Underwater Garbage Detection Remotely-Operated Vehicle (ROV) Based on Arduino and YOLO V7

Xuanye Dai*

Indiana Academy for Science, Mathematics and Humanities, Indiana, America

*Corresponding author: daixuanye40@gmail.com

Abstract. In response to solid waste pollution in water bodies, people generally use surface vessels with salvage nets for artificial purification, but are powerless to deal with areas that have sunk to the bottom of the water or are inaccessible to ships. Therefore, this project has designed a robot that can identify and even salvage solid waste pollution underwater. In order to solve the problem of multi thruster control for underwater robots, this project maps multi-channel joystick signals to multiple thrusters based on the principle of linear superposition. In order to have the ability to identify underwater garbage, real-time video data from underwater cameras is transmitted wirelessly, allowing laptops to use their self trained YOLO V7 weight model to statistically analyze the types and quantities of underwater garbage. In the future, this project can also design an independent garbage collection device to more conveniently solve the problem of underwater solid waste pollution.

Keywords: Underwater robot; linear thrust allocation; YOLO image recognition.

1. Introduction

In recent years, the expansion of water pollution has tremendously harmed the marine ecosystems, soon pervading the whole planet(Figure 1). To be more candid, the abundant submersed toxins could directly contaminate our daily aquatic products, endanger the marine species, and threaten the safety of drinking water. To address this serious issue, our project hammers at developing an advanced underwater robot capable of detecting subaquatic plastic debris[1,2].

This design mainly applies several vital mathematical methods and computational algorithms. In order to remote control effectively, our devise leverages the mathematical matrices model of thruster allocation[3]. With the setting of three degrees aimed to achieve omnidirectional motion, This paper decide to configure four channels with two on the x-axis and the other two on the y-axis. According to the distribution of the channels, our paper conceive the structure and present the whole in detail applying the 3D modeling[4]. Besides, our paper leverage Arduino and C++ for the control system. Spontaneously, our paper harness Yolo V7 to our image recognition system.To set the tone, this project will analyze the a mathematical model for thrust allocation in multi-thruster underwater robots.

Throughout this article, our paper initially detailed the trudge of conquering technical hurdles, including Arduino, PWM signal, ESC, Brushless motor, and the design of multi-thruster[5]. Then, this project designed the blueprint through Fusion360. Plus, we utilized PVC to set up the mother ship. Finally, to achieve subaquatic garbage detection, this project determined to exploit 5.8G image transmission system. Thus, this project could transmitt submersed video information to a laptop for real-time detection of underwater litters.

Figure 1: Severe Environmental Pollution Caused by Marine Plastic Waste

2. Theoretical Basis

2.1. Degrees of freedom of the robot

While various types of robots employ different propulsion methods such as wheels (for land), air propellers (for aerial), underwater thrusters (for underwater/surface), and rocket engines (for space), their fundamental control principles involve coordinating these specific propulsion systems in various operational states to achieve specific motion states for the entire robot[6].

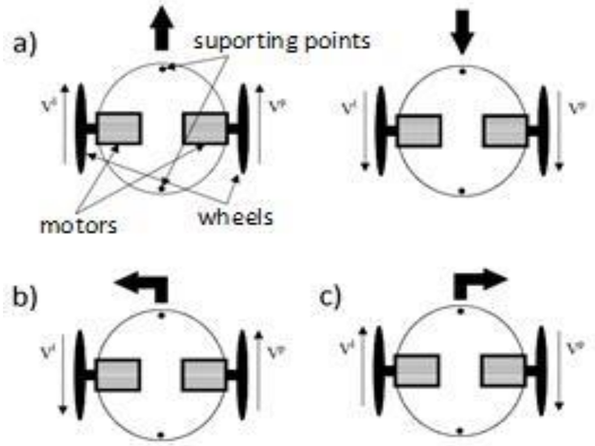

Figure 2: Differential Drive Principle of Land Robots

As shown in Figure 2, taking the example of the simplest differential drive with two wheels on land,both wheels simultaneuosly have the magnitude and direction for angular velocity. When the anglular velocity magnitude and direction are the same, the robot moves in a straight line. However, when the speed magnitude or direction differs between the left and right wheels, the robot performs various non-linear movements.

