Binocular intelligent following robot based on YOLO-LITE

Yuanyuan Zheng[1], and *Jun* Ge[1,*]

Nanjing University of Posts and Telecommunications School of Modern Posts, Nanjing, 210003, China

Abstract. In order to solve the problem that the deep neural network model is large in scale, the calculation time is too long, and the real-time performance is severely limited when combined with embedded devices, so studied the intelligent follower robot system based on YOLO-LITE algorithm combined with Raspberry Pi 3B+. The system mainly includes camera processing, target detection and other modules. Obtained the internal and external parameters of the camera through calibration, and according to these parameters to correct the binocular camera. Recognized and located the target in each frame of image, calculated the distance from the camera to the target and the center location error, and driven the car to move. The experimental results show that the following car has excellent real-time performance, the average detection frame rate can reach 20Fps, and the average detection accuracy can reach more than 80%.

1 Overview

In recent years, intelligent service robots have been greatly developed by riding on the ride of the booming artificial intelligence technology. Various service robots, such as cleaning robots, smart wheelchairs, etc., are constantly improving people's quality and level of life,the intelligent follower robot is an important part and functional module of the service robot, and the implementation technology of the follower robot is also diverse, mainly including infrared-based follow-up[1], and ultrasonic-based follow-up[2].The realization of intelligent following robot in this paper depends on good vision target detection technology

On the other hand, in recent years, deep learning technology has been developed rapidly, and neural networks have also been widely used in the field of visual target detection. Different from traditional methods, deep learning is to use deep convolutional neural networks to extract high-level semantic information of the target to be detected and tracked [3]. At the same time, with the continuous improvement of the performance of computers and various embedded devices, as well as the continuous efforts of experts and research institutions at home and abroad in the field of visual target detection, many new research results have been obtained in this field, and target detection algorithms have also been obtained. Continuous optimization. Google's SSD Mobilenet V1 target detection model can

[*] Corresponding author: 2724531334@qq.com

run at 5.8fps (Frame Per Second) on non-GPU devices, and mAP (Mean Average Precision) can also reach 21% [4]. Joseph Redmon et al. [5] of the University of Washington proposed the YOLOv2 target detection algorithm. This algorithm draws on the idea of Faster R-CNN, introduces Anchor, and solves the inaccurate positioning of the YOLO algorithm itself. Compared with the method based on Region Proposal, the recall The disadvantage of lower rate greatly simplifies the network. Divvala S et al. [6] of the Alle Artificial Intelligence Research Institute in the United States proposed the Tiny-YOLO algorithm. The innovation of the algorithm is to convert the detection and positioning problem into a regression problem. The image can be obtained by processing the image once. The positions of all targets contained in the image. Li et al. [7] trained multiple networks based on Fast R-CNN [8] to detect pedestrians of different scales, and combined the results of all networks to produce the final result.

Although the above work has done a lot of exploration and research on visual target detection, but in practical applications, many neural networks have larger model scales and calculations. The real-time performance is severely restricted due to problems such as excessively long process time. In addition, the embedded device is well combined with the visual target detection algorithm, and there are still fewer intelligent follow-up robot systems that use cameras to capture and track targets. The research in this article is an intelligent follower robot system based on Raspberry Pi 3B+ and YOLO-LITE algorithm.

2 Hardware design

The hardware part of this system is mainly composed of video acquisition module, display output module, Raspberry Pi 3B+ microprocessor and its peripheral peripheral configuration. This system is designed on the Raspberry Pi 3B+ microprocessor and embedded Linux system platform. First, the video image frame is obtained through the USB binocular camera. After the obtained binocular image is cut, the left camera The video frame is transferred to the Raspberry Pi 3B+ microprocessor (this article uses the left camera to obtain the video frame image, the right camera can also be used, and the left and right cameras are used for target ranging), through the YOLO-LITE deep learning algorithm It performs functions such as target recognition, positioning and tracking. Then use tools such as SSH and VNC to remotely log in to the Raspberry Pi, and display the processed results remotely.

