Safe Driving System Based-on STM32

Chen Tong and Shuqun Wang

November 19, 2019

A safe driving system based-on STM32

Tong Chen, Wang Shuqun*

(Southwest University of Nationalities, Chengdu, Sichuan, 610225)

Abstract: With the increasing popularity of private cars, the incidence of traffic accidents has increased year by year. Among them, drunk driving and fatigue driving account for a large part of the reason. Aiming at this situation, the safe driving system offers a practical and effective solution. The system consists of three modules: anti-collision, anti-drunk driving and anti-fatigue driving: STM32 is used as the core to expand outward. The HC-SR04 ultrasonic range finder is used to detect the distance from the front obstacle. The side-heated semiconductor alcohol gas sensor MQ-3 is used to detect the alcohol concentration in the air. Collect and re-processe and recognize the face information with the Raspberry Pi, and issue a prompt when the driving time reaches the upper limit for continuous driving. As a result, this system achieves the three major functions: anti-collision, anti-drunk driving and anti-fatigue driving. The safe driving system puts prevention first, which can reduce the incidence of traffic accidents effectively.

Keywords: anti-collision, anti-drunk driving, anti-fatigue driving, STM32, HC-SR04 ultrasonic range finder, MQ-3 alcohol sensor, raspberry pi face recognition

CLC number: [] Document code: [] Document number: []

Introduction: According to statistics, in the annual traffic accidents, the number of traffic accidents caused by drivers' unqualified driving skills, bad driving habits, drunk driving and fatigue driving accounts for more than half of the total[1]. Traffic accident not only causes a large number of casualties, but also brings misfortune to countless families and affects social and economic development and stability. As for drunk driving, the current laws and regulations can only test the vehicles passing by at a fixed location to detect whether the driver has drunk, which can not prevent drunk driving from the source. If we test whether the driver has drunk or has reached the standard of drunk driving when the driver opens the door and gets on the vehicle, and then take corresponding measures, we can fundamentally prevent drunk driving. At the same time, fatigue driving is also a major cause of traffic accidents. In the process of driving, the insufficient rest and excessive fatigue will result in inattention and slow response. If the driver can not make the right response in time, the accident will happen easily. At present, there is no good solution for effectively controlling fatigue driving on the market. The safe driving system intends to use face recognition to record the continuous driving time of the driver to judge whether the driver is fatigued or not. When the upper limit is reached, an alarm will be given to alert the driver to be cautious and rest in time, which can prevent fatigue driving to a certain extent.

1. System frame structure design

The safe driving system is designed to use STM32F103C8T6 as the main control chip, which controls HC-SR04 ultrasonic range finder, MQ-3 alcohol sensor and Raspberry pi 3b+ to form three functional modules: anti-collision, anti-drunk driving and anti-fatigue driving.

The STM32F103C8T6 microcontroller uses a high-performance ARM Cortex-M3 32-bit processor with 64K bytes of SRAM. The CPU can read and write with 0 bytes. Therefore, there is no need to increase the external data memory, which can save system cost and improve system stability and reliability. And STM32F103C8T6 supports three low-power modes: sleep, shutdown and standby. Compared with the same series of chips, the performance has a significant advantage[2].

Taking into account the complexity of image acquisition and image processing algorithms, as well as the size, installation and cost requirements of the application scenarios of fatigue driving detection, the Raspberry Pi development board (raspberry pi 3b+) is used as the control unit for fatigue driving. It is an ARM-based micro development board with rich peripheral interfaces: 4 USB ports, 1 network interface and 1 HDMI interface. It also supports Raspbian, Kali Linux, RISC OS and other operating systems, and the price is cheap. At the same time, Python is used as the main development language. The Python language is simple, easy to read and easy to extend, and has rich and powerful libraries, such as scientific computing extension libraries: NumPy, SciPy, and matplotlib, which are ideal for studying image processing[3].

The overall block diagram of the system is shown in Figure 1.

