

Intelligent In-Vehicle Safety System Based on Yolov5

Biyun Chen, Wei Cai, Jinwen Zhu

College of Computer Information and Engineering, Yancheng Teachers University, Yancheng, China

Email: chenby@yctu.edu.cn

How to cite this paper: Chen, B.Y., Cai, W. and Zhu, J.W. (2024) Intelligent In-Vehicle Safety System Based on Yolov5. *Journal of Computer and Communications*, 12, 207-218.

<https://doi.org/10.4236/jcc.2024.123013>

Received: February 7, 2024

Accepted: March 25, 2024

Published: March 28, 2024

Copyright © 2024 by author(s) and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

In order to reduce the occurrence of traffic accidents and assist drivers to avoid dangerous driving. This paper presents a smart in-vehicle safety system that utilises the Yolov5 algorithm. Yolov5 algorithm is used to anticipate driver fatigue and distraction behaviours, and remind drivers to pay attention to safe driving in time. The system continuously splits the frames and analyses the frame content through the video feedback from the front camera, compared to the traditional machine learning, Yolov5's mosaic data is enhanced, resulting in a batch size enhancement of 92.3%, and it also uses the Drop Block mechanism to prevent overfitting. The hardware of this system uses STM32 microcontroller and uses system DMA interrupt control and buzzer alarm device to warn about dangerous driving behaviour.

Keywords

Behaviour Detection, STM32, Pyside2, Yolov5, Dlib Open Source Library, Perclos Model

1. Introduction

The China Logistics Big Data Truck Driver Report points out that one of the main causes of accidents during cargo transport is truck driver fatigue. In addition to fatigue driving, another important cause of traffic accidents is distracted driving [1]; take looking at mobile phones while driving as an example: it takes a minimum of 3 seconds to look at a mobile phone while driving, and if you are driving at a speed of 60 km/h, you will drive 50 metres in 3 seconds, and this 50 metres is completely blind driving. According to data, the probability of an accident occurring while looking at a mobile phone is about 23 times that of an accident occurring under normal driving conditions, the probability of an accident occurring while opening a mobile phone is 2.8 times that of an accident occur-

ring under normal conditions [2], and the reaction time for texting while driving is also about 35% slower than the normal reaction time [3]. Distracted driving is no less harmful than fatigued driving. Some data show that distracted driving not only slows down the driver's perceptual ability, resulting in a delay or even a complete failure to receive important traffic information, but also causes a delay in decision-making, resulting in a delay in taking normal actions or adopting incorrect steering, throttle, and braking manoeuvres due to panic. At the same time, the act of driving without a seatbelt is also a major cause of car accidents. The protective role of seat belts is mainly to prevent the driver and passengers from being thrown out of the vehicle in the event of danger, and to prevent the driver and passengers from violently hitting the steering wheel, dashboard, and windscreen in the event of a car accident [4]. There are statistics that in a frontal collision, if the seat belt is fastened correctly, then the probability of survival of the passengers in the car can be increased to 60%, of which, if a frontal collision occurs, fastening the seat belt can reduce the lethality rate by about half, if a side collision occurs, it can make the mortality rate reduced to 56%, and if a rollover occurs, it can reduce the mortality rate by 80%. In the most recent study, unbelted passengers in the rear slammed into the front seats in a crash, causing massive impact or secondary injuries to the driver or front passenger, making them 500% more likely to die in a crash [5].

2. Related Work

How to conduct safe and effective monitoring of anti-fatigue driving and anti-distracted driving is an important topic of research at home and abroad. Existing driving behaviour detection methods can be mainly divided into three categories: detection based on the driver's ECG signal, detection based on the vehicle's driving parameters, and detection based on the driver's facial features. However, most of the research still stays in the theoretical and experimental state, and even if the theory is made into a product, it is difficult to achieve marketability. As shown by a large amount of information, many current domestic and foreign studies are mainly based on the following technologies:

2.1. Based on the Driver's ECG Signal

When drivers are in the state of fatigue driving, the ECG signal will drop regularly [6], and by adopting the ECG signal processing technology through the external detecting device, it is possible to judge the driver's emotional activity and the data of the duration of driving time, so as to judge whether the driver is driving dangerously or not.

