need to think about how to construct rule-based (COLREGs) collision avoidance decision models using deep reinforcement learning. This chapter proposed a new idea for handling decision neural network input data, it is different from any traditional AIS data application. Whether it is ship dynamic data saved in a single text format or image or video format data in frame-based units, this chapter proposed method can handle such historical data by combining a rule-based network and a neural-based network, make our input data diverse and large-scale training. even in the case of a complex situation, it is still able to train smoothly and make sure have a true scene.

With the rapid development of artificial intelligence, it seems obvious that the neural network (neural-based decision-making network) can solve various problems. However, in the process of improving the input efficiency of input data and improving the training effect, combining rule-based decisions will produce better results. In real ship navigation, no one can guarantee that all ships are unmanned and that all ships can voyage in accordance with the COLREGs. In future research, when processing radar data in frame-based units or VHF data based on analog signals, the combined rule-based method and neural-based method for data preprocessing will still have the advantages of guidance direction.

5.5 Summary

This chapter uses deep reinforcement learning to evaluate the performance of decision-making efficiency under different AIS data input shapes. The results show that the decision neural network trained with AIS data has good robustness and a

high ability to achieve collision avoidance. The proposed new training data input mode for risk perception and decision neural network. At present, the research of deep reinforcement learning for navigation of autonomous ships also focuses on the construction of models and the construction of simulation scenarios. The authors use real AIS data and combines traditional concepts of ship domain, to construct a collision avoidance decision neural network. The training results show that the AIS data can be used as input data to the neural network, and then this chapter using a hypothetical agent, to take action in a dynamic and rule-based real sea navigation environment, eventually reach its destination. And pointed out that in future further research (small targets such as fishing boats do not have AIS equipment), using the same methodology, includes instructive guidance for processing radar, camera, ENC, etc., respond to different risk perception tasks in different scenarios. It has important implications for fully unmanned navigation.

The historical AIS data itself can integrate rule-based and neural-based data properties. When using our trained perception model for future risk perception, this model can not only calculate the appropriate collision avoidance decision-making actions based on the outputs of the neural network but also predict the possible state of the target ship and converge the actions of the own ship according to the COLREGs. This data processing method can fundamentally abandon seafarer participation and make it possible to achieve fully unmanned decision-making. In the situation of seafarer absence, our model still can judge whether the actions of the target ship comply with COLLEGs, it also can judge whether the actions of the target ship have good seamanship.

5.6 References

- [1] Felski A, Jaskólski K, Banyś P. (2015): Comprehensive assessment of automatic identification system (AIS) data application to anti-collision manoeuvring. *The Journal of Navigation*, 68(4): 697-717.

- [2] Zhao L, Shi G, Yang J. (2018): Ship trajectories pre-processing based on AIS data. *The Journal of Navigation*, 71(5): 1210-1230.
- [3] Fujii Y, Tanaka K. (1971): Traffic capacity. *The Journal of navigation*, 24(4): 543-552.
- [4] Goodwin E M. (1975): A statistical study of ship domains. *The Journal of navigation*, 28(3): 328-344.

Chapter 6

Ocean Target Detection for Risk Perception Oriented Ship Navigation

6.1 Introduction

Autonomous ships, particularly unmanned ships, will play an important role in the future of the global maritime transportation industry. Ocean-going ships possess the characteristics of long line sight from the bridge, the fishing boat's trajectory is difficult to predict, frequent communication with encounter ships or port authorities in busy waters, and limited ship performance in restricted waters. For the special ocean environment, a diverse and multi-scenario ocean target detection dataset has a great significance for ship navigation, especially for ocean-going autonomous ships, shore-based weak control unmanned ship, target detection model with proprietary ocean dataset is necessary, a visual perception model and decision strategy for risk perception is required.

On the night having complicated conditions, if the decision-maker is human, with shipborne aid equipment and experience, it is relatively easy to recognize the background light from the shore, ship signal light, or other light from the targets at sea. In busy waters, in accordance with the relevant IMO regulations, ships must follow the traffic separation scheme (TSS) if available; the human is responsible for distinguishing between navigation lights or other light spots. Therefore, if the decision-maker is intelligent agent, autonomous ships must be able to solve the challenge faced by maritime transportation. They must distinguish the light from the encounter ship, the light from the coastal background or the light from the navigation beacon.

Compared with driverless cars on city roads, the environment of ship navigation in open water is simpler. However, it also has its challenges, such as the detection

of tiny fishing boat, which are easy to miss even by experienced seafarers. For unmanned ship navigation or current manned assistance pilot systems, although there is some detection equipment such as automatic identification system (AIS) and radar, vision-based perception is still indispensable. These two main types of sensors have limitations. For AIS, there are several targets in the navigation environment without data feedback, such as some ocean construction and tiny fishing boats that do not have AIS equipment installed. Some special-purpose nautical moving targets also deliberately shield their AIS signals, such as warships. On the other hand, radar signal is vulnerable to weather interference, and the effect is weakened. In addition, when there are large obstructions, it is easy to miss some tiny targets blocked by large ones.

According to the nautical reality, for different navigational water areas, the visual-based perception strategy is different. The detection characteristics for different water areas are summarized in Table 6.1.

Table 6.1 Detection characteristics and strategies for different scenarios

Navigation area	Detection characteristics
Open sea	Normal speed, simple encounter situation, few targets, drilling platforms or other stationary ocean targets will appear in certain areas.
Traffic separation scheme (TSS)	Speed control, navigation light detection is particularly important, many different ship types, Fishing boats, special operation ships are intensive, many head-on encounter situations, attention background lights from the coastal area.
Fishing operation	Speed control, fishing boat have different maneuverability, many cross and takeover situation, visual perception is especially important.
Canal water	Speed control, navigation beacon colors are different according to different navigation areas, attention background lights from the shore construction.

In general, visual target perception includes dynamic target detection, such as tiny fishing boats, encounter ships, etc., static target detection such as navigation light, beacon, etc., and navigable area detection. The purpose of this paper is to build a proprietary dataset of ocean targets, use a deep learning method to detect real-time

dynamic and static targets at sea, and implement risk perception for autonomous ships navigation or assistant navigation.

Section 1 introduces the problems that may be encountered in the visual perception of autonomous ships and the importance of building ocean target datasets. Section 2 reviews the current research status of ship detection and computer vision, introduces the main classification neural network, and compares the accuracy and parameter amount of each model. Section 3 compares the ocean target classification performance with various neural networks and selects the training model according to the ocean target characteristics, introduces the specific methods and principles of deep residual learning. Section 4 is based on the dataset collected in this study and the selected neural network framework to perform the training experiments. Using the modified YOLO v3 model for target detection, the author verified the accuracy and efficiency of the model with different strategies. Section 5 discusses the training model and experimental results combined with a real validation set. The conclusions are presented in Section 6.

6.2 Literature review

6.2.1 Ship visual perception tasks and motivations

For a ship's visual perception, most related research uses deep learning to evaluate and imitate the seafarer's risk perception processing, which can represent reasonable actions in similar scenarios in the case of unmanned or assisted navigation. For example, Ping et al. focused on modeling subjective risk perception by drivers using a deep learning method [1]. Integrating environmental features with human risk perception dataset to model individual drivers' subjective risk perception. Kim et al. proposed a visual simultaneous localization and mapping (SLAM) robotic area coverage problem and showed the results of a hybrid simulation using synthetic images and real images in an autonomous underwater hull inspection application [2]. Jeon et al. analyzed and optimized the

delay from camera capture to its target detection module output [3]. Under the premise of ensuring detection accuracy, optimizing the agent's risk perception is a good direction to explore. However, research on large-scale ocean target detection and risk perception from the perspective of the ship bridge is still not conclusive.

Crisp reviewed the Synthetic Aperture Radar (SAR) for detecting ships [4]. The detection of ship wakes is addressed using a basic system of low-to-medium resolution single-channel SAR images, and proposes improvements and alternatives. Shen et al. proposed the detection and classification of ships based on radiated noise, and the detection and classification are implemented by auditory inspired convolutional neural networks trained from raw underwater acoustic signals [5]. Noroozi used silhouette images to extract ship outline features, to identify and classify ships, and silhouette images capture the main external ship features [6]. The advantage is low consumption and quick training, but few training samples lead to low generalization ability.

Tang et al., Zou et al. and Yang et al. proposed high-resolution optical satellite images for automatic ship detection. The detection results are greatly affected by weather conditions (such as clouds and waves) [7] [8] [9]. Higher resolution creates a larger amount of data, and a longer distance captures a smaller target ship, which makes processing more difficult. If it is used for autonomous ship navigation or assist navigation, the real-time performance is insufficient, and the cost is high. Barnum presents ship detection by high-frequency (HF) skywave backscatter over-the-horizon radar (OTHR) [10]. It can achieve maximum sea clutter spectrum purity and thus obtain a greater target signal-to-clutter ratio (SCR). However, the OTHR has limited navigational assistance capabilities for long-distance ocean navigation.

