Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

A Real-time Driver Fatigue Detection Method Based on Two-Stage Convolutional Neural Network ⋆

Hu He * **Xiaoyong Zhang** * **Fu Jiang** * **Chenglong Wang** *
Yingze Yang * **Weirong Liu** * **Jun Peng** *

* *School of Computer Science and Engineering, Central South University, Changsha, China 410083 (corresponding author e-mail: jiangfu0912@csu.edu.cn)*

Abstract: Fatigue-related traffic accidents have a higher mortality rate and cause more significant damage to the environment. To ensure driving safety, a real-time driver fatigue detection method based on convolutional neural network (CNN) is proposed in this paper. The proposed fatigue driving detection method is cascaded by two CNN-based stages, including a detecting phase and classifying phase. The Location Detection Network is designed to extract facial features and localize the driver's eyes and mouth regions. Then the State Recognition Network is training to recognize the driver's eyes and mouth status. Simulations show that the proposed method has good effect of real time process and high accuracy of detection. Experiments conducted on Raspberry Pi 4 embedded system indicate that the proposed method has a good performance in the real driving environment.

Keywords: Driving safety; Driver fatigue detection; Facial feature; Convolutional neural network; Location Detection Network; State Recognition Network

1. INTRODUCTION

The rapid development of the transportation system has also increased the number of casualties. Fatigued driving has always been a driver's occupational hazard, which is a significant cause of road traffic accidents and has an important impact on road safety (Sikander and Anwar, 2018). 16% of fatal traffic accidents and 13% of collisions that cause injuries are related to fatigue driving (Asbridge et al., 2012). The AAA Foundation published a report on traffic safety (Arnold and Tefft, 2015), which stated that drivers' attention and decision-making ability during fatigue driving would be affected and cause accidents. They appeal to the public to pay attention to the danger and seriousness of fatigue driving.

Many methods and experiments have been applied for the driver fatigue detection. Currently, fatigue driving detection can be divided into three categories according to the input characteristics: physiological parameters, vehicle data, and facial features.

The driver fatigue detection method based on monitoring physiological parameters is closely related to the physiological status of the driver. Some physiological features can be utilized as fatigue representation, such as electrocardiogram (ECG) (Fu and Wang, 2014), electroencephalogram (EEG) (Simon et al., 2011), electromyography (EMG) (Zhang et al., 2013) and electrooculogram (EOG) (Lal and Craig, 2002). The driver fatigue level can be determined by the changes of acquired physiological features. These methods have been shown to have good accuracy. However, to measure these parameters, the driver is required to wear the appropriate detection equipment during the driving process, which is a driver's intrusion mechanism that interferes with the driver's normal driving.

Fatigue driving detection method by analyzing vehicle data is an indirect detection method. Data information sharing between vehicles in (Al-Sultan et al., 2013) is used to detect the driver's abnormal behavior. (Sandberg and Wahde, 2008) analyzed time-series data such as vehicle speed and steering wheel angle for fatigue driving detection, which can predict fatigue-related lane departure six seconds in advance. Obtaining and analyzing the real-time data on the vehicle is susceptible to the driver's driving habits and the external environment, so the detection accuracy is closely related to the driver and the driving environment.

Facial fatigue characteristics include head posture, yawning cycle, blink frequency, etc. (Mandal et al., 2016) is based on the adaptive integration of multiple models for detecting eyes. It can be used to estimate driver's fatigue state but the multiple models are very complicated. In (Alioua et al., 2014), yawning is used for driver fatigue detection. Support Vector Machine (SVM) and gradient edge detectors are used to locate the mouth. The method needs more features to be included to improve accuracy.

Facial features can be learned through deep learning techniques. Convolutional neural network (CNN) has shown the high accuracy and efficiency in object detection and recognition. Various object detection algorithms based on

⋆ This work is partially supported by National Natural Science Foundation of China (Grant Nos. 61873353 and 61672539).
Corresponding author: Fu Jiang. E-mail: jiangfu0912@csu.edu.cn

(a) (b) (c) (d)

Fig. 1. Original and gamma correction renderings: (a) Underexposed image. (b) Underexposed image after gamma correction. (c) Overexposed image. (d) Overexposed image after gamma correction.