In terms of image collection, this article directly uses a drive-free USB binocular camera, which is convenient and quick. Raspberry Pi 3B+ is an ARM-based micro computer motherboard, small in size, suitable for various small smart devices, with complete computer processing functions, and supporting large-scale data calculations, graphics and image processing applications[9].

3 Software design

The design is a target detection and tracking system based on Raspberry Pi 3B+ microprocessor. The system is generally divided into four modules: the camera processing module of the binocular camera, the target detection module, the ranging module and the vehicle body movement. The overall principle block diagram of the system is shown as in Fig.2.

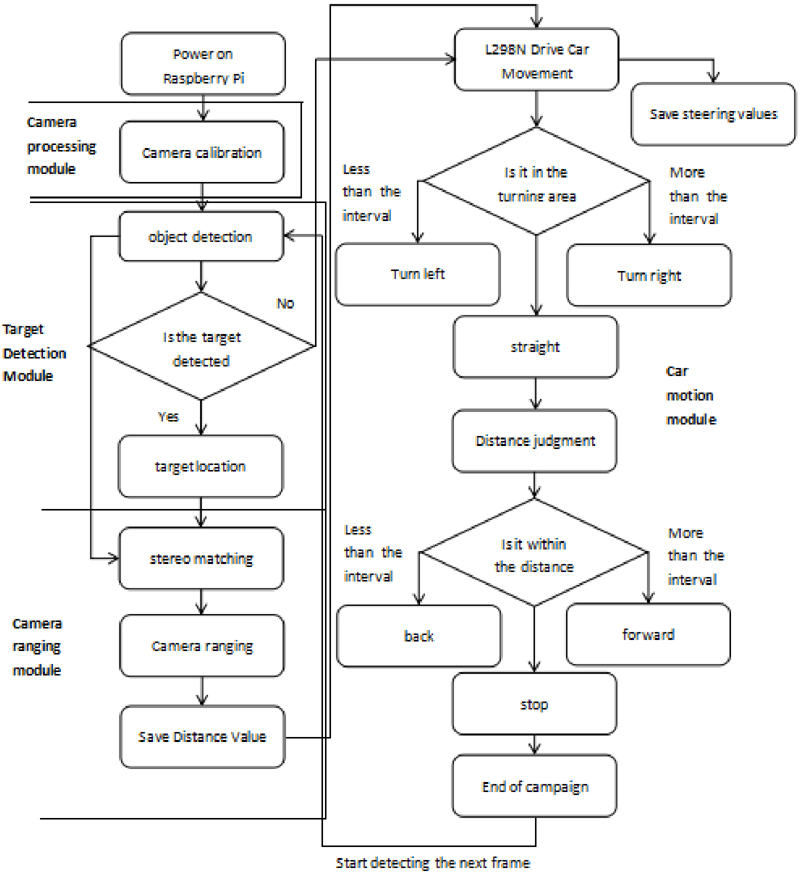


Fig.1. Overall system schematic

4 Design and implementation of each module

The main functional modules of the follower robot system designed in this article are shown in Figure 1.

4.1 Camera processing module

The main work of the camera processing module is camera calibration. The input of camera calibration is the image coordinates of all internal corner points on the image of the calibration plate and the three-dimensional coordinates of all internal corner points on the image of the calibration plate [10] (in general, it is assumed that the image is located at Z=0 on flat surface). The output of camera calibration is the camera's internal and external parameter coefficients [11].

This article uses Opencv combined with Zhang Zhengyou's algorithm to achieve the specific process of camera calibration as shown in Figure 2:

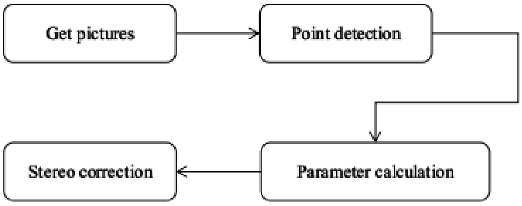


Fig.2. Camera calibration flow chart.

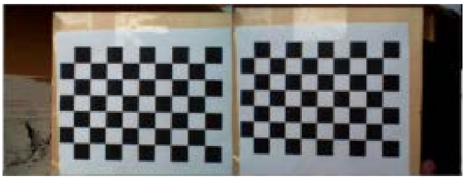


Fig.3. Checkerboard required for calibration.