Fefilatyev et al., Wijnhoven et al. and Zhang et al. illustrate different methods, such as online learning, from the buoy camera system, ships, and other non-fixed platforms, to present visual marine surveillance [11, 12] [13] [14]. A new horizon detection scheme has been developed for complex sea areas, which is helpful for

the position and category perception of long-distance ship navigation. However, for the aforementioned various practical problems of navigation, especially in order to improve the detection accuracy and the ability of real-time perception, the scale, categories, and application scenarios of marine datasets have not been addressed yet.

In order to address these problems, the motivation of this study is threefold: the first comprises nautical practice, constructing an ocean target environment using our collected dataset; the second is ship classification, using the transfer learning method to train the accuracy of various classification models, especially for the classification of oil tankers and bulk carriers, which have similar features but completely different maneuverability, the influence of shipwreck pollution also differs. The author collect numerous ocean target data to improve classification accuracy and help make appropriate collision avoidance decisions in dangerous situations. The third is to avoid missing long-distance and tiny targets and have multi-scene adaptability ocean target detection. Some appropriate strategies and a training model based on a deep residual network have been selected to solve tiny target detection insufficient and real-time perception insufficient for ship navigation.

6.2.2. Deep learning-based target detection framework

For a deep learning-based target detection model, the mainstream current target detection algorithms can be roughly divided into two categories: (1) Real-time orientation, this type of detection algorithm does not require a region proposal stage and directly generates objects through one stage. Fast output the class probability and position coordinate, the typical models are YOLO, SSD, and CornerNet [15] [16] [17]; (2) precision-oriented, this type of detection algorithm divides the detection problem into two stages. The first stage is called region proposals, which generates candidate regions that contain the approximate location information of the target, and then the second stage classifies and refines the candidate regions. These typical models are R-CNN, Fast R-CNN, and Faster R-

CNN [18] [19]. The main performance indicators of the target detection model are detection accuracy and efficiency, where accuracy considers the positioning and classification accuracy of targets. In general, the precision-oriented model displays an advantage in accuracy, while the real-time model has an advantage in speed. However, with the development of research, both types of models are improved in two aspects, and both can achieve better accuracy and speed results. The accuracy of the YOLO algorithm does not exceed that of Faster R-CNN, but a tradeoff between speed and accuracy. After many improvements, YOLO v3 not only exhibits a certain accuracy, but also maintains a high training and detection speed. YOLO v3 is favored in many tasks that require high edge computing and real-time performance.

6.2.3. Current status of visual perception

Table 2 summarizes the achievements of the popular object detection algorithms on the ImageNet dataset in recent years. Krizhevsky et al. proposed the early classics of AlexNet in 2012 and Simonyan et al. laboratory present VGG models are pioneering works, after various improvements, as many object detection models have emerged [20] [21]. The research direction is to increase the number of neural network layers while reducing the number of parameters for model training weights. The latest research has increasingly focused on shrinking models and reducing the parameters. It can be seen that there are MobileNet, Inception, etc., but unfortunately, some sacrifices will be made in performance, such as detection accuracy rate [22] [23]. The most famous miniaturization model is EfficientNet, which was proposed by Google, it has only 66 M parameters, but it can reach 97.10% recognition accuracy in the ImageNet dataset. It is close to the current state of the art score. A detailed comparison is shown in Table 2. Among them, the target detection deep residual learning method proposed by He et al., and the series models constructed by ResNet have an important influence on classification and target detection [24]. For example, the ResNeXt model, DenseNet series models, and inception models of different versions [25] [26]. There is a clear improvement in the detection accuracy rate.

Table 6.2 Comparison of Image classification performance on ImageNet.

MODE NAME	TOP 1 ACCURACY	TOP 5 ACCURACY	PARAMS (M)
AlexNet	63.30%	84.60%	60.00
VGG-16	74.40%	91.90%	138.00
VGG-19	74.50%	92.00%	144.00
ResNet-50	81.20%	96.00%	25.00
ResNet-101	78.25%	93.95%	40.00
ResNet-152	79.10%	94.60%	60.00
ResNeXt-101	80.90%	95.60%	83.60
Inception ResNet V2	80.10%	95.10%	55.80
FixResNet-50	79.80%	94.90%	25.60
DenseNet-201	78.54%	94.46%	20.00
DenseNet-264	79.20%	94.71%	25.60
EfficientNet-L2	88.40%	98.70%	480.00
EfficientNet-B7	84.40%	97.10%	66.00
MobileNet-V2	72.56%	90.81%	3.34
Inception V1	69.80%	89.90%	5.00
Inception V2	74.80%	92.20%	11.20
Inception V3	78.80%	94.40%	23.80

6.3 Ocean target classification using a deep residual network

6.3.1 Vision based target detection process

Sort out the visual perception process: a. Camera input, b. Image preprocessing, c. Neural network training and recognition, d. Multiple branches (traffic light recognition, lane line recognition, 2D target recognition to 3D), e. Output results (target type, distance, speed, course of detected target, etc.). Input ocean image data from camera sensors and perform detection, classification, and segmentation based on each frame calculation, and finally use multi-frame information for calculation of multi-target tracking and perception, output-related results.

The core of the above cycle is still a neural network. Its accuracy, training speed, and hardware resource utilization are all indicators that need to be measured and considered. Target detection is most likely to be falsely detected or missed, channel line detection fitting. Some targets such as the navigation aid lights in the traffic separation system are easy to cover in the background light or confused with the encounter ship lights. The complex sea environment detection is difficult, and the passing space boundary has a high accuracy rate and real-time detection. The ocean target classification section of this paper mainly introduces the core methods and features of deep residual learning, and makes relevant experimental comparisons, verifying the classification performance of various models.

6.3.2 Introduction to a deep residual network

He et al. proposed a deep residual network (ResNet) in 2016 [24]. As the number of network layers increases, the problems of gradient explosion and gradient disappearance severely restrict the performance of the neural network. Researchers have alleviated this problem to a certain extent by proposing methods including batch normalization, but the network is still unable to attain the demand. The phenomenon of gradient disappearance makes the network difficult to train. Although the depth of the network is deepened, it is actually impossible to train the network effectively. A network with insufficient training cannot only improve performance but also reduce performance.

The author proposes the construction of identity mapping to solve above problem. ResNet was won first in the classification task of the ImageNet competition in 2015, because it is "simple and practical". Subsequently, several methods were established based on ResNet50 or ResNet101. ResNet is used in detection, segmentation, identification, and other fields. It is also used in Alpha zero, thereby highlighting the ease of usage aspect [27]. The sign of solving the above problem is to increase the number of network layers, but the training error does not increase. The purpose of ResNet is to solve the "degeneration" problem. The design of the residual block makes it easy to learn identity mapping. Even if there are many

blocks stacked, ResNet can make redundant blocks learn into identity mapping without performance degradation. Therefore, the "actual depth" of the network is determined during the training process, ResNet possesses the ability of depth adaptation.

The following three characteristics can be summarized after using ResNet: 1. The deep residual network does not have an advantage in model representation. Therefore, ResNet cannot better characterize a certain aspect, but it allows a deeper characterization of more models layer by layer. 2. The residual network makes the operation of the feed-forward / backpropagation algorithm very smooth. It makes it much easier to optimize deeper models. 3. The “shortcut” connections do not generate additional parameters; it does not increase the complexity of the calculation. Quick connect simply performs identity mapping and adds their output to the output of the overlay. With the back-propagated method of stochastic gradient descent (SGD), the entire network can still be trained as an end-to-end model.

6.3.3. Comparison of ocean target classification tasks

Classification of targets based on visual perception is a simple task. In this study, classification and cross-validation of classification was conducted for various types of ships. Among these, the classification accuracy of oil tankers and large bulk carriers attracted attention because they have many similarities in appearance, but are vastly different in maneuverability and risk management ability. For example, when there is a risk of collision or fire, the two types of ships cause completely different degrees of damage to the ocean environment. In a real voyage environment, it needs to have different decision-making strategies and operation processing in risk assessment.

The author used ImageNet pre-trained models for transfer learning. In this study, the annotated image data of oil tankers and bulk carriers were selected from our ocean target dataset. Oil tankers and bulk carriers, respectively, include 854 training datasets and 350 verification datasets, a total of 2408. The models were

initialized by the ImageNet classification models and then fine-tuned on the ocean target dataset. The author experimented with training common models under the same conditions. As shown in Fig. 6.1, considering the training time and accuracy rate, at our data scale, ResNet18 achieved the best results. At the same time, it also proves that the new network structure may not necessarily achieve good results on our dataset, such as Ma et al. proposed Shufflenet v2, which is guided by the memory access cost and platform characteristics [28]. In the classification training task of ocean targets, the training speed and accuracy rate cannot be considered an ideal model. The dataset and the code for our approach will be publicly available.