CNN have become ubiquitous, like Faster RCNN (Ren et al., 2015), single-shot multibox detector (SSD) (Liu et al., 2016), you only look once (YOLO) (Redmon et al., 2016), etc. To take advantage of the high accuracy and high efficiency of CNN, we propose a CNN-based driver fatigue detection method. It includes a Location Detection Network and a State Recognition Network. The Location Detection Network extracts eyes and mouth regions instead of the driver's whole face. Extracting local facial regions can reduce network training parameters and undesirable noise effects. The State Recognition Network is responsible for judging the state of the eyes and mouth regions. The integration of the driver's eyes state and mouth state for a period of time can determine whether the driver is driving fatigue.

The contributions of this paper are three-fold. First, gamma correction is added on the frame image preprocessing to achieve automatic grayscale correction of the uneven illuminance image. Second, we propose a real-time fatigue driving detection method based on a two-stage convolutional neural network. Third, results on the computer and the embedded device Raspberry Pi 4 are provided to show the proposed method can achieve real-time requirements with high detection accuracy.

The remainder of this paper is organized as follows. In Section II, we introduce gamma correction in the frame image preprocessing. In Section III, we propose a real-time fatigue driving detection method based on a two-stage convolutional neural network. In Section IV, we evaluate the performance of the proposed method using a computer and Raspberry Pi 4. We conclude the paper in Section V.

2. FRAME IMAGE PREPROCESSING

During the fatigue detection process, the driver's face would be unclear and the quality of the frame image would be affected when the illumination intensity changed drastically. The frame image can easily be underexposed or overexposed. Fig. 1(a) shows the cases of underexposure and Fig. 1(c) shows the case of overexposure. The grayscale distribution of such images is uneven. It greatly reduces the quality of the frame image, which will have a serious negative impact on the subsequent use of real-time frame images to analyze whether the driver is fatigued or not. To solve this problem, gamma correction is added in the frame image preprocessing to reduce the influence of uneven gray distribution on the image and improve frame image quality.

Cathode Ray Tube (CRT) display was once widely used worldwide. The relationship between the input voltage of CRT display and screen brightness is nonlinear. There is a power-law curve relationship between them. The power-law curve is described by the following gamma correction formula:

$$V_{\text{out}} = V_{\text{in}}^{\gamma}, \tag{1}$$

where the non-negative real input value V_{in} is raised to the power γ to get the output value V_{out}.

Color input in the camera will eventually diminish brightness when displaying to the display, which will affect the quality of imaging. The camera must perform gamma correction to keep the imaging quality. Therefore, the camera need to introduce a nonlinear distortion opposite to the display. We call it the gamma value of the camera, which is $1/\gamma$. The relevant formula is as follows:

$$V_{out} = (V_{in})^{1/\gamma}. \tag{2}$$

The gamma correction of the camera and the gamma correction of the display cancel each other out, which can reduce the influence of the uneven grayscale distribution of the image. From the equation (1) and the equation (2), the equation (3) can be obtained:

$$V_{display} = \left(V_{camera}^{1/\gamma_1}\right)^{\gamma_2} = V_{camera}^{\gamma_2/\gamma_1}. \tag{3}$$

Equation (3) shows that when γ_1 is equal to γ_2, the display can display the image of the original scene perfectly. Gamma correction is essentially a power function on grayscale. Therefore, gamma correction is also applied to image processing to achieve image contrast enhancement, smoothing the details of dark or light tones.

In the real driving condition, since the eyes and mouth regions acquired through the frame image are hardly partial overexposed or partial underexposed, gamma correction can be performed on the entire frame image to improve image contrast. As shown in Fig. 1(b) and Fig. 1(d), the gray value of the input frame image is transformed into a roughly similar desired gray value by the gamma correction.

