Advances in Computer, Signals and Systems (2024)
Clausius Scientific Press, Canada

DOI: 10.23977/acss.2024.080517
ISSN 2371-8838 Vol. 8 Num. 5

Research on Pothole Detection and Avoidance Unmanned Vehicle System Based on YOLOv8 and Raspberry Pi

Fang Zeping[1,a,*], Wu Na[2,b], Liu Wenyuan[1,c]

[1]School of Automation and Electrical Engineering, Zhongyuan University of Technology, Zhengzhou, China
[2]Zhengzhou Railway Vocational and Technical College, Zhengzhou, China

[a]3943@zut.edu.cn, [b]wwwnnn3418@163.com, [c]2926611914@qq.com
[*]Corresponding author

Keywords: Unmanned vehicle, Road potholes, Detection, Avoid, Raspberry Pi, YOLOv8

Abstract: In order to reduce the harm of road potholes to the safe driving of unmanned vehicles, it is necessary to create an efficient and accurate road pothole detection and avoidance strategy. Therefore, this paper proposes a road pothole detection and avoidance unmanned vehicle (PDA-UV) system based on YOLOv8 and Raspberry Pi. The system mainly includes unmanned vehicle, road pothole detection, avoidance motion controller and image sensor. YOLOv8 is used as a road pothole detection algorithm. The motion controller of unmanned vehicle takes Raspberry Pi 4B/4G as the core and four Mecanum wheels as the motion mechanism of unmanned vehicle. Firstly, the system obtains the road pothole image through the camera; Then, the road pothole detection model is obtained after training with YOLOv8 algorithm, and the collected road scenes are tested. Finally, the road pothole detection model is deployed to Raspberry Pi 4B/4G, and the real-time motion control of the unmanned vehicle is carried out according to the identified road pothole results, so as to realize the avoidance function of the unmanned vehicle to the road pothole. In this paper, the experimental results of road potholes detection and avoiding single road potholes are given. The experimental results show that the unmanned vehicle can accurately detect road potholes and realize the avoidance motion control of a single road pothole according to the preset trajectory at low speed.

1. Introduction

Unmanned Vehicle (UV) and intelligent transportation systems (ITS) will undoubtedly affect and change people's quality of life and future travel modes. UV and ITS make people's traffic more coordinated, safe and intelligent [1]. Advanced sensor technology, machine learning and artificial intelligence have played a prominent role in the latest progress of ITS [2]. Efficient road infrastructure plays a vital role in a country's economic development, thus improving people's living standards. Due to extreme weather events, different loads and aging, pavement defects such as potholes and cracks appear repeatedly [3].

Road potholes are considerable structural road damage caused by rainwater infiltration into the pavement [4]. Road potholes are harmful to safe transportation, which will make vehicles out of balance and become the main factor inducing accidents. Road potholes will cause vehicle damage, poor ride comfort and passengers in danger [4-6]. When a manned vehicle passes through these damaged roads, the driver can judge the road conditions in real time and control the safe driving of the vehicle. However, for unmanned vehicles that do not have the ability to deal with damaged roads at present, road potholes will affect the safe driving of unmanned vehicles because of the lack of real-time control of vehicles like drivers. In this way, an unsolved challenge in UV design is to create an efficient and accurate road pothole detection and avoidance strategy. Most of the existing research focuses on road pothole detection methods, however, the strategies and experiments to avoid road pothole collision are rarely studied. Therefore, it is very important to study and formulate a road pothole detection and avoidance strategy to jointly optimize safety and ride comfort.

In reference [7], Global Positioning System (GPS) is used to collect data and locate road potholes. Reference [8] proposed a model combining neural network, on-board diagnosis (OBD) data and methods based on visual data. In order to improve the accuracy of the system and reduce the influence of environmental factors, reference [9] proposed a pothole detection system based on the Internet of Things using Kinect sensors. Another method uses context information to provide high-quality data in a secure manner [10]. In order to reduce the computational complexity, researchers proposed image segmentation technology. Four important image segmentation algorithms are image threshold, Canny edge detection, K-means clustering and fuzzy C-means clustering. A stereo vision system with two-square weighted robust least squares approximation is used to determine the size and depth of potholes. Reference [13] proposes a system based on hybrid deep learning stereo vision to optimize the visual detection method. Reference [14] mainly uses YOLOv8 algorithm to detect road potholes, and uses and analyzes the effects of different variants of YOLOv8 (YOLOv8x, YOLOv8l, YOLOv8m, YOLOv8, YOLOv8n) to detect road potholes. The paper also gives the effect of detecting road potholes with YOLOv3, YOLOv4, YOLOv4-tiny, YOLOv5s, SSD, Fast R-CNN and other algorithms.