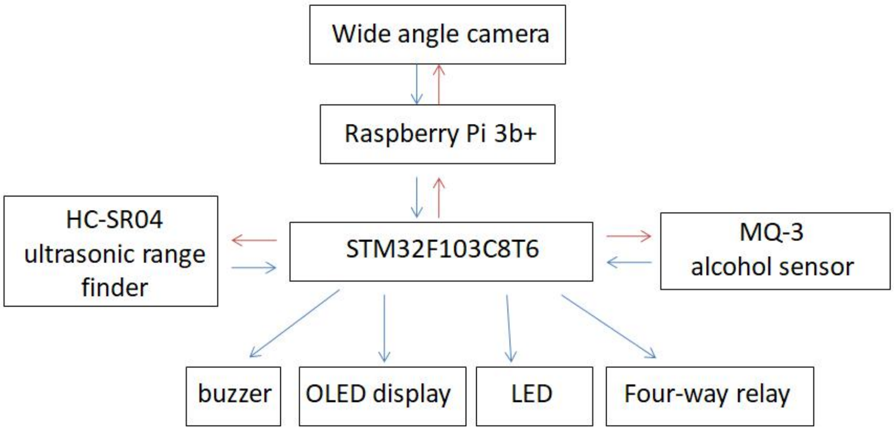

Figure 1 system overall block diagram

（1） Anti-collision module

The anti-collision module mainly uses an ultrasonic range finder based on STM32 single-chip microcomputer to monitor the distance between the car and the front obstacle in real time, and set buzzer alarms of different frequencies according to different distances to remind the driver to avoid obstacles and achieve the purpose of collision avoidance.

（2） Anti-drunk driving module

It is prescribed that the alcohol content in the driver's blood is greater than (equal to) 20 mg / 100 ml and less than 80 mg / 100 ml, which belongs to drunk driving. According to the actual situation, it can be approximated that in the case of drinking slowly, driving within 8.6 hours after drinking 3 bottles of beer will violate the standard[4].

In anti-drinking driving module, we mainly use the alcohol sensor based on STM32 single-chip microcomputer. When the door is opened, the sensor device enters the working state to detect the air alcohol concentration in the car. If the drunk driving concentration standard is reached, the power supply of the car ignition device will be cut off, and the car can not be started to prevent the drunk driving.

（3） Anti-fatigue driving module

According to the Traffic Safety Law, the driver shall not drive continuously for more than 4 hours without a break. When the driver drives continuously for more than four hours, the mental function and physiological function will change, the driver's attention will be difficult to concentrate, the reaction speed will slow down, there will be deviation when judging the road conditions, and there will be various problems such as blurred vision[5].

Therefore, the anti-fatigue driving module adopts the method of face recognition, uses the raspberry pi wide-angle camera to collect and recognize the face data, and collects the face information every 20 minutes for comparison. When it is judged that the same driver drives continuously for up to four hours, an alert is issued to alert the driver to drive carefully or to stop as soon as possible to have a rest.

2. System hardware design

Four-way relay, ultrasonic sensor, alcohol sensor, OLED display, LED light and buzzer are controlled by STM32F103C8T6 single-chip microcomputer; face recognition module is achieved by Raspberry pi 3b+ and wide-angle camera.

Figure 2 shows the control circuit diagram of the STM32 microcontroller and ultrasonic range finder and alcohol sensor.