2.2. Vehicle-Based Driving Parameters

Vehicle driving parameters are mainly steering wheel movement, speed and direction of travel, the movement of the car steering wheel is closely related to the driver's driving status: the steering wheel if the left and right sway is normal

driving, if the steering wheel does not move for a long time there is a possibility of dangerous driving; speed and direction of travel mostly using infrared detection of the road travelling line, during the day due to good light, the detection effect will also be better.

2.3. Based on the Driver's Facial Features

Facial features when driving dangerously are significant reference factors for human physiological performance [7]. When the human body is fatigued, the eyes blink frequently and the eye closure time is long, through the eye closure time and the number of blinks can judge whether there is a fatigue driving condition; through the machine learning and deep learning can judge whether the driver has a distraction condition.

This system adopts multi-ocular camera to monitor the driver in real time. Through the number of blinks and yawns of the driver, the fatigue level of the driver is predicted; through the Yolov5 trained model, the distracted driving behaviour of the driver is detected, which mainly includes the three behaviours of smoking, drinking, and playing with mobile phones; and through the interrupt controller of the STM32 to simulate whether or not to wear a seatbelt during the driving process. When the system detects the dangerous driving information mentioned above, the system will control the STM32 buzzer to alert.

3. Related Work

This system is based on PC as the main detection device and STM32 microcontroller as the behavioural warning device, and through the connection of modules, it completes the warning of dangerous driving of drivers. This system is mainly based on the PC camera to obtain the driver's facial features and other driving behaviour data [8], counting the number of blinks and the number of times the driver opens his mouth, so as to warn of fatigue driving; through the deep learning, to analyse whether the driver is playing with mobile phones, smoking, drinking water and other three behaviours [9], so as to warn of distracted driving; through the simulation of the driving behaviour characteristics, the key control of the STM32 is simulated as the "seat belt works or not" event in the driver, so as to warn of distracted driving. By simulating driving behaviour characteristics, the STM32 key control is simulated as the event of "whether the seatbelt works or not" in driving, so as to warn of dangerous driving. The system will record the detected data in chronological order for the administrator's reference.

Specific function description: This system takes PC as the core collection device and STM32 microcontroller as the behavioural warning device, with the help of Yolov5 technology and STM32 interrupt control technology, it can realize the collection of driver's distracted state and dangerous driving behaviour, when the driver has fatigue, distraction and dangerous driving state, the system will control the buzzer of the microcontroller and the LED light to remind the

driver. The system will control the buzzer of the microcontroller and the LED light to remind the driver when the driver is tired, distracted and dangerous driving. The driving behaviour characteristics of the driver will be recorded in real time for the administrator's reference.

4. Algorithm Design of Distracted Driving Behaviour Monitoring System Based on Yolov5

This system uses Dlib algorithm to detect the key points of the face, and Perclos model to calculate the number of blinks and yawns of the driver to analyse the fatigue level of the driver, in addition to the fatigue state will exist dangerous driving behaviour. At the same time, in the process of driving, the distraction state will also affect the driver driving, this system will use Yolov5 to detect the driver's distraction characteristics. [10] Yolov5 compared to other versions of the relatively large changes, first of all, it is the use of mosaic data augmentation, randomly selecting the centre of a large picture, in the four corners of the centre of the point of placing a picture to a certain extent to enhance the batch size; secondly, it adopts the Disk Size; secondly, it adopts the Disk Size model, which is the most effective way to detect the key points of the face. size; secondly, it uses DropBlock mechanism to prevent overfitting; then in order to make the labels smooth, it adds Label Smoothing to prevent the neural network from being overconfident, and the last point is that it adds a loss function.

4.1. Blink Count and Mouth Count Modules

This system uses the Dlib algorithm for face keypoint detection. Dlib algorithm is an open source library for machine learning which contains a rich set of machine learning and deep learning algorithms [11]. It is easy to use, including header files directly, and does not depend on other libraries. Currently heavily used in industry and academia, including image processing for virtual reality, intelligent processing of embedded development devices, mobile phones and large-scale high-performance computing environments for real-world applications. dlib open source library has two key functions: `dlib.get_frontal_face_detector()` and `dlib.shape_predictor(predictor_pacth)`. The first function detects the built-in face, using the HOG pyramid, and detects the boundaries of the face region. The latter is used to detect the coordinates of feature points within the boundaries of the face region and outputs the coordinates of those feature points. It requires pretrained models to work properly. By taking the pretrained model `shape_predictor_68_face_landmarks.dat` as a base, the coordinates of the feature point locations can be obtained, and by connecting all the feature points in a face pattern, a coordinate map as shown in **Figure 1** below can be obtained.

