Research on contactless control of elevator based on machine vision

Hengcheng Yu[1, a], Zhengyu Chen[2]

[1]Yancheng Institute of Technolog Yancheng, Jiangsu, China

[2]Jinling Institute of Technology Nanjing, Jiangsu, China

[a]984614161@qq.com

Abstract. Aiming at the problem of cross-infection caused by elevator public buttons during the COVID-19 epidemic, a non-contact elevator button control gesture recognition system based on machine vision is designed. In order to improve the detection speed of gesture recognition, combined with the Spatial Pyramid Pooling (SPP) and replaced the Backbone in YOLOv5 with the lightweight model ShuffleNetV2, an improved YOLOv5_shff algorithm was proposed. After testing, in the task of recognizing gestures, the detection speed of the YOLOv5_shff algorithm is 14% higher than the original model, and the detection accuracy is 0.1% higher than the original model. Taking the improved YOLOv5_shff algorithm as the core, a gesture recognition system that can be applied to elevator button control is designed. The experimental data shows that the gesture recognition accuracy for controlling elevator buttons reaches 99.3%, which can meet the requirements of non-contact control of public elevators.

Keywords: Gesture recognition, Machine vision, Non-contact elevator button, YOLOv5, COVID-19.

1. Introduction

The COVID-19 is raging around the world and avoiding frequent contact with public facilities is an effective way to prevent the spread of the virus.

In the field of human interaction, gestures have many application scenarios, such as non-contact control, smart home, real-time sign language translation[1]. In daily life, people hope to adopt non-contact human-computer interaction with certain public facilities that cannot be avoided, such as buttons in public elevators.[2, 3] Non-contact human-computer interaction generally refers to the information interaction between people and devices through voice recognition, virtual buttons and gesture detection. However, voice recognition cannot avoid the risk of droplet transmission. [4]Virtual buttons often need to be equipped with electronic screens, which are expensive high. [5]The non-contact public elevator key control solution proposed in this paper realizes non-contact human-computer interaction by specifying corresponding logical operations for different gestures, and only needs to configure a monocular camera to realize gesture cloud detection. The scheme has the advantages of low investment cost, fast detection speed and strong robustness.

2. Related research

Traditional gesture recognition is generally based on sensors and requires human-computer interaction through wearable devices. Liu et al. [6] proposed a wearable body tracking device based on micro-flow sensors and micro-accelerometers to detect the posture of human limbs in motion engineering. Wong et al. [7] proposed an effective and

low-cost capacitive sensor device for gesture recognition and designed a wearable capacitive sensing unit. However, the above-mentioned sensor-based gesture recognition will bring discomfort to the user and be expensive. With the successful application of convolutional neural networks [8] in object detection, classification, researchers have shifted the research focus of gesture recognition to the field of computer vision.

Mao et al. [9] proposed a lightweight object detection method based on YOLOv3, using deep separable convolution for its backbone network Darknet-53, and suppressing the boundary effect of

its convolution kernel through dimensional transformation for its remaining network. The detection speed is reduced by half. Wu et al. [10] combined the channel pruning algorithm on the basis of YOLOv4 to greatly reduce the model size and the number of parameters and improve the detection speed. Some scholars have also done further research on the target detection algorithm. Wang Kun et al. [11, 12] used the MobileNet model to decompose the basic neural network layer of the SSD algorithm to reduce the size of the model. However, when multiple people detect, the detection accuracy is slightly lower than the original model, and the accuracy rate has decreased.

The main work of this paper is as follows: First, replace the Backbone of YOLOv5 with the lightweight model ShuffleNetv2; Second, add the Spatial Pyramid Pooling (SPP) module to enhance the accurate positioning ability of the network and improve the detection speed while maintaining the detection accuracy. Then, on the public dataset handpose_x_gesture_v1[13] and the self-built dataset named hand_num, compared with the original YOLOv5 algorithm, the mean average precision (mAP@0.5) of the improved algorithm is 99.3%, which can improve the speed and accuracy of gesture multi-scale and rotational occlusion detection. Finally, Experimental results show that the YOLOv5_shff algorithm can control the elevator quickly and accurately in a non-contact manner.

