Gesture recognition real-time control system based on YOLOV4

Gesture recognition real-time control system based on YOLOV4

Zhuowen Zheng*

College of Communication Engineering, Hangzhou Dianzi University, Hang Zhou, 310000, China

*Corresponding author's e-mail: zhengzhuowen@hdu.edu.cn

Abstract. With the development of industrial information technology in recent years, gesture control has attracted wide attention from scholars. Various gesture control methods have emerged, such as visual control, wearable device control, magnetic field feature extraction control. Based on one of the visual gesture control methods, this paper proposes a visual gesture control method applied to music box control by combining YOLOv4 object detection network. We design seven main gestures, reconstruct gesture datasets, and retrain the YOLOv4 object detection network by the means of the self-built datasets, further build a music box gesture control system. In this paper, we obtain the recognition accuracy of 97.8% for the object detection network in the gesture control system after a series of experiments, and recruit eight volunteers to conduct experimental tests on the self-built gesture-controlled music box system, mainly to quantify the time of executing a single command, attention concentration, etc. The results show that compared with the traditional control method, the visual gesture control method ensures the accuracy while has a faster response speed and takes up less of the user's attention.

1. Introduction

With the development of modern information technology, various new electronic control modes have been widely used. The mainstream control modes are controller control, speech recognition control and visual recognition control. These control modes rely on the senses of the controller and are an extension of his or her consciousness.

Along with the development of computer arithmetic and artificial intelligence technology, humans are gradually getting tired of controllers in daily life, and button control is no longer the direction of mainstream control system development attributed to its own property defects and limitations.Meanwhile, gesture control based on computer vision is receiving more and more attention. In contrast to traditional control methods, this control method does not require a medium and is more convenient, practical and natural because of the intuitive nature of gestures. At present, there are three main solutions for gesture control systems: the use of wearable devices such as data gloves, which often require specialized data collection devices and are not widely available due to their high price despite providing the best accuracy and speed; the use of 3D gesture recognition (ultrasound, magnetic field) with depth-informed pictures,which requires additional feature extraction of key parts of the hand compared to the first method,it will bring extra recognition time and compute loss as well;the use of 2D gesture recognition with original visual photographs, which has good real-time performance, accuracy, wide application scenarios, and low hardware requirements (often only an optical lens is required). This paper will combine the object detection network with a camera to realize gesture

recognition control of local music box.