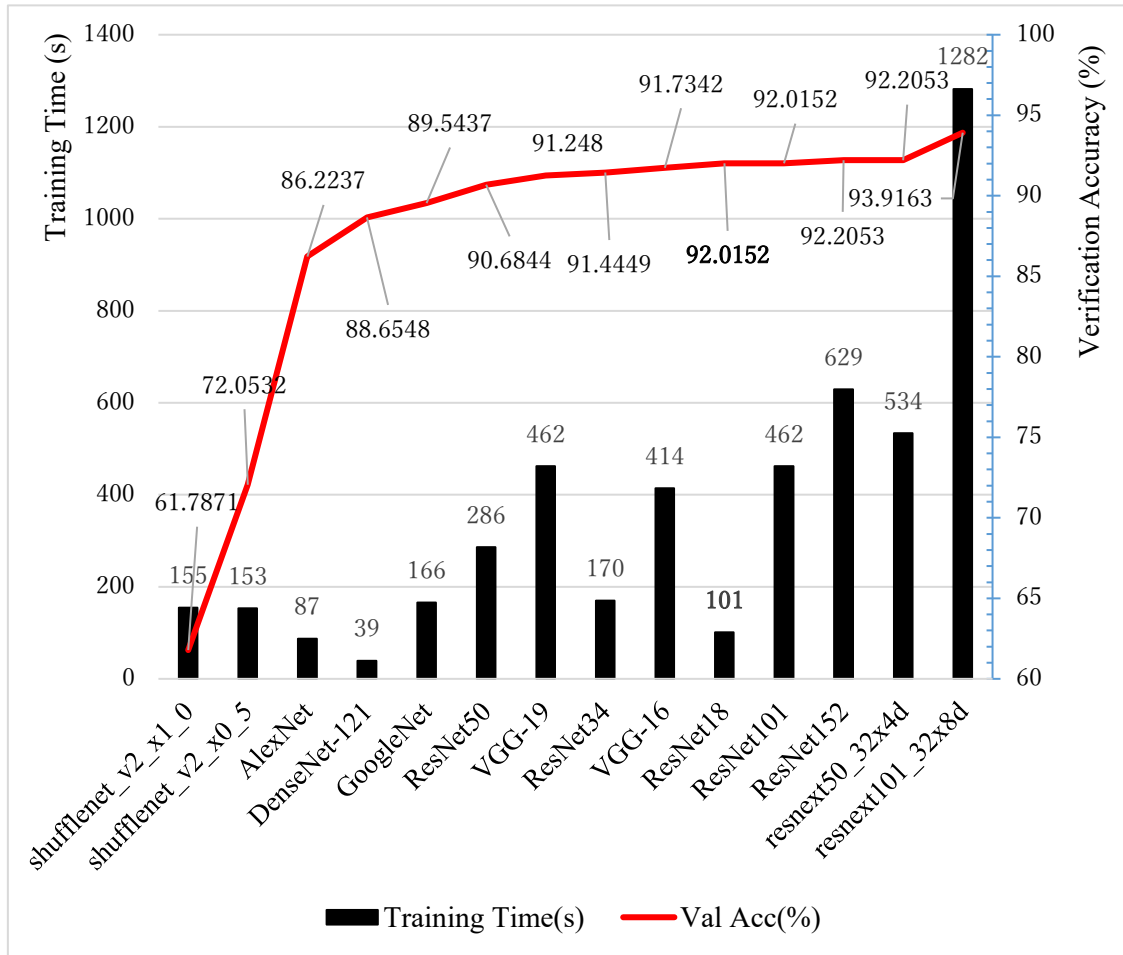


Fig.6.1 Comparison of the accuracy and training time with various network

6.4 Ocean tiny target detection experiment

In this section, the author have introduced our data sources, proposed the structure and design of the training dataset provided innovative labeling strategies, and also compared the training process and training results using a modified YOLO v3 model.

6.4.1 Data set structure design and detection strategy

Majority of the ocean target data is sourced from a ship bridge camera. The bridge camera captures data, including the ship entering the port from Shanghai, Hong Kong, Tokyo, New York, and other world-famous terminals as well as from the Suez Canal, Panama Canal, and other highly reusable navigation scenes. Some of the data used for ocean target classification comes from the practice and research ship "Fukae Maru", which belongs to Kobe University, Mitsui O.S.K. Lines container merchant shipping company, etc. It is also downloaded from the Internet. In some special scenarios, such as some situations that have collision risk, data that are difficult to obtain in a real navigation environment. Zhang et al. proposed a new method that uses the generative adversarial network (GAN) model to create image data for different navigation latent risk scenarios, which can greatly extend the diversity of ocean target datasets [29].

As shown in Fig. 6.2., the ocean target dataset can be divided into four major categories: ship types, ocean obstacles, coastal architecture, and navigation signs. In addition, by adding a light spot and person, a total of 25 small categories, the output classes only the categories in bold font in Fig. 2. In this study, the dataset focuses on the ocean target classification and detection, regarding other tasks such as segmentation, grasping the ship's motion situation, etc. This dataset is generally a subset of the entire ocean navigation dataset.

The dataset used for the target detection model does not include all possible ship types because rare ship types such as timber ships rarely appear in the dataset. For long-distance target detection, our strategy is based on the actual distance of the

target from the own ship. If the target is too far from the own ship, the author first track it as a floating object. Then, with the distance closer, judge and subdivide into different target categories, and then evaluate the risk of such targets. Our classification criteria also include shore-based architectures. The light spots from the shore architecture at night may adversely affect the detection target, so these targets will also include our detection range. Some targets used to guide navigation, such as lighthouse, but long-distance lighthouse is easy to hide in the background noise, so a large number of labeled samples is required, which is one of the focuses of this study. The night environment has a great influence on the image captured by the camera, especially in harsh weather environments, so more labeled samples in such extreme environments are needed. Light spot also adopts different strategies according to the distance. If the light spot is far away, the first step is locked and tracked in real-time, more precise classification according to the change in distance, and then take appropriate decisions.

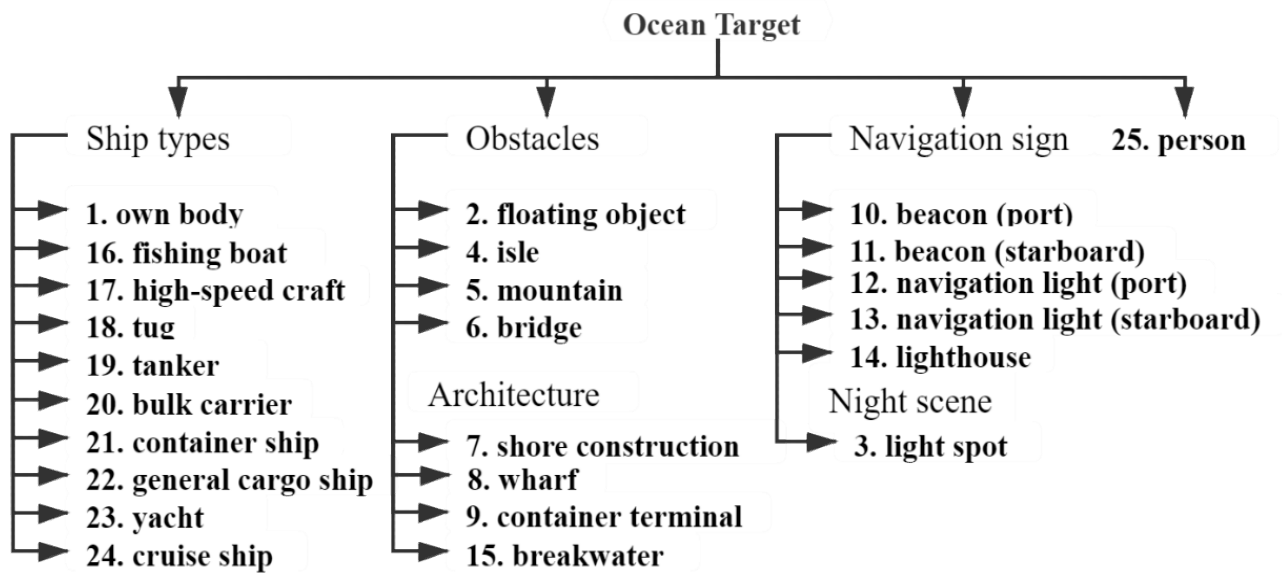


Fig.6.2 Ocean targets dataset hierarchy. Leaf nodes correspond to 25 classes

Compared with the target classification strategy, labeled dataset deployed in different files, the training model accesses the specified folder to obtain image information, and read the position and size information of the corresponding image's annotation frame simultaneously. The main folders include Annotations,

ImageSets, and JPEGImages. Detailed information about the folder is provided in Table 6.3.

