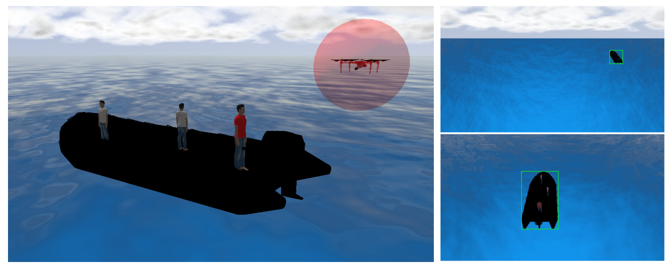
This article has been accepted for publication in a future proceedings of this conference, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/IROS60139.2025

2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2025)
October 19-25, 2025. Hangzhou, China

Autonomous UAV Control for Maritime Applications using Deep Reinforcement Learning-based Image Optimisation

Yuanqing Yang[1], Emmanouil Spyrakos-Papastavridis[1], Mingfeng Wang[2], and Yansha Deng[1*]

***Abstract*—In this paper, we present an autonomous control system for Unmanned Aerial Vehicles (UAVs), specifically designed to inspect a detected suspicious vessel and capture information-rich images in a maritime environment. The maritime environment is ever-changing and uncertain, making it challenging to perform maritime monitoring tasks efficiently and reliably. The proposed UAV control system consists of multiple modules, including path planning, vessel searching, image processing, and image optimization. A novel image optimization approach utilizing deep reinforcement learning (DRL) is proposed to enhance the quality of the captured images by jointly controlling the movement of the UAV and camera orientation. The effectiveness and efficiency of the proposed system were validated and evaluated by searching the vessel and optimizing the captured images in the self-developed simulation environment in Gazebo.**



(a) (b)

Fig. 1. UAV-based maritime monitoring simulation environment: (a) Overview of the vessel and UAV model in the simulation environment. (b) The original image (top) and the optimized image (bottom) from the UAV onboard camera.

I. INTRODUCTION

In recent years, the topic of maritime domain awareness (MDA) has received increasing attention due to the higher frequency of maritime crime incidents, such as smuggling and illegal fishing [1]. Conventional MDA technology integrates information from multiple sensors, including long-distance radars, Automatic Identification System (AIS), satellites and other sensors to gain a thorough comprehension of the maritime environment [2], [3]. As these sensors can only detect objects from a distance, they only allow observation of the rough shape of the target objects (e.g., vessels), which results in a challenge to capture a high-resolution image of the target object for further analysis with conventional MDA. A promising technology that may bridge this gap is that of unmanned aerial vehicles (UAVs), which are well-suited for remote sensing applications. Due to the exceptional mobility and low cost of UAVs, they have been extensively utilized in diverse areas, such as power line inspection [4], traffic monitoring [5], and forest fire monitoring [6].

In the field of MDA, numerous studies have been conducted on the utilization of UAVs for maritime monitoring [7]. However, most studies only employ UAVs as a tool for data collection, disregarding the significance of controlling the UAVs to acquire optimized data [8], [9]. In [10], an off-shelf UAV was employed to monitor marine megafauna by taking high-resolution videos (4K, 30fps). To collect sufficient data, a flight path was devised to cover the largest swept area with minimal overlap without changing flight height, velocity, and camera tilt angle during flight. Although the gathered data can be used to identify marine species, additional improvements would be required to recognize specific species. In [11], the authors employed a UAV to identify an unmanned surface vehicle (USV) on water. The primary focus was on the object recognition algorithms, with no consideration given to the control of the UAV. There are instances where the USV appears at the edge of the footage, which increases the difficulty of the recognition task. Other studies have also explored optimizing the flight path of the UAV [12]. In [13], the authors studied the coverage path planning problem to enhance the efficiency of a multi-UAV system in maritime search and rescue operations. The focus was on the optimization of flight paths for different UAVs without considering the actual scene observed from the cameras. In this case, human intervention is required to obtain the desired images in practical applications.

To our knowledge, a limited number of studies have been conducted on the topic of UAV control optimization for capturing enhanced images in maritime environments. As it is crucial to effectively collect the desired information, there is an imperative demand to develop a comprehensive UAV control system to automatically search for a designated target and to capture optimized images, which can make a significant impact, such as collecting evidence of maritime crime.

To fill this gap, we propose an autonomous UAV control

*This research is supported by UK Research and Innovation (UKRI) under Grant 10014507 (Corresponding author: Yansha Deng).