In a word, the existing research mainly focuses on road pothole detection using high-complexity technology and standardized data sets, rather than road pothole avoidance mechanism. In addition, the existing work does not provide the rider with the means to help the road pothole detection or avoid the accuracy. Reference [14] shows that road pothole detection based on YOLOv8 algorithm has outstanding advantages. Therefore, in order to further study the road pothole detection and avoidance for unmanned vehicles, this paper designs a road pothole detection and avoidance system based on YOLOv8 and Raspberry Pi. The system obtains the images of road potholes through the camera, and obtains the road pothole detection model after training with YOLOv8 algorithm, and tests the collected road scenes. Finally, the road pothole detection model is deployed to Raspberry Pi 4B/4G, and the real-time motion control of the unmanned vehicle is carried out according to the identified road pothole results, so as to realize the avoidance function of the unmanned vehicle to the road pothole.

2. Unmanned vehicle design scheme

The design scheme of unmanned vehicle is shown in Figure 1. The unmanned vehicle mainly includes Raspberry Pi 4B/4G, camera, four DC motors, four Mecanum wheels, PCA9685, two pieces of TB6612FNG and a frame. Take Raspberry Pi 4B as the control core unit. In order to meet the high flexibility requirements of unmanned vehicles, Mecanum wheels with omni-directional motion are adopted. DC motor drives Mecanum wheel to rotate. PCA9685 and TB6612FNG are used as the direction and speed control and driving chips of DC motor. LED and speaker are used for light and sound reminder. The camera is installed in the front of the unmanned vehicle, which is used to acquire the road scene image and identify the road potholes in the image by YOLOv8 algorithm. According to the identified information of road potholes, Raspberry Pi generates the trajectory of unmanned vehicles to avoid road potholes, and controls the unmanned vehicles to complete the avoidance movement.

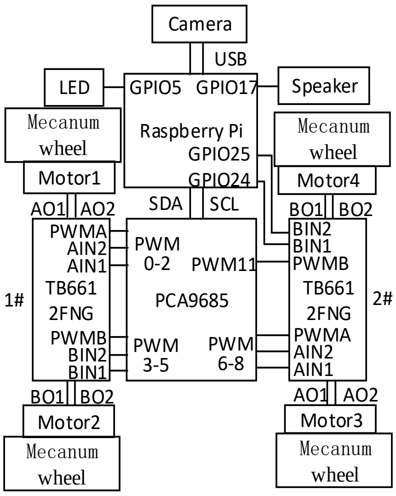


Figure 1: Design scheme diagram.

3. Hardware design of unmanned vehicle based on Raspberry Pi

3.1. Raspberry Pi

The road pothole detection algorithm designed in this paper will be applied to low-cost, low-power and small embedded devices, and will be loaded on unmanned vehicles, thus contributing to the popularization of the system [15]. At present, mainstream embedded products include Raspberry Pi and NVIDIA series products. Because the road pothole detection algorithm in this paper is real-time detection of video stream, the computing power of single CPU is difficult to meet the requirements. In addition, although NVIDIA series products have integrated GPU based on NVIDIA CUDA, and the calculation speed is very fast, the cost is very high. Raspberry Pi is a general embedded device controlled by microcomputer in the industry. It also integrates various resources such as sensing and communication, and has higher performance than microcontroller and lower cost than NVIDIA products. It has the characteristics of light weight, low power consumption, strong performance and low cost. In addition, Raspberry Pi edge computing nodes can assign tasks to neighboring nodes to reduce load, delay and improve resource utilization. Raspberry Pi can be combined with neural network computing stick 2(NCS2) in the later stage, which can significantly improve the computing power, so this research chooses Raspberry Pi 4B/4G as the core hardware. The hardware parameters of Raspberry Pi 4B/4G are listed in Table 1.

Table 1: Table of Main Parameters of Raspberry Pi 4B/4G.

Parameter name	value	Parameter name	value
processor	64-bit ARMv8	Memory	4GB
GPU	0.5GHz	USB	2xUSB3.0 2xUSB2.0
wireless	80211ac(2.4/2.5GHz)	Power	USB Type-C
Bluetooth	5.0	HDMI	2 micro HDMI interface
size	85x56x17mm	audio frequency	3.5mm audio jack