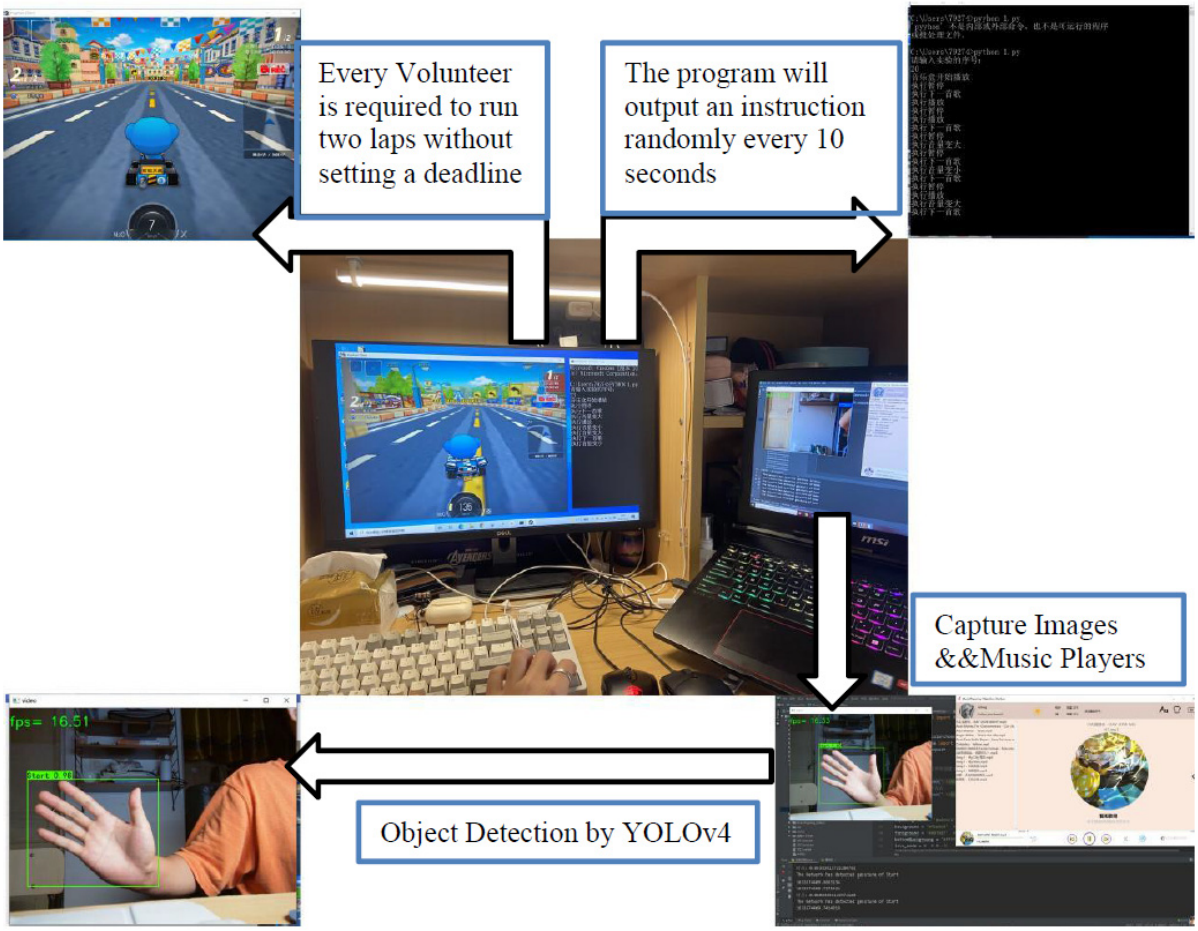

Figure1. Schematic diagram of a music box system based on visual gesture control

The main contributions of this paper are: i) the design of gesture datasets for the control of basic functions of the music box; ii) a certain degree of localization of the neural network based on the YOLOv4 network; iii) the construction of music playback management system based on a gesture recognition.

2. Related Work

In the past decade, natural image object detection algorithms have been based on traditional handcrafted features. With the rise of CNN[1], object detection has entered a deep learning stage since 2013. During this period, deep learning object detection algorithms are mainly divided into two categories: Two stage and One stage.

Two Stage:firstly create a region called region proposal (RP, a pre-selected box that may contain the object to be examined), and then classify the samples by convolutional neural network. Tow stage object detection algorithms commonly used include R-CNN[2], Fast R-CNN[3], Faster R-CNN[4] and R-FCN[5] and so on. For the two main tasks of object detection - object classification and localization -

R-CNN draws on the idea of sliding windows to recognize the areas. But because RP is created through rather slow selective search arithmetic and repetitive convolution operation，R-CNN is very slow and occupy large memory.Fast R-CNN improves this problem by adding ROI pooling and a multi-task loss function. Instead of selective search, Faster R-CNN directly creates the region to be detected by a Region Proposal Network (RPN), which reduces the object detection time from 2s to 10ms.

One stage: predict object classification and location by extracting features directly in the network without using RP. Common one-stage object detection algorithms include: YOLOv1[6], YOLOv2[7], YOLOv3[8], YOLOv4[9], SSD[10], etc. To address the common drawback of slow computing speed of two-stage object detection algorithms, YOLO creatively proposes one-stage, which classifies and localizes objects in one step. SSD balances the strengths and weaknesses of YOLO and Faster RCNN by using dense sampling, multi-size feature map, multiple anchor boxes, and NMS filtering.

Object detection algorithms are often used in gesture recognition as the backbone. Gesture recognition techniques can be roughly classified into three levels: 2D hand recognition, 2D gesture recognition, and 3D gesture recognition. The first two gesture recognition technologies are based entirely on the 2D layer, and they only require 2D information without depth information as input. Instead, the third type of gesture recognition technology is based on a 3D layer. The fundamental difference between 3D gesture recognition and 2D gesture recognition is that 3D gesture recognition requires an input containing depth information, which makes 3D gesture recognition much more complex than 2D gesture recognition in terms of both hardware and software.