[1]Yuanqing Yang, Emmanouil Spyrakos-Papastavridis, and Yansha Deng are with the Department of Engineering, Faculty of Natural, Mathematical Engineering Sciences, King's College London, WC2R 2LS, UK {yuanqing.yang, emmanouil.spyrakos, yansha.deng}@kcl.ac.uk

[2]Mingfeng Wang is with the Department of Mechanical and Aerospace Engineering, Brunel University of London, UB8 3PH, UK mingfeng.wang@brunel.ac.uk

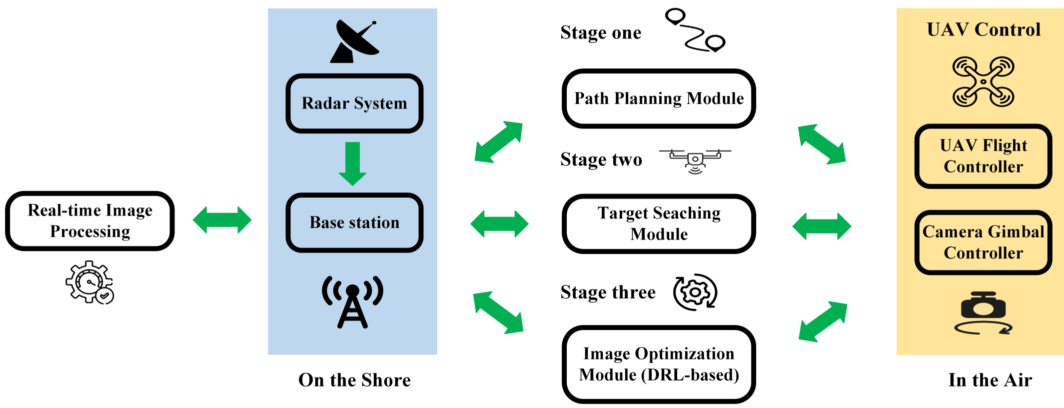


Fig. 2. An overview of the system architecture.

system, which relies on the robot operating system (ROS). The proposed system offers good compatibility and adaptability by utilizing a modular design strategy, where separate modules are responsible for specific tasks. It comprises multiple modules such as path planning, target searching, real-time image processing and image optimization. Based on this, we present a method that utilizes deep reinforcement learning (DRL) to enhance the quality of the captured images by controlling the position of the UAV and adjusting the orientation of the camera. The proposed autonomous UAV control system is implemented and evaluated using the Gazebo simulation environment as shown in Fig. 1.

The outline of this paper is as follows. In Section II we introduce the problem and the simulation environment. In section III we describe the architecture of the proposed system. In section IV we present the proposed pinhole camera model-based and DRL-based methods for image quality optimization. In section V we provide the simulation results and section VI concludes the paper.

II. PROBLEM DESCRIPTION

We study a marine application scenario where an Unmanned Aerial Vehicle (UAV) is employed to inspect suspicious vessels that disable their AIS and cease broadcasting their identity and position. This is a complex task that involves path planning, vessel searching, localization, and image optimization. However, it differs from traditional UAV tasks as our ultimate goal is to capture information-rich and valuable images that contain the desired information, by means of manipulating the UAV.

Traditional techniques usually employ onboard sensors, such as LiDAR and RGB-D cameras, to locate and track the target object [14], [15]. However, a few of these sensors have restricted ranges and are not suitable for maritime applications. It is noted that high-precision sensors are typically expensive and require a considerable amount of energy to sustain operation throughout the mission. Moreover, incorporating extra onboard sensors could increase the overall load of the UAV. These factors can potentially reduce the flight duration of the UAV and lead to failure of the task, as well as significantly increase the cost. To address these problems, it is preferable to utilise a UAV with essential but minimal equipment.

A promising solution can be relying solely on the onboard monocular camera, which is an essential component of a UAV. Extensive research has been conducted on utilizing the monocular camera for the purpose of detecting and tracking objects [16], [17], [18]. Nevertheless, the detection accuracy of monocular-based methods still remains limited. Hence, we present a DRL-based approach to optimize UAV control and improve the quality of the captured images.

In order to alleviate the challenges associated with implementing complex UAV control algorithms in an actual marine environment, we established a 3D simulation environment to replicate the effects and phenomena present in such an environment using the Gazebo simulator, as shown in Fig. 1. The simulation environment consists of a DJI Matrice 100 drone, a vessel, an ocean surface and multiple human models. By utilizing this maritime simulation environment, we aim to acquire image data that resembles that of real-world scenarios. We implement the control of the UAV and the gimballed camera using ROS. Moreover, we are able to monitor the real-time maritime scene through the onboard camera.

Our contributions can be summarized as: (1) We propose an autonomous UAV control framework for maritime applications based on ROS. (2) To optimize the captured image, we propose a DRL-based approach by jointly controlling the movement of the UAV and the orientation of the gimbal. (3) To evaluate the effectiveness of the proposed method, we develop a maritime simulation environment using the Gazebo simulator.