3.2. Unmanned vehicle movement direction control principle

The motion mechanism of unmanned vehicle has four Mecanum wheels, which are two left-handed wheels and two right-handed wheels respectively. The combination of four Mecanum wheels is shown in Figure 2. The steering cooperation of four Mecanum wheels can generate force in any direction and also generate rotating moment, and the arm of the rotating force is also relatively long, so the unmanned vehicle can realize omni-directional motion [16]. Each Mecanum wheel is connected with a DC motor fixed on the bottom plate through a coupling, and each Mecanum wheel is independently driven by the DC motor. Each Mecanum wheel can rotate forward, reverse and stop by driving a DC motor. By controlling the rotation direction and stop of four DC motors, the unmanned vehicle can move in different directions. In Figure 2, the arrow direction represents the moving direction of the unmanned vehicle. "1", "0" and "-"represent forward rotation, reverse rotation and stop of DC motor respectively. The different combinations of four digits "1", "0" and "-"represent the different moving directions of the unmanned vehicle, such as "1111", "0101" and "-11-" respectively represent the unmanned vehicle's forward movement, left movement and left front movement.

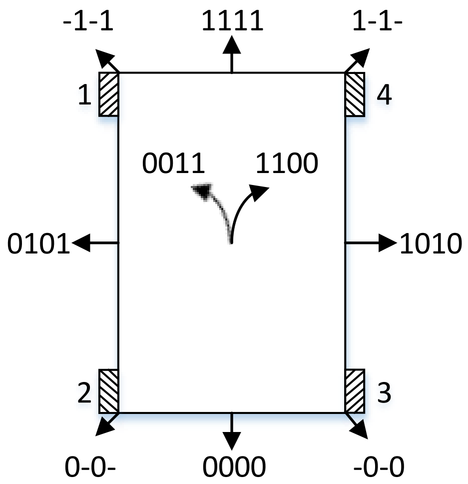


Figure 2: Schematic diagram of unmanned vehicle motion direction control.

3.3. Unmanned vehicle motion control principle

3.3.1. Driving control principle of four Mecanum wheels

The control principle of four Mecanum wheels is shown in Figure 1. PCA9685 chip realizes the output of PWM signal. PCA9685 chip is connected with Raspberry Pi through I2C interface, so as to receive the control command from Raspberry Pi and program it online. PCA9685 chip can output PWM0-PWM15 signal with variable duty cycle through programming. TB6612FNG is a DC motor drive integrated circuit, and one TB6612FNG chip can drive two DC motors. Because there are four DC motors driving Mecanum wheels on the unmanned vehicle, two TB6612FNG are needed here. Taking the driving control principle of DC motor 1 on unmanned vehicle as an example, the driving control principle of Mecanum wheel is analyzed. PWMA is the PWM signal input pin of TB6612FNG chip, which is connected to the PWM0 pin of PCA9685. The duty ratio of PWM0 pin output is set by programming, so as to adjust the speed of DC motor 1. AIN1 and AIN2 are motor rotation direction control pins, which are connected to PWM1 and PWM2 pins of PCA9685 respectively. By programming PWM1 and PWM2 of PCA9685 to output high and low levels, the rotation direction of DC motor is controlled. When the duty cycle is 100%, the PWM pin output of

PCA9685 is high; When the duty cycle is 0%, the PWM pin output of PCA9685 is low. The driving and control principles of DC motor 1~ 4 are the same as those of DC motor 1, so they are not detailed here [17].

3.3.2. The principle of controlling motor 4 through Raspberry Pi GPIO

As shown in Figure 1, the direction and speed of motor 4 are controlled by BIN1, BIN2, and PWMB of TB6612FNG chip 2. The GPO24 and GPO25 of Raspberry Pi are connected to the BIN1 and BIN2 pins of the TB6612FNG chip 2, and the rotation direction of the motor 4 is controlled by setting the high and low levels of GPO24 and GPO25 outputs.

The GPIO port of Raspberry Pi is a universal input and output port. We can run python code to make these ports output high and low levels, or input the information of external hardware to Raspberry Pi through GPIO port. The unmanned vehicle sets the port of Raspberry Pi through python's integrated library RPi.GPIO. Raspberry Pi has two types of pin numbering rules: BOARD number and BCM number, and BCM number rule is adopted here [18]. Specify BCM numbering rules through python statement GPIO.setmode (GPIO.BCM). For Raspberry Pi, GPIO245 and GPIO25 are output pins, and GPIO24 and GPIO25 are set as output pins through codes GPIO.setup(24,GPIO.OUT) and GPIO.setup(25,GPIO.OUT). Set the output high level or level of GPIO24/GPIO25 through the code GPIO.output(24/25,1/0).

4. Road potholes detection based on YOLOv8

4.1. YOLOv8 algorithm

YOLOv8 is one of the main single-phase target detection algorithms of YOLO, including five different versions: YOLOv8n, YOLOv8s, YOLOv8m, YOLOv28l and YOLOv8x. The overall structure of YOLOv8 consists of spine, neck and head. However, it has a new architecture, an improved convolution layer (backbone) and a more advanced detection head, and now it not only supports target detection tasks, but also supports image classification and instance segmentation tasks [19].