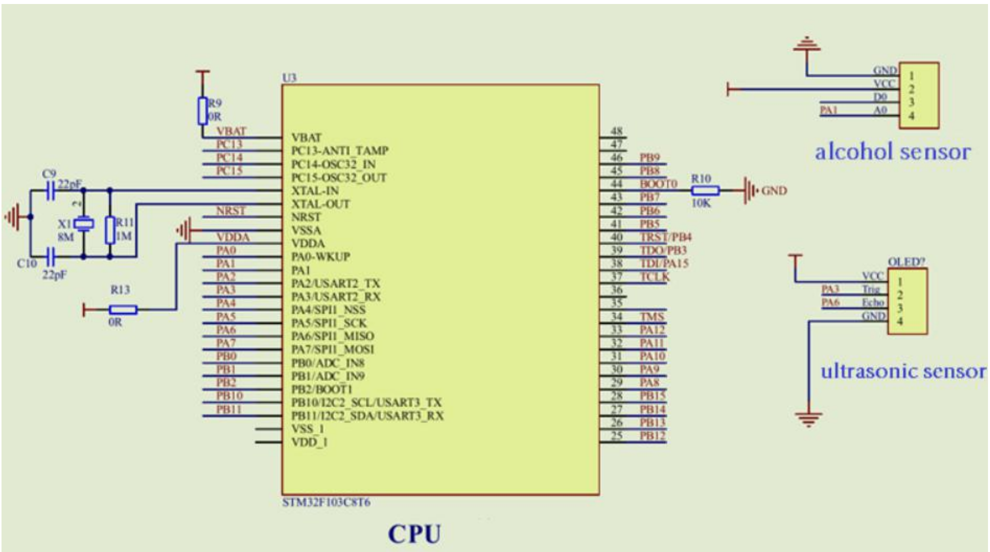

Figure 2 CPU control circuit

As the main control CPU, STM32 is connected to the minimum system, and then directly connected to the A0 output of the alcohol sensor through the GPIO port to exchange information of analog signals; the Trig end and Echo end of the ultrasonic range finder are connected with the corresponding IO port of the main control chip to achieve the transmission and reception of ultrasonic signals.

2.1 Ultrasonic ranging

This system uses the HC-SR04 ultrasonic range finder. The HC-SR04 ultrasonic ranging module provides 2cm-400cm non-contact distance sensing capability with a range accuracy of 3mm[6]. The module includes ultrasonic transmitter, receiver, and control circuit. The working principle of HC-SR04: the EM78P153 microcontroller is the core of processing. When the trigger signal input of the EM78P153 receives a high-level signal of minimum 10μs sent from the STM32 microcontroller, the EM78P153 emits a 40 kHz square wave, the square wave will be amplified by the LMC6034IM operation and be converted into 40 kHz sound wave by ultrasonic probe; the sound wave is reflected by the object and received by the receiving probe. After being processed by the EM78P153, a high level signal is sent back to the STM32 MCU through the echo signal output terminal. The STM32 MCU starts timing when the high level signal is issued and ends timing when it receives the high level signal, the timing is ended. The time interval between the two signals is the propagation time of the ultrasonic wave[7].

2.2 Alcohol sensor

This system uses MQ3 alcohol sensor. MQ3 alcohol sensor has high sensitivity and good selectivity to ethanol vapor[8], and its conductivity increases with the increase of the alcohol concentration in the air. The working mode is to convert the alcohol concentration into a level signal. Connect the A0 end of the component to the analog voltage output terminal, that is, to the ADC channel 1 PA1 end of the STM32F103C8T6. The D0 terminal is connected to the TTL high and low level output terminals, and the VCC and GND terminals are respectively connected to the positive and negative power terminals.

The design of the hardware physical board and the display of the oled display are shown in Figure 3:

Figure 3 oled display L: distance from the front obstacle C: air alcohol concentration

2.3 Raspberry Pi and Wide Angle Camera

The Raspberry Pi is a Linux 32-bit operating system based on Debian. It uses a Micro SD card as a memory hard disk. It is small in size but has the basic functions of a personal computer. It brings up 96 GPIO interfaces which can be used to connect multiple underlying layers. And it has SATA and HDMI interfaces for high-definition output[9].

The Raspberry Pi wide-angle camera provides 2592×1944 static picture resolution, using the photosensitive chip OV5647. It supports 1080p30, 720p60 and 640×480p60/90 video recording. Insert the wide-angle camera into the camera interface on the Raspberry Pi 3B+ board to enable the camera function.

3. Software design

3.1 Ultrasonic Ranging Module

According to the ultrasonic ranging principle, the Wave_SRD_Init function is written in the wave.c file for ultrasonic initialization. Then, the TIM_GetCounter function is called to obtain the count value of the timer, and use the formula length = TIM_GetCounter(TIM3)*340/200.0 to get the ultrasonic distance, and the value is stored in the length variable. After that, call the Wave_SRD_Strat function, set the Trig to a high level, and continue to trigger greater than 10us, triggering the ultrasonic module to work. The display of the distance is achieved by calling the OLED_ShowString function. Finally, the program judges the range of the value of length, and when the distance is greater than 60cm, the buzzer will not give an alarm; when between 40-60cm, the buzzer alarms slowly; when between 20-40cm, the buzzer alarms rapidly; when less than 20cm, the buzzer keeps alarming and the brake relay works.