III. SYSTEM ARCHITECTURE OVERVIEW

As shown in Fig. 2, the proposed UAV control framework consists of a radar system, a base station, and a UAV with a gimballed camera. The UAV is also equipped with a GPS sensor to obtain its real-time position and altitude. It is assumed that the radar system can detect the suspicious vessel and provide an estimated position at a consistent frequency. Meanwhile, we assume that there is stable wireless communication between the base station and UAV. Upon receiving the estimated vessel position, the control system initiates the path planning module and generates commands to direct the UAV towards the target area (Stage one). Once the UAV reaches the target area, the system activates the target searching module, which employs the coverage-based path planning method to guide the UAV in exploring the search region (Stage two). A real-time image processing module is proposed to identify the suspicious vessel. Upon detection, the image optimization module is activated to improve the quality of the captured image (Stage three). If either no vessel is detected or the desired image is acquired, the UAV will return to the base station. It is important to note that during stage one, the real-time image processing module functions at a low frequency to enhance the efficiency of the system. If a vessel is detected, the system will proceed to stage three and bypass stage two. This relies on the accuracy of the radar system.

A. *Path Planning Module*

In contrast to the intricate urban environment, the maritime environment is relatively unobstructed and expansive. This allows us to simplify the problem of path planning in three dimensions to a two-dimensional problem. The UAV initially ascends to a designated cruising altitude and performs path planning and target searching tasks on this two-dimensional plane. In our case, we employ the classic A* algorithm to generate an optimal path from the UAV to the estimated vessel position [19].

The corresponding workspace is defined as a grid map with dimensions of 500 by 500 cells, each of which is represented by a node class denoted by N. The heuristic function is defined as the Manhattan distance between the current node $n \in N$ and the goal node $g \in N$:

$$\begin{aligned} f(n) &= g(n) + h(n), \\ h(n) &= |x_n - x_g| + |y_n - y_g|, \end{aligned} \tag{1}$$

where $g(n)$ represents the cost of the shortest path discovered from the starting node to node n at position (x_n, y_n), $h(n)$ represents the heuristic estimate of the cost from the node $n \in N$ to the goal node g at position (x_g, y_g), and $f(n)$ represents the sum of the costs. In each iteration, the algorithm explores adjacent nodes and selects the node with the lowest total cost, and then terminates either when it arrives at the goal node or when there are no more nodes left to explore. In this manner, we can find the shortest path from the UAV to the suspicious vessel, which can be considered as a series of waypoints.

B. *Target Searching Module*

This paper utilizes a grid-based coverage path planning algorithm as the method for target searching. This approach has been extensively employed in UAV applications to ensure efficient and comprehensive data collection and surveillance [20]. It partitions the search region into separate grids and designates the centre of each grid as a waypoint, which is similar to the aforementioned path planning module. The UAV is subsequently directed to navigate through each waypoint. The onboard camera is tilted at a 45-degree angle in a downward vertical direction to capture a broader field of view.

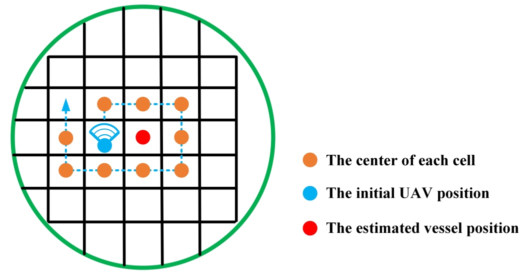


Fig. 3. An illustration of the grid-based coverage method.

The search region is defined as a circular area with a radius of r, and the centre represents the estimated position of the vessel, as shown in Fig. 3. In order to efficiently explore the search region, the UAV follows the extended square search path, described in [21]. The UAV traverses a square pattern and gradually expands outward from the initial point. This approach enables the UAV to systematically cover the designated search region and collect sufficient data.

C. *Real-time Image Processing Module*

During the UAV path planning and search operation, the real-time image processing module operates concurrently. The control system uses the onboard camera to capture real-time RGB images, which are then transmitted to the base station for further processing. In this study, we utilize the OpenCV library to identify the vessel and determine its presence in the image. The recognition results can subsequently be used to optimize UAV control and enhance image quality. Fig. 4 illustrates a schematic diagram of the image processing procedure.

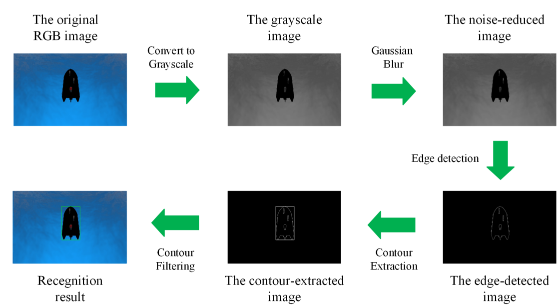


Fig. 4. The detailed procedure for identifying a vessel in an RGB image using OpenCV.

The detailed procedure for image processing is as follows: To begin with, we convert the RGB image to grayscale to decrease computational complexity and speed up processing, where a Gaussian blur function is applied to reduce noise. Next, we use an edge detection algorithm to extract edges from the image and distinguish the vessel from the sea. Subsequently, we extract contours from the edge-detected image. To identify the potential vessel, we set an area-related threshold denoted as I. If a detected contour's area exceeds the threshold, we classify it as a vessel and assign the centre of the contour as the position of the vessel. It is worth noting that we only consider the case of one vessel within the search area in this paper. After conducting numerous tests, we set I as 50 to ensure the system can detect the vessel even at a considerable distance. In the end, we place a bounding box around the identified vessel.