Where the right eye is open and closed:

$$\text{EyeOpenness}_{\text{right}} = \text{normalize} \left(\frac{P_{42} \cdot y + P_{41} \cdot y - P_{38} \cdot y - P_{39} \cdot y}{P_{40} \cdot x - P_{37} \cdot x} \right) \quad (1)$$

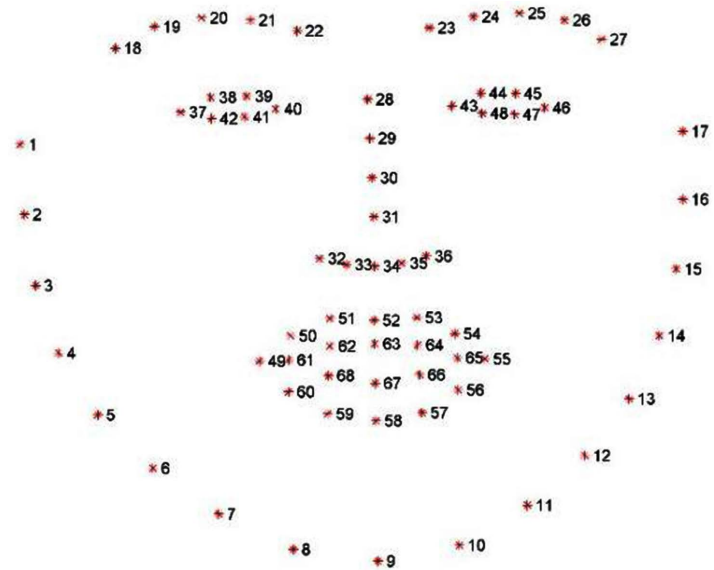


Figure 1. Coordinate diagram of face feature points.

The left eye is open and closed:

$$\text{EyeOpenness}_{\text{left}} = \text{normalize} \left(\frac{P_{47} \cdot y + P_{48} \cdot y - P_{44} \cdot y - P_{45} \cdot y}{P_{46} \cdot x - P_{43} \cdot x} \right) \quad (2)$$

Mouth opening and closing for:

$$\text{MouthOpenness} = \text{normalize} \left(\frac{P_{59} \cdot y + P_{57} \cdot y - P_{51} \cdot y - P_{53} \cdot y}{P_{66} \cdot x - P_{49} \cdot x} \right) \quad (3)$$

4.2. Fatigue Prediction Module

In the study of fatigue response, the degree of eye closure was found to be closely related to the degree of fatigue and was identified as the most secure information. Perclos model calculates the number of hours of eye closure per unit of time as a percentage of the total time, and the Perclos method is considered to be the most effective, accurate, and image-recognition-based fatigue assessment for on-board automobiles. Usually there are 3 types of Perclos determination methods, namely p70, p80 and EM [12].

- p70: indicates that when more than 70% and less than 80% of the pupil area is covered by the eyelid, the eye is considered to be in a closed state, and the percentage of time spent with the eye closed is calculated for each unit of time.
- p80: indicates that when more than 80% of the pupil area is covered by the eyelid, the eye is considered to be closed, and the percentage of time spent with the eye closed is calculated for each unit of time.
- EM: indicates that when more than 50% and less than 70% of the pupil area is covered by the eyelids, the eyes are considered to be in a closed state, and the proportion of time spent with the eyes closed per unit of time is calculated.