3.2 Alcohol Sensor Module

Because the MQ3 component requires 20s to warm-up, the OLED display screen does not show the concentration value during the first 20s of the component startup, and the ignition relay works to control the car can not be started. Call the OLED_ShowCHinese and OLED_ShowString functions in the header file 20 seconds after the module starts, and connect the D1, D0, DC, and CS ports of the OLED display screen to the PA5, PB0, PB1, and PA4 ports of the STM32F103C8T6. The screen shows the measured airborne alcohol concentration at this time.

Specific alcohol concentration detection process: First, call the Adc_Init and OLED_Init functions to initialize the MQ3 alcohol ADC value and the OLED. Then, call the Get_Adc_Average function to get the ADC value and store it in the TVOC_AD variable. Since the value of TVOC_AD obtained at this time is corresponding to the value converted by the current voltage value on the pin relative to 3.3V and 4095 ($2^{12}-1$), the alcohol concentration is obtained by using the formula avalue=(float)TVOC_AD*(3.3/4095). And the value will be displayed on the OLED screen. The positive electrode of the red LED and the positive electrode of the green LED are respectively connected to the PC13 and PC14 ports of the STM32F103C8T6. Finally, when the alcohol concentration is less than 20mg/100ml, the green LED lights up, the ignition relay, the seat belt relay, the sound and light alarm relay do not work, the car can starts normally; when the alcohol concentration is greater than 20mg/100ml, the red LED lights up, and the four-way relay works, so that the car can not be activated.

Module implementation flow chart:

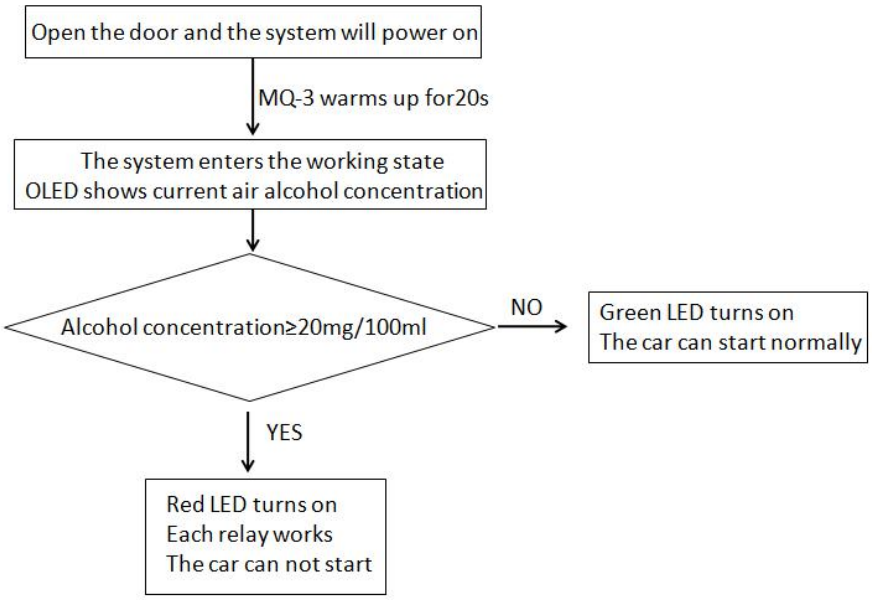

Figure 4 Alcohol detection module flow chart

3.3 Face Recognition Module

The software development environment needs to prepare OpenCV and Python 3[10]. Use the command $ source ~/.profile to open a new terminal to run the source to ensure that the system variables have been set up correctly, then enter on the virtual environment, open the python compiler, and run version 3.5 or above. Afterwards, have OpenCV library installed in the Raspberry pi, test to confirm that camera is working properly, then write in the code of face recognition. A complete face recognition project needs to go through three stages: first, create a dataset , where we will store for each ID, a group of photos in gray with the portion that was used for face detecting. Second, train all user data directly by a specific OpenCV function, so that the recognizer is able to recognize each person's face and their respective ids. Finally, capture a fresh face on the camera and if this person had his face captured and trained before, the recognizer will make a prediction returning its id and an index, shown how confident the recognizer is with this match. The process block diagram is shown in Figure 5.