3. DRIVER FATIGUE DETECTION METHOD

In this section, we propose a driver fatigue detection method based on convolutional neural network. The framework of the proposed method is shown in Fig. 2. Location Detection Network and State Recognition Network

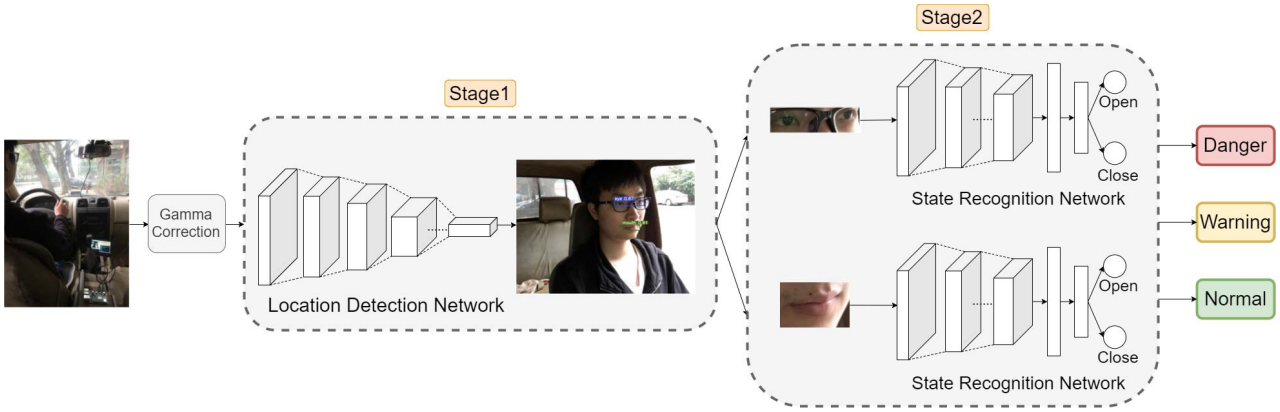


Fig. 2. Fatigue Driving Detection Method.

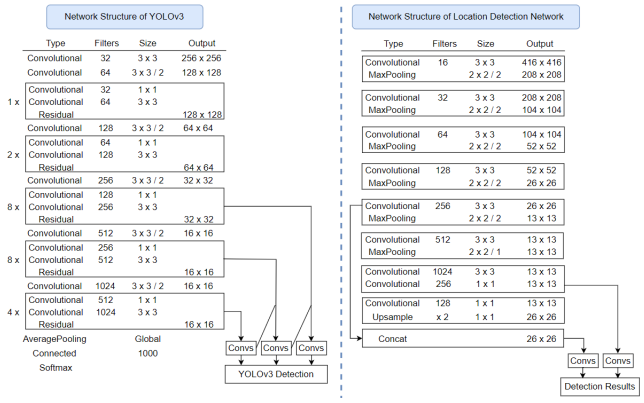


Fig. 3. YOLOv3 Network Structure compared with Location Detection Network structure.

are proposed for the implementation of fatigue detection. A driver fatigue criteria with three levels will be given to determine whether the driver is driving fatigued.

3.1 Location Detection Network

Location Detection Network is inspired by YOLOv3, which is an object detection algorithm based on CNN. As shown in Fig. 3, YOLOv3 uses convolutional layers and residual layers as the feature extraction network (backbone network) and uses a multi-scale detection method for different size object detection. For the proposed driver fatigue detection method, we focus on detecting two facial features (i.e., eyes and mouth) and we have to guarantee the detection speed and accuracy for the real-time driving condition. The Location Detection Network is designed with reference to the backbone network and multi-scale detection method of YOLOv3.

Location Detection Network is built with convolution layers and pooling layers, as shown in Fig. 3. We use 6 convolutional layers and 6 pooling layers as the feature extraction network. Using a smaller number of convolution layers is that we focus on two facial features extraction and can ensure that training with a smaller size of dataset is less prone to over-fitting problems. The residual layers are removed because the residual layers in the shallow network do not greatly optimize the accuracy but slow down the detection speed. The reason for adding the pooling layer

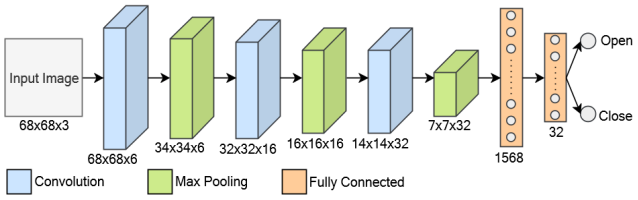


Fig. 4. State Recognition Network structure.

is that pooling is an image downsampling operation that speeds up the model computation.