Among them, vision-based 2D real-time gesture recognition is widely used in daily life, and deep learning gesture recognition often requires a high classification accuracy, fast response time, and low resource consumption of the network. Real-time gesture recognition systems require simultaneous detection and classification of continuous video streams.

3. System design

3.1. Gesture setup

3.1.1. Introduction of Gesture datasets

Gestures are the specific movements and positions that occur when people use their arms. It is one of the first communication tools used by humans and is still used in a number of countries. Because of its intuitiveness and simplicity, under conventional circumstances,gestures can convey messages most directly to different language users, different races, and different ages of people, and they can also help convey messages to people who are visually or hearing impaired or deaf and dumb. Therefore, a series of gesture-based command datasets have been designed to be universally applicable to all groups of people without distinction.

The following seven gestures are selected as the set of gesture commands for controlling the sound box by investigating various gestures. The reason is that these gestures are very common in daily life, easy to remember, and have a high degree of differentiation between each other, so that each finger can be fully deployed, which helps the training process of the object detection network and can yield better training models and results.

Figure2. Diagram of gesture datasets

3.1.2 The process of collecting the Gesture datasets

This paper captures the seven gestures described above in a complex multi-textured indoor environment with the iPhone 11's own optical camera. The average shooting time of each gesture is 120 seconds. For the captured video, image processing is carried out, and the frame extraction of the gesture set video is conducted at the interval of 0.3 seconds, that is, three frames of images can be extracted from the video per second, ensuring the diversity of the gesture set. Each gesture contains

350 images, which covers seven gestures in a multi-texture background, and the whole gesture set contains a total of 2450 diverse photos. For the obtained raw gesture data, the photographs are annotated by the labeling software using VOC format. In order to ensure the training results of the network, gesture datasets is divided into training set, verification set and test set at 7:2:1, and gesture datasets is then rotated, symmetrized, and scaled to achieve the effect of data augmentation,and then all the augmented photographs are set to a size of 416X416 and input into the network.

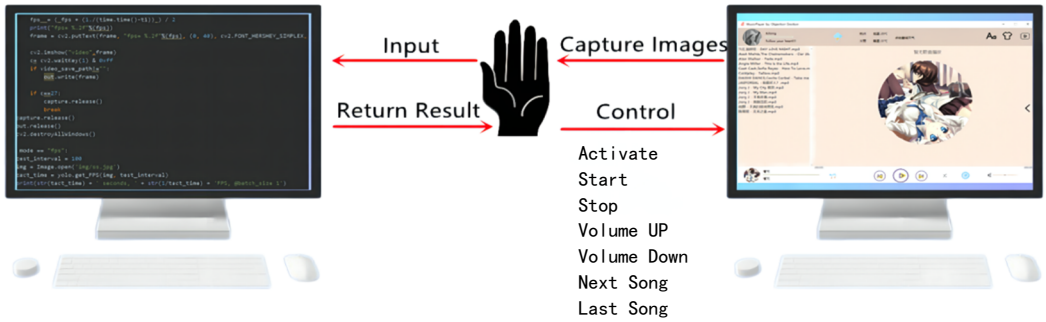

Figure3.Schematic diagram of the gesture recognition system

3.2. Hardware System

This paper uses a host computer equipped with a NAVID GTX1070 GPU and an i7-8750H as the core processing platform, and the OPEN-CV SDK is used to adjust the external camera to capture photographs of users' hands in real time for control. The captured hand pictures are image processed, resized to 416X416, and sent to a YOLOv4 object detection network with resenet-50[11] as the backbone , then the network extracts features,classify the input data, predict results, and finally outputs the instruction meaning contained in the gesture and the corresponding score value.