Experiments have shown that p80 is more accurate, and the standard used in this system is also p80. The principle of the calculation is to predict the probabil-

ity of fatigue based on the aspect ratio of the eyes. When the human eye is in the open state, according to the eye aspect ratio will fluctuate up and down a certain value. When the human eye is in the closed state, the eye aspect ratio will decrease sharply, and the theoretical value will be infinitely close to zero. At that time, the face detection model was not that accurate. So it was assumed that the eyes were in a closed state when their aspect ratio was below a certain threshold. In order to be able to accurately detect the number of blinks and yawns of a driver, this system is required to process a continuous recorded video in separate frames. Due to the relatively fast blinking speed, the blinking action is usually completed within 1-3 frames. If the absolute value of the difference between the aspect ratio of the eyes in the current frame and the previous frame is greater than 0.38, the driver is considered to be in a fatigue state. Where the eye aspect ratio (EAR) is calculated by the following formula: [13]

$$\text{EAR} = \frac{P_{42} \cdot y + P_{41} \cdot y - P_{38} \cdot y - P_{39} \cdot y}{2(P_{40} \cdot x - P_{37} \cdot x)} \quad (4)$$

In the design process of the code, it is calculated whether the value of eye aspect ratio is greater than the threshold value (0.15) for each frame of the video, and if it exceeds 50 times in a row, it is considered to be in the state of “fatigue”, and the Perclos model includes the eye aspect ratio and the degree of mouth opening and closing, and the system will score the Perclos model when it is greater than 0.38; otherwise, the driver is judged to be awake. In the specific implementation process, the system will score the Perclos model after 150 frames, and when the Perclos model is greater than 0.38, the system will judge that the driver is in a fatigue state; otherwise, it will judge that the driver is in a wakeful state.

4.3. Distraction Prediction Module

In addition to the fatigue state will exist dangerous driving behaviour, during the driving process, the distraction state will also affect the driver driving [14], this system will use Yolov5 to detect the driver’s distraction characteristics. Yolov5 compared with other versions made a relatively big change, first of all, it is that it adopts the mosaic data augmentation, randomly selects the centre point on a large map, and places a picture in each of the four corners of the centre point, which enhances the batch size to some extent; secondly it uses DropBlock mechanism to prevent overfitting; then to make the labels smooth, it adds Label Smoothing to prevent the neural network from being overconfident; and the last point is that it adds a loss function.

The distraction prediction of the system needs to be done in the Anconda environment.

Firstly, the source code of Yolov5 needs to be downloaded and unpacked, and the dependency packages contained in the requirements.txt file should be downloaded with a single click by entering the folder via CMD and using the pip method. The folder also contains the train.py file, which is also the startup file for

training the yolo model. The sample image for model training is shown shown in **Figure 2**.

In labels, the model will convert the dataset format to yolo_txt format, *i.e.*, each xml annotation extracts the bbox information in txt format, each image corresponds to a txt file, each line of the file contains information about a target, including class, x_center, y_center, width, height format. Under the model folder in the yolov5 directory is the configuration file of the model, this side provides s, m, l, x versions, gradually increasing, as the architecture increases, the training time is also gradually increasing. The training log generated by the training process of yolov5 is shown in **Figure 3**.