Table 6.3 Annotated ocean target image set deployment

Folder name	Characteristic
Annotations	Label file for detection task, xml file format
ImageSets	Store the split dataset, include train, val, test
JPEGImages	Store image files in .jpg format

ImageSets folder store distribution files of datasets contains three subfolders: Layout, Main, and Segmentation, where the Main folder stores dataset files for classification and detection. As shown in Table 6.4, the dataset Train / validation / test used for ocean target detection has a total of 1380 images. Each image contains more than 1 and less than 45 target-labeling boxes, and a total of 8719 labeled target box labels.

Table 6.4 Training samples description in the main folder

File name	Description	Images number
train.txt	Image name for training	880
val.txt	Image name for verification	200
trainval.txt	train and val collection	1080
test.txt	Image name for testing	300

6.4.2 Bounding box strategy and IoU for long-distance tiny target

As shown in Fig. 6.3, for some long-distance targets, there are not enough feature descriptions to recognize whether the target belongs to a moving ship or a fixed ocean obstacle. Yu et al. indicate that a public dataset such as MS COCO or ImageNet is short of tiny targets, an effective Scale Match approach proposed that aligns the object scales between two datasets for favorable tiny-object representation [30]. In maritime practice, the distance changing causes the target size to change greatly, and missing targets will cause unpredictable losses or accidents. Bounding box strategies are necessary for target detection, and feature extension should be marked to make the target easier to detect.



Fig.6.3 Bounding box labeling strategy in task 1



Fig.6.4 Bounding box labeling strategy in task 2

The strategy of this study is to change the size of the bounding box to increase the target feature. The target background in the ocean is usually monotone, changing the bounding box size to obtain more features, obtaining the contrast between the tiny target and ocean background, making it easy to detect tiny targets at long distances. First, the author fix the dimensions of the input image and divide it into same-size grids. The cells in each grid can output the target category and the coordinates of the bounding box. The bounding box can be expressed by the coordinates of the upper left corner of the box (x, y) and the width and height of

the rectangle (w, h). The input image is divided into the same size grid, and there are many small cells. Check the center of the target in the image falls in, in which the cell falls in which the cell is responsible for predicting this target. As shown in Fig. 6.3, task 1 is the labeled image obtained by the normal labeling strategy. In Fig. 6.4, task 2 is to intentionally increase the size of the bounding box, and it obtain another labeling dataset. Finally, the author compares the training results and evaluate the tiny target detection efficiency. In the training phase, if the target center falls on this cell, the cell is labeled with the target's label (including x, y, w, h, and class). In other words, by setting the training label in this way, the cell is taught to predict the target in the image. In the test phase because the model has been learned to predict the target that falls in the cell during the training phase, the test phase is to compare whether the labeled box corresponds to the predicted cell. If it corresponds, the predicted target is correct.



Fig.6.5 Example of intersection over union (IoU)

Target detection defines the intersection over union (IoU), as shown in Fig. 6.5. The red box (target box) is the bounding box where the real target is located, and the green box (anchor box) is that the bounding box slide in the image to predict the size and position of the target, and IoU is the ratio of the area of overlap to the area of union. As shown in formula 1.

$$IoU = \frac{|B^p \cap B^{gt}|}{|B^p \cup B^{gt}|} \quad (1)$$

Where $B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground truth, and $B^p = (x^p, y^p, w^p, h^p)$ is the predicted box. The expression of IoU Loss can be written as formula 2:

$$L_{IoU} = 1 - \frac{|B^p \cap B^{gt}|}{|B^p \cup B^{gt}|} \quad (2)$$

From the perspective of IoU to solve the regression loss, its disadvantage is that when two boxes do not intersect, L_{IoU} is always 1 and cannot give a specific optimization direction. L_{GIoU} can solve this problem:

$$L_{GIoU} = 1 - IoU + \frac{|C - B^p \cup B^{gt}|}{|C|} \quad (3)$$

Where C represents the smallest rectangle containing two boxes, so that the situation where the two boxes no intersection also can be optimized. However, there is still a problem with L_{GIoU} . When two boxes intersect, the L_{GIoU} degenerates into a L_{IoU} . As a result, optimization becomes very difficult when the prediction box and ground truth box are included, especially in the horizontal and vertical directions. Therefore, Zheng et al. mentioned method has been used. Distance-IoU Loss as shown formula 4 [31].

$$L_{DIOU} = 1 - IoU + \frac{\rho^2(B^p, B^{gt})}{c^2} \quad (4)$$

Among them, B^p and B^{gt} in the loss function respectively represent the center points of the anchor box and the target box, and represent the calculation of the Euclidean distance between the two center points, and c represents the diagonal distance of the smallest rectangle that can cover both the anchor box and the target box. Therefore, the normalized distance between the anchor box and the target box is modeled in DIOU, and the normalized distance between the anchor box and the target box is directly minimized to achieve a faster convergence speed.

Advantages of Distance-IoU Loss: 1. Similar to the GIoU loss, the DIOU loss can provide the movement direction for the bounding box when it does not overlap with the target box. 2. The DIOU loss can directly minimize the distance between two target boxes, so it converges much faster than the GIoU loss. 3. For the case

of two boxes in the horizontal and vertical directions, the DIoU loss can make the regression loss drop very quickly, while the GIoU loss almost degenerates into the IoU loss.

6.4.3 Model training using modified YOLO v3 framework

This experiment uses the YOLO v3 framework provided by Redmon et al. to train the ocean target dataset. YOLO's backbone uses ResNet, and it can achieve real-time target detection above 35 fps [32]. After the R-CNN becoming more and more mature, the Yolo surpasses the master, which is inseparable from its high performance and the use of regression ideas for target detection.

Yolo paper mentioned the following three advantages: 1. Yolo directly returns the position and gives up a complicated framework; the detection efficiency is fast. 2. Lower false detection rate from the background, Yolo will make predictions based on the entire image information, while other sliding window detection frameworks only infer based on partial image information. 3. The image features learned by Yolo have good versatility. The author tried to use the natural picture dataset for training, and used art works to make predictions, and Yolo's detection effect is better.

The author use Darknet as a benchmark. Except for changing the labeling strategy of the bounding box size and the optimization techniques of IoU, the framework contents of others remain unchanged. Train our dataset on GTX1080Ti, divided into YOLO v3 original task, task1, task2 with different labeling strategies, DIoU-Task 1, DIoU-Task 2 with different IoU optimization methods, a total of five training tasks, a single training time of 94 – 118 h.

6.4.4 Target recognition model performance evaluation criterion

This section presents the ocean target detection performance and evaluation criteria. Each model evaluates performance on the same "validation / test" dataset, performance measurement uses various statistics such as accuracy, precision, recall, etc. The selected statistics are usually for specific application scenarios and

examples. The precision calculation is the correct sample number divided by the total number. Recall is the correct number of data divided by the ground truth of the positive data number. For each application scenario, it is important to choose a fair-minded evaluation criterion that can objectively compare the model's performance. The most commonly used evaluation indicator in object detection is the mean average precision (mAP). Each time an image is detected, whether it is a positive or negative sample, the current precision and recall are calculated. Suppose there are M positive examples in every N samples in the detection, then M recall values ($1 / M, 2 / M, \dots, M / M$) will be recorded. For each recall value r , corresponding to ($r' > r$), the maximum precision can be calculated. Then, average these max precisions to obtain the final AP value. In multi-class detection, the average value of each class of AP is taken as mAP.

6.4.5 Training result verification and testing

Our dataset is divided into Task 1 and Task 2 training according to the different labeling strategies of the bounding box. Fig. 6.6 shows a comparison of the IoU values of Task1 and Task 2 in the start and end stages of the training process. As shown in Fig. 6.6, even if the author adopt different strategies for the labeling strategy, it will not significantly impact the value of IoU, which proves that our labeling strategy is desirable.

Because the background of ocean floating objects in most cases is a single color, if viewed from the perspective of the ship bridge. The larger callout bounding box can not only capture the characteristics of the target completely, but also maximize the difference between the target and the background. Such a labeling strategy is conducive to target detection.