First, this article uses a 1.3 megapixel USB drive-free binocular camera to shoot checkerboard images at different positions, different angles, and different postures, as shown in Figure 3. In order to ensure the accuracy of the parameters, 20 pictures were collected by the left and right cameras [12]. Then, corner detection is performed on the left and right checkerboards, as shown in Figure 4 and Figure 5.

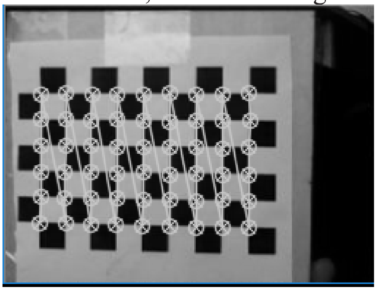


Fig. 4. Right checkerboard corner detection.

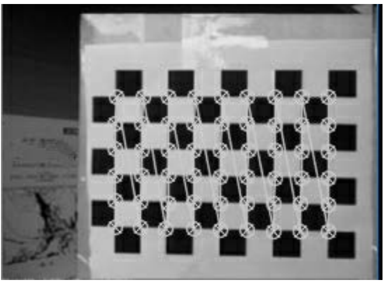
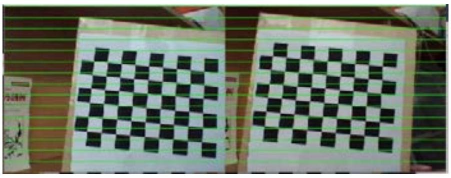


Fig. 5. Left checkerboard corner detection.

Secondly, each image of the left and right cameras is calibrated separately. Then, calculate the internal parameter matrix of the bi-objective centering. Finally, perform stereo correction on the calibrated result, calculate the external parameter matrix, and perform distortion correction. Since the tangential distortion is small, the main consideration in this paper is the radial distortion[13]. At the same time, the use of epipolar constraint for binocular correction can make the feature points lie on the epipolar line in both images[14], which greatly accelerates the calculation speed and reduces errors matches[15]. The effect pictures before and after binocular correction are shown in Figure 6 and Figure 7:



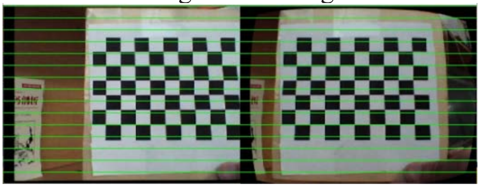


Fig.6. Effect picture before binocular correction. **Fig.7.** Effect picture after binocular correction.

After the binocular calibration is completed, the binocular vision calibration is completed. The parameter results of each camera calibration in this article are shown below.

$$M_1 = \begin{bmatrix} 805.99 & 0 & 396.65 \\ 0 & 807.77 & 23062 \\ 0 & 0 & 1 \end{bmatrix} \qquad M_2 = \begin{bmatrix} 810.11 & 0 & 349.99 \\ 0 & 810.85 & 237.40 \\ 0 & 0 & 1 \end{bmatrix}$$

$$D_1 = [-0.5003 \quad 0.3033 \quad 0.0069 \quad -0.0079 \quad 0.0000]$$
$$D_2 = [-0.4654 \quad 0.1629 \quad 0.0042 \quad -0.0012 \quad 0.0000]$$

$$R = \begin{bmatrix} 0.999900 & -0.003388 & 0.013702 \\ 0.003412 & 0.999993 & -0.001677 \\ -0.013677 & 0.001723 & 0.999905 \end{bmatrix} T = \begin{bmatrix} -109.11 & 0.1857 & 2.3577 \end{bmatrix}$$

Among them, M_1, M_2, D_1, D_2, R and T are the internal parameters, distortion parameters, rotation matrix, and translation matrix of the left and right cameras, respectively.

4.2 Target detection module

After the camera calibration and binocular correction are completed, the camera passes the acquired frame of image data to the target detection module, which mainly recognizes and tracks the target in the image.