A scale of detection result is obtained through two convolutional layers after the backbone network. The feature map here for detection has 32 times downsampling compared to the input image. As shown in Fig. 3, the input image size is 416 x 416 and the feature map size is 13 x 13. The feature map contains the basic object information, like edges, colors, primary position information, etc. The feature map of the ninth layer upsamples and then concatenate with the fifth layer feature map. The fusion feature map has 16 times downsampling compared to the input image. It contains the semantic information of the eyes and mouth regions on the driver's face.

3.2 State Recognition Network

The eyes and mouth regions are obtained by the Location Detection Network. State Recognition Network is used to determine the state of the mouth and eye regions, i.e., open or closed.

State Recognition Network structure is shown in Fig. 4. Eyes and mouth regions obtained from the Location Detection Network are in a different size. We resize the eyes and mouth regions to 68 x 68 and then feed them to the State Recognition Network. The convolution kernels' size is 3 x 3 and the step size is 1. To keep edge information of the image, padding is used to fill all zeros around the image after the first convolution. The maximum pooling layer has a convolution kernel size of 2 x 2 and a step size of 2. The features extracted by the previous convolutional layer and the maximum pooling layer are integrated by the fully connected layer. Eyes and mouth states are classified by the softmax layer finally.

3.3 Driver Fatigue Criteria

PERCLOS refers to the percentage of total eyes closure time in a certain period of time, which is the most effective indicator for visual fatigue detection. In this paper, N frames are collected through the video stream in the time period T, and the PERCLOS value is defined as:

$$PERCLOS = \frac{N_{eye}}{N} \times 100\%, \tag{4}$$

where N_{eye} represents the number of closed eyes frames in the N frames. People blink average ten to twenty times per minute. The normal blink time of people is between 0.2s and 0.3s, and the PERCLOS value is between 3.3% and 10%. If the blink time is between 0.5s and 3s, the driver can be regarded as a fatigued state and the PERCLOS value is between 16.7% and 100%. In this paper, in order to distinguish fatigue state and normal driving state accurately, when PERCLOS value is greater than 20%, the driver is considered to be in a state of fatigue.

During fatigue driving, the driver may fall asleep, his eyes will close for a long time. When driver yawns, his mouth will open for a long time. The continuous closed eyes frame time F_e and the continuous opened mouth frame time F_m are defined by the following formula:

$$F_e = (E_{end} - E_{start}) \times \frac{T}{N}, \tag{5}$$

$$F_m = (M_{end} - M_{start}) \times \frac{T}{N}, \tag{6}$$

where E_{end} represents the end number of continuous closed eye frames, E_{start} represents the starting frame of continuous closed eye frames, M_{end} represents the end frame of continuous closed mouth frames, M_{start} represents the starting frame of continuous closed mouth frames and $\frac{T}{N}$ represents the time interval between the every two selected frames.

In this paper, the driver's fatigue state is divided into three levels. The first type is the danger level, where the continuous eye-closing time is more than three seconds. When the driver's eyes are closed for more than three seconds, they will unable to respond correctly when an accident occurs. The system should remind the driver to pay attention to road conditions. The second type is the warning level, where the PERCLOS value exceeds the threshold value of 20% or the continuous eye-closing time is between 0.5s and 3s or the continuous mouth opening time reaches 4s or more. The driver may be fatigued. The driver should pay attention to have a rest and avoid fatigue driving. The third type is the normal level. The driver is driving normally.

4. EXPERIMENTS

In this section, we will first introduce the training data set, which consists of the YawDD data set and the self-build data set. Second, we compare the accuracy and speed on the Location Detection Network with existing methods on the computer. And we show the State Recognition Network has good classification performance after training. Finally, we evaluate the performance of the proposed driver fatigue detection method on the embedded device Raspberry Pi 4.

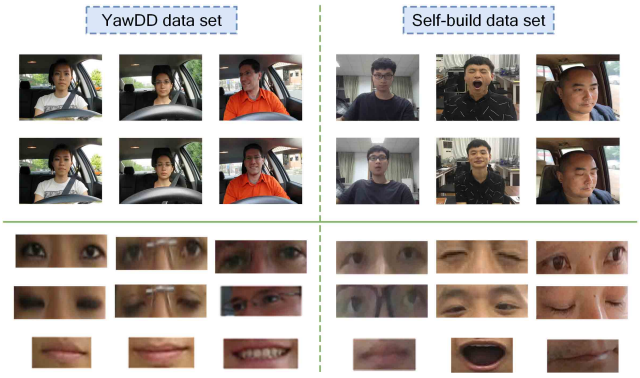


Fig. 5. YawDD data set and self-build data set.