3. System model

3.1 Improved YOLOv5 model

YOLOv5 is a more advanced target detection and recognition algorithm. Based on YOLOv4, it draws on the idea of CSPNet, uses the improved CSPNet as the backbone network, and predicts images at multiple scales to improve prediction precision. At the same time, it uses the native architecture of Pytorch, making the network scale smaller than YOLOv4. In this paper, the original algorithm is modified: The Backbone of YOLOv5 is replaced with the lightweight model ShuffleNetV2; The Spatial Pyramid Pooling (SPP) module is added to improve the detection speed while maintaining the detection accuracy. As shown in Figure 1 Backbone's yellow and blue background modules.

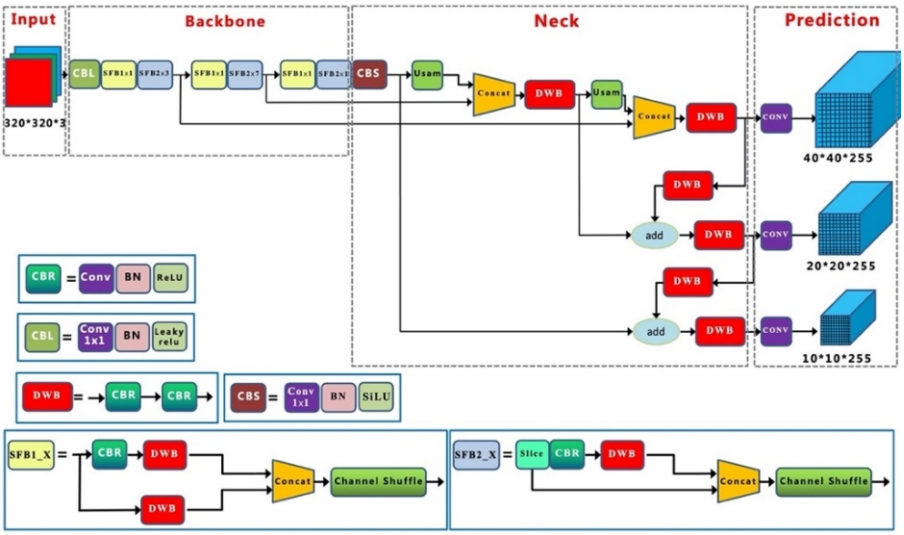


Figure 1. Improving network structure of YOLOv5

3.2 ShuffleNetV2 network

Ma et al. [11] proposed 4 efficient and lightweight network criteria in ShuffleNetV2, keeping the large number of channels and the number of channels unchanged, that is not using too much convolution and not too many groups, by this proposes a new block design based on ShuffleNetV1, as shown in Figure 2, the two basic unit modules of ShuffleNetV2.

In the ShuffleNet V2 network, the Memory Access Cost (MAC) is the smallest when the convolution input and output channels are equal, and the grouped convolution reduces the computational complexity through sparse connections between channels, but the convolution with too many groups will increase the MAC. Therefore, in order to make the model more efficient, the key is to maintain equal-width channels and not use dense convolution operations and too many group convolutions. As shown in Figure 2, the ShuffleNetV2 network is divided into two groups by channel division operations, using 1×1 Convolution replaces point-by-point group convolution, and the number of input and output channels before and after convolution in the branch remains unchanged. After the branches are spliced, channel shuffling is used to enhance the information exchange between branches.