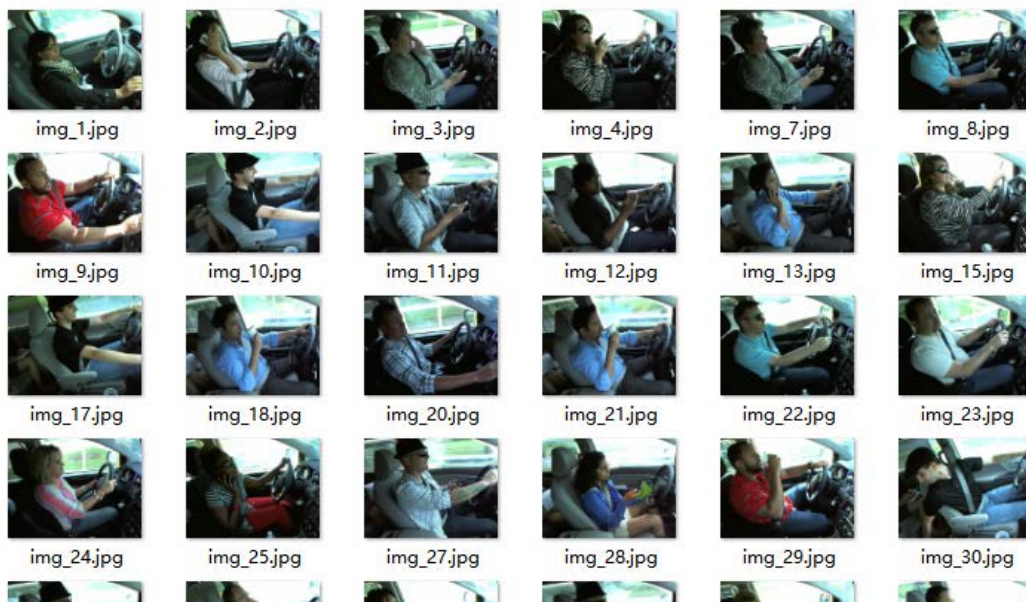


Figure 2. Training sample diagram.

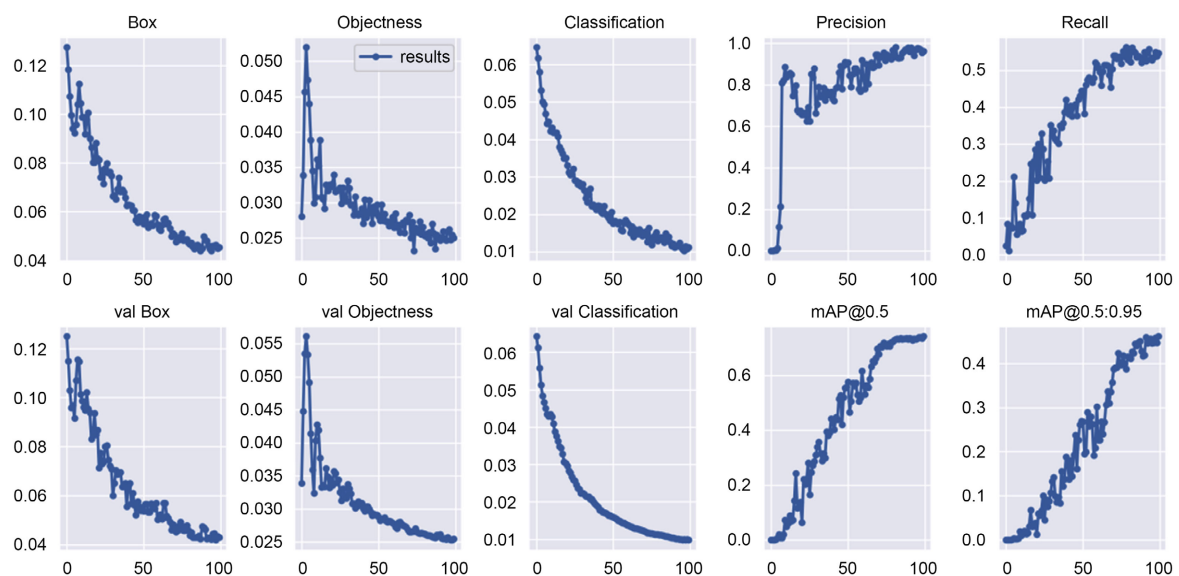


Figure 3. Training log diagram.

When the camera is turned on, the camera will pass frame into the detection function `myframe.frameTest()`, at this time `ret` and `frame` will be returned as a function value, which `ret` for the detection results, `ret`'s format is [lab, eye, mouth], lab for yolo's recognition results contain "phone" "smoke" "drink", eye is the aspect ratio of the eyes, mouth is the degree of mouth opening and closing, `frame` is the frame frame painted with the identification results of the identification of the frame, distraction detection results in 15 frames as a cycle, when the system detects distraction behaviour will be identified in red font. The key code of this design is shown below.

5. System Testing

5.1. Hardware Circuit Test

1) Circuit check

Before power-on debugging, this paper in accordance with the circuit diagram, in a certain order, level by level corresponding to the check, at this time, the power supply is not connected to the wrong, and no short-circuit phenomenon.

2) Power-on observation

Power supply using USB access to the PC port for power supply, at this time, no short-circuit phenomenon at the power supply. After the power supply is switched on, the buzzer of the development board will sound, there is no abnormality in the circuit at this time.