4.2.1 Core algorithm

The target detection algorithm used in this paper is YOLO-LITE, which is a simplified version of Tiny-YOLOv2, Tiny-YOLOv2 consists of 9 convolutional layers, a total of 3181 filters and 6.97 billion FLOPS [6] (floating-point operations per second). YOLO-LITE without BN (Batch Normalization) layer consists of only 7 layers, with a total of 749 filters and 482 million FLOPS. Compared with Tiny-YOLOv2, YOLO-LITE has no BN layer, which reduces FLOPS by more than 14 times.

4.2.2 Algorithm experiment

In order to reflect the real-time performance of YOLO-LITE, this article uses three algorithms of YOLOv2, YOLO-LITE and Tiny-YOLOv2 on the Raspberry Pi to compare the performance of the three models on the PASCAL VOC data set[16]. The comparison is shown in Table 1.

Table 1. Performance comparison of some models on the VOC dataset on Raspberry Pi 3B+.

Algorithm Model	mAP	FPS
YOLO-LITE	12.26%	20
Tiny-YOLOv2	23.7%	5.6
YOLOv2	48.1%	1.6

It can be seen from the above table that although the mAP of YOLO-LITE is lower than that of YOLOv2 and Tiny-YOLOv2, it can still meet the performance requirements of model detection. At the same time, when YOLO-LITE is running, FPS is 3.8 times faster than Tiny-YOLOV2 and 13 times faster than YOLOV2, which greatly improves the operating efficiency of the model. It can be seen from the above table that although the mAP of YOLO-LITE is lower than that of YOLOv2 and Tiny-YOLOv2, it can still meet the performance requirements of model detection. At the same time, when YOLO-LITE is running, FPS is 3.8 times faster than Tiny-YOLOV2 and 13 times faster than YOLOV2, which greatly improves the efficiency of target detection.

4.3 Camera ranging module

In this module, through the stereo matching of the camera, and according to the result of camera calibration and the position in the camera coordinate system where the target is located in the target detection module, the coordinates of the world coordinate system where the target is located are measured, and then the target position is obtained through depth calculation. Depth information in the binocular camera to complete the camera ranging.

4.3.1 Stereo matching and depth calculation

In order to avoid the occurrence of obstructions or the image is not smooth, the matching process will become messy, and in severe cases, it will also lead to errors in matching, this paper uses a fast and effective block matching stereo algorithm to stereo match the image.

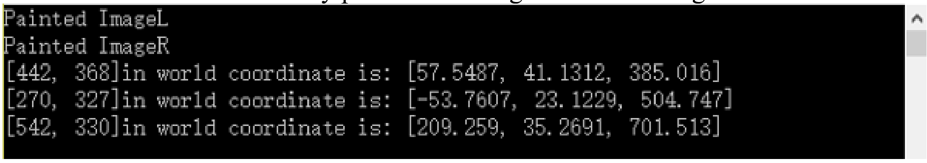
In order to accurately obtain the distance of a point in three-dimensional space, the parameters we need to obtain are focal length, parallax, and camera center distance.

Through Equation 1, the spatial coordinates of the target object can be obtained. Then through the camera's external parameters and parallax, internal parameter values, so as to finally complete the preparation work needed to find the three-dimensional coordinates of a point.

As shown in Equation 6, it is the distance conversion formula[17],

$$\frac{T-(x^l-x^r)}{Z-f}=\frac{T}{Z}\Rightarrow Z=\frac{fT}{x^l-x^r} \tag{1}$$

The distance information of any point in the image is shown in Figure 8 below:



```
Painted ImageL
Painted ImageR
[442, 368]in world coordinate is: [57.5487, 41.1312, 385.016]
[270, 327]in world coordinate is: [-53.7607, 23.1229, 504.747]
[542, 330]in world coordinate is: [209.259, 35.2691, 701.513]
```

Fig.8. Real-time display of distance measurement based on console.

4.4 Car Motion Module

This article mainly uses the L298N motor drive module combined with the Raspberry Pi 3B+ to control the car body to move accordingly after obtaining the distance of the target.