Clearly, land-based robots are considered to move in a two-dimensional plane, whereas underwater robots operate in three-dimensional space. In addition to forward and backward movement and turning, underwater robots can also move vertically and rotate along various axes. To better study the motion capabilities of robots, this project will introduce the concepts of three-dimensional coordinates and degrees of freedom, demonstrating the necessary motion freedoms that our robot should possess.

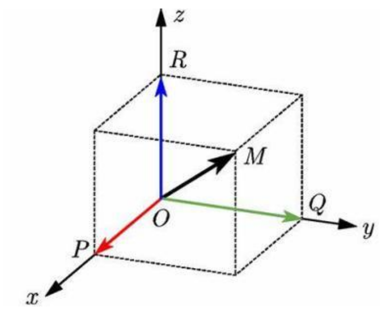

Figure 3: Position Representation in Cartesian Coordinate System

Given that a robot is moving in three-dimensional space, regardless of its own rotation but mainly focusing on the positional changes in space, this project can take advantage of a Cartesian coordinate system to represent the robot's position state, as depicted in Figure 3. The position of point M can always be represented using three points P, Q, and R.

Similarly, when the frame of reference is not a static third-person view but rather aligned with the robot's specific orientation, the robot's translational motion state (where the robot itself undergoes no rotation, akin to the linear motion depicted in Figure 2) can be represented by the superposition of three mutually perpendicular velocity vectors.

However, when we consider that the robot not only displays translational motion but also rotational motion around its own axis (such as the left-right turning motion of the robot depicted in Figure 2), we need to utilize a new mathematical model to describe the robot's potential motion states.

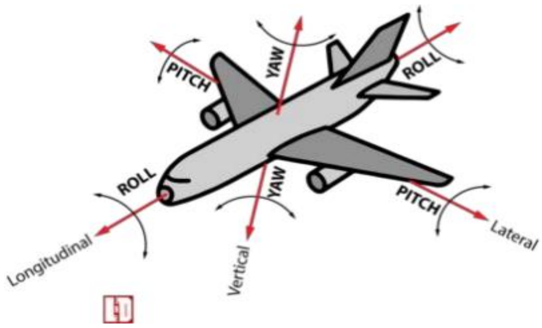

Figure 4: 6 Degrees of Freedom in Three-Dimensional Space

As shown in Figure 4, this project first consider that underwater robots and airplanes flying in the sky share similarities (both operate in three-dimensional space). Therefore, referring to the description above regarding translational motion states of these objects, this project define the forward direction of the airplane/robot as the x-axis, the vertical direction as the z-axis, and the y-axis as a coordinate axis perpendicular to the xz-plane, pointing in the left-right direction.Correspondingly, rotational movements around the x, y, and z axes can be defined as roll, pitch, and yaw, respectively.

According to the definition of degrees of freedom in mechanics (the number of independent variables required to fully describe the motion of a mechanical system), we can consider that when the reference frame is placed on the robot or airplane, and does not change relative to the object's position, the object's motion state can be described using six degrees of freedom (x, y, z, roll, pitch, yaw). Henceforth, the robot's control system only needs to be designed with reference to the forces required to generate these six degrees of freedom[6,7].

2.2. Thrust Allocation and Linear Systems

2.2.1 Applications and Basic Principles of Thrust Allocation

Thrust allocation is primarily applied in aerospace and underwater robotics, where it is necessary to utilize the thrust generated by propellers to perform vector synthesis in three-dimensional space, enabling the entire device to move in the desired direction.

To futher illustrate, thrust allocation generally relies on the coordinated operation of multiple thrusters in terms of thrust magnitude and thrust direction to correspond to the control commands input by the operator. Therefore, the thrust allocation system can be viewed as a single-input, multiple-output control system. Therefore, the thrust allocation system can be viewed as a single-input, multiple-output control system[7].