3) Indicator debugging

When the development board for the first time when the power supply, the simulated scene for the driver did not fasten the seat belt, the STM32 development board buzzer will sound the alarm, when the driver presses the `KEY_PRESS` button, the simulated scene for the driver to fasten the seat belt, the buzzer will stop the alarm. When the driver opens the camera, the system will decompose the dynamic video into different frames, and analyse each frame image specifically, when the driver has the state of smoking, drinking, playing mobile phone, fatigue, and distraction, the system calls the buzzer of STM32 for alarm prompt.

5.2. General Users

5.2.1. General User Login Module Is Shown in Figure 4

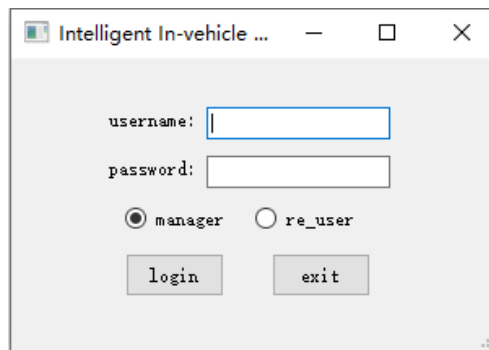


Figure 4. Ordinary user login.

The user name of the administrator and ordinary user is zjw, and the password is 123. Users can select the corresponding mode according to different modes, when the selected mode is “Ordinary user”, users can test the related warning module; when the selected mode is “Administrator”, users can annotate the driving information of the ordinary user module. When the selected mode is “Administrator”, the user can annotate the driving information of the ordinary user module.

5.2.2. Normal User Interface Is Shown in Figure 5

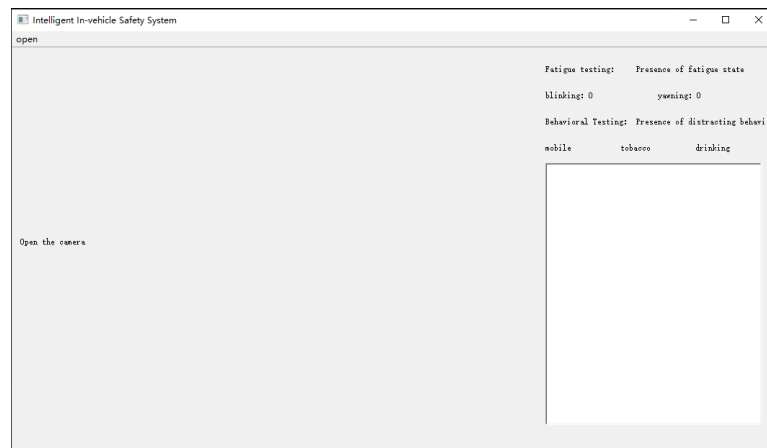


Figure 5. General user interface.

In the common user interface, its main interface layout is to provide camera privileges on the left side and to analyse the information of image frames on the right side, in which the camera privileges rely on the cv2.VideoCapture(0) function to be implemented, and the information of image frames mainly contains the number of blinking times, the number of yawning times, behavioural detections, distraction detections, fatigue detections and so on.

5.2.3. Fatigue Detection and Distraction Detection Are Available to the Average User by Turning on the Camera Is Shown in Figure 6

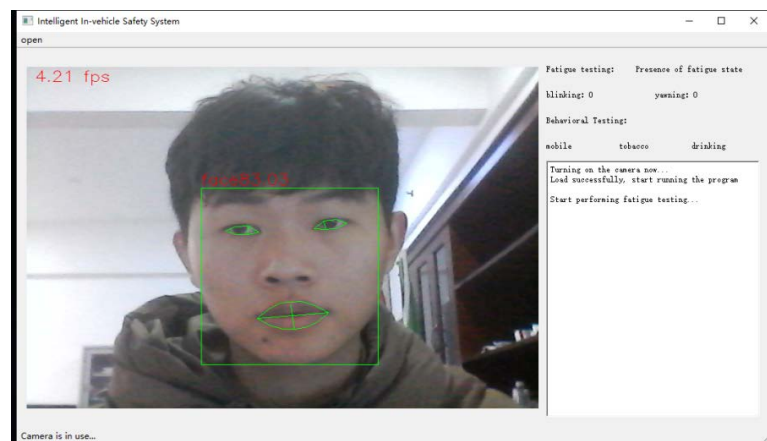


Figure 6. Fatigue, distraction detection.