D. Image Quality Optimization Module

This module aims to enhance the quality of the captured images by controlling the movement of the UAV and the orientation of the gimballed camera, as illustrated in Fig. 5. The UAV can be operated in a plane with four possible actions (forward, backward, left, right) while the gimbal can be manipulated with two rotations, i.e., pitch and yaw, which are denoted by Euler angles α and β (in radians) respectively. Moreover, we assume that the gimballed camera can maintain its orientation consistently and neglect any jitter caused by the UAV during the flight. This is because the gimbal can compensate for vibrations and ensure the production of smooth footage while the UAV is in motion.

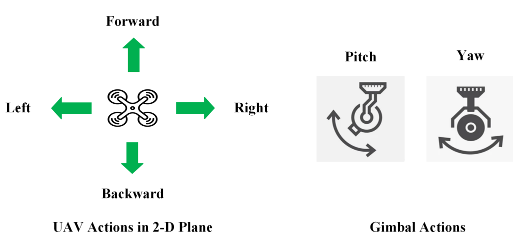


Fig. 5. The action set for the UAV and the gimbal.

In this case, we are able to increase the likelihood of acquiring more valuable information, such as passenger details and the type and purpose of the vessel, by capturing a greater number of pixels associated with the vessel. Based on this idea, we propose two criteria for evaluating image quality: one is whether the vessel is positioned in the centre of the image while the other is whether the vessel occupies a sufficient proportion of the entire image. In essence, the evaluation of image quality is target-oriented and depends on the information we can extract from the image. To fulfil the two aforementioned criteria, we propose two image optimization methods: a pinhole camera model method as the baseline, and a DRL-based method. These approaches are illustrated in section IV.

IV. TWO PROPOSED METHODS FOR IMAGE QUALITY OPTIMIZATION

A. The Pinhole Camera Model-Based Method

In this subsection, we propose a baseline method that utilizes the pinhole camera model [22] to estimate the actual vessel's position. The pinhole camera model is a widely used mathematical model that maps the pixel position, denoted as (x_v, y_v), in an image to the 3D world, as shown in Fig. 6, where the captured image resolution is denoted as $(X_{\mathrm{r}}, Y_{\mathrm{r}})$ and determined by the camera specification.

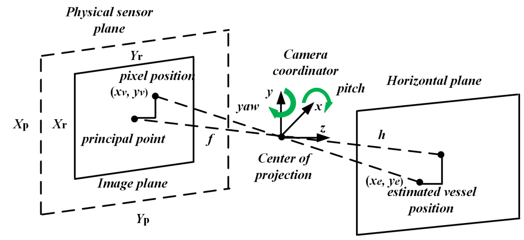


Fig. 6. Illustration of the pinhole camera model.

Based on this model and the camera parameters, we can obtain the estimated position of the vessel, which is defined as (x_e, y_e), on the horizontal plane as:

$$\begin{aligned} x_e &= h_t tan(\alpha + arctan(\frac{(x_v - \frac{X_{\mathrm{r}}}{2})(\frac{X_{\mathrm{p}}}{X_{\mathrm{r}}})}{f_x})) + x_u, \\ y_e &= h_t tan(\beta + arctan(\frac{(y_v - \frac{Y_{\mathrm{r}}}{2})(\frac{Y_{\mathrm{p}}}{Y_{\mathrm{r}}})}{f_y})) + y_u, \end{aligned} \tag{2}$$

where (x_u, y_u) represents the horizontal position of the UAV, h_t represents the altitude of the UAV, θ_h represents the horizontal field of view (HFoV) of the onboard camera, $(X_{\mathrm{p}}, Y_{\mathrm{p}})$ represents the actual physical size of the camera sensor, and f_x and f_y represent the focal lengths along the x-axis and y-axis, respectively, as expressed in Eq. 3.

$$\begin{aligned} f_x &= \frac{\frac{X_{\mathrm{p}}}{2}}{tan(\frac{\theta_h}{2})}, \\ f_y &= \frac{\frac{Y_{\mathrm{p}}}{2}}{tan(\frac{\theta_v}{2})}, \end{aligned} \tag{3}$$

where $\theta_v = 2arctan((tan(\frac{\theta_h}{2})\frac{Y_{\mathrm{r}}}{X_{\mathrm{r}}})$ denotes the vertical field of view (VFoV). In the simulation, the default camera orientation is as follows: the pitch angle is $\frac{\pi}{4}$, while the yaw and row angles are both 0. As we replicate the position control method of the DJI Matrice 100 drone that uses the offsets in the x, y, and z axes as input, the control command for the UAV can be expressed as:

$$\begin{aligned} &\Delta x = h_t tan(\alpha) - h_t tan(\alpha - arctan(\frac{(x_v - \frac{X_r}{2})(\frac{X_{\mathrm{p}}}{X_{\mathrm{r}}})}{f_x})), \\ &\Delta y = h_t(\frac{(y_v - \frac{Y_r}{2})(\frac{Y_{\mathrm{p}}}{Y_{\mathrm{r}}})}{f_y}), \\ &\Delta z = h_d - h_t, \end{aligned} \tag{4}$$

where h_d represents the desired UAV altitude and Δx, Δy, Δz represent the offsets of the UAV along the x, y, and z axes.