4.2. Road pothole detection and pothole information based on YOLOv8

The operating systems used in this study are Windows 11 Professional Edition and Ubuntu 16, the CPU model is 11th generation Intel(R) Core(TM)i5-11800F, the GPU is NVIDIA GeForce RTX 3060, and the compilation language is Python 3.8.17. YOLOv8 version is ultralytics8.0.135. The road pothole detection process based on YOLOv8 is shown in Figure 3. It mainly includes three steps: data set processing, model training and road pothole detection. In this project, the model training mainly includes three steps: installing ultralytics (pip Install ultralytics), creating YOLOv8 model (model = YOLO ("YOLOV8m.pt")) and model training (model.train(data, epochs, batch)). YOLO model training needs to set parameters, and YOLO model has multiple parameters and configurations for model training of data sets. In the above model training instruction, data indicates the location of the training data set file. The data set is a .yaml file. The data set file contains the images and labels needed for training and verification. The epochs is the number of training rounds, and its default value is 100. This parameter sets how many times the model will be trained, and each round will traverse the whole training data set. The batch is the number of images in each batch. This parameter determines the number of images contained in each batch. Other parameters take default values. In this paper, data is a data set of road potholes, with an epochs value of 250 and a batch value of 8. After the model training is completed, an optimal training result model file, best.pt, can be obtained, which is the model file with the minimum loss value and used for subsequent road pothole detection. The main process of road pothole detection includes loading model, road pothole prediction (model = yolo (path =' best.pt', task =' detect') and outputting road pothole information (results = model (img_pothole)). The path is to load the model obtained above. In order to detect road potholes, the task parameter is designated as' detect', which means to detect targets. The img_pothole specifies the image file path and image file of the road potholes to be detected. In this paper, the image file is a real-time road scene image collected by the camera on the unmanned vehicle.

The results are the object list of road pothole detection results. Each result contains boxes, masks, keypoints and probs information. In this paper, boxes provides road pothole detection information, which mainly provides data, xywh, xywhn, xyxyn and xyxyxyn. Data contains the information of the bounding box of road potholes, that is, the coordinates (x, y), length and width of the left vertex of the bounding box. The xywh is the center coordinate (x, y), width and height of the bounding box. The xywhn is the normalized bounding box coordinates and dimensions, and the value range is usually between 0 and 1. The xyxy is the upper left corner coordinate and the lower right corner coordinate of the bounding box. The bounding box coordinates of xyxynxyn normalized xyxy format.

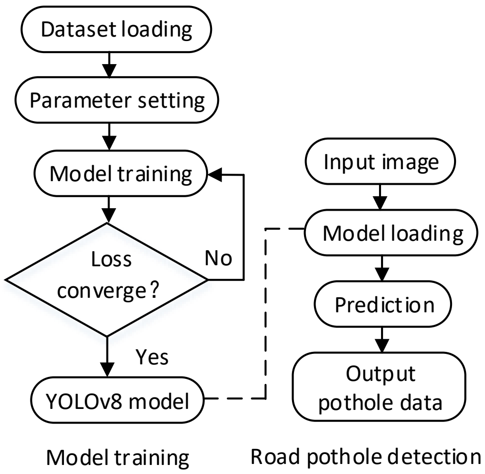
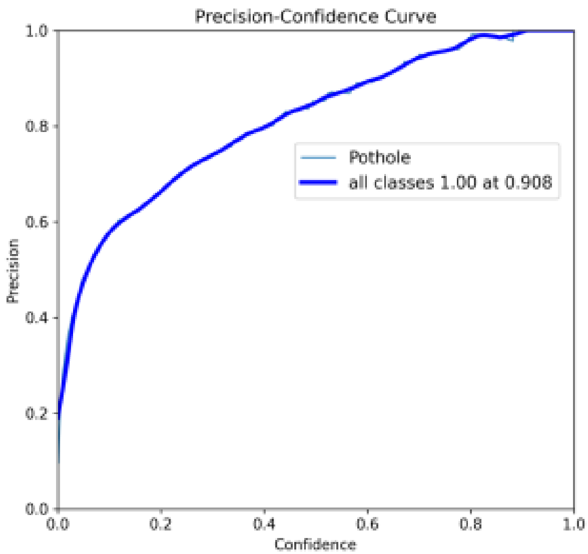