2.2.2 Linear Relationship Between Remote Control Signals and Thruster States

First, our work demonstrated the case where a single control command corresponds to a single thruster as Figure 5:

Figure 5: Single Joystick and Single Thruster

In this case, the joystick relies on a potentiometer to output a continuous analog signal. After analog-to-digital(AD) conversion , this signal is converted into a continuous numerical value ranging from 0 to 1023 in the MCU's 10-bit AD sampler. The variation in this numerical value is linearly related to the rotational angle of the joystick along that axis[7].

Besides, the thruster requires a bidirectional brushless electronic speed controller (ESC) for driving. The PWM signal range for controlling the bidirectional ESC is typically 1000-2000 microseconds, and the output power of the ESC is linearly proportional to this signal range. Assuming our single-channel control system goes from the joystick to the MCU, where the MCU outputs a continuously varying PWM signal, and this PWM signal controls the ESC to continuously adjust power, we can consider the entire system's input-output relationship to be linear.

Plus, in systems based on vector synthesis decomposition, the relationship between each vector and the final output vector depends on the trigonometric values of their angles. When the directions of the decomposed and synthesized vectors remain unchanged, these trigonometric values are fixed. Therefore, given the directions of the input vector and the output vector, their relationship can be simply viewed as a linear superposition.

Consequently, this project can treat both the input and output values as linear equations, utilizing the principle of linear superposition to achieve the desired control output.

2.3. Mathematical Model

In this program, our work decided to use a remote control with two joysticks and employ mode switching to control both the mother ship and the submersible. The mother ship has only two thrusters (as illustrated in the diagram below the mother ship introduction), allowing it to achieve horizontal movement along the x-axis and yaw rotation. The submersible has four thrusters, enabling movement along the x-axis and z-axis as well as yaw rotation[8].

First, as Figure 6, this project listed the various states of the remote control and the thrusters (define their state as+1 when moving forward, -1 when moving backward, and 0 when stopping). This paper assume that the submersible has four thrusters, two on the x-axis (M1, M2) and two on the z-axis (M3, M4), enabling movement along these axis and yaw rotation. The remote control has two joysticks: the y-axis movement (forward/backward) of the left joystick controls (Ly) the ROV to move forward/backward, the x-axis movement (left/right) of the left joystick controls (Lx) the ROV to yaw rotation, and the y-axis movement (forward/backward) of the right joystick controls (Ry) the z-axis movement:

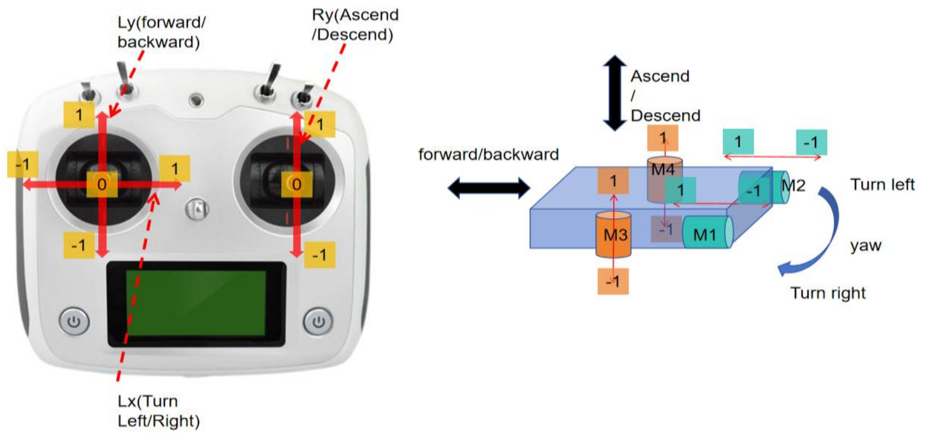

Figure 6: State definition of remote control joysticks and ROV thrusters under different control commands

Table 1: ROV Output Status Table

Action	L_y	L_x	R_y	M_1	M_2	M_3	M_4
Forward	1	0	0	1	1	0	0
Backward	-1	0	0	-1	-1	0	0
Ascend	0	1	0	0	0	1	1
Descend	0	-1	0	0	0	-1	-1
Turn Left	0	0	-1	-1	1	0	0
Turn Right	0	0	1	1	-1	0	0