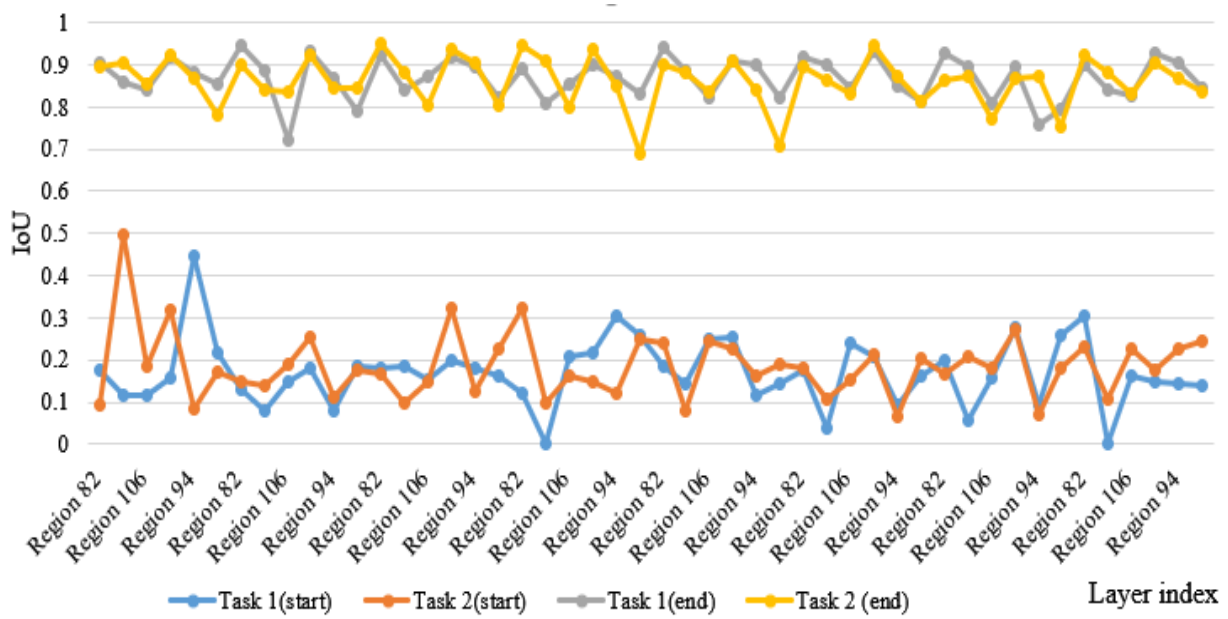


Fig.6.6 IoU distribution of task1 and task2 in the start and end phases

Table 6.5 is a record result example of Task 2 at the end of training. From the data in the table, a stable training process can be determined. The region number is the index of the yolk layer in the configuration file, 82 represents the large target layer, 94 is the medium target layer, and 106 is the small target layer. “Avg IoU” means that in the current iteration, the average intersection ratio of the predicted box and the marked box is as large as possible. The expected value is 1, which indicates that our predicted bounding box and the manually marked bounding box completely coincide. It can be seen from the table that in this training episode, our predicted box IoU can reach 0.876829. “Class” indicates the classification accuracy of the marked target, the larger the better, the expected value is 1, and the average value of this experiment reaches 0.998109. “obj” represents the larger the better, the expected value is 1. “No obj” indicates the smaller the better. “.5R” represents recall, when threshold of IoU equal 0.5, where recall equal detected positive samples divide by actual positive samples. “.75R” is the recall when the threshold of IoU is equal to 0.75. “Count” denotes the number of positive samples.

Table 6.5 Task 2 records at the end of one training process

Layer index	Avg IoU	Class	Obj	No Obj	.5R	.75R	count
Region 82	0.946220	0.999843	0.999989	0.004368	1.000000	1.000000	5
Region 94	0.899563	0.999912	1.000000	0.000344	1.000000	1.000000	1
Region 106	0.740491	0.999787	0.968657	0.001200	0.925000	0.550000	40
Region 82	0.930161	0.999835	0.999949	0.005983	1.000000	1.000000	7
Region 94	0.933708	0.999955	1.000000	0.000669	1.000000	1.000000	3
Region 106	0.857121	0.999994	0.999160	0.001451	1.000000	0.875000	24
Region 82	0.939510	0.999820	0.999996	0.005817	1.000000	1.000000	7
Region 94	0.858488	0.999972	0.999820	0.000892	1.000000	1.000000	4
Region 106	0.834058	0.999828	0.991600	0.001622	1.000000	0.857143	42
Region 82	0.955714	0.999929	0.998747	0.005378	1.000000	1.000000	6
Region 94	0.844861	0.972345	0.998523	0.001829	1.000000	0.800000	10
Region 106	0.836841	0.999911	0.991464	0.001748	1.000000	0.914286	35
Region 82	0.900130	0.994027	0.999449	0.007208	1.000000	1.000000	6
Region 94	0.851039	0.996243	0.999997	0.000526	1.000000	1.000000	3
Region 106	0.828446	0.999544	0.996865	0.001547	1.000000	0.878788	33
Region 82	0.944598	0.999984	0.999986	0.003675	1.000000	1.000000	4
Region 94	0.849926	0.999944	0.999997	0.000733	1.000000	1.000000	4
Region 106	0.840797	0.999825	0.997381	0.001458	1.000000	0.914286	35
Region 82	0.952194	0.999667	0.999982	0.006621	1.000000	1.000000	7
Region 94	0.889266	0.999998	0.999917	0.001055	1.000000	1.000000	6
Region 106	0.780283	0.999941	0.998098	0.000890	0.920000	0.720000	25

As shown in Fig. 6.7, the abscissa represents the number of images tested. The ordinate on the left is the number of sample targets in the test image, the orange line represents the total number of targets, and the red line represents the number of samples predicted correctly. The ordinate on the right represents the average recall value. There will be some fluctuations when the number of samples is small, but the value of recall will remain around 90% after becoming stable.

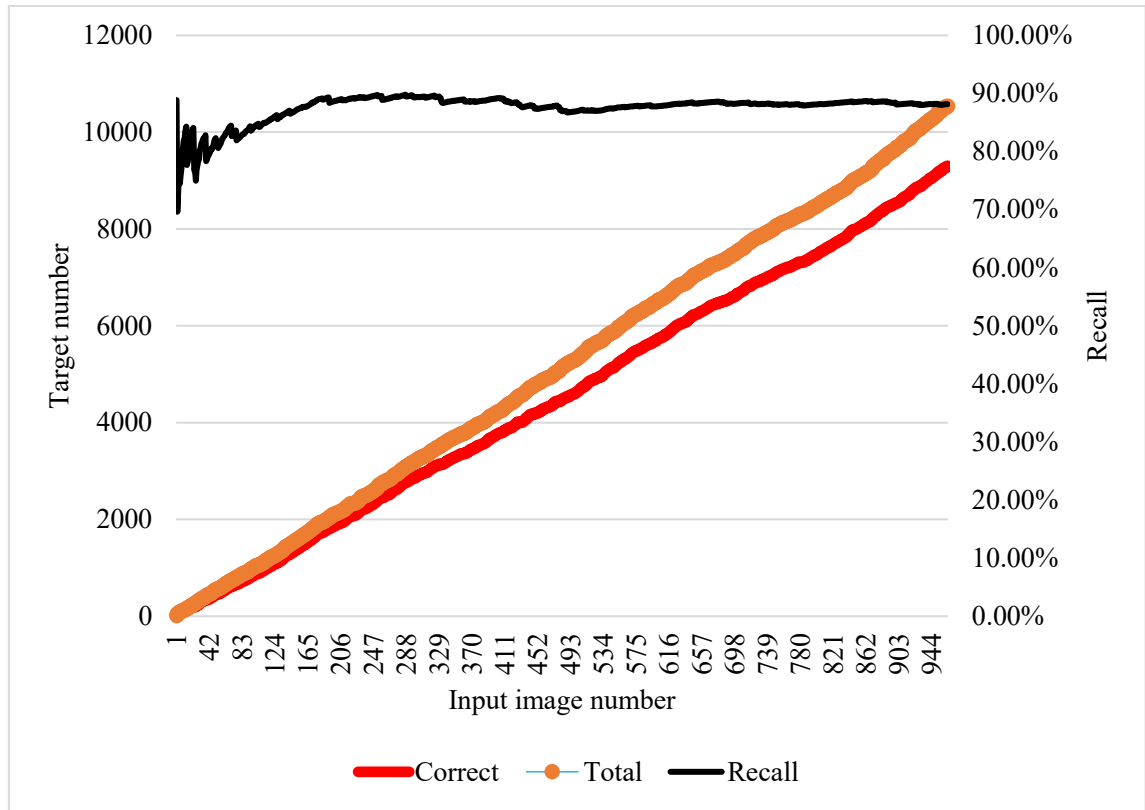


Fig. 6.7 Task 2 test results for recall

Table 6.6 shows the training time and mAP values for different tasks. YOLO v3 is the original framework training time and mAP obtained by randomly marking the ocean target dataset, which is similar to the results obtained by Task1. Task 1 and Task 2 due to different labeling strategies, it can be seen that the mAP obtained by Task 2 is significantly higher than that of Task 1. Task1 and DIoU-Task 1 have the same labeling strategy, and the training dataset is consistent. The training time is shorter, but the accuracy is higher. The time spent on the five training sessions is reduced according to the improvement of the IoU strategy. The training results indicate that the mAP shows a linear change, indicating that our labeling strategy and IoU improvement can help improve the efficiency and performance of ocean tiny target detection.