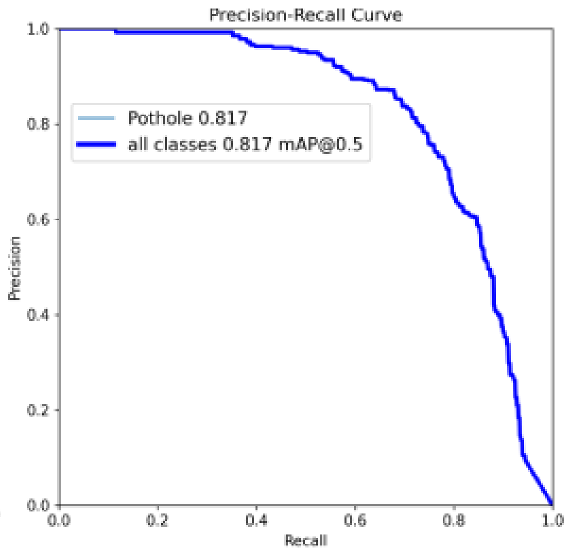
Figure 3: Road pothole detection process diagram based on YOLOv8.

4.3. Training results

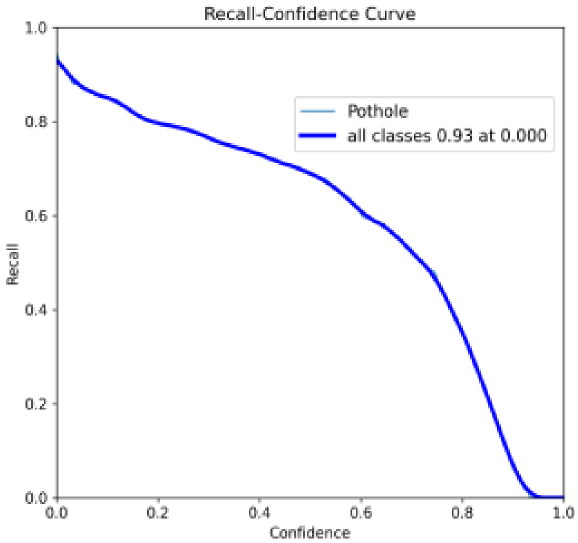
The data in this paper comes from the open source data set RDD2022[20]. This data set contains various types of road defects, such as cracks, potholes and road breaks. The data set is divided into training image set and test image set. After labeling these image sets, the YOLOv8 model is trained, and the training result is shown in Figure 4. According to the result of the image, the accuracy value for detecting road potholes is 90.8%. In addition, Figure 5 is a loss function diagram. Figure 6 is the result of the confusion matrix model used. Confusion matrix is a method to summarize the performance of classification algorithm. Diagonal line shows the importance of prediction results in confusion matrix; Horizontal and vertical lines indicate false negatives and false positives, respectively.



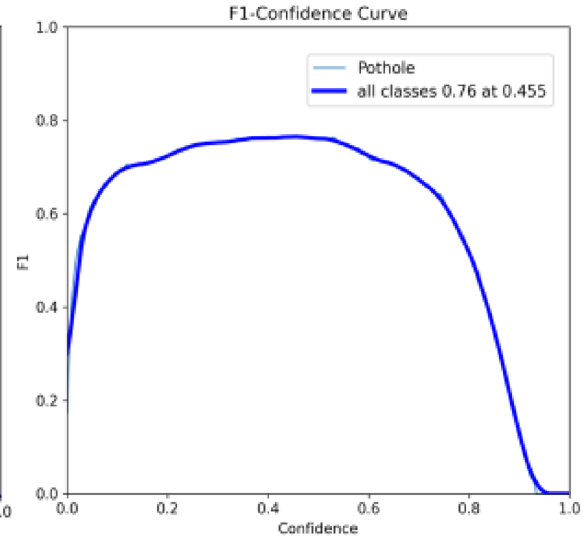
(a) Confidence-accuracy



(b) Accuracy-recall curve



(c) Recall-confidence curve



(d) F1- confidence curve

Figure 4: Training results.