B. The DRL-based Method

In this subsection, we propose a DRL-based method to optimize the quality of the captured image. Deep reinforcement learning is a machine learning approach that aims to maximize rewards in the long term through trial and error. In the image optimization module, the reinforcement learning (RL) agent acts as a controller to choose appropriate actions based on the observed environmental information.

A reinforcement learning framework typically includes several essential elements, such as state $s \in \mathcal{S}$, action $a \in \mathcal{A}$ and reward $r \in \mathcal{R}$. The state s is defined as a vector containing the position and area of the vessel in the image, as well as the orientation of the camera. The action a is defined as a vector including the offsets of the UAV, as well as the changes in the direction of the gimbal.

$$s_t = \{x_v,\ y_v,\ c_t\ ,\alpha\ ,\beta\}, \tag{5}$$

$$a_t = \{\Delta x, \Delta y, \Delta z, \Delta\alpha, \Delta\beta\}, \tag{6}$$

where c_t represents the area of the identified vessel at time step t. Here, each offset Δx, Δy, and Δz can be either positive or negative unity. $\Delta\alpha$ and $\Delta\beta$ represent the changes in pitch and yaw directions, respectively. Each variable has two possibilities: either $+\frac{\pi}{90}$ or $-\frac{\pi}{90}$. The range of pitch is within $[\frac{\pi}{6}, \frac{\pi}{2}]$ and the range of yaw is within $[-\frac{\pi}{4}, \frac{\pi}{4}]$.

At each time step, the RL agent chooses an appropriate action a_t from the action set $\mathcal{A}$ based on the current state s_t. After the UAV performs action a_t, the agent receives a scalar reward r_t and observes a new state s_{t+1} from the environment. Given that our primary objective is to position the vessel at the centre of the image and expand its size, we split the learning process into two phases. The first phase involves moving the vessel towards the centre of the image. In this scenario, the reward function depends on the distance between the vessel and the centre of the image, denoted as d. The distance d is normalized as follows:

$$d = \sqrt{(x_v - \frac{X_{\mathrm{r}}}{2})^2 + (y_v - \frac{Y_{\mathrm{r}}}{2})^2} / \sqrt{(\frac{X_{\mathrm{r}}}{2})^2 + (\frac{Y_{\mathrm{r}}}{2})^2} \tag{7}$$

$$r = \begin{cases} -d & \text{if } d > d_h, \\ 1 & \text{otherwise.} \end{cases} \tag{8}$$

where d_h is the distance threshold which is set to 0.1. If d is less than d_h, we assume that the vessel has reached the centre of the image. The second phase involves enlarging the size of the vessel in the image. The reward function is based on the percentage of the vessel's area relative to the entire image. To encourage the vessel to remain in the middle area of the image, a small penalty p_d will be applied if the vessel is positioned excessively far from the centre of the image. If $d > 0.5$, then p_d equals -0.1; Otherwise, p_d is set to 0.

$$r = \begin{cases} \dfrac{c_i}{kX_{\mathrm{r}}Y_{\mathrm{r}}} - 1 + p_d & \text{if } c_i < c_h, \\ 1 & \text{otherwise.} \end{cases} \tag{9}$$

where k is the factor that determines the desired vessel area and $c_h = kX_{\mathrm{r}}Y_{\mathrm{r}}$ represents the area threshold. We set k to 0.02 in order to achieve a vessel area of $128 \times 72 \times 2$ pixels. If c_i exceeds the area threshold, the agent receives a positive reward of +1 and the iteration terminates.

In this paper, we employ a well-established deep reinforcement learning method known as Deep Q-learning (DQN) [23]. DQN leverages convolutional neural networks to approximate the action-value function $Q(s, a; \theta)$ and utilizes the ϵ-greedy method to balance exploration and exploitation. At each step, the agent has a probability of $1 - \epsilon$ to choose the action with the highest Q value and a probability of ϵ to randomly choose an action. The loss function used to train the Q-function approximation can be expressed as:

$$L(\theta_i) = \mathbb{E}[(Q(s, a; \theta_i) - (r + \gamma max_{a'} Q(s', a'; \theta_{i-1}))^2], \tag{10}$$

where θ_i represents the parameters of the Q-network for iteration i. The parameters from the previous iteration θ_{i-1} are kept constant when optimizing the loss function $L_i(\theta_i)$ in order to enhance learning stability. To update the Q-network, the gradient of the loss function with respect to the network parameters is calculated as follows:

$$\begin{aligned} \nabla L(\theta_i) = \mathbb{E}_{s,a}[r + \gamma max Q(s', a'; \theta_{i-1}) - \\ Q(s, a; \theta_i)] \nabla_{\theta_i} Q(s, a; \theta_i)], \end{aligned} \tag{11}$$

The neural network parameter $\theta(s, a; \theta)$ is updated using the RMSProp optimizer. The detailed DQN algorithm is shown in Algorithm 1.