Since the input signals separately control forward/backward and left/right movements, our project want the thrusters M_1 and M_2 to be simultaneously controlled by both signals from the joystick as Table 1. Thus, for mixed control, we can assume:

$$M_1 = a \times L_y + b \times L_x$$
$$M_2 = c \times L_y + d \times L_x \quad (1)$$

Based on the table, we can derive the following equations:

$$M_1 = L_y + L_x$$
$$M_2 = L_y - L_x \quad (2)$$

2.4. Image Recognition Framework YOLO V7

2.4.1 Introduction of YOLO V7

As Figure 7, YOLOv7 is a state-of-the-art real-time object detector that surpasses all known object detectors in both speed and accuracy in the range from 5 FPS to 160 FPS. It has the highest accuracy (56.8% AP) among all known real-time object detectors with 30 FPS or higher on GPU V100. Moreover, YOLOv7 outperforms other object detectors such as YOLOR, YOLOX, Scaled-YOLOv4, YOLOv5, and many others in speed and accuracy. The model is trained on the MS COCO dataset from scratch without using any other datasets or pre-trained weights. Source code for YOLOv7 is available on GitHub[9].

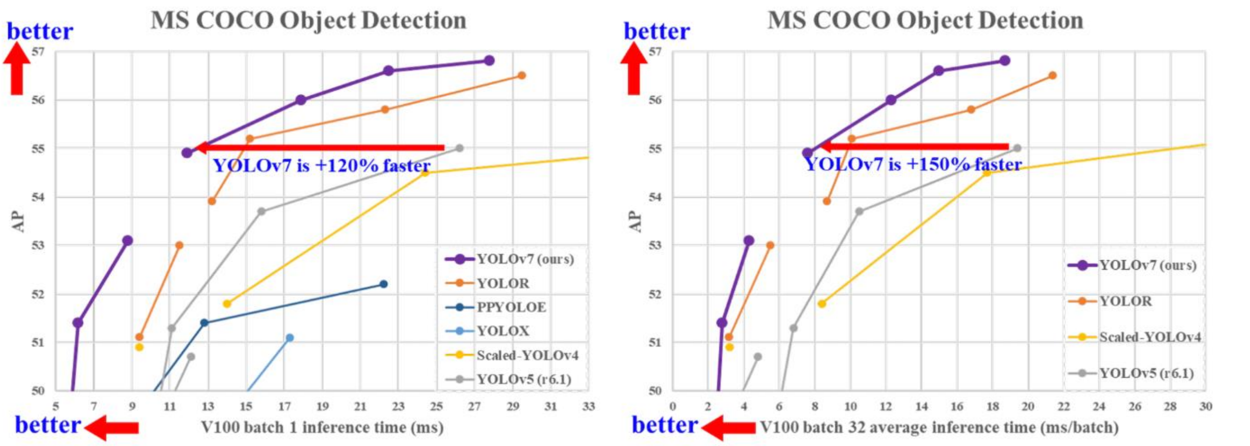

Figure 7: Improvement of YOLO V7

2.4.2 Training Process of YOLO V7

First, this project need to prepare the dataset. The purpose of this project is to address the issue of underwater environmental pollution detection. Therefore, a dataset comprising 1000 images of various underwater environments and plastic waste has been collected and annotated. Using the ImageLabel tool, the images have been manually labeled to distinguish between five categories: plastic bottles, plastic bags, discarded fishing nets, and other types of plastic waste. Precise annotation of the images is required to accurately identify the location and category of the waste, which is essential for effective training of the model[10,11].

Secondly, load the pre-trained model. Due to hardware limitations, our work chose an RTX 3060 GPU for the environment setup. After installing Python 3.9, we configured the PyTorch framework and downloaded and installed the YOLO V7 image recognition framework. To accelerate training, this project need to load the pre-trained weights "yolov7_s.pt" in the code. Given the GPU's performance constraints, I proceeded with training by setting the number of categories to 5 while keeping the default settings. The estimated training time is 22 hours.

The final step is validation. After completing the training, this project will use an underwater wired camera to capture images, which will be transmitted to a laptop via a USB interface. Upon validation, the project will be able to detect underwater waste effectively.