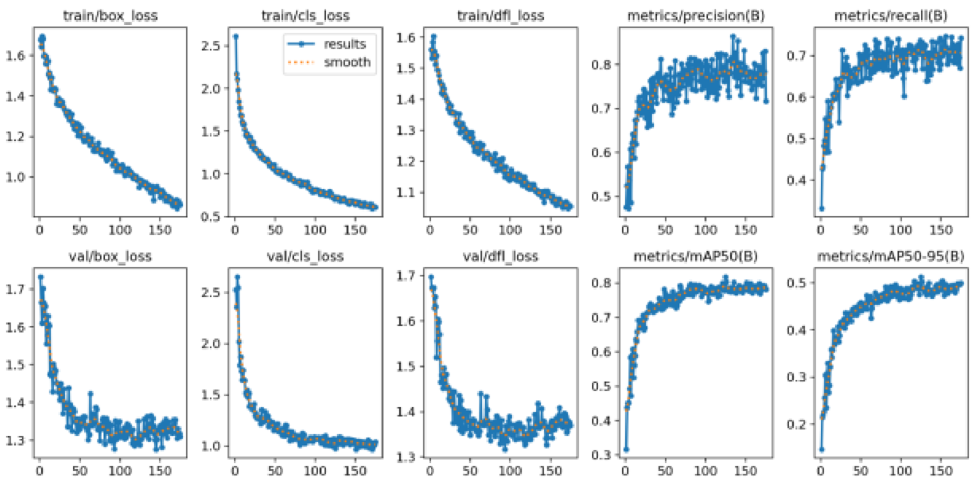


Figure 5: Loss function.

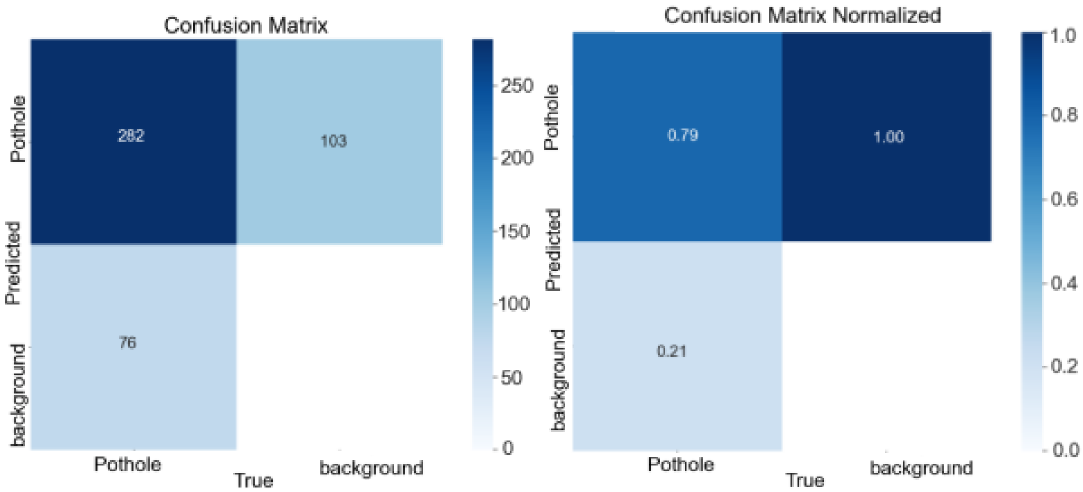


Figure 6: Confusion matrix.

4.4. Experimental results

In order to ensure the validity and accuracy in the actual road environment, the model obtained in Section 4.3 is verified. Take the verification result of an image, as shown in Figure 7. Pothole in Figure 7 represents road potholes.

Figure 7: Verification results.

5. Avoidance experiment

Based on the above road pothole detection methods, next, we carry out experimental tests.

5.1. Unmanned vehicle

The unmanned vehicle used for experimental test is shown in Figure 8. In Figure 8, 1 represents Raspberry Pi, 2 represents Raspberry Pi expansion circuit board, 3 represents camera, and 4 represents Mecanum wheel.

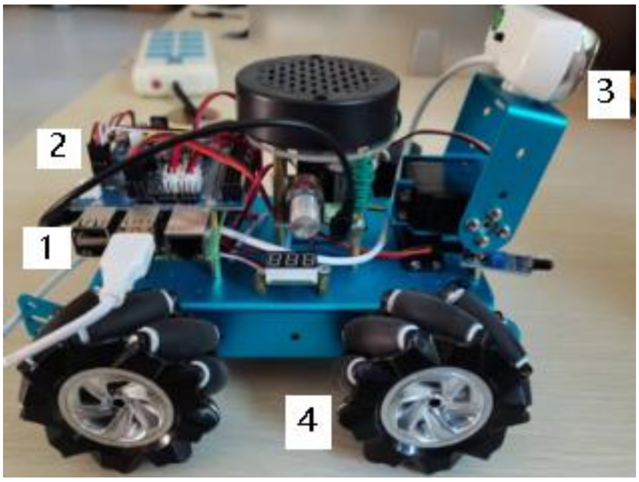


Figure 8: Unmanned vehicle.

5.2. Avoidance strategy

According to the types of potholes listed in Table 2, road potholes are classified. According to the area of road potholes, the unmanned vehicle generates its trajectory to avoid road potholes. As listed in Table 2, the types of road potholes are defined according to the number of pixels occupied [21].

