

# Computer vision based obstacle detection and target tracking for autonomous vehicles

Ruoyu Fang<sup>1</sup>, and Cheng Cai<sup>1,\*</sup>

<sup>1</sup>School of Electronic Information Engineering, Shanghai Dianji University, Shanghai, China

**Keywords:** Target tracking, obstacle detection, target tracking, deep learning, PID

**Abstract.** Obstacle detection and target tracking are two major issues for intelligent autonomous vehicles. This paper proposes a new scheme to achieve target tracking and real-time obstacle detection of obstacles based on computer vision. ResNet-18 deep learning neural network is utilized for obstacle detection and Yolo-v3 deep learning neural network is employed for real-time target tracking. These two trained models can be deployed on an autonomous vehicle equipped with an NVIDIA Jetson Nano motherboard. The autonomous vehicle moves to avoid obstacles and follow tracked targets by camera. Adjusting the steering and movement of the autonomous vehicle according to the PID algorithm during the movement, therefore, will help the proposed vehicle achieve stable and precise tracking.

## 1 Introduction

The current target tracking technology is applied in various fields, and usually the device moves with the movement of people. Today, autonomous obstacle avoidance and target tracking are mainly based on laser ranging radar. This paper proposes a method based on computer vision to achieve obstacle avoidance and target tracking.

We use an autonomous vehicle equipped with a drive board and a power board, and use the NVIDIA Jetson Nano motherboard to drive the autonomous vehicle to move through programming. We can run the neural network framework on Jetson nano through python language.

We use the Resnet-18 two-classifier model to train the model to complete the identification of obstacles, and use the COCO data set to train the Yolo-v3 model and set the tracking label. After setting the tracking label, we use the PID algorithm to adjust the rotation and driving of the autonomous vehicle, so that the autonomous vehicle can accurately track the target.

---

\* Corresponding author: [caic@sdju.edu.cn](mailto:caic@sdju.edu.cn)

## 2 Implement

### 2.1 Deploy device

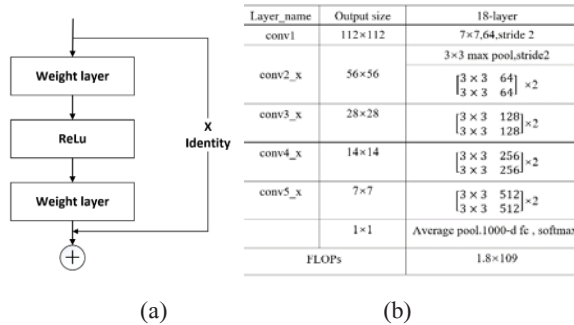
Jetson Nano is installed on the autonomous vehicle equipped with two motors and a battery. Wireless network card is installed so that we can remotely control the movement of the autonomous vehicle through the computer. Then, we can program to control the movement of the autonomous vehicle and deploy the neural network framework.

### 2.2 Avoid obstacles

The traditional autonomous obstacle avoidance is realized by laser ranging radar. This paper chooses the Resnet-18 two classification model. With the help of the jump structure, the information loss in the layer-by-layer feature extraction process of the deep learning model is reduced, and the gradient disappearance and gradient explosion in the deep network training process can be significantly solved. The structure of each residual block is shown in Fig.3. The residual unit can be expressed as:

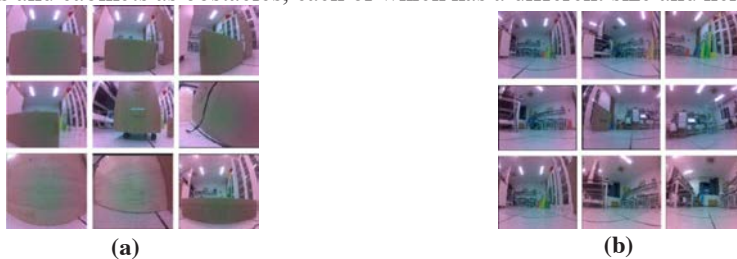
$$x_{i+1} = f(F(x_i, W_i) + h(x_i)) \tag{1}$$

In the formula:  $x_i$ ,  $x_{i+1}$  and  $W_i$  represent the input, output and weight matrix of the  $i$ -th residual unit respectively;  $f(\cdot)$  is the Relu nonlinear activation function;  $h(\cdot)$  is the direct mapping part;  $F(\cdot)$  is the residual mapping part.



**Fig.1.** Residual block and Resnet-18 model.

This experiment uses the Raspberry Pi CSI camera. We initialize the camera on the workstation to collect the required data set. This paper divides the data set into two categories: one is free, which means there are no obstacles, and the other is Block, which means that the car needs to avoid obstacles, and then collect 200 pictures for each category. We use cartons and cabinets as obstacles, each of which has a different size and height.



**Fig.2.** Block dataset and Free dataset.

We use the method of transfer learning and the weights of the pre-trained models that have been learned, and then input a small number of data sets to achieve high accuracy. This paper uses ResNet-18 network for transfer learning. These collected data sets are imported into the model for training, and the trained model is used to predict obstacles.

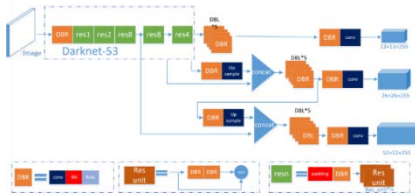
Train Loss	Acc	Val Loss	Acc
0.2505	0.8852	0.1785	0.9412

**Fig. 3.** Best accuracy.

With the help of the real-time image collected by the camera, the input image is predicted, and the threshold value of encountering obstacles is set. If the probability of detecting the obstacle exceeds the threshold value, the autonomous vehicle is set to avoid obstacles and rotate. Then we enable the car to avoid obstacles autonomously.