Algorithm 1 The DRL-based Method for Image Quality Optimization

Initialize replay memory $\mathcal{D}$ with capacity N
Initialize Q-network with weights θ and target network with weights θ^-
Set target network weights $\theta^- \leftarrow \theta$
Initialize exploration rate ϵ
Initialize time step $t \leftarrow 0$
while training is not complete **do**
    Observe current state s_t from the Gazebo simulation environment
    With probability ϵ select a random action a_t, otherwise select $a_t = \arg\max_a Q(s_t, a; \theta)$
    Execute UAV action a_t, observe reward r_t and next state s_{t+1}
    Store (s_t, a_t, r_t, s_{t+1}) in $\mathcal{D}$
    Sample a random minibatch m of transitions (s_i, a_i, r_i, s_{i+1}) from $\mathcal{D}$
    Compute target Q-values for each transition:
        $y_i = \begin{cases} r_i & \text{if episode terminates at step } i+1 \\ r_i + \gamma \max_{a'} Q(s_{i+1}, a'; \theta^-) & \text{otherwise} \end{cases}$
    Compute the loss:
        $L(\theta) = \frac{1}{m} \sum_i (Q(s_i, a_i; \theta) - y_i)^2$
    Update Q-network weights θ by minimizing the loss:
        $\theta \leftarrow \theta - \alpha \nabla_\theta L(\theta)$
    Every C time steps, update the target network weights:
        $\theta^- \leftarrow \theta$
    Reduce the exploration rate ϵ over time
    Increment time step $t \leftarrow t + 1$
end while

V. SIMULATION SETUP AND RESULTS

A. *Simulation Setup*

In the Gazebo environment, the camera settings are configured to closely resemble the actual onboard camera of the DJI Matrice 100 drone. The settings are as follows: the horizontal

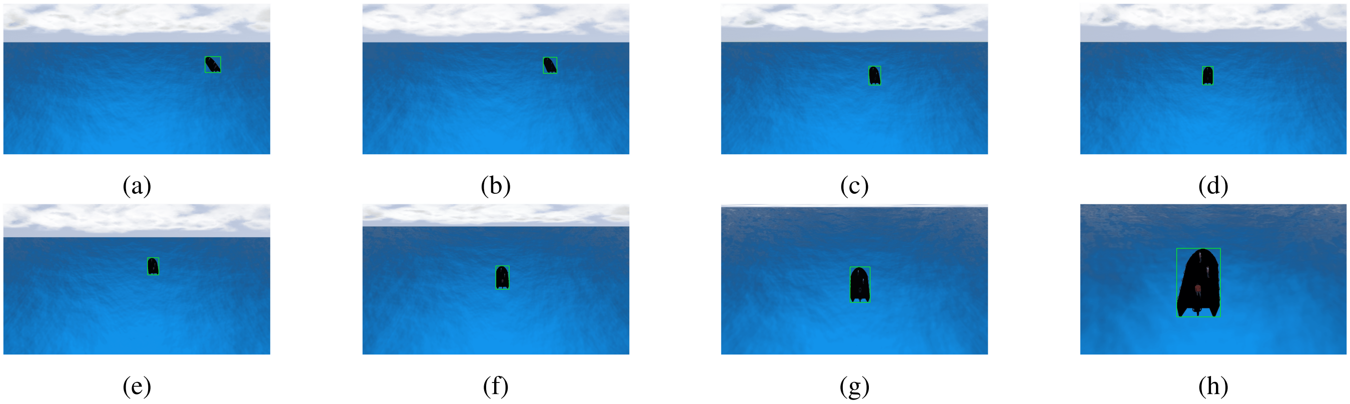


Fig. 7. The illustration of image quality optimization results: (a)∼(d) Phase One and (e)∼(h) Phase Two.

field of view (HFOV) is set to 94 degrees, the physical size of the camera sensor is set to 1/2.3-inch (approximately 6.17 mm $\times$ 4.55 mm), and the image resolution is set to 1280 $\times$ 640. As GPS devices typically have a margin of error of a few meters, a random error is introduced to the altitude of the UAV to simulate this error, denoted as $e_u \in [-2, -2]$ units. In practical situations, the camera may experience errors regarding focal length and lens distortion, which are also considered in simulation. We make the assumption that the UAV travels at a constant speed and moves one unit per timestep. In the DQN algorithm, we set the learning rate as 0.001, the batch size as 32, and the reward discount rate γ as 0.9. The greedy rate ϵ gradually decreases from 1 to 0 during training. The maximum number of episodes is set to 600 and the maximum step is set to 50.