Table 2: Types of road potholes in pixels.

Pothole category	Occupied pixels
Small	area $\leq 32^2$
Medium	$32^2 <$ area $\leq 96^2$
Large	area $> 96^2$

5.3. Avoidance results and analysis

5.3.1. Road pothole test results

We get a road scene image through the camera and identify the road potholes, as shown in Figure 9. Figure 9(a) is the original image, and Figure 9(b) is the road pothole recognition result.

(a) Recognition result of original image

(b) recognition result

Figure 9: Identification results of road potholes.

5.3.2. Method for extracting avoidance results

The method for extraction avoidance result can display the avoidance motion of unmanned vehicles in the form of images. Firstly, we record a video file of the unmanned vehicle avoiding road potholes. Then, we write a program in Matlab environment to convert the video file into a series of image frames. Finally, we superimpose the image frames according to equation (1) to generate a visualization result.

In Formula (1), the current image frame I_2 and the previous accumulated image frame I_1 are superimposed. In the process of superposition, the pixel value of the same pixel point takes the accumulated image frame as the priority, and partial pixel superposition and partial pixel retention methods are adopted, specifically as follows

$$D(i,j,k) = \begin{cases} I_2(i,j,k) & I_1(i,j,k) > I_2(i,j,k) \\ I_1(i,j,k) & I_1(i,j,k) < I_2(i,j,k) \end{cases} \tag{1}$$

Where i, j and k are the index positions of pixel points, and their sizes are related to the size of the image. D is a temporary image frame, and $D(i,j,k)$ is a temporary pixel value. When the pixel value of the current image frame is smaller than that of the accumulated image frame, updating the pixel value of the accumulated image frame with the pixel value of the current image frame; On the contrary, the accumulated image frames are taken first, that is, the pixel values of the accumulated image frames are reserved. According to equation (1), the algorithm traverses the pixel values of the current image frame and assigns pixel frame D to pixel frame I. We need to repeat the previous process and continue to overlay with the next image frame until all image frames are overlaid.

5.3.3. Avoidance results

Record part of the video of the experiment process, and according to Formula (1), after Matlab programming, the experimental results of unmanned vehicles avoiding road potholes are shown in Figure 10. The point P is a road pothole. According to the information of road pothole recognition, the road pothole is located in the right front of the unmanned vehicle, so it is easier for the unmanned vehicle to avoid the pothole when moving to the left. Here, the left translation movement of the unmanned vehicle is realized by taking advantage of the omni-directional movement of the Mecanum wheel. In order to obtain a clear avoidance movement process, every 30 frames of images are taken from time t1 to time t4. Road potholes were identified at time t1. At t2, the road potholes began to move to the left. At time t3, pan to the far left. At t4, the movement of avoiding road potholes is completed.

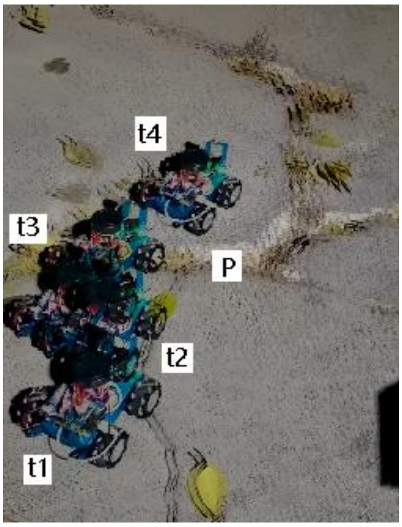


Figure 10: Experimental results of unmanned vehicles avoiding road potholes.

6. Conclusions

This paper studies an efficient and accurate road pothole detection and avoidance strategy. The YOLOv8 algorithm is adopted as the road pothole identification algorithm. Based on Raspberry Pi 4B/4G, a road pothole detection and avoidance unmanned vehicle system is designed. In this paper, the experimental results of road potholes detection and avoiding a single road pothole are given. The experimental results show that at low speed, the unmanned vehicle can not only accurately detect a single road pothole, but also realize the avoidance motion control of a single road pothole according to the preset trajectory. In the future, under the condition of higher speed, a higher performance processor will be used to detect and avoid road potholes. The research results in this paper are of great significance to reduce the influence of road potholes on the safe driving of unmanned vehicles.

Acknowledgements

This work is supported by the 2021 Science and Technology Public Relations Project of the Henan Provincial Department of Science and Technology.

References

[1] Raja G, Anbalagan S, Senthilkumar S, Dev K, Qureshi NM. SPAS: Smart pothole-avoidance strategy for autonomous vehicles [J]. IEEE Transactions on Intelligent Transportation Systems. 2022, 23(10):19827-19836.