### 2.3 Target tracking

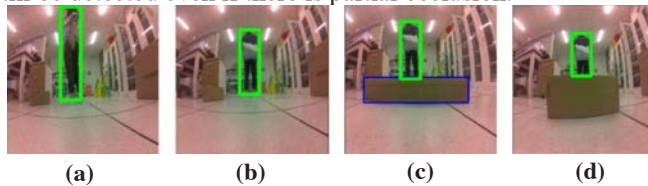
We need to use a target detection algorithm to detect the object that we want to track, and then let our autonomous vehicle track the target. We choose the Yolo-v3 model for target detection. Yolo-v3 is a classic one-stage detection model for deep learning target detection. Yolo-v3 target detection algorithm has the advantages of fast speed and high accuracy, which meet the requirements of detection algorithms. Yolo-v3 uses the Darknet-53 as the network, and uses non-maximum suppression to remove redundant anchor frames. This paper uses COCO data sets to train the model.



**Fig. 4.** YOLOV3 structure.

Set people as tracking targets. Through the camera, we can see the target detected by the autonomous vehicle, and the target object frame becomes green. The autonomous vehicle turns to the target and moves to the target object.

In Fig.6, we can detect the target accurately after setting the target tag for tracking, and the target can still be detected even if there is partial occlusion.



**Fig. 5.** Target detected.

### 2.4 PID algorithm

By calculating the central coordinates of the tracking target and finding the target closest to the centre of the image, we try to make the detected target closest to the centre of the field of view, and then guide the autonomous vehicle to move to the target. In order to keep the target in the centre of the field of view, we set the autonomous vehicle to adjust at a fixed angle. But the robot cannot accurately track the target, and the rotation range is too large,

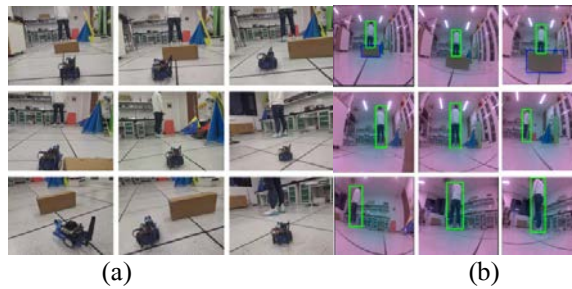
which can easily lead to the loss of the target in the autonomous vehicle's field of view. The robot swings back and forth, and it can't move smoothly to the target. So we introduce the PID algorithm to make the robot track the object more smoothly.

The PID algorithm is proportional, integral, and derivative. We use PID algorithm to calculate the control quantity for control.

$$C = K_p \times er + K_d \times er_{rate} + K_i \times \int_0^T er * dt \quad (2)$$

After loading the PID algorithm, the autonomous vehicles. can track the target stably.

### 3. Result display



**Fig.6.** Result presentation.

We set up an experimental scene and set the tracking target as a moving person, and use cardboard boxes as obstacles. In Fig. 6. we can see the effect of our detection. The robot can avoid obstacles. After bypassing the obstacles, the target is detected and adjusted by the PID algorithm so that the robot can approach the target smoothly.

### 4.Conclusion

In this paper, we use computer vision based for target tracking and autonomous obstacle avoidance. We use Resnet-18 as a two-classifier to avoid obstacles, YOLOV3 as a target detection framework for target tracking, and the PID algorithm to make the robot approach the target smoothly. In the industrial field, as robots follow people's movement, human operations become quite convenient. With the development of computer vision and artificial intelligence, there will be more and more inventions based on computer vision. Artificial intelligence has a bright future.

Supported by Natural Science Foundation of China (Grant No.6207024186)

### References

1. Z. Zhou, "Attention Based Stack ResNet for Citywide Traffic Accident Prediction," 2019 20th IEEE International Conference on Mobile Data Management (MDM), Hong Kong, Hong Kong, 2019, pp. 369-370, doi: 10.1109/MDM.2019.00-27.
2. F. Miao, Y. Tian and L. Jin, "Vehicle Direction Detection Based on YOLOv3," 2019 11th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), Hangzhou, China, 2019, pp. 268-271, doi: 10.1109/IHMSC.2019.10157.
3. A. A. Süzen, B. Duman and B. Şen, "Benchmark Analysis of Jetson TX2, Jetson Nano and Raspberry PI using Deep-CNN," 2020 International Congress on

- Human-Computer Interaction, Optimization and Robotic. Applications (HORA), Ankara, Turkey, 2020, pp. 1-5, doi: 10.1109/HORA49412.2020.9152915.
4. P. Inthanon and S. Mungsing, "Detection of Drowsiness from Facial Images in Real-Time Video Media using Nvidia Jetson Nano," 2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Phuket, Thailand, 2020, pp. 246-249, doi: 10.1109/ECTI-CON49241.2020.9158235.
  5. X. Ou et al., "Moving Object Detection Method via ResNet-18 With Encoder–Decoder Structure in Complex Scenes," in IEEE Access, vol. 7, pp. 108152-108160, 2019, doi: 10.1109/ACCESS.2019.2931922.
  6. "IEE Colloquium on Getting the Best Out of PID in Machine Control (Ref. No.1996/287)," IEE Colloquium on Getting the Best Our of PID in Machine Control, London, UK, 1996, pp. 0\_1-.