3. Our Work

3.1. Structural Principle

As Figure 8, this paper chose two pairs of thrusters, one oriented horizontally and the other vertically, to achieve the basic three degrees of freedom (x, yaw, z). This configuration allows the sensors at the front of the submersible to effectively detect objects without any blind spots.

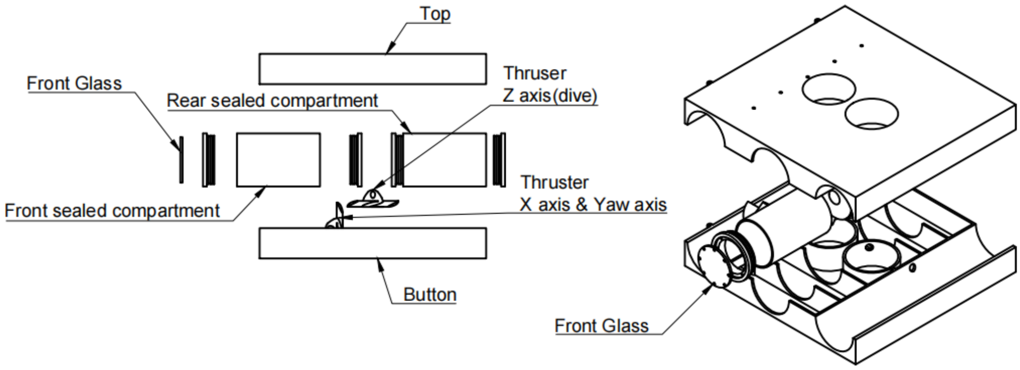

Figure 8: Submersible Design Diagram

To achieve successful remote control, our work utilized PWM signals. Using C++ and the Arduino compiler, we conducted research and designed linear equations for thrust control, which were then combined to create the final remote control system.

In addition to the thrusters, this paper meticulously crafted the housing. Specifically, our work chose ABS for 3D printing using the Fused Deposition Modeling (FDM) process and used bolts for fastening.

For circuit control, to facilitate the simultaneous remote control of both the mothership and the submersible, our work implemented a mode-switching mechanism to select the control target.

3.2. Design and Fabrication of Underwater Robot Blueprints

This paper applies Autodesk software to design the submersible model, and the model is shown above. The model is symmetrical, featuring a vertical duct and a horizontal duct on each side. Additionally, this paper customized rotors and stators for the propellers to fit motors available online and positioned them at the center of each duct. Openings were created in the middle of both ends of the submersible to accommodate cameras.

The model's housing is divided into two parts, upper and lower. To lay out internal wiring, our work drilled small holes in certain sections of the model (not reflected in the Autodesk design). To ensure proper submersion and buoyancy control, this project prepared foam blocks (used) and some iron weight blocks (unused).

3.3. Image Transmission and Mothership Design and Fabrication

As Figure 9, following the principle of simplicity, this project constructed the mothership using a raft structure. At the bottom, this paper incorporated two thicker plastic pipes to provide buoyancy. These pipes have small holes at both ends, connecting to four thinner pipes, with a platform established at the upper ends of these four pipes to accommodate batteries. Once the model was established, this project used 3D printing technology to create physical objects, as depicted in the figure.