To valid and evaluate the proposed system, we have two simulation phases, i.e., centreing the vessel and expanding the vessel's size, where two control modes are considered: (1) We solely control the movement of the UAV, while the gimbal's orientation remains constant at a 45-degree angle pointing vertically downwards. (2) The gimbal's yaw and pitch orientations can be jointly controlled with the UAV's movement. In this case, we implemented three control methods (baseline, DRL without gimbal and DRL with gimbal) in both simulation phases as discussed below.

B. *Phase One: Centering the Vessel*

In Phase One, the UAV operates on a 2-D horizontal plane with a constant altitude set of 50 units. At the beginning of each episode, the vessel is initialised at different locations within the image. Fig. 7 (a) displays an initial view captured by the onboard camera, indicating that the vessel is located at the edge of the footage. The episode terminates when the vessel arrives at the centre of the image.

Fig. 8 illustrates the comparison of results obtained through three different methods. We tested 50 trials with different initial vessel positions. It can be observed that the baseline method has the longest execution time required to centre the vessel, and it also exhibits the lowest success rate. This can be attributed to the presence of errors from the sensor and the camera model. Both DRL-based methods achieve a reliable performance with an exceptionally high success rate of up to 100%. It is worth noting that the DRL-based method with gimbal control can significantly reduce the total execution time, which is approximately 30 % faster than the baseline. This is due to the agent having a wider range of possible actions and can exhibit more diverse behaviours. A successful process for Phase One is illustrated in Figs. 7 (a) - (d), where the vessel gradually moves towards the centre of the footage.

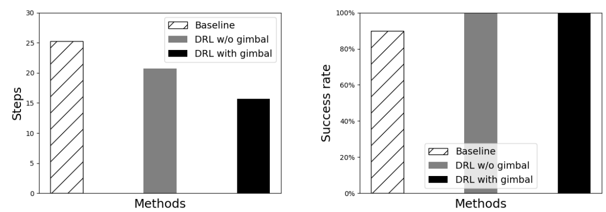


Fig. 8. The simulation results of Phase One for different methods: (a) The execution time (b) The success rate.

C. *Phase Two: Expanding Vessel's Size*

In Phase Two, the system aims to enlarge the size of the pixel associated with the vessel. As we cannot replicate the camera's zoom-in function in the Gazebo simulator, the UAV is instructed to gradually descend to a minimum altitude of 20 units. The descent speed is set as one unit per timestep. At the beginning of each episode, the vessel is initialised at the middle section of the image with an altitude of 50 units and a direction of a 45-degree angle pointing vertically downwards. The episode terminates when the vessel's area reaches the target threshold.

The results pertaining to three different methods are compared in Fig. 9. The DRL-based method with gimbal control achieves the shortest execution time, while the other methods require more than double the time to accomplish the task. The baseline method needs the longest execution time and achieves the lowest success rate around 70%. It can be observed that the DRL-based methods outperform the pinhole camera model-based method in terms of both execution time and success rate. Meanwhile, the DRL-based method with gimbal control is around 60 % more efficient than the DRL-based method without gimbal control. A successful Phase Two execution process is shown in Figs. 7 (e) - (h). The agent successfully learned a policy to approach the vessel and increase its size. Consequently, we can extract more details about the vessel.

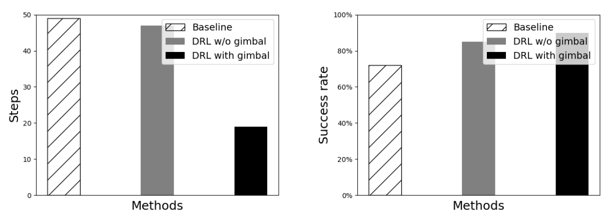


Fig. 9. The simulation results of Phase Two for different methods: (a) The execution time (b) The success rate.

VI. CONCLUSIONS

In this paper, we proposed a novel autonomous control system for unmanned aerial vehicles (UAVs) utilized in maritime monitoring tasks. The system comprises multiple modules, such as path planning, target searching, image processing and image optimization. Two image quality optimization methods are proposed to optimize the captured images including the pinhole camera model-based and the DRL-based methods. We evaluate the proposed methods via the maritime Gazebo simulation environment. The results show that the proposed DRL-based method can efficiently centre the target vessel and expand its size in the footage. Future work is planned to test the proposed UAV control system by implementing it on an actual quadrotor drone within a maritime environment.

REFERENCES

[1] Marin Chintoan-Uta and Joaquim Ramos Silva. Global maritime domain awareness: A sustainable development perspective. *WMU Journal of Maritime Affairs*, 16:37–52, 2017.

[2] Brian J Tetreault. Use of the automatic identification system (ais) for maritime domain awareness (mda). In *Proceedings of Oceans 2005 Mts/Ieee*, pages 1590–1594. Ieee, 2005.

[3] Urška Kanjir, Harm Greidanus, and Krištof Oštir. Vessel detection and classification from spaceborne optical images: A literature survey. *Remote sensing of environment*, 207:1–26, 2018.

[4] Guang Zhou, Jinwei Yuan, I-Ling Yen, and Farokh Bastani. Robust real-time uav based power line detection and tracking. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 744–748. IEEE, 2016.