Table 1. Details about the hardware system

Network Framework	Pytorch 1.6.0
Compiler software	Pycharm
Operating system	Window10
GPU	GTX 1070
CPU	i7-8750H
Camera	HIKVISION E11

3.3. Application

This paper designs a music player based on gesture control. Through the investigation and comparison of music players in the market, it is found that most music players have following seven functions: software start, play, stop, volume up, volume down, play next song and play last song. Based on the above basic functions, the seven results of gesture recognition correspond to the seven control commands of the local music box, and the results of the network are transmitted to the music control

terminal. The whole music box gesture control system is divided into three modules: acquisition module, gesture recognition and analysis module and application control module. The acquisition module mainly captures real-time images of the user's environment and hand posture through an external camera, and transmit the captured images to the gesture recognition and analysis module after image processing and size change operations. Gesture recognition and analysis module is mainly composed of a gesture recognition network based on YOLOv4 object detection network and its corresponding components,In this module,after feature extraction, feature classification and prediction, the image outputs user gesture meanings, and transmits relevant meanings to the application control module to form control instructions, which can control the music box and its image interface in real time, realizing stable music box control.This method of controlling the music box by gesture reduces the user's attention consumption when controlling the music box, especially in dangerous events requiring highly attention concentration such as driving a car or cutting vegetables, thus ensuring the user's safety to a certain extent.

Figure4. Gesture recognition output control instruction process

3.4. Experimental design

The experiment recruits 8 volunteers, including 4 men and 4 women. Considering young people's strong ability to accept new things and fast response, the age range of volunteers to participate in the experiment is 20-23 years old. The experiment was divided into two modules. Volunteers were asked to run around a fixed map in a car driving simulation game on a computer --- to simulate driving in real life --- while controlling the car's progress, they were asked to control a local music box based on instructions that randomly popped up on the screen. Each instruction appears at ten seconds intervals (to reflect the randomness and discontinuity of the instruction). In module 1, volunteers will use the object detection gesture recognition system trained in this paper to control the local music box. In module 2, volunteers will control the local music box directly using the mouse. The recording parameters of this experiment are gender, age, the way of controlling the music box, the average time spent executing a single command, the number of car collisions in the experiment, and the time spent finishing the game. The efficiency of the two control modes is compared from the micro and macro perspectives respectively. The number of car collisions in the experiment is to quantify the occupation of driver attention by the two control modes.

After the experiment, volunteers will receive a questionnaire and give their answers on a scale of 1-7, with 1 being the minimum and 7 being the maximum. Specific questions in the questionnaire are as follows:

- The degree to which the two control options affect your opening process
- Which option do you prefer?
- If gesture recognition controls are to be applied to automobiles in the future, would you be willing to use them?

4. Experiment And Result

4.1. Experiment Method

This paper uses 245 original photographs and their labels as the test set, and the set is tested under the

mAP0.5 and mAP0.75 standards. 7 categories are used in the test set, namely, Activate, Start, Stop, Up, Down, Next Song, and Last Song. This seven categories will be presented below in categories 1-7 in order:

Table2. Self-trained object detection network performance in test sets

(a). Under the mAP0.5 standard

Standardized	Categories	AP	F1	Precision	Recall
mAP0.5	1	99.40%	98%	95.48%	100%
mAP0.5	2	100%	100%	99.35%	99.10%
mAP0.5	3	99.10%	100%	100%	100%
mAP0.5	4	100%	100%	100%	100%
mAP0.5	5	100%	100%	100%	100%
mAP0.5	6	100%	99%	98.03%	100%
mAP0.5	7	99.33%	100%	100%	99.33%
mAP0.5	99.70%				

(b). Under the mAP0.75 standard

Standardized	Categories	AP	F1	Precision	Recall
mAP0.75	1	97.35%	96%	93.55%	97.97%
mAP0.75	2	97.32%	98%	97.39%	98.03%
mAP0.75	3	98.20%	99%	99.09%	98.20%
mAP0.75	4	99.33%	99%	99.33%	99.33%
mAP0.75	5	100%	100%	100%	100%
mAP0.75	6	99.33%	98%	97.37%	99.33%
mAP0.75	7	96.93%	98%	97.99%	97.33%
mAP0.75	98.35%				