[2] G. Raja, A. Ganapathisubramaniyan, S. Anbalagan, S. B. M. Baskaran, K. Raja, and A. K. Bashir, Intelligent reward-based data offloading in next-generation vehicular networks [J]. IEEE Internet Things J., 2020, 7(5): 3747–3758.

[3] Ahmed A, Ashfaque M, Ulhaq MU, Mathavan S, Kamal K, Rahman M. Pothole 3D reconstruction with a novel imaging system and structure from motion techniques. IEEE Transactions on Intelligent Transportation Systems. 2021, 23(5): 4685-4694.

[4] Fan R, Ozgunalp U, Wang Y, Liu M, Pitas I. Rethinking road surface 3-d reconstruction and pothole detection: From perspective transformation to disparity map segmentation. IEEE Transactions on Cybernetics. 2021, 52(7):5799-5808.

[5] S. Shah and C. Deshmukh, Pothole and bump detection using convolution neural networks[C]. In Proc. IEEE Transp. Electrific. Conf. (ITEC India), 2019(10): 1–4.

[6] A. Kumar, D. J. Kalita, and V. P. Singh, A modern pothole detection technique using deep learning[C]. In Proc. 2nd Int. Conf. Data, Eng. Appl. (IDEA), 2020(2): 1–5.

[7] M. M. Garcillanosa, J. M. L. Pacheco, R. E. Reyes, and J. J. P. San Juan, Smart detection and reporting of potholes via image-processing using Raspberry-pi microcontroller[C]. In Proc. 10th Int. Conf. Knowl. Smart Technol. (KST), 2018(1): 191–195.

[8] K. S. Ashwini, G. Bhagwat, T. Sharma, and P. S. Pagala, Triggerbased pothole detection using smartphone and OBD-II[C]. In Proc. IEEE Int. Conf. Electron., Comput. Commun. Technol. (CONECCT), 2020(7): 1–6.

[9] D. R. Reddy, G. P. C. Goud, and C. D. Naidu, Internet of Things based pothole detection system using Kinect sensor[C]. In Proc. 3rd Int. Conf., 2019(10): 232–236.

[10] A. Vora, L. Reznik, and I. Khokhlov, Mobile road pothole classification and reporting with data quality estimates[C]. In Proc. 4th Int. Conf. Mobile Secure Services (MobiSecServ), 2018(2): 1–6.

[11] A. Ahmed, S. Islam, and A. Chakrabarty, Identification and comparative analysis of potholes using image processing techniques[C]. In Proc. IEEE Region 10 Symp. (TENSYMP), 2019(6): 497–502.

[12] Y. Li, C. Papachristou, and D. Weyer, Road pothole detection system based on stereo vision[C]. In Proc. IEEE Nat. Aerosp. Electron. Conf., 2018(6): 292–297.

[13] A. Dhiman and R. Klette, Pothole detection using computer vision and learning [J]. IEEE Trans Intell. Transp. Syst., 2020, 21(8): 3536–3550.

[14] Kumari S, Gautam A, Basak S, et al. YOLOv8 based deep learning method for potholes detection[C]. 2023 IEEE International Conference on Computer Vision and Machine Intelligence (CVMI). 2023.10465038.

[15] Luo Y, Ci Y, Jiang S, Wei X. A novel lightweight real-time traffic sign detection method based on an embedded device and YOLOv8 [J]. Journal of Real-Time Image Processing. 2024, 21(2):24.

[16] Si Wenzhan.Structural design and motion control system research of intelligent omnidirectional mobile platform

[D].Jiangsu Ocean University, 2022.
[17] Q. Wang, S. Wang and H. Ni, Design of an odor search robot system based on open sampling system[C]. 2021 33rd Chinese Control and Decision Conference (CCDC), Kunming, China, 2021, 3383-3388.
[18] Li Sensen, Zhu Shiwei, Shi Liyu, et al. Automatic humidity control device based on RPi [J]. Electronic world, 2020: 91-93.
[19] Zhao M, Su Y, Wang J, Liu X, Wang K, Liu Z, Liu M, Guo Z. MED-YOLOv8s: a new real-time road crack, pothole, and patch detection model [J]. Journal of Real-Time Image Processing. 2024, 21(2):26.
[20] Arya, D., Maeda, H., Ghosh, S.K., et al. Crowdsensing-based road damage detection challenge[C]. (CRDDC'2022). In: Proceedings of the 2022 IEEE International Conference on Big Data (Big Data), 2022.
[21] Kuan CW, Chen WH, Lin YC. Pothole detection and avoidance via deep learning on edge devices[C]. In 2020 international automatic control conference (CACS) 2020(10): 1-6.