4.4.1 Vehicle body control

L298 can drive two motors, OUT1, OUT2 and OUT3, OUT4 can be connected to each motor, this experimental device we choose to drive a motor. Pins 5, 7, 10 and 12 are connected to the input control level to control the forward and reverse rotation of the motor. EnA and EnB are connected to the control enable terminal to control the stop of the motor.

The driver board can drive two DC motors. It is valid when the enable terminals ENA and ENB are at high level. The control mode and DC motor status table are shown below. If you want to perform PWM speed regulation on a DC motor, you need to set IN1 and IN2, determine the direction of rotation of the motor, and then output PWM pulses to the enable terminal to achieve speed regulation[18]. Note that when the enable signal is 0, the motor is in a free stop state. When the enable signal is 1, IN1 and IN2 are 00 or 11, the motor is in a braking state to prevent the motor from rotating. The control logic is shown in Table 2:

Table 2. L298N logic table.

Electrical machinery	Rotation mode	Control end IN1	Control end IN2	Control end IN3	Control end IN4	Input PWM signal change pulse width adjustable speed	
						Speed regulating terminal A	Speed regulating terminal B
M1	corotation	high	low	/	/	high	/
	reversal	low	high	/	/	high	/
	stop	low	low	/	/	high	/
M2	corotation	/	/	high	low	/	high
	reversal	/	/	low	high	/	high
	stop	low	low	/	/	/	high

4.4.2 Car body steering motion

Under the control of the L298N motor drive module, the car body performs corresponding motions. In addition to linear motion, the system studied in this article also has the function of left and right steering. The indicator for judging the steering of the trolley in this article is the center location error CLE (center location error) .

The CLE is the distance between the abscissa of the world coordinate of the center point of the target frame obtained by the target detection algorithm and the abscissa of the world coordinate of the marked target center. When the offset value is within the turning range, the car will keep going straight, otherwise it will make a corresponding turning movement. The specific steering judgment method is shown in Equation 2.

$$\begin{cases} CLE<-30cm \quad turn \quad left \\ CLE>+30cm \quad turn \quad right \end{cases} \tag{2}$$

4.5 Car body following experiment

The main content of this article is the research and application of the intelligent follower car system based on Raspberry Pi.

This article mainly measures the following performance of the trolley from three aspects: the delay time of the trolley, the detection accuracy of the trolley and the following distance of the trolley.

The delay time of the trolley is an indicator to measure the phenomenon of the trolley stuck. This article mainly analyzes the impact of the image resolution on the delay time of the trolley. In addition, the detection accuracy of the trolley is an index used in this paper to measure the reliability of the detection of the trolley. Among them, the detection accuracy of the car is measured by the average confidence level ($\frac{1}{n}\sum_{k=i}^{n} con_i \qquad (1 \le i \le n)$ where *n* is the number of effective image frames, that is, the total number of image frames involved in the target detection, *i* is the current image frame number,con_i is the confidence of the current image frame target detection) In this paper, the average value of the confidence is used as one of the indicators to measure the detection accuracy of the car to eliminate the influence of abnormal phenomena in which some frames are not detected or the target confidence is lower than the set confidence threshold. The following distance of the car mainly measures the performance of the car following from the real-time distance and error between the car and the target, and the trajectory of the target and the car.

4.5.1 The impact of image resolution on the experiment

There is a close relationship between image resolution and image size. The higher the image resolution, the more pixels it contains, the greater the amount of information in the image. The image resolution also has a certain influence on the delay time of the trolley.

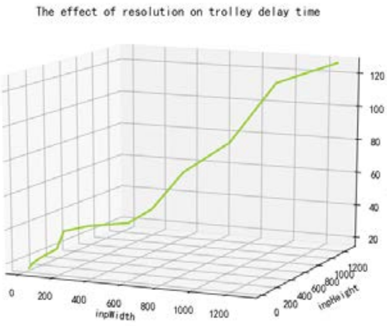


Fig.9. The effect of resolution on the delay.

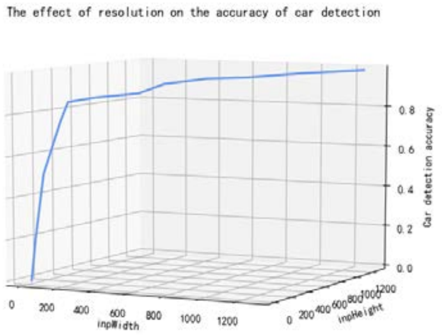


Fig.10. The effect of resolution on the detected accuracy.