Table 6.6 Displaying the accuracy tradeoff on the mAP at .5 IoU metric.

Method	YOLO v3	Task 1	Task 2	DIoU-Task 1	DIoU-Task 2
Train time (h)	112	105	118	94	97
mAP-50	50.8	49.5	52.4	51.6	53.1

6.5 Consideration

6.5.1 Ocean target detection in different scenarios

The perception strategy of this study can be summarized as follows: ships are not classified by category, but by their maneuverability and the risk level to their own ship. When a tiny target appears in the distance, if the resolution is not clear, our strategy is selected as a floating object, keeps tracking, and then updates the detection result. The float object may be a small fishing boat, or it may be another type of ship or a fixed obstacle. Shore architecture is divided into navigation aids and obstacles, which are detected and perceived, respectively. Night lights are divided into navigation lights, obstacle spots, and shore-based lights.

The classification of the dataset accounts for changes in the environment during the night and day, which are likely to seriously impact visual perception. At night, the navigation beacon and the lights on the fishing boat are easily confused, and tiny targets in the vicinity are easily covered by large light sources in the distance. The distance between the bow and the stern of a large ship is too long, and the lights between the bow and stern are too far, which is easily considered to be two scattered targets. In this study, ship wakes based on images and videos considered whether a ship belongs to a high-speed ship, so as to appropriately increase its level of risk perception. Human detection is easy to miss, but human safety is the imperative at all times, which requires careful consideration. Therefore, when designing our dataset structure, the author separated people into one category. (Yu et al., 2020) [29] also mentioned that during the maritime rescue process, the detection of people who fell into the water at a long distance, which provided new ideas for the collection and training of tiny target datasets.



Fig.6.8 The detection effect of small targets at long distances at night

Our ocean target dataset includes various scenes. As shown in Fig. 6.8, the port lighting environment at night is complex, long-distance, and many tiny targets. The dark light environment and over-exposure environment adversely affect target detection. Our model was prepared to identify the light from the shore and the light emitted by the floating objects near the shore.



Fig.6.9 The recognition effect of the port and starboard lights

The traffic separation system water area usually requires special attention to the navigation lights and other floating objects lights nearby at night. As shown in Fig. 6.9, our training results can clearly distinguish the ship's target, navigation lights, and lights from the coastal background. In particular, the identification of the port and starboard navigation lights is clearly visible.

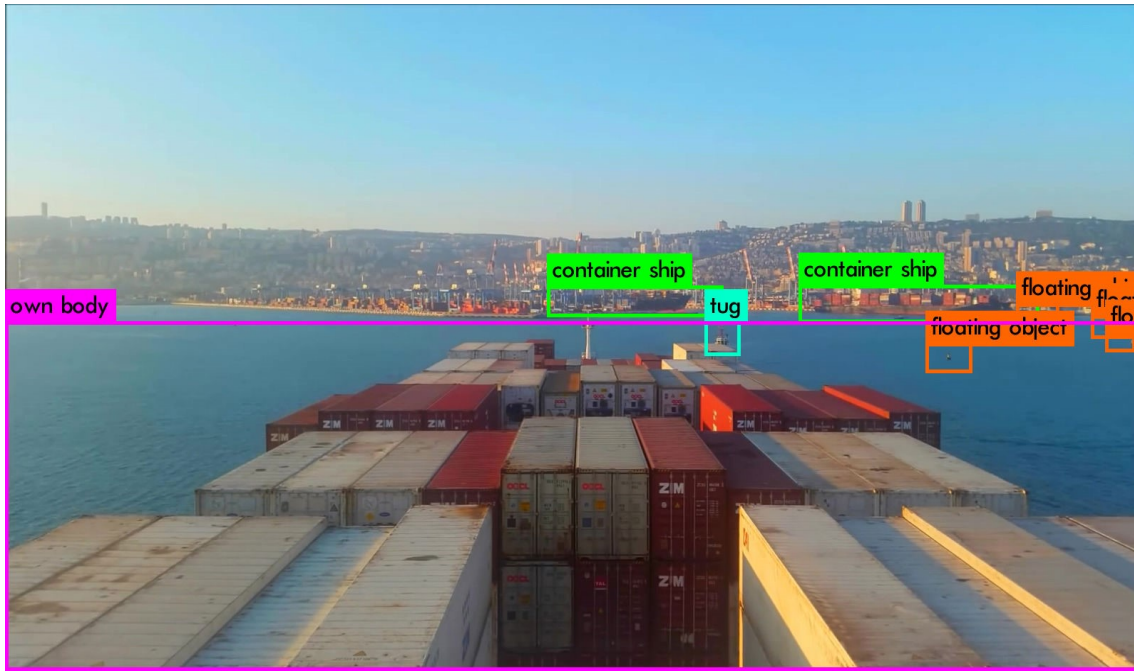


Fig.6.10 The ability to detect partially occluded objects is remarkable

In Fig.6.10, the tug that is not easy to find in the bow of the ship is detected. Although only a part of the tug mast is exposed, the tug features learned by our model ensure accurate recognition performance. In addition, on the shore with complicated background noise in Fig.6.10, a partially blocked container ship can also be detected.

When there is a lot of noise in the background, it is particularly easy to miss some sea floating objects with similar characteristics. For example, as shown in Fig.6.10, the feature of many buildings on the shore is similar to the shape of a container ship, which has always been a difficult issue in target detection. Our model can handle similar targets well and get correct detection results.

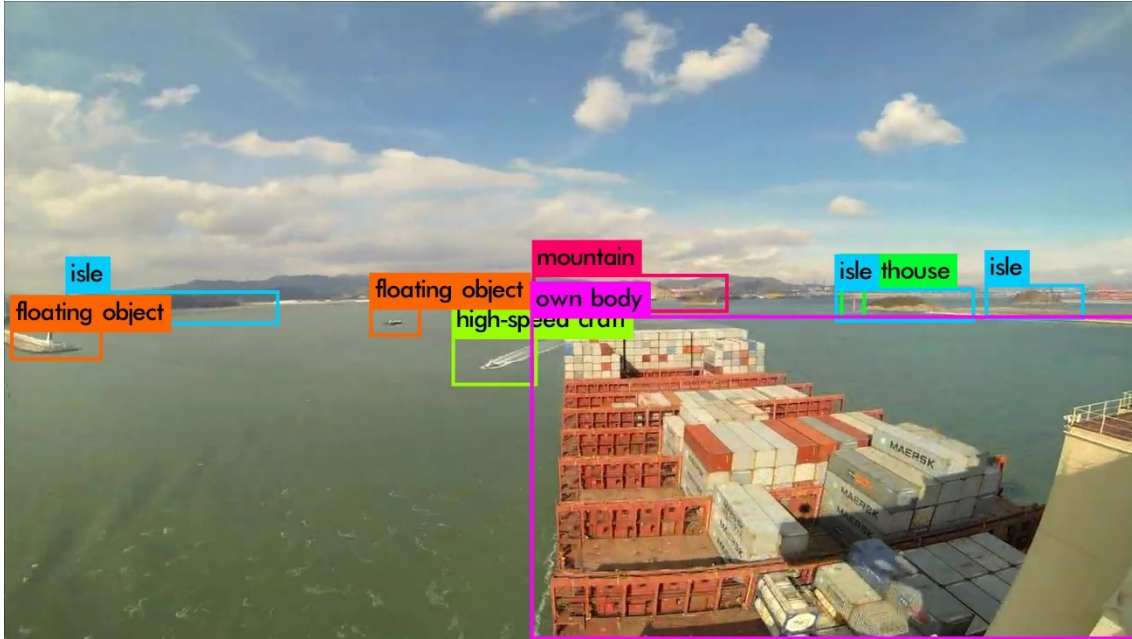


Fig.6.11 The ability to detect small targets from complex background noise

As shown in Fig. 6.11, in the port or/and canal area, the traffic flow is complicated, and there are many types of ships. In particular, the trajectory of the fishing boat is difficult to predict, which requires the detection model to miss any possible targets. Our model can locate targets in an environment with complex background noise, as shown in Fig.6.11. On the right side, at the end of the breakwater, lighthouses, which are important meaningful assistant architecture are still detected. This is a great help for autonomous navigation and assisted navigation.