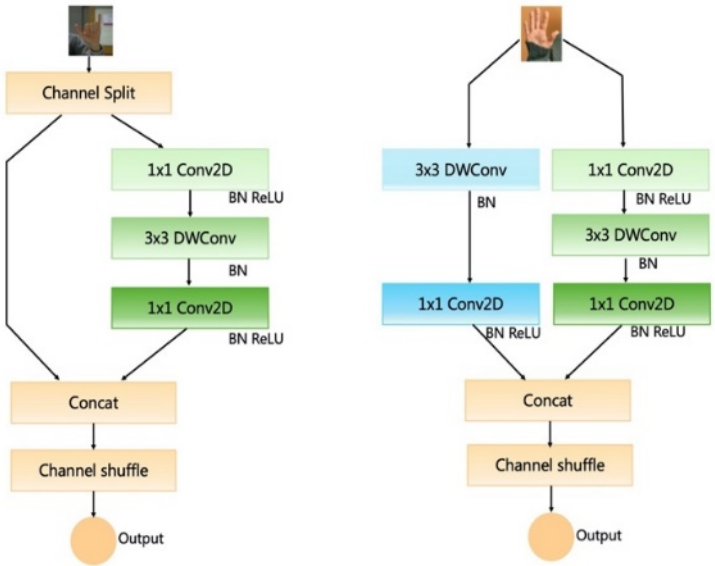


Figure 2. ShuffleNet V2 Unit.

4. Experiments and Results

The experiment is done on a PC, the main configuration of the PC is Windows 10 operating system, equipped with NVIDIA RTX-3060Ti graphics card. The implementation framework is Pytorch and uses OpenCV for image display.

4.1 Experimental data

The experimental data comes from the public dataset handpose_x_gesture_v1 and the hand_num dataset. Table 1 gives the hyperparameter settings of the lightweight YOLOv5 model. Before the formal training of this paper, three generations of warm-up learning were first carried out, in which the warm-up learning momentum was 0.8, and the warm-up initial learning rate was 0.1. The purpose was to make the model gradually stabilize after warm-up learning, and then perform formal gesture recognition. The effect is better, and the rest of the hyperparameter settings are shown in Table 1.

4.2 Experimental evaluation index

The purpose of the method in this paper is to reduce the memory ratio of the model while ensuring the accuracy and speed of the model. While evaluating the test performance of the model through mAP (mean average precision) and FPS (frames per second), the training time of the model is also considered and model memory ratio. The specific expression is as follows:

$$P = \frac{TP}{TP+FP}, \tag{1}$$

$$R = \frac{TP}{TP+FN}, \tag{2}$$

$$\mathrm{AP} = \int_0^1 P(R)dR, \tag{3}$$

$$mAP = \frac{\sum_{q=1}^{Q} AP(q)}{Q}. \quad (4)$$

P is the precision, the ratio of the number of positive samples that are correctly predicted to the number of positive samples that are predicted to be positive. R is the recall, the ratio of the number of correctly predicted positive samples to

Table 1. Hyperparameter settings.

hyperparameters	lr0	lrf	weight_decay	warmup_epochs	warmup_momentum
scratch	0.01	0.2	0.0005	3.0	0.8
tune	0.0032	0.12	0.00036	2.0	0.5
finetune	0.012	0.15	0.0005	3.0	0.8

the number of all positive samples. AP is the recognition accuracy of each class, and mAP is the recognition accuracy of all classes in the sample.

4.3 Gesture recognition algorithm process design

The hardware required for this algorithm is simple, only a computer, a camera and a corresponding connecting line are required, and the tester only needs to point his hand at the camera to make corresponding gestures. The overall design of the test algorithm is shown in Figure 3, and its core part is designed based on the improved YOLOv5. This algorithm is divided into one-handed mode and two-handed mode. The one-handed mode can identify the elevator's ascent, descent, door opening, door closing, and gestures leading to floors 1 to 9; in the two-hand mode detection, the number detected by the left hand represents ten digits, and the right hand the detected number represents the one-digit number, and when both hands are used at the same time, you can make a two-digit gesture and issue a command to reach the 10th floor or higher.

4.4 Model Results and Analysis

During training, mAP is calculated on the test set every 4 epochs. Figure 4 shows the mAP curve and loss curve changes of the gesture detection and recognition model during training. At the same time, the YOLOv5 and YOLOv5_shff training curves are compared. It can be seen from Figure 4 that in the initial stage of model training, the model learning efficiency is high and the training curve convergence speed is fast. As the training progresses, the slope of the training curve gradually decreases. Finally, when the number of training epoch reaches about 260, the model learning efficiency gradually reaches saturation and the loss is around 0.006. The mAP basically reached 99.3% at the iteration number of 280 and remained stable.