Figure 9 reflects the influence of different resolutions on the delay time of the trolley. It can be seen from the figure that, on the whole, multi-threading can greatly reduce the stall phenomenon. At the same time, as the resolution increases, the delay time of the trolley shows an upward trend. In the case of small resolution, the delay time of the trolley is also small. This is mainly because the resolution is too low, the image has too little information, and the target cannot be successfully detected, so it takes less time. With the improvement of image resolution, the amount of image information continues to increase, and the required operating time will increase accordingly.

At the same time, there is a close relationship between the image resolution and the detection accuracy of the car.

Figure 10 reflects the impact of different resolutions on the detection accuracy of the car. It can be seen from the figure that the detection accuracy of the car is also improving as the image resolution increases. This is because as the image resolution increases, the pixels contained in the image It is also increasing, and the amount of information contained in the image has also been correspondingly improved. Therefore, the detection accuracy of the car is continuously improved as the resolution of the image increases.

However, the resolution of the image is not as high as possible. Although it will improve the detection accuracy of the car, it will greatly increase the delay time of the car and aggravate the carton phenomenon during the movement of the car. This is extremely disadvantageous. Therefore, according to the above experiment, this article finds an optimal resolution (256X256). Under this resolution, the detection accuracy of the system is 80%, and the delay time is only 20ms. The difference between the delay time of the car and the detection accuracy of the car Maintain an optimal balance point between.

4.5.2 Car tracking experiment

when the car is tracking the target, in order to prevent the target from being too close to the car and causing the car to be damaged, this paper sets a safe distance range (95cm—105cm).

If the range is larger, the car will approach the target; if it is smaller than the range, the car will be far away from the target.

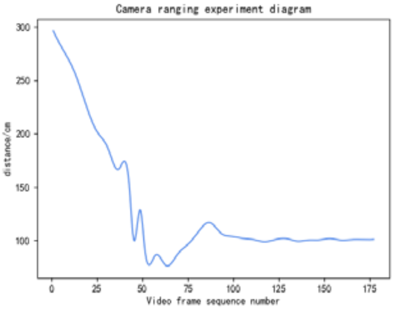


Fig.11. Camera ranging experiment diagram.

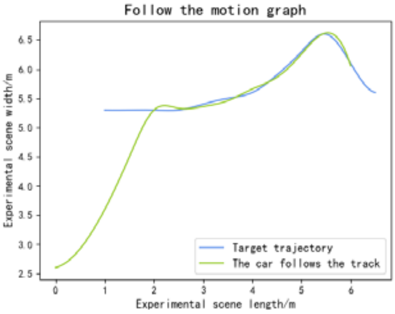


Fig.12. Follow the motion trajectory.

Figure 11 reflects the distance between the car and the target during the car tracking movement. It can be found that when the distance between the car and the target is greater than 100cm, the distance between the car and the target shows a decreasing trend. When the distance between the car and the target is less than 100cm, the distance between the car and the target is increasing, and finally the distance between the car and the target will gradually converge to Around 100cm. This shows that although the target is far and close to the trolley, the trolley continuously adjusts the movement state according to the safe distance range, and finally stops within the safe distance. In general, the tracking movement is relatively stable and can meet the delay time of the trolley Requirement for the balance between detection accuracy.

Figure 12 shows the trajectory diagram of the car and the target. The following experiment is carried out indoors. The horizontal and vertical coordinates are the length and width of the experimental site respectively. Through the trajectory of the car and the target, you can more intuitively see how the car tracks the target The process further reflects the real-time and reliability of the car's tracking movement.