4.1 Data Collection

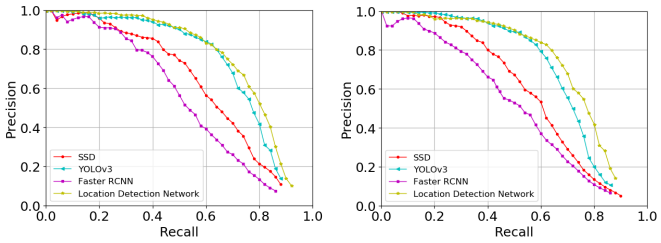
The proposed driver fatigue detection method is consisted of Location Detection Network and State Recognition Network. We need to collect samples to train the two networks. As shown in Fig. 5, the training data set is composed of the YawDD data set and the self-built data set. The YawDD data set (Abtahi et al., 2014) is a driver video data set with 351 video sequences, which is used to design yawn detection algorithms and test models. We select 20 videos from the YawDD data set to convert into frame images. The self-built data set contains 1020 images, which is collected from 10 volunteers. We use *labelImg*, an open-source graphical image annotation tool to mark the driver's eyes and mouth position information. The frame images with eyes and mouth position information are used for the Location Detection Network training. Eyes and mouth regions are used to train the State Recognition Network.

4.2 Experiments on Computer

The experiment uses the CPU model Intel(R) Core(TM) i5-8300H, a main frequency of 2.3 GHz, a memory of 16 G, and the GTX1060 GPU as the experimental platform. In our experiments, the pytorch platform is used to build the convolutional neural networks.

We evaluate the Location Detection Network's detection accuracy and detection speed with Faster RCNN, SSD, and YOLOv3. Fig. 6 shows the result of PR (precision-recall) curve on the YawDD data set and the self-build data set. The proposed Location Detection Network has obvious advantages in eyes and mouth regions position detection compared with Faster RCNN and SSD, slightly better than YOLOv3. Combined with Table 1, the Location Detection Network detection speed is much faster than the other three methods. Fig. 7 shows the detection results by Location Detection Network on the two data sets.

Eyes and mouth regions, which are consisted by the YawDD data set and self-build data set, are set to train the State Recognition Network. we set the batch size to 64 and the learning rate is set to 0.001. 100 epochs are trained on the data set in this paper. Fig. 8(a) shows



(a) (b)

Fig. 6. P-R curves on the two data sets: (a) YawDD data set. (b) self-build data set.

Table 1. Average detection speed on YawDD data set and self-build data set

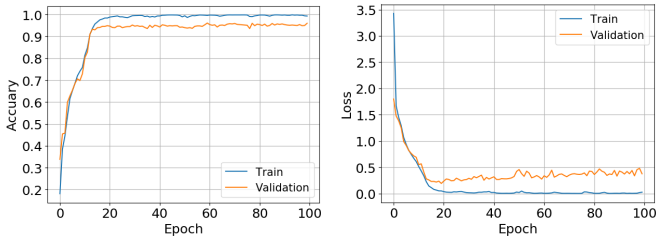
Network	Frames Per Second
SSD	17
Faster RCNN	5
YOLOv3	22
Location Detection Network	41

(a) (b) (c)

(d) (e) (f)

Fig. 7. Detection results of the Location Detection Network on the two data sets: (a-c) self-build data set, (d-f) YawDD data set.

that the classification accuracy of the training set is 98.3% compared with 93.83% on the validation set, which can be seen that the State Recognition Network has not been over-fitting. The training loss and validation loss are shown in Fig. 8(b). The State Recognition Network model with the smallest loss on the validation set is saved as the final training model.



(a) (b)

Fig. 8. Training results of the State Recognition Network. (a) model accuracy on training set and validation set. (b) model loss on training set and validation set.