[5] Hailong Huang, Andrey V Savkin, and Chao Huang. Decentralized autonomous navigation of a uav network for road traffic monitoring. *IEEE Transactions on Aerospace and Electronic Systems*, 57(4):2558–2564, 2021.

[6] Liyana Adilla binti Burhanuddin, Xiaonan Liu, Yansha Deng, Ursula Challita, and András Zahemszky. Qoe optimization for live video streaming in uav-to-uav communications via deep reinforcement learning. *IEEE Transactions on Vehicular Technology*, 71(5):5358–5370, 2022.

[7] Zongyao Yang, Xueying Yu, Simon Dedman, Massimiliano Rosso, Jingmin Zhu, Jiaqi Yang, Yuxiang Xia, Yichao Tian, Guangping Zhang, and Jingzhen Wang. Uav remote sensing applications in marine monitoring: Knowledge visualization and review. *Science of The Total Environment*, 838:155939, 2022.

[8] Carlos A Trasviña-Moreno, Rubén Blasco, Álvaro Marco, Roberto Casas, and Armando Trasviña-Castro. Unmanned aerial vehicle based wireless sensor network for marine-coastal environment monitoring. *Sensors*, 17(3):460, 2017.

[9] Jessica Schaub, Brian PV Hunt, Evgeny A Pakhomov, Keith Holmes, Yuhao Lu, and Lucy Quayle. Using unmanned aerial vehicles (uavs) to measure jellyfish aggregations. *Marine Ecology Progress Series*, 591:29–36, 2018.

[10] Amanda BD Giacomo, Jonathas Barreto, João Batista Teixeira, Lorena Oliveira, Luciano Cajaíba, Jean-Christophe Joyeux, Nelson Barcelos, and Agnaldo S Martins. Using drones and rov to assess the vulnerability of marine megafauna to the fundão tailings dam collapse. *Science of the Total Environment*, 800:149302, 2021.

[11] Jinshan Wu, Ronghui Li, Jiawen Li, Meichang Zou, and Zijian Huang. Cooperative unmanned surface vehicles and unmanned aerial vehicles platform as a tool for coastal monitoring activities. *Ocean & Coastal Management*, 232:106421, 2023.

[12] Tingting Yang, Zhi Jiang, Ruijin Sun, Nan Cheng, and Hailong Feng. Maritime search and rescue based on group mobile computing for unmanned aerial vehicles and unmanned surface vehicles. *IEEE transactions on industrial informatics*, 16(12):7700–7708, 2020.

[13] Sung Won Cho, Hyun Ji Park, Hanseob Lee, David Hyunchul Shim, and Sun-Young Kim. Coverage path planning for multiple unmanned aerial vehicles in maritime search and rescue operations. *Computers & Industrial Engineering*, 161:107612, 2021.

[14] Ha Sier, Xianjia Yu, Iacopo Catalano, Jorge Pena Queralta, Zhuo Zou, and Tomi Westerlund. Uav tracking with lidar as a camera sensor in gnss-denied environments. In *2023 International Conference on Localization and GNSS (ICL-GNSS)*, pages 1–7. IEEE, 2023.

[15] Zhefan Xu, Xiaoyang Zhan, Baihan Chen, Yumeng Xiu, Chenhao Yang, and Kenji Shimada. A real-time dynamic obstacle tracking and mapping system for uav navigation and collision avoidance with an rgb-d camera. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10645–10651. IEEE, 2023.

[16] Abdulla Al-Kaff, Fernando García, David Martín, Arturo De La Escalera, and José María Armingol. Obstacle detection and avoidance system based on monocular camera and size expansion algorithm for uavs. *Sensors*, 17(5):1061, 2017.

[17] Xiaoyue Zhao, Fangling Pu, Zhihang Wang, Hongyu Chen, and Zhaozhuo Xu. Detection, tracking, and geolocation of moving vehicle from uav using monocular camera. *IEEE Access*, 7:101160–101170, 2019.

[18] Ziyue Lin, Wenbo Xu, and Wei Wang. A moving target tracking system of quadrotors with visual-inertial localization. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3296–3302. IEEE, 2023.

[19] Luca De Filippis, Giorgio Guglieri, and Fulvia Quagliotti. Path planning strategies for uavs in 3d environments. *Journal of Intelligent & Robotic Systems*, 65:247–264, 2012.

[20] Tauã M Cabreira, Lisane B Brisolara, and Ferreira Jr Paulo R. Survey on coverage path planning with unmanned aerial vehicles. *Drones*, 3(1):4, 2019.

[21] Jiqiang Li, Guoqing Zhang, Changyan Jiang, and Weidong Zhang. A survey of maritime unmanned search system: Theory, applications and future directions. *Ocean Engineering*, 285:115359, 2023.

[22] Andrea Tonini, Paula Redweik, Marco Painho, and Mauro Castelli. Remote estimation of target height from unmanned aerial vehicle (uav) images. *Remote Sensing*, 12(21):3602, 2020.

[23] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.