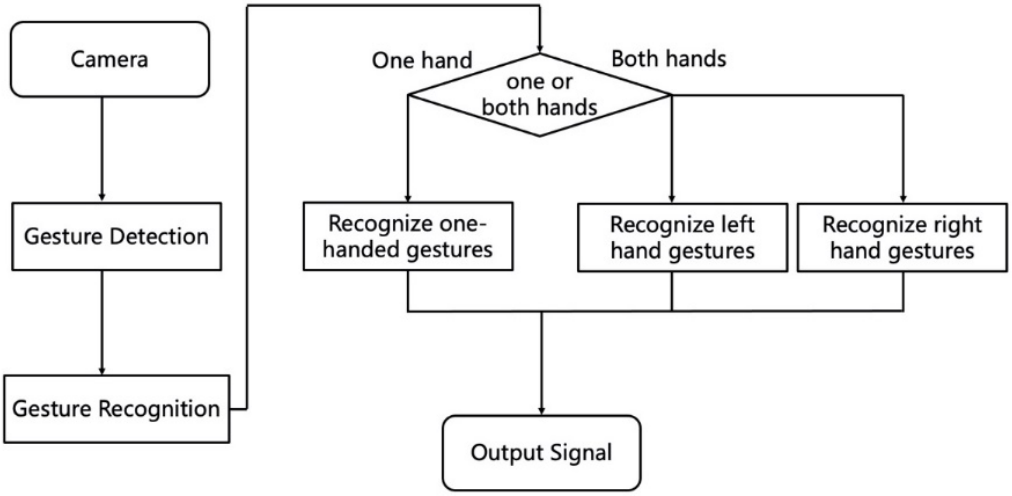
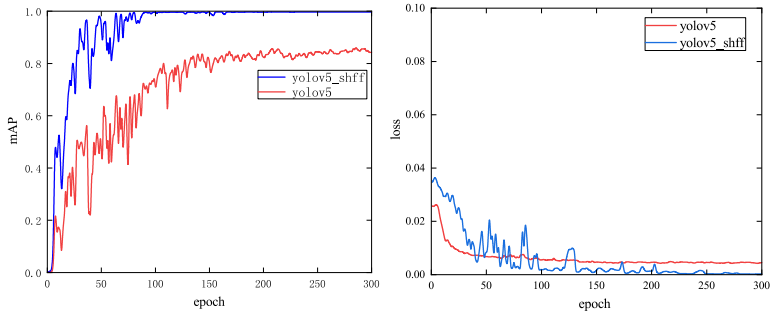


Figure 3. System flow chart.

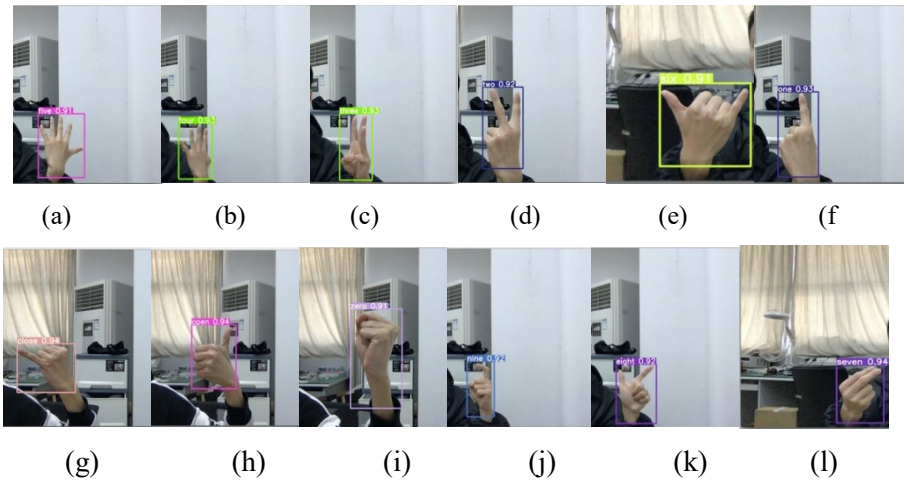
The mAP and model loss performance of testing during training are shown in (a) and (b) in Figure 4.



(a) Map (b) model loss

Figure 4. mAP and loss curve of gesture detection model

The gesture recognition diagram detected based on the YOLOv5_shff algorithm is shown in Figure 5. (a)~(i) represent the gestures of numbers 1~9, respectively



(a) (b) (c) (d) (e) (f

(g) (h) (i) (j) (k) (l)

Figure 5. Gesture Recognition Diagram.