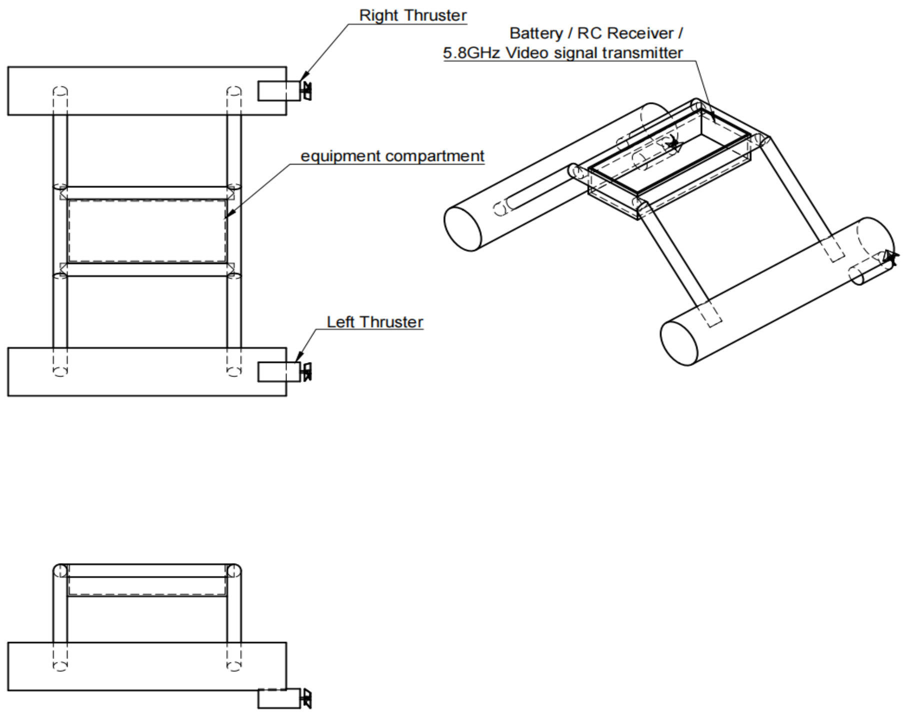

Figure 9: Mothership Design Diagram

3.4. Training of the YOLO V7 Garbage Detection Model

To parctice a model for underwater garbage detection using the YOLO V7 framework, this project chose to download an open-source underwater garbage dataset and converted it into the YOLO-specific data annotation format using Python programming. Our work conducted training using the 1660S GPU, averaging 12 hours per iteration. Ultimately, our work obtained a functional image recognition model as Figure 10:

```
                                        image_name  x_min  y_min  x_max  \
0    000000_jpg.rf.beffaf3b548106ccf1da5dc629bc9504...     75    278    130
1    000000_jpg.rf.beffaf3b548106ccf1da5dc629bc9504...    155     31    181
2    000000_JPG.rf.d3371cb3d63a59c5ba6730368b7905af...    840   1122   1568
3    000000_jpg.rf.e662cb85f63817325956fea222d0990f...     78     97    352
4    000000_jpg.rf.ee75fdf06813399a8376c6ff7056423a...    687   1213    904
..                                                ...    ...    ...    ...
464  000128_JPG.rf.eb71904df1426bfc208c7f961ae52d22...    156   1360   1214
465  000129_JPG.rf.eb4bb302dd9580081c49afc5465f58ce...    571   1160    880
466  000129_JPG.rf.eb4bb302dd9580081c49afc5465f58ce...    995    903   1418
467  000129_JPG.rf.eb4bb302dd9580081c49afc5465f58ce...    623   1589   1852
468  000129_JPG.rf.eb4bb302dd9580081c49afc5465f58ce...    423    612    942

     y_max class_name
0      294      metal
1       56      metal
2     2074    plastic
3      246    plastic
4     1432    plastic
..     ...        ...
464   2627      paper
465   1546      metal
466   1584      paper
467   2622      paper
468    984      paper

[469 rows x 6 columns]
469
```

Figure 10: Annotation and Classification of the Dataset

After that, this project analyzed the dataset as Figure 11:

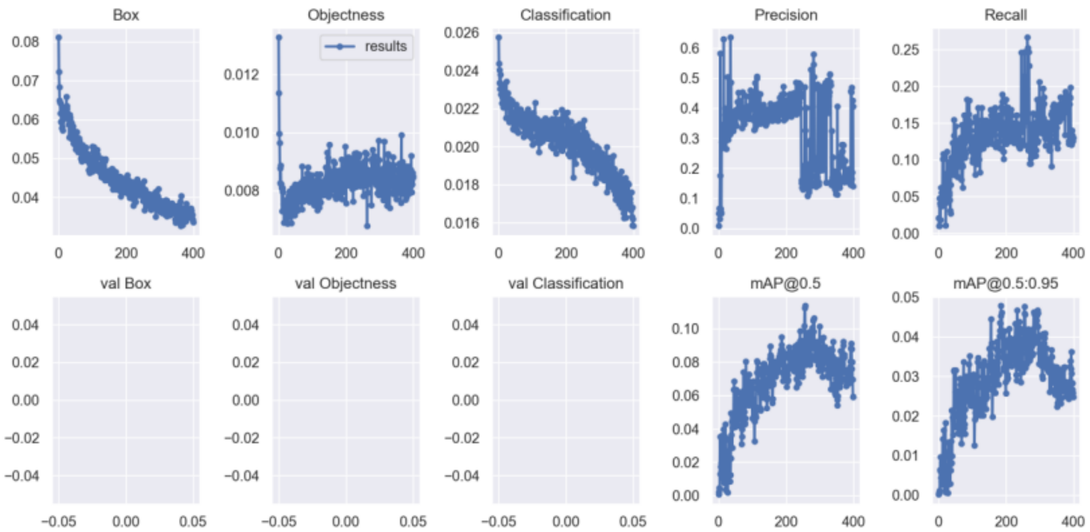

Figure 11: Parameter Convergence Status

The results are shown Figure 12:

Figure 12: Training Results

Finally, we achieved the classification and recognition of various types of plastic waste by training the YOLO V7 framework. As the model needs to run on a high-performance graphics card, we implemented the preset functions through the aforementioned image transmission system.

4. Conclusion

After field testing, this article demonstrates the possibility of solving the control model between input and output through state analysis of multi thruster robots, establishing corresponding linear equation systems, and verifying the feasibility of all ideas through practical engineering practice. Due to the AI model trained for automatic recognition and classification of underwater garbage, it can currently only run efficiently on high-performance laptops. Therefore, a wireless image transmission system is used to transmit real-time image information from the ROV perspective to the laptop. Through the above activities, the author gained a deep understanding of engineering related knowledge and skills, and implemented their initial ideas through practice.

In 2020, the EPA(Enviromental Protection Agency) reported that over 40% of lakes in the United States were polluted enough to be unsafe for fishing, swimming, or aquatic life. Besides, according to WHO(World Health Organzation), around 80% of urban wastewater globally is discharged untreated into the environment, leading to widespread water pollution issues. Therefore, in such a tense and urgent context, our underwater robotics project stands out due to its simple yet effective linear thrust allocation, YOLO image recognition, and 3D printing capabilities. These innovations reduce the costs of environmental management and enhance its efficiency, striving for a better future for our planet!

Reference

[1] Zhang C, Guo L, Luo Q, et al. Research on marine plastic garbage governance in Northwest Pacific Region from the perspective of cooperative game[J]. Journal of Cleaner Production, 2022, 354: 131636.

[2] Lebreton L. The status and fate of oceanic garbage patches[J]. Nature Reviews Earth & Environment, 2022, 3(11): 730-732.

[3] Christ R D, Wernli Sr R L. The ROV manual: a user guide for remotely operated vehicles[M]. Butterworth-Heinemann, 2013.

[4] Shahrubudin N, Lee T C, Ramlan R. An overview on 3D printing technology: Technological, materials, and applications[J]. Procedia manufacturing, 2019, 35: 1286-1296.

[5] Teague J, Allen M J, Scott T B. The potential of low-cost ROV for use in deep-sea mineral, ore prospecting and monitoring[J]. Ocean Engineering, 2018, 147: 333-339.

[6] Kabanov A, Kramar V, Ermakov I. Design and modeling of an experimental rov with six degrees of freedom[J]. Drones, 2021, 5(4): 113.

[7] Aguirre-Castro O A, Inzunza-González E, García-Guerrero E E, et al. Design and Construction of an ROV for Underwater Exploration[J]. Sensors, 2019, 19(24): 5387.

[8] Fairman F W. Linear control theory: the state space approach[M]. John Wiley & Sons, 1998.

[9] Wang C Y, Bochkovskiy A, Liao H Y M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023: 7464-7475.

[10] Faisal M, Chaudhury S, Sankaran K S, et al. Faster R-CNN Algorithm for Detection of Plastic Garbage in the Ocean: A Case for Turtle Preservation[J]. Mathematical Problems in Engineering, 2022, 2022(1): 3639222.

[11] Deng H, Ergu D, Liu F, et al. An embeddable algorithm for automatic garbage detection based on complex marine environment[J]. Sensors, 2021, 21(19): 6391.