5.2.4. Blink Count and Yawn Count Is Shown in Figure 7

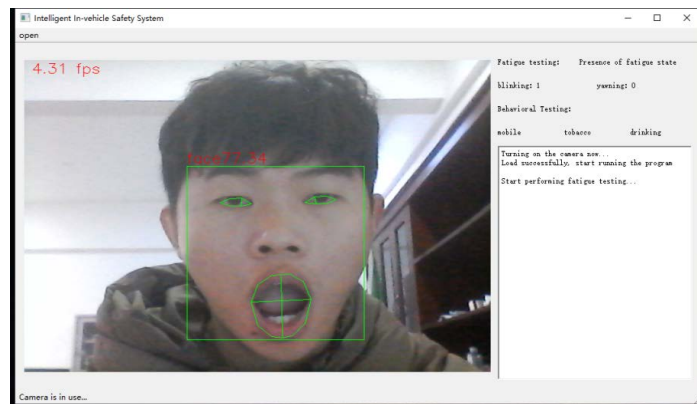


Figure 7. Blink and yawn count.

In dynamic detection, the value of the eye aspect ratio is calculated for each frame of the video to detect whether the value is greater than a threshold value, and if it exceeds 50 consecutive times, it is determined to be “fatigue”.

5.2.5. Smoking Test Is Shown in Figure 8



Figure 8. Smoking inspection.

5.2.6. Drinking Water Test Is Shown in Figure 9

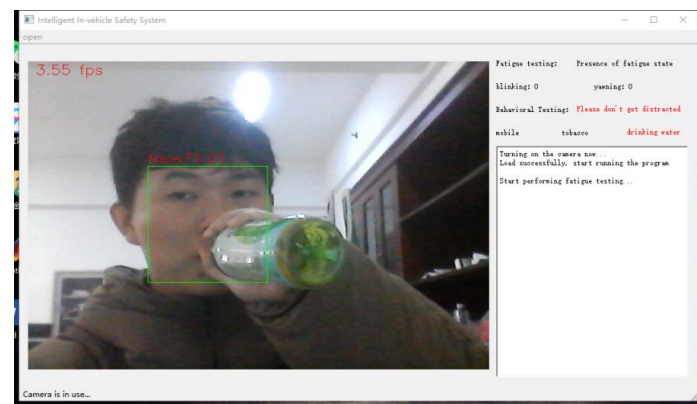


Figure 9. Drinking check.

5.2.7. Play Mobile Phone Detection Is Shown in Figure 10

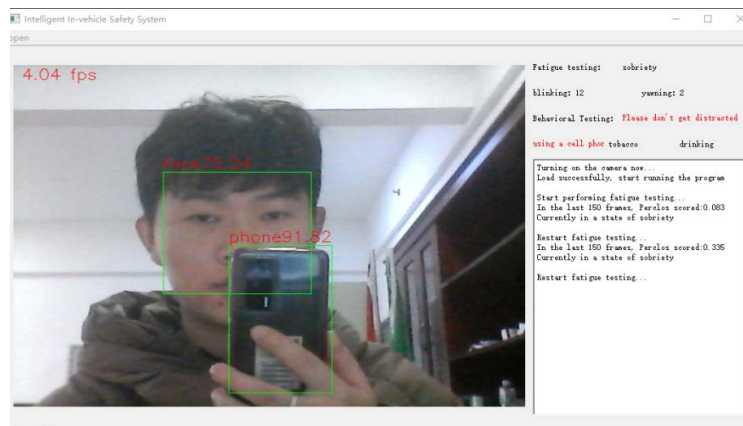


Figure 10. Playing mobile phone detection.

6. Conclusion

This system is based on YOLO V5 to design a piece of intelligent supervision system to assist safe driving, through the number of blinks and yawns of the driver to complete the detection of fatigue driving and make a warning [15]. The model predicts three distracted driving behaviours: smoking, drinking, and mobile phone use. It also simulates whether the driver is wearing a seatbelt while driving through the interrupt controller of STM32. When the system detects fatigue driving, distracted driving, not wearing a seat belt behaviour, the system will control the STM32 buzzer to be prompted. At present, the development board can also be involved in other data such as luminosity monitoring, the user interface needs to be further embellished, the design mainly uses the front camera on the PC for the analysis of the driver's behaviour, in the subsequent development process, you can also add an external camera to sense the surrounding data, for example, analysis of whether to run a red light, analysis of whether there are pedestrians around the body, and so on.