(j) represents the gesture of number 0, (k) represents the rising gesture or Door opening gesture, (l) means descending gesture or door closing gesture, when used inside the elevator, because up and down buttons are not required, (k) (l) indicate door opening and closing gestures, when used outside the elevator, (k) (l) for up and down gestures. Based on the above gestures, the non-contact control of the elevator can be basically completed. For example, if the user wants to take the elevator to go up to the eighth floor, he only needs to make the gestures shown in (k) and (h) in sequence. If the user needs to reach an even-numbered floor, he needs to make a gesture corresponding to the floor number with both hands.

5. In conclusion

This paper optimizes the YOLOv5 model and proposes the YOLOv5_shff algorithm to improve the detection speed. The main work is summarized as follows: First, replace the Backbone of YOLOv5 with the lightweight model ShuffleNetV2; Second, add the Spatial Pyramid Pooling (SPP) module to enhance the accurate positioning ability of the network and improve the detection speed while maintaining detection accuracy.

The application of the proposed algorithm to the elevator control system can effectively avoid the risk of cross-infection in public facilities during the COVID-19 epidemic. The self-built hand_num gesture dataset can enrich the existing dataset, allowing gestures to be applied in more human-computer interaction scenarios.

Acknowledgment

This work was supported by Postgraduate Research and Practice Innovation Project of Yancheng Institute of Technology. The project number is 1142249(SJCX21_XY025).

References

[1] Bose S R, Kumar V S. Efficient inception V2 based deep convolutional neural network for real-time hand action recognition [J]. IET Image Processing, 2020, 14(4): 688-696.

[2] Lu Z, Qin S, Li L, et al. One-shot learning hand gesture recognition based on lightweight 3D convolutional neural networks for portable applications on mobile systems [J]. IEEE Access, 2019, 7: 131732-131748.

[3] Oudah M, Al-Naji A, Chahl J. Hand gesture recognition based on computer vision: a review of techniques[J]. journal of Imaging, 2020, 6(8): 73.

[4] Sun Y, Armengol-Urpi A, Kantareddy S N R, et al. Magichand: Interact with iot devices in augmented reality environment[C]//2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR). IEEE, 2019: 1738-1743.

[5] Katti J, Kulkarni A, Pachange A, et al. Contactless Elevator Based on Hand Gestures During Covid 19 Like Pandemics[C]//2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS). IEEE, 2021, 1: 672-676.

[6] Liu S Q, Zhang J C, Zhu R. A wearable human motion tracking device using micro flow sensor incorporating a micro accelerometer [J]. IEEE Transactions on Biomedical Engineering, 2019, 67(4): 940-948.

[7] Wong W, Juwono F H, Khoo B T T. Multi-features capacitive hand gesture recognition sensor: a machine learning approach [J]. IEEE Sensors Journal, 2021, 21(6): 8441-8450.

[8] Valueva M V, Nagornov N, Lyakhov P A, et al. Application of the residue number system to reduce hardware costs of the convolutional neural network implementation [J]. Mathematics and Computers in Simulation, 2020, 177: 232-243.

[9] Mao Q-C, Sun H-M, Liu Y-B, et al. Mini-YOLOv3: real-time object detector for embedded applications [J]. Ieee Access, 2019, 7: 133529-133538.

[10] Wu D, Lv S, Jiang M, et al. Using channel pruning-based YOLO v4 deep learning algorithm for the real-time and accurate detection of apple flowers in natural environments [J]. Computers and Electronics in Agriculture, 2020, 178: 105742.

[11] Ma N, Zhang X, Zheng H T, et al. Shufflenet v2: Practical guidelines for efficient cnn architecture design[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 116-131.

[12] Howard A G, Zhu M, Chen B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications [J]. arXiv preprint arXiv:17040.4861, 2017, CVPR 2017.

[13] Fronteddu G, Porcu S, Floris A, et al. A Dynamic Hand Gesture Recognition Dataset for Human-Computer Interfaces [J]. Computer Networks, 2022: 108781.