5 Conclusion

In order to solve the problems of large scale of deep neural network model, long calculation time, and serious limitation of real-time performance when combined with embedded devices, this paper studies the intelligent follower robot system based on the YOLO-LITE algorithm and the Raspberry Pi 3B+ . The internal and external parameters of the binocular camera are obtained through calibration, and the binocular camera is calibrated according to these parameters, and the real-time distance and center position offset of the target are calculated. In addition, this article measures the performance of the trolley from two aspects: the real-time and accuracy of the trolley. The analysis results show that with the improvement of resolution, although the delay time of the trolley continues to increase, the detection accuracy of the trolley is also continuously improved. This article finds an optimal resolution. The average detection accuracy is over 80%, and the delay time is only 20ms, the detection speed also reaches 20fps, maintaining a good balance between the real-time performance of the car and the detection accuracy of the car.

In the next step, functional modules such as path planning and obstacle avoidance of the car will be added, and improvements will be made to the YOLO-LITE target detection algorithm to further improve the real-time performance and detection accuracy of the car.

Project sponsored by NUPTSF(Grant No.NY220063)

References

1. D. Hermann,R. Galeazzi,J.C. Andersen,M. Blanke. Smart Sensor Based Obstacle Detection for High-Speed Unmanned Surface Vehicle[J]. IFAC Papers OnLine,vol.**16**,48(2015)
2. Ryan Marks,Alastair Clarke,Carol A Featherston,Rhys Pullin. Optimization of acousto-ultrasonic sensor networks using genetic algorithms based on experimental and numerical data sets:[J]. SAGE PublicationsSage UK: London, England,vol.**11**,13,(2017)
3. Xie Juanying, Liu Ran. Research progress of target detection algorithms based on deep learning [J/OL]. Journal of Shaanxi Normal University (Natural Science Edition), vol.**05**,1-9.(2019)
4. A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand,M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” CoRR, vol. abs/1704.04861,(2017)
5. Joseph Redmon, Ali Farhadi. YOLO9000:Better, Faster, Stronger. arXiv preprint arXiv:1612.08242v1, 2016. 11,25
6. Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.Las Vegas, NV, USA. (2016). 779–788.
7. LI J, LIANG X, SHEN S, et al. Scale-aware Fast R-CNN for pedestrian detection [J]. IEEE Transactions on Multimedia,(2018).20,**4** : 985－996.
8. GIRSHICK. R. Fast R-CNN [C] // Proceedings of the 2015 IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE, 2015: 1440－1448.
9. Gao Feng, Chen Xiong, Chen Wanqiu. Video detection and tracking system based on Raspberry Pi B+ microprocessor [J]. Television Technology, 39(19): 105-108.(2015)
10. Cai Yao, Yu Tao, leaf kumyoung. Binocular stereo vision, camera calibration and distortion correction software and application of [J]. computer, 9-10,19(2012)
11. Chen C, Yang Q, Li D (2018) Moving objective tracking based on 3D coordinates. Applied Science and Technology, vol. **45**, pp. 23-28.(2018)
12. Qu Hua, Wu Chaona. Step-by-step calibration and accuracy analysis of binocular stereo vision[J]. Journal of Tianjin Polytechnic University, 3(2018)
13. Tang Zhongwei,Grompone von Gioi Rafael,Monasse Pascal,Morel Jean-Michel. A Precision Analysis of Camera Distortion Models.[J]. IEEE transactions on image processing : a publication of the IEEE Signal Processing Society,26,6(2017)
14. Men Y, Zhang G, Men C, Li X, Ma N (2015) A Stereo Matching Algorithm Based on Four-Moded Census and Relative Confidence Plane Fitting. Chinese Journal of Electronics, pp. 807-812.4(2015)
15. Yang J H, Liu W, Liu Y, et al. Calibration of binocular vision measurement system[J]. Optics & Precision Engineering, pp. 300-338.24,**2**(2016)
16. [16] PASCAL, “The pascal visual object classes home page l, Last accessed on 2018-07-18. 3(2018)
17. X. Wang, Y. Bian, F. Liu, and F. Wu. Optimization of structural parameters of binocular vision system in remote 3D coordinate measurement[J]. Optics and Precision.Engineering,2902-2908.23,10(2015)
18. Yin Liuliu, Han Sen, Wang Fang, Li Yuchen, Sun Hao, Li Chunjie, Wang Quanzhao. Design and Application of DC Motor Speed Regulation System Based on L298N [J]. Information Technology,104-106(2017)