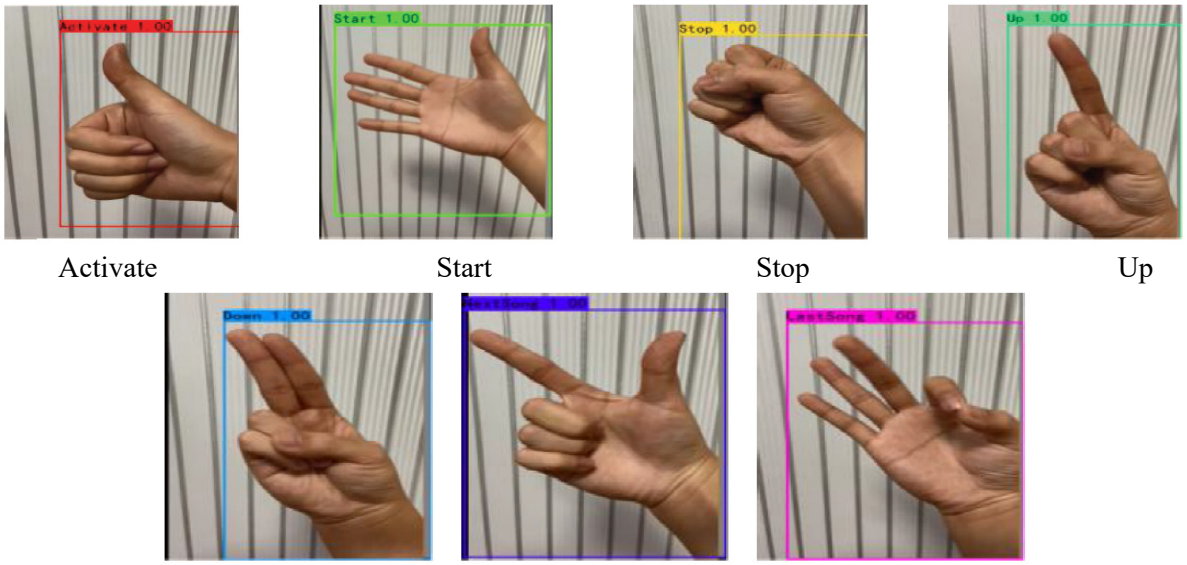

Figure5. Detection result of gesture datasets

4.2 Experiment Result

The HIKVISION E11 camera captures the image with the size of 1980X1080 in real time. After image preprocessing and adjusting the size to 416X416, the image is sent to the trained object detection

network.The average detection time per image is 0.055s. For the whole gesture recognition &music box system, in the real-time detection mode of the network,the maximum FPS value can reach 18, the minimum FPS value is 16.5, and the average FPS is 17, the accuracy of gesture recognition is 97.8%.
In order to avoid the gesture error caused by the network being too sensitive, the network adopts the anti-shake algorithm: for each photo continuously input into the network, the object detection network will give the corresponding object detection score, and when the score is less than 0.55, the network will discard the photo. The gesture recognition music box system takes 15 consecutive frames of pictures as the judgment basis of a single instruction. When 80% of the detection results of pictures are the same gesture, the instruction of the gesture will be executed.