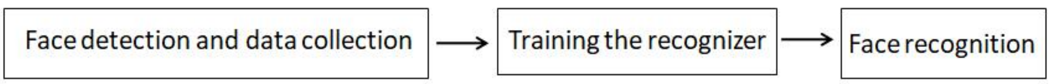

Figure 5 Block diagram of the face recognition process

After identifying successfully, a command is sent to the relay, and the face information is collected every 20 minutes for comparison. If the detection judges that the driver is the same one for 12 consecutive times (that is, the driver has been driving for 4 hours continuously), the information will be returned to the Raspberry Pi voice module, prompting the driver to drive carefully and rest in time.

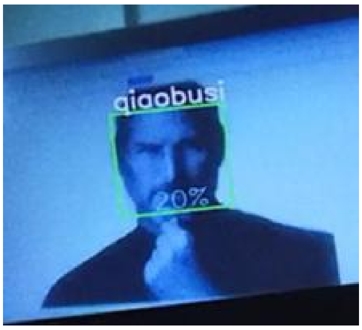

Figure 6 Identification results

4 Conclusion

Based on the main cause of traffic accidents, we design a safe driving system which is anti-collision, anti-drunk driving and anti-fatigue driving by using the STM32 as the core to control the ultrasonic range finder, alcohol sensor, Raspberry pi and camera. The safe driving system realizes three functions: First, when the distance between the car and the front obstacle reaches the preset distance, the buzzer will alarm. Second, when the driver opens the door after drinking, causing the alcohol concentration in the car to exceed the standard, the vehicle can not be activated. Thirdly, when the driver drives continuously for four hours without parking, the voice module will send a reminder. This system can eliminate drunk driving fundamentally, prevent fatigue driving, and reduce the incidence of traffic accidents.

Acknowledgment

This paper is supported by the innovation and entrepreneurship training program (s20191 0656174).

References

[1] Zhang Chaoming. Analysis of factors affecting safe driving of vehicles and preventive me asures [j]. Modern Economic Information, 2019 (09): 356.

[2]Qin Yongyi. Design of vehicle alcohol detection system based on STM32 [d]. Qufu Normal Un iversity, 2014.

[3] Lin Hong, Wu Liangfeng. Real-time monitoring system of eye fatigue based on Raspberry Pi [J]. Journal of Minjiang University, 2019, 40(02): 76-83.

[4] Su Qing, Zhou Yiji. Prediction of alcohol content in human body after drinking and analy sis of safe driving time [j]. Computer Knowledge and Technology, 2019, 15 (24): 290-292.

[5] Kang Xizhao. Influencing factors of driving safely and and prevention analysis [j]. Hebe i Agricultural Machinery, 2019 (09): 54.

[6] Li Jun, Shen Junze. Design of ultrasonic rangfinder based on HC-SR04 [j]. Single Chip Mi crocomputer and Embedded System Application, 2011, 11 (10): 77-78.

[7] Zhang Kesheng, Long Changxi. Acoustic velocity measurement by time difference method based on HC-SR04 module [j]. Machinery and Electronics, 2018, 36 (02): 54-57.

[8] Lv Xiaoying. Design of alcohol detection system based on MQ-3 [j]. Technology horizon, 2019 (21): 31-32.

[9] Zhu Lin, Zhang Feng, Shen Zexuan, Xia Yuanman. Raspberry Pie 3b series face recognition laboratory access control system [j]. MCU and embedded system applications, 2019,19 (04): 69-71+76 .
[10] Ma Shuai. Design and implementation of face recognition access control system based on Raspberry Pi [d]. Dalian Jiaotong University, 2018.