4.3 Experiments on Raspberry Pi 4

Most of the object detection algorithms based on convolutional neural network are deployed on the computer with multiple GPUs. The platform has the mighty computing power and can deploy large-scale network models. However, due to the relatively large size, it is difficult to achieve industrial applications even if real-time object detection is achieved on the platforms. Therefore, deploying model to the embedded device is important for realizing intelligent transportation.

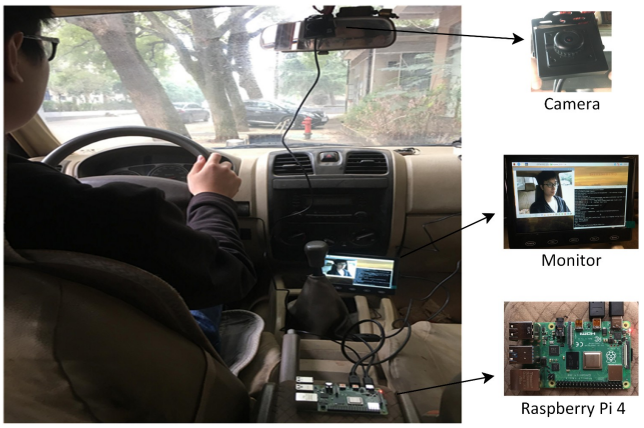


Fig. 9. Raspberry Pi 4 with driver fatigue detection model in the real driving environment.

The embedded device Raspberry Pi 4 is popular for its low-cost, portability, and connectivity features. The Raspberry Pi 4 has 4 GB RAM and 1.5 GHz processing speed, which provides desktop performance comparable to entry-level x86 PC systems. We choose Raspberry Pi 4 to build the proposed driver fatigue detection model to verify the real-time performance and accuracy.

Table 2. Average time cost on Raspberry Pi 4

Process	Time Cost(ms)
Frame Image Preprocessing	4.8
Stage1(Location Detection Network)	65.8
Stage2(State Recognition Network)	25.7
Total	96.3

In Fig. 9, the Raspberry Pi 4 with a camera and a monitor is put to a car. The proposed driver fatigue method has three processes, including the frame image preprocessing, facial feature extraction (stage1) and facial feature recognition (stage2). As shown in Table 2, we record the average processing time cost of the three processes on Raspberry Pi 4. Location Detection Network costs much of the time because it needs many computations. According to Table 2, analyzing a frame image costs 96.3ms. Therefore, the proposed driver fatigue detection speed achieves 10.4 FPS on Raspberry Pi 4, which meets real-time requirements for embedded device.

To verify the accuracy of the proposed method, we record the eyes state and mouth state in the selected frames on the Raspberry Pi 4. It shows that the average accuracy of the records for driver fatigue detection is 94.7%, which meets our requirements in terms of fatigue detection accuracy. Fig. 10 shows the record results. When the state is equal to 1, it means the eyes or the mouth is open. When the state is equal to 0, it means the eyes or the mouth is closed. Fig. 10(a) indicates that the driver is in a normal driving condition. The number of closed eyes frames for the driver is about three frames. The warning level is shown in

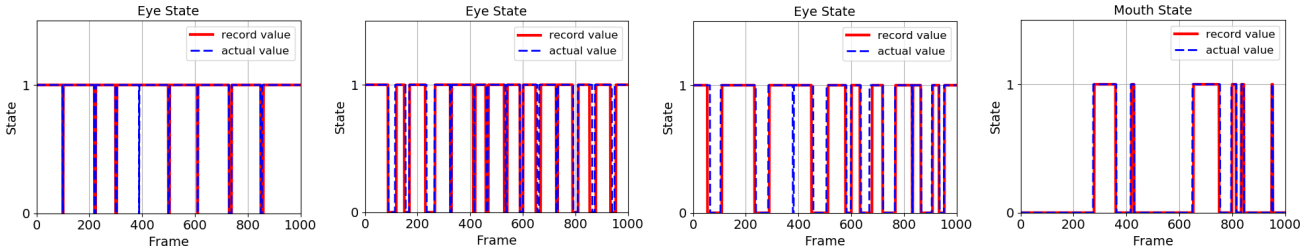


Fig. 10. Eye state and mouth state with different fatigue level on Raspberry Pi 4: (a) eye state with normal level. (b) eye state with warning level. (c) eye state with danger level. (d) mouth state with warning level.