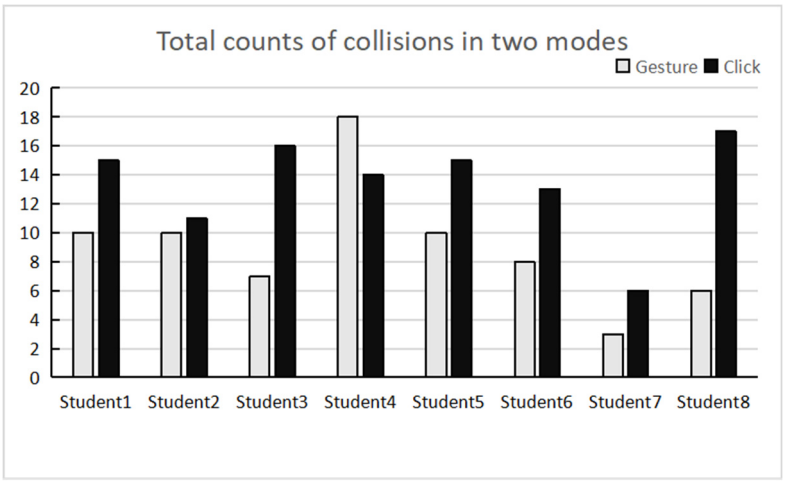

Figure6: Time taken to complete a single experiment in both control modes

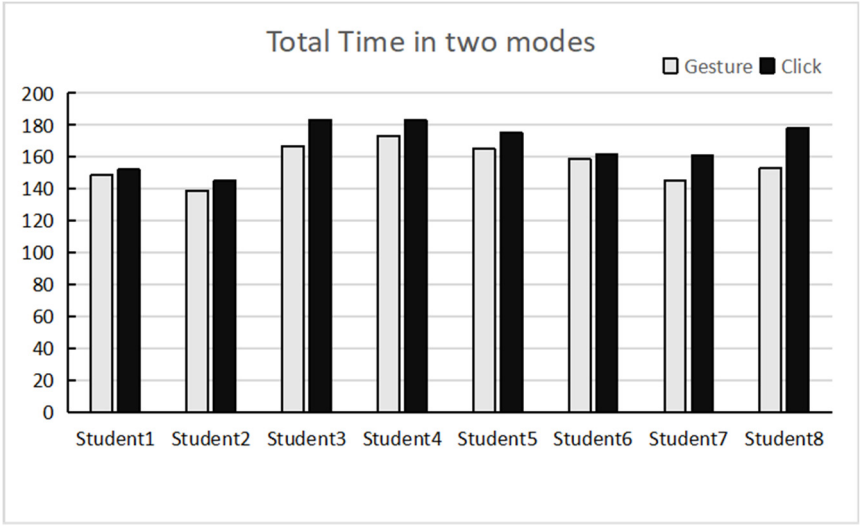

Figure7: The number of collisions between the two control modes in a single experiment

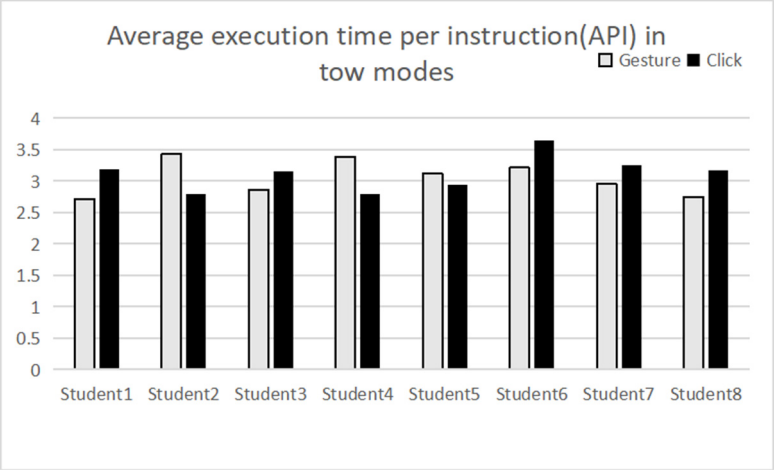

Figure8: Average single instruction execution time in two control modes

Table3. Volunteer questionnaire results

NO.	Gender	degree of sanctification with Gesture Control	degree of sanctification with Click Control	Modes preference	Willing to adapt gesture control on automobile
1	Male	5	4	Gesture	4
2	Male	5	3	Gesture	5
3	Female	4	2	Gesture	6
4	Female	4	3	Click	4
5	Female	5	2	Gesture	5
6	Female	5	2	Gesture	5
7	Male	7	4	Gesture	6
8	Male	5	3	Gesture	6

According to data statistics, the average number of collisions of all experimental volunteers in a single experiment was 11.19, among which the average number of collisions in gesture control mode was 9 with a variance of 19.14, and the average number of collisions in mouse click control mode was 13.38 with a variance of 12.27. The average completion time of all experimental volunteers in a single experiment was 161.81s, in which the average completion time in gesture control mode was 156.25s with a variance of 138.21, and the average completion time in mouse click control mode was 167.38s with a variance of 209.31. The average response time of all volunteers to a single command in a single experiment was 3.59s, in which the average response time to a single command in gesture control mode was 3.06s, and the variance was 0.076. The average response time of single instruction in mouse click control mode is 4.18s, and the variance is 0.0783.