Funding

This research was funded by 2023 General Project of Educational Teaching Reform Subjects in Yancheng Teachers University (Grant No. 2023YCTCJGY34), Yancheng Teachers University 2023 Curriculum Civics Demonstration Course Project.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Kong, R. (2003) The Neglected Causes of Car Accidents. *The Friends of Agricultural Equipment*, No. 6, 46-47.

- [2] Qin, B.B., Peng, L.K., Lu, X.M. and Qian, J.B. (2021) Research Progress on Driver Distracted Driving Detection. *Computer Applications*, **41**, 2330-2337.
- [3] Jiao, S.J., Liu, L.Y. and Liu, Q. (2021) A Hybrid Deep Learning Model for Recognising Actions of Distracted Drivers. *Sensors*, **21**, 7424. <https://doi.org/10.3390/s21217424>
- [4] Peng, Y.C., Cheng, L.Y., *et al.* (2021) Examining Bayesian Network Modeling in Identification of Dangerous Driving Behaviour. *PLOS ONE*, **16**, e0252484. <https://doi.org/10.1371/journal.pone.0252484>
- [5] Kuang, W.T., Mao, K.C., Huang, J.C. and Li, H.B. (2016) Fatigue Driving Detection Based on Gaussian Eye-White Model. *Chinese Journal of Image Graphics*, **21**, 1515-1522.
- [6] Zhang, H. (2021) Research on Fatigue Driving Detection and Its Early Warning Method Based on Driver Physiological Signals. Master's Thesis, Jilin University, Changchun. <https://doi.org/10.27162/d.cnki.gilin.2021.005168>
- [7] Yang, H. (2020) Fatigue Driving Detection Based on Facial Fine Motion Recognition and Deep Learning. Master's Thesis, Nanchang University, Nanchang. <https://doi.org/10.27232/d.cnki.gnchu.2020.003221>
- [8] Hu, Y.Y. (2020) A Video Processing Approach Based on Python + OpenCV. *Computer Products and Distribution*, No, 8, 139+141.
- [9] Utaminigrum, F., Purwanto, A.D., Masruri, M.R.R., *et al.* (2021) Eye Movement and Blink Detection for Selecting Menu On-Screen Display Using Probability Analysis Based on Facial Landmark. *International Journal of Innovative Computing, Information and Control*, **17**, 1287-1303.
- [10] Yan, F. and Lei, L. (2023) A Violation Behavior Detection Algorithm Based on YoloV5 and Dilb for Online Video Surveillance. *Optoelectronic Technology*, **43**, 276-282.
- [11] Abtahi, S., Omidyeganeh, M., Shirmohammadi, S., *et al.* (2014) YawDD: A Yawning Detection Dataset. *Proceedings of the 5th ACM Multimedia Systems Conference*, March 2014, 24-28. <https://doi.org/10.1145/2557642.2563678>
- [12] Hsu, R.-L., Abdel-Mottaleb, M. and Jain, A.K. (2002) Face Detection in Color Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**, 696-706. <https://doi.org/10.1109/34.1000242>
- [13] Yan, J.-J., Kuo, H.-H., *et al.* (2016) Real-Time Driver Drowsiness Detection System Based on PERCLOS and Grayscale Image Processing. 2016 *International Symposium on Computer, Consumer and Control (IS3C)*, Xi'an, 4-6 July 2016, 243-246. <https://doi.org/10.1109/IS3C.2016.72>
- [14] Huang, X.K. (2021) Research on Distracted Driving Behaviour Detection Based on Deep Learning. Master's Thesis, Chongqing University of Posts and Telecommunications, Chongqing. <https://doi.org/10.27675/d.cnki.gcydx.2021.000773>
- [15] Bertl, J., Ewing, G., Kosiol, C., *et al.* (2017) Approximate Maximum Likelihood Estimation for Population Genetic Inference. *Statistical Applications in Genetics and Molecular Biology*, **16**, 291-312. <https://doi.org/10.1515/sagmb-2017-0016>