6.5.2 Improve end-to-end risk perception efficiency

Using camera images as the input, an intelligent agent to make appropriate decisions and control, which can be regarded as an ideal end-to-end perception-decision process. Camera data contain rich color information; therefore, it is necessary to rely on computer vision technology for obstacle recognition, navigation light detection, lane detection, and traffic sign detection. However, there is a big difference between camera-based detection and other perception equipment, which is distance. There is no meaning if the unmanned agent merely has knowledge of an obstacle in front of it. The distance or 3D coordinates of the

obstacle is required. This problem can be easily combatted using laser obstacle detection because the laser itself contains distance information, but it is difficult to calculate the distance using only the image information.

There are multiple methods for calculating distance information:

- I. Laser ranging. The principle aims to calculate the distance based on the laser reflected back time. The distance calculated in this way is the most accurate. Real-time detection of targets and automatic laser data fusion, and adaptive capabilities are required.
- II. Monocular depth estimation. The principle is that using a monocular camera image as input, a CNN model of depth estimation used for prediction, each pixel depth as output. The advantage of this method is that the frequency can be higher, but has a high estimated error.
- III. Binocular distance measurements. The distance is calculated based on the slight difference seen by the two lenses. The disadvantage is that the distance error is large if the object is too far.

The main disadvantage of these methods is that structured light is greatly affected by natural light, and it needs to be trained with neural networks in a large number of nautical actual environments. For this, a large amount of real navigation datasets is particularly important. In addition, a more accurate description of the environment in which the intelligent agent is located is required, which is significant for end-to-end risk perception. This research provides a feasible idea for future research.

6.6 Summary

Our ocean target datasets were divided into two categories: 1. Various types of ocean target datasets for identification and classification, where each image was cut into a single target and saved; 2. Full view image / video datasets collected

from ship bridge perspectives in ship actual navigation. This contributes to the automatic classification of the targets of autonomous ships navigation, ship navigation assistance systems, and tracking latent risk targets, especially for the risk perception of the navigation status of ships, such as tiny targets. This study uses transfer learning to train different models and obtain the classification accuracy of the ocean targets. The detection efficiency of various neural network models has been compared with respect to the model parameters, training accuracy, and time consumption.

To detect long-distance tiny targets, different bounding boxes were designed. To adapt to the characteristics of ocean targets, different classification strategies were adopted. To achieve a faster convergence speed and minimize the normalized distance between the anchor box and the target box, different intersection over union loss functions were adopted. Using a deep residual network and modified YOLO v3 target detection framework, the real-time detection efficiency of our dataset has been implemented and verified. The performance and efficiency of the target recognition model on the actual collected ocean target recognition dataset has been improved. The author provided a relatively complete and reliable target recognition capability for ship autonomous navigation or assistant navigation.

6.7 References

- [1] Ping, P., Sheng, Y., Qin, W., Miyajima, C., & Takeda, K. (2018). Modeling driver risk perception on city roads using deep learning. *IEEE Access*, 6, 68850-68866.
- [2] Kim, A., & Eustice, R. M. (2013, May). Perception-driven navigation: Active visual SLAM for robotic area coverage. In *2013 IEEE International Conference on Robotics and Automation* (pp. 3196-3203). IEEE.
- [3] Jeon, W., Kang, K., & Kim, J. C. End-to-End Delay Analysis and Optimization of Object Detection Module for Autonomous Driving.

- [4] Crisp, D. J. (2004). The state-of-the-art in ship detection in synthetic aperture radar imagery (No. DSTO-RR-0272). Defence Science And Technology Organisation Salisbury (Australia) Info Sciences Lab.
- [5] Shen, S., Yang, H., Li, J., Xu, G., & Sheng, M. (2018). Auditory inspired convolutional neural networks for ship type classification with raw hydrophone data. *Entropy*, 20(12), 990.
- [6] Noroozi, M. R., Ramezani, A., & Aghababae, M. (2014). Automatic ship types classification in silhouette images. *International Journal of Engineering and Advanced Technoogy (IJEAT)*, 4(1), 52-56.
- [7] Tang, J., Deng, C., Huang, G. B., & Zhao, B. (2014). Compressed-domain ship detection on spaceborne optical image using deep neural network and extreme learning machine. *IEEE Transactions on Geoscience and Remote Sensing*, 53(3), 1174-1185.
- [8] Yang, G., Li, B., Ji, S., Gao, F., & Xu, Q. (2013). Ship detection from optical satellite images based on sea surface analysis. *IEEE Geoscience and Remote Sensing Letters*, 11(3), 641-645.
- [9] Zou, Z., & Shi, Z. (2016). Ship detection in spaceborne optical image with SVD networks. *IEEE Transactions on Geoscience and Remote Sensing*, 54(10), 5832-5845.
- [10] Barnum, J. (1986). Ship detection with high-resolution HF skywave radar. *IEEE Journal of Oceanic Engineering*, 11(2), 196-209.
- [11] Fefilatyev, S., Goldgof, D. B., & Langebrake, L. (2007, October). Toward detection of marine vehicles on horizon from buoy camera. In *Unmanned/Unattended Sensors and Sensor Networks IV* (Vol. 6736, p. 67360O). International Society for Optics and Photonics.
- [12] Fefilatyev, S., Goldgof, D., Shreve, M., & Lembke, C. (2012). Detection and tracking of ships in open sea with rapidly moving buoy-mounted camera system. *Ocean Engineering*, 54, 1-12.
- [13] Wijnhoven, R., van Rens, K., Jaspers, E. G., & de With, P. H. (2010). Online learning for ship detection in maritime surveillance. In *Proc. of 31th Symposium on Information Theory in the Benelux* (pp. 73-80).
- [14] Zhang, Y., Li, Q. Z., & Zang, F. N. (2017). Ship detection for visual maritime surveillance from non-stationary platforms. *Ocean Engineering*, 141, 53-63.

- [15] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).
- [16] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016, October). Ssd: Single shot multibox detector. In European conference on computer vision (pp. 21-37). Springer, Cham.
- [17] Law, H., & Deng, J. (2018). Cornernet: Detecting objects as paired keypoints. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 734-750).
- [18] Girshick, R. (2015). Fast r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 1440-1448).
- [19] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems (pp. 91-99).
- [20] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).
- [21] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
- [22] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4510-4520).
- [23] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2818-2826).
- [24] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).
- [25] Xie, S., Girshick, R., Dollár, P., Tu, Z., & He, K. (2017). Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1492-1500).

- [26] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700-4708).
- [27] Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., ... & Chen, Y. (2017). Mastering the game of go without human knowledge. *Nature*, 550(7676), 354-359.
- [28] Ma, N., Zhang, X., Zheng, H. T., & Sun, J. (2018). Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 116-131).
- [29] Zhang, R. L., & Furusho, M. (2019). Developing Generative Adversarial Nets to Extend Training Sets and Optimize Discrete Actions.
- [30] Yu, X., Gong, Y., Jiang, N., Ye, Q., & Han, Z. (2020). Scale Match for Tiny Person Detection. In the IEEE Winter Conference on Applications of Computer Vision (pp. 1257-1265).
- [31] Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., & Ren, D. (2019). Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. arXiv preprint arXiv:1911.08287.
- [32] Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.

Chapter 7

Conclusion

This work regards the unmanned ship as an intelligent agent that can autonomously decision-making, observe and perceive the environment in which the ship navigation at sea, and abstract the observed environment into a state, which is used to train the decision-making neural network, the output is appropriate discrete actions, then, realize the autonomy of the ship navigation. The focus of the research is to use mathematics and computer tools to describe the environment of the ship as abstractly as possible. As an observed state, the ship selects actions according to the strategy function approximated by the designed neural network, obtains the reward of environmental feedback, and then according to the transition probability of the state obtain the next state, and iterative training obtains multiple trajectories. Optimize this action strategy function so that the agent learns to make appropriate actions in a similar sea environment.

1. Decision transfer.

Treat the ship as an intelligent agent that can independently decision-making. According to the division of automaticity level, in all cases below level 5, all decisions are made by the agent itself; If there are some special circumstances that cannot be handled, in all aspects of ship decision-making (level 5 and above) human-in-the-loop is designed to transfer the final decision-making to the shore-based control center. The process of this research is mainly to minimize human involvement and transfer more decision-making tasks to human-out-of-the-loop. Chapters 1 and 2 complete the introduction and research.

2. Describe the environment (state).

The focus of this part is to use various sensors to perceive the real environment of ship navigation. For example, using AIS to sense the navigation state of the

encountering ship, using a camera to capture the surrounding tiny fishing boats, and using radar to detect surrounding obstacles. Then, the perception tasks such as the recognition of the targets in the surrounding environment of the ship, enable our agents to describe the environment more accurately. In some special circumstances, such as collision and other extreme situations, it is difficult to obtain relevant data from real navigation environment. This study used GAN method to generate such environmental image data to ensure that the agent has more data to train the decision model. You can find the detailed implementation process in Chapter 3.