After the questionnaire survey, the participants were evaluated from 1 to 7. Because the mouse click control requires frequent change of perspective for instruction execution and sight loss, the volunteers gave a satisfaction score of 2.875 for the mouse control mode. The volunteers gave the gesture control a satisfaction rating of 5.0 for ease of use and no need to look away. 87.5% of volunteers thought gesture control was better than mouse click control; The volunteers gave an expectation of 5.125 for whether they expected the technology to be used in driving in the future.

In conclusion, on the premise of significantly improving the average response time of a single command (26.8%), the gesture control mode significantly reduced the number of collisions in a single experiment by 32.7%, and the variance of average number of collisions in the gesture control mode was larger than that in the mouse click control mode. This may be due to the difference in the speed at

which each volunteer accepted gesture control and adopted it. In addition, gesture control mode significantly reduced the total time of a single experiment from 167.38s to 156.25s, reducing by 6.6%.

5. Conclusion

The experiments in this paper show that the local music box control system based on gesture control can reduce the number of collisions during driving, increase the operation speed, reduce the occupation of attention, and improve safety and reliability while speeding up the command execution when the user is driving a simulated car.

The experiment of simulating driving a car is to mimic the event that requires highly attention concentration. The application of gesture-controlled operating system can be very wide, with dangerous properties of the assembly line production line, high-speed car driving, people's daily life home control, etc. The operating system of gesture control can facilitate the use of the controller, reduce the occupation of attention and ensure the safety during the control on the premise of ensuring the control timeliness and control accuracy. For these specific occasions, systematic customization can reduce the loss caused by artificial inattention in the future. For example, if it is applied to the driving platform, it is expected to reduce the frequency of traffic accidents.

Of course, some problems are encountered during the experiments: gesture Up and gesture Down are easily confused, which is attributed to the fact that the only difference between the two gestures is that the former has the index finger pointing upward and the latter has the index and middle finger pointing upward, and the network does not capture the gap between the latter's index and middle finger and mistakenly sees the two fingers as one finger, which leads to the appearance of misinterpreting gesture Down as gesture Up. To address this problem, we can take the approach of optimizing the datasets by taking photos of gesture Up and gesture Down at several different angles to increase the spatial information of the datasets, which allows the object detection network to learn more gesture features.

References

[1] Technicolor T, Related S, Technicolor T, et al. ImageNet Classification with Deep Convolutional Neural Networks [50].
[2] Girshick R, Donahue J, Darrell T, et al. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation[C]// IEEE Computer Society. IEEE Computer Society, 2013.
[3] Girshick R. Fast R-CNN[J]. Computer Science, 2015.
[4] Ren S, He K, Girshick R, et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017, 39(6):1137-1149.
[5] Dai J, Li Y, He K, et al. R-FCN: Object Detection via Region-based Fully Convolutional Networks[J]. Curran Associates Inc. 2016.
[6] Redmon J, Divvala S, Girshick R, et al. You Only Look Once: Unified, Real-Time Object Detection[J]. Computer Vision & Pattern Recognition, 2016.
[7] Redmon J, Farhadi A. YOLO9000: Better, Faster, Stronger[C]// IEEE Conference on Computer Vision & Pattern Recognition. IEEE, 2017:6517-6525.
[8] Redmon J, Farhadi A. YOLOv3: An Incremental Improvement[J]. arXiv e-prints, 2018.
[9] Bochkovskiy A, Wang C Y, Liao H. YOLOv4: Optimal Speed and Accuracy of Object Detection[J]. 2020.
[10] Liu W, Anguelov D, Erhan D, et al. SSD: Single Shot MultiBox Detector[J]. Springer, Cham, 2016.
[11] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[J]. IEEE, 2016.