Fig. 10(b) and Fig. 10(d). Fig. 10(b) presents that a total of 216 frames in 1000 frames the eyes are closed, which means the PERCLOS value is greater than 20%. Fig. 10(d) shows that the continuous opened mouth frame number is 81, which means the yawning time of the driver is about eight seconds. The driver at the warning level needs to have a rest and prevent fatigue driving. The danger level is shown in Fig. 10(c). The continuous eye-closing frame number is 56, which means the continuous eye-closing time of the driver is about six seconds and the driver has fallen asleep. It's very dangerous for the driver and should take appropriate measures to assist the driver drive safely.

5. CONCLUSION

In this paper, a real-time driver fatigue detection method is proposed to ensure driving safety. Gamma correction is introduced first for improving the frame image contrast. In order to achieve high detection accuracy and fast detection speed, a two-stage convolutional neural network method is proposed for driver fatigue detection. The method includes a Location Detection Network and a State Recognition Network. Experiments on the computer show that the Location Detection Network has a significant superiority in detection speed over existing methods and the State Recognition Network has 93.83% classification accuracy. Experiments on the Raspberry Pi 4 indicate that the proposed driver fatigue detection model achieves 10.4 FPS detection speed with 94.7% accuracy, which has great reference value for industrial applications. In the future work, we will fuse different kinds of features, such as physiological parameters, vehicle data, and facial features for driver fatigue detection.

ACKNOWLEDGEMENTS

This work is partially supported by National Natural Science Foundation of China (Grant Nos. 61873353 and 61672539).

REFERENCES

Abtahi, S., Omidyeganeh, M., Shirmohammadi, S., and Hariri, B. (2014). Yawdd: A yawning detection dataset. In *Proceedings of the 5th ACM Multimedia Systems Conference*, 24–28. ACM.

Al-Sultan, S., Al-Bayatti, A.H., and Zedan, H. (2013). Context-aware driver behavior detection system in intelligent transportation systems. *IEEE transactions on vehicular technology*, 62(9), 4264–4275.

Alioua, N., Amine, A., and Rziza, M. (2014). Driver's fatigue detection based on yawning extraction. *International journal of vehicular technology*, 2014.

Arnold, L.S. and Tefft, B.C. (2015). Prevalence of self-reported drowsy driving, united states: 2015.

Asbridge, M., Hayden, J.A., and Cartwright, J.L. (2012). Acute cannabis consumption and motor vehicle collision risk: systematic review of observational studies and meta-analysis. *Bmj*, 344, e536.

Fu, R. and Wang, H. (2014). Detection of driving fatigue by using noncontact emg and ecg signals measurement system. *International journal of neural systems*, 24(03), 1450006.

Lal, S.K. and Craig, A. (2002). Driver fatigue: electroencephalography and psychological assessment. *Psychophysiology*, 39(3), 313–321.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., and Berg, A.C. (2016). Ssd: Single shot multibox detector. In *European conference on computer vision*, 21–37. Springer.

Mandal, B., Li, L., Wang, G.S., and Lin, J. (2016). Towards detection of bus driver fatigue based on robust visual analysis of eye state. *IEEE Transactions on Intelligent Transportation Systems*, 18(3), 545–557.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 779–788.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, 91–99.

Sandberg, D. and Wahde, M. (2008). Particle swarm optimization of feedforward neural networks for the detection of drowsy driving. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, 788–793. IEEE.

Sikander, G. and Anwar, S. (2018). Driver fatigue detection systems: A review. *IEEE Transactions on Intelligent Transportation Systems*, 20(6), 2339–2352.

Simon, M., Schmidt, E.A., Kincses, W.E., Fritzsche, M., Bruns, A., Aufmuth, C., Bogdan, M., Rosenstiel, W., and Schrauf, M. (2011). Eeg alpha spindle measures as indicators of driver fatigue under real traffic conditions. *Clinical Neurophysiology*, 122(6), 1168–1178.

Zhang, C., Wang, H., and Fu, R. (2013). Automated detection of driver fatigue based on entropy and complexity measures. *IEEE Transactions on Intelligent Transportation Systems*, 15(1), 168–177.