3. Training decision-making neural network.

The principles and applications of the Markov decision process with Markov properties are introduced. This study mainly uses QGIS software to visualize the AIS data on the graphical interface, select the connected 4 frames of images as the state space, constructing deep reinforcement learning model, give the reward function by the distance to the terminal state, use a random strategy gradient, define the discrete action space, training for decision-making neural networks. The 4th chapter is the principle explanation and mathematical description, and the 5th chapter is the implementation process using DQN model.

4. Target detection.

Use the real navigation data from the perspective of the ship bridge to train the target ship classification neural network and the target detection neural network, to improve the ship's ability of risk perception. This chapter 6 designs a unique labeling strategy to better deal with the detection problem of long-distance targets and tiny targets. In real nautical practice, the detection of tiny targets is very important. If tiny targets such as small fishing boats are missed, it is easy to cause dangerous accidents such as collisions etc. In addition, a more detailed perception of the environment can enable faster end-to-end training.

Acknowledgement

In the beginning, I would like to express my deepest and sincere gratitude to my advisor, supervisor Prof. FURUSHO Masao, who guide me well throughout the research work, thanks for supporting me during these past six years. His immense seamanship knowledge, freedom of academic research space, research motivation and patience have given me more power and spirit to excel in the research writing. Conducting the academic study regarding such a difficult topic couldn't be as simple as he made this for me. He is my mentor and a better advisor for my doctorate study beyond the imagination. I would also like to offer my special gratitude to Professor Hideo OKAMURA, Professor Nobukazu WAKABAYASHI and Professor Katsutoshi HIRAYAMA who are pleased to examine and review this dissertation book. Thank you for your academic help and suggestions in my past research work.

I am very grateful with the financial support from MEXT Scholarship, Umeki Scholarship, which support gives me more opportunities to communicate with researchers from other countries and regions, it improves my language skills, logical thinking ability. And this work was supported by the Sasakawa Scientific Research grant from the Japan Science Society, grant No.2019-2036. Provided me with strong computing power support for my research. This work also thanks to Mitsui O.S.K.Lines Co. for provided a large number of actual ship navigation images and video data. The guidance provided by Shiragami san, Nakao san, Otaki san and other office members. In particularly, I appreciate and thank all my friends in my laboratory: Gao san, Fukuda san, Matsue san, Ueno san, Akase kun, Asakura san, Ryo san, Ludfi san, Eka san, Manik san, etc., and all my friends in KICC.

Finally, my special thanks are extended to my lovely parents, Father Wudi ZHANG, Mother Xiuying DAI, and all my family member, who always give their endless support, pray, and anything that I cannot appreciate in words. Because of their prayer, their helps, and their love, I can stand until this time.

List of publication

Journals:

- ① Zhang Ruolan, Furusho M. Developing Generative Adversarial Nets to Extend Training Sets and Optimize Discrete Actions[J]. International Journal on Marine Navigation and Safety of Sea Transportation, Vol.13, No.4, pp.875-880, 2019. DOI: 10.12716/1001.13.04.22
- ② Zhang Ruolan, Furusho Masao. Using the Scene Generated by the GAN Model to Improve the DQN Based Unmanned Ship Decision-Making Algorithm. [J]. European journal of navigation, 16, No.3, pp. 12-17, 2018.
- ③ Zhang Ruolan, Furusho Masao. Conversion timing of seafarer's decision-making for unmanned shipnavigation[J]. International Journal on Marine Navigation and Safety of Sea Transportation, Vol.11, No.3, pp. 463-468, 2017. DOI 10.12716/1001.11.03.11

Proceedings:

- ① ZHANG Ruolan○, FURUSHO Masao. Using a Markov Processes Approach Unmanned Vessel Decision Model. Proceeding of Asia Navigation Conference. Zhanjiang, China. November, 2017.
- ② ZHANG Ruolan○, FURUSHO Masao. Developing Decision-Making Algorithm for Unmanned Vessel Navigation Using Markov Processes. Maritime Safety International Conference (MASTIC 2018). IEEE Indonesia, Bali, Indonesia. July 2018. DOI: 10.23977/mastic.023
- ③ ZHANG Ruolan○, FURUSHO Masao. Using the Scene Generated by the GAN Model to Improve the DQN Based Unmanned Ship Decision-Making Algorithm. 13th International Symposium on Integrated Ship's Information Systems & Marine Traffic Engineering Conference. Berlin, Germany. September, 2018.
- ④ ZHANG Ruolan○, FURUSHO Masao. Use of GAN-DQN Algorithm Generated Navigation Data to Aid in Unmanned Ships Training Autonomous Decision-Making

Model. International Association of Institutes of Navigation (The 16th IAIN World Congress 2018). Tokyo, Japan. November, 2018.

- ⑤ ZHANG Ruolan○, FURUSHO Masao. Automatically Replanning Navigation Route by Using Conditional Generative Adversarial Networks. Proceeding of Asia Navigation Conference. Tokyo, Japan. November, 2018.
- ⑥ ZHANG Ruolan○, FURUSHO Masao. Using the CGAN model Extend Encounter Targets Image Training Set. 5th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS), Grete, Greece. 3-5 May, 2019. DOI:10.5220/0007676803270332
- ⑦ ZHANG Ruolan○, FURUSHO Masao. Developing generative adversarial nets to extend training sets and optimize discrete actions. 13th International Conference on Marine Navigation and Safety of Sea Transportation (TransNav 2019). Gdynia, Poland. 12-14 June, 2019.
- ⑧ ZHANG Ruolan○, FURUSHO Masao, GAO Xinjia. Constructing an Autonomous Decision Approach in a Real AIS Environment Using Deep Reinforcement Learning. Busan, South Korea. 21-23, November, 2019

List of figures

Fig.1.1	Doctoral research map.....	4
Fig.1.2	Dissertation structure	16
Fig.2.1	Target detection, tracking, loss process from the bridge view	24
Fig.2.2	Automaticity level assessment for each process stage	25
Fig.2.3	An example of unmanned ship maneuvering cycle	26
Fig.2.4	A flowchart of conversion timing optimization of decision-making	27
Fig.3.1	Example shoes the state that own ship encounters an obstacle (target)	32
Fig.3.2	Example of the own ship encounters an obstacle state	33
Fig.3.3	Simplified single-target, single-task Markov trajectory process.....	37
Fig.3.4	Training model results (approach 10 steps to avoid obstacles).....	39
Fig.3.5	Embodiment decision-making from the perspective of action selection	40
Fig.4.1	Applying GAN generate various target ships with different background.....	45
Fig.4.2	Target ship image data-generating process	48
Fig.4.3	Example of generated lifeboats with different back-grounds	48
Fig.4.4	Large-scale data generated with different backgrounds. (Lifeboat).....	49
Fig.4.5	Large-scale data generated with different ship status. (Lifeboat)	50
Fig.4.6	Lifeboat large-scale dataset (Lifeboat).....	51
Fig.4.7	Ocean liner Large-scale generated dataset (1)	52
Fig.4.8	Ocean liner Large-scale generated dataset (2)	53
Fig.4.9	Policy gradient convergence of discrete data	55
Fig.5.1	Framework of combined rule and neural based network	60
Fig.5.2	AIS traditional applications for VTS ship management	61
Fig.5.3	Mosaic AIS data with divided ship domain	63
Fig.5.4	Example of the AIS data with heading, ship domain and ship types	64
Fig.5.5	Sequent of the frame motion and moving state at the same area	65
Fig.5.6	Training example of AIS image dataset	66
Fig.5.7	Unstable collision avoidance actions (6,000 times)	69
Fig.5.8	Stable collision avoidance actions (500,000 times)	69
Fig.6.1	Comparison of the accuracy and training time with various network.....	82
Fig.6.2	Ocean targets dataset hierarchy. Leaf nodes correspond to 25 classes	84
Fig.6.3	Bounding box labeling strategy in task 1	86
Fig.6.4	Bounding box labeling strategy in task 2	86
Fig.6.5	Example of intersection over union (IoU)	87
Fig.6.6	IoU distribution of task1 and task2 in the start and end phases	91

Fig.6.8	The detection effect of small targets at long distances at night.....	95
Fig.6.9	The recognition effect of the port and starboard lights	95
Fig.6.10	The ability to detect partially occluded objects is remarkable	96
Fig.6.11	The ability to detect small targets from complex background noise.....	97

List of tables

Table 2.1	Automaticity selection by ship decision & shore-based monitoring	23
Table 2.2	Multiple Target Tracking in a Markov Decision Process (MDP)	25
Table 6.1	Detection characteristics and strategies for different scenarios.....	74
Table 6.2	Comparison of Image classification performance on ImageNet.	79
Table 6.3	Annotated ocean target image set deployment	85
Table 6.4	Training samples description in the main folder	85
Table 6.5	Task 2 records at the end of one training process	92
Table 6.6	Displaying the accuracy tradeoff on the mAP at .5 IoU metric.....	94