American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS)
ISSN (Print) 2313-4410, ISSN (Online) 2313-4402

http://asrjetsjournal.org/

Real-time Fatigue Driving Recognition System Based on Deep Learning and Embedded Platform

Lei Geng[a], ZhiQiang Hu[b*], ZhiTao Xiao[c]

[a,b,c]*School of Electronic and Information Engineering, Tianjin Polytechnic University, Tianjin 300387, China*

[a]*Email: 15620730094@163.com*

Abstract

The frequent occurrence of automobile traffic accidents seriously threatens the safety of human life and property. Therefore, fatigue driving detection has important social value and research significance. In consideration of the market demand of intelligent assistant driving system, we design a real-time driver fatigue detection system based on deep learning and ARM platform, which uses Samsung 6818A53 series ARM as the driver fatigue real-time detection platform. In order to reduce the interference caused by the change of light and the occlusion of sunglasses in the actual driving environment, the driver's face image is captured by USB infrared camera. Firstly, face detection and alignment are carried out by multi-task cascaded convolutional neural network; Then the eye region is obtained according to geometric relationship between the feature points; Moreover, the driver's eye state is identified by Convolutional Neural Network (CNN); Finally, fatigue judgment is made based on PERCLOS criterion. The system has been tested in the experimental simulation environment and the actual driving environment. The experimental results show that detection speed of the system can reach more than 20 frames per second, which meets the requirement of real-time detection.

Keywords: Fatigue Driving; Embedded Platform; CNN; PERCLOS; Real-time Detection.

1. Introduction

With the rapid development of global economy and large-scale construction of road traffic facilities, transportation industry has become one of the fastest growing industries in recent years. However, while automobiles bring convenience to human life, road traffic accidents occur frequently. The research shows that if the driver can make an emergency response in advance before the accident happens, the traffic accident can be avoided to a large extent.

----

* Corresponding author.

Therefore, the design of a safety assistant driving system which can automatically identify the driver's fatigue state is of great social significance and economic value to ensure the safety of life and property of road personnel, which is also in line with the current trend of the development of safety and intelligence in the automotive industry. At present, the mechanisms in fatigue driving detection has been categorized into three broad methods [1,2], including physiological-based, vehicle-based, and vision-based approaches.

The detection of human physical signals plays an important part in modern medical treatment and research. For first method, researchers use the measurement of physiological activities to detect fatigue. These signals characteristics are acquired from various sensors attached to the body in some particular areas such as head (analyzing electroencephalogram-EEG [3]), chest (obtaining electrocardiograph-ECG [4]), muscle (acquiring electric signals from muscle cells which called electromyogram-EMG) [5], fingers, and eyes (detecting the change of eyes' resting potential and generating electrooculogram-EOG) [6], etc. However, these approaches are invasive, unfriendly, and require contact with the driver's body. Sometimes it can receive an accurate detection result, but not very practical in most cases. The method relying on vehicle operating status information is used to measure the behaviors of vehicles, such as speed, steering wheel angles [7], and lane departure detection [8], etc. But these methods are affected by driving conditions, driving experience, driving habit, vehicle type. These parameters are greatly influenced by drivers' individual differences, especially road conditions. A third way to identify the drowsy driver is to apply cameras and machine vision algorithms for capturing and analyzing visual information while driving, such as blink frequency, yawning [9], head movement, gaze direction [10], and especially facial expression [11], which can provide observable cues in changing facial features. A significant evaluation index called "PERCLOS [12]", referring to the percentage of the eye closing time over a specific time period, is widely used to identify fatigue. The third method has been well received, owing to its non-intrusive, low-cost and friendly peculiarity in monitoring the driving state, and the visual behavior of drowsy drivers is significantly different compared to sober-minded drivers.

In recent years, deep learning [13] algorithms represented by Convolutional Neural Networks (CNNs) have been successfully applied in substantial visual tasks: target categorization, object segmentation, target detection, etc, due to its strong feature extraction ability and robustness. Many excellent CNN models are also advanced and have achieved good results such as AlexNet [14], VGG-Net [15], ResNet [16], DenseNet [17], SE-Net [18], etc. In addition, some techniques are used to optimize network and heighten learning ability. Inception architectures [19] show that the network can achieve competitive accuracy by embedding multi-scale processes in its modules. Batch normalization operation, which is brought forward by Ioffe, S and his colleagues (2015) [20], not only can speed up the convergence of the model, but also abate the gradient dispersion of deep convolution network and prevent over-fitting. Transfer learning idea [21] was raised to initialize the target dataset network weights, using the pre-trained CNN model from the big dataset to fine-tune the network weights of small-scale target dataset.

Lately, deep learning with visual features to judge fatigue driving has become a popular trend. However, based on this method, there are still some problems to be solved, which are as follows:

- Lack of professional data sets for fatigue driving detection.
- Great interference in extracting the driver's visual features is caused by the change of light during actual driving.
- When the driver wears myopia glasses or sunglasses, the eye area will be occluded.
- The deep learning model occupies a large amount of memory, while the memory of embedded devices is limited. A large number of real-time data processing challenges the application of fatigue detection technology.

In this paper, deep learning is introduced into the field of fatigue detection. The main contributions of this work are as follows:

- On one hand, using the USB infrared camera to collect driver's face image can reduce the interference caused by the change of light. On the other hand, a clearer eye image can be got through sunglasses.
- Face detection and feature point location are carried out by multi-task cascaded convolutional neural network (MTCNN). Furthermore, eye region is obtained according to the geometric relationship between eye feature points.
- Eye state is classified by well-designed convolutional neural network. Besides, fatigue driving is judged by PERCLOS criterion.
- The fatigue detection algorithm is transplanted to ARM platform for real-time monitoring of driving state.

Compared to traditional methods, a higher classification accuracy is achieved by our system.

2. Methods

In this section, we mainly describe our approach to detect fatigue driving.

2.1 Software platform and hardware platform

The system uses Flying Ok6818-C development board as the embedded development platform, which is shown in Figure 1. The development platform is based on ARM Cortex-A53 and adopts core board combined with floor structure. The main model of core board is FET6818-C and the CPU model is S5P6818XC0-LA40. USB infrared camera is connected to Ok6818-C development board. So, a real-time video input channel is built. The collected images are processed by image algorithm on Ok6818-C embedded platform. Moreover, the driver's monitor screen is output to the display screen through MIPI interface in real time, which is convenient for observing the test results. The hardware structure of the system is shown in Figure 2.

The programming environment of this system is 64-bit Ubunt operating system, version 14.04. The kernel version is Linux 3.4.39. The development tools include QT, Tslib, Opencv, Cmake, Code::Blocks, cross-platform cross-compiler, etc. NCNN is a highly optimized forward-computing neural network framework of high-performance for mobile phones, which does not require third-party dependence and is cross-platform. In addition, when running on mobile phones, the speed of CPU is faster than all known open source frameworks. Thus, NCNN is selected as the deep learning framework. Based on NCNN, we can easily transplant deep learning algorithms to embedded platforms, mobile phones and other mobile terminals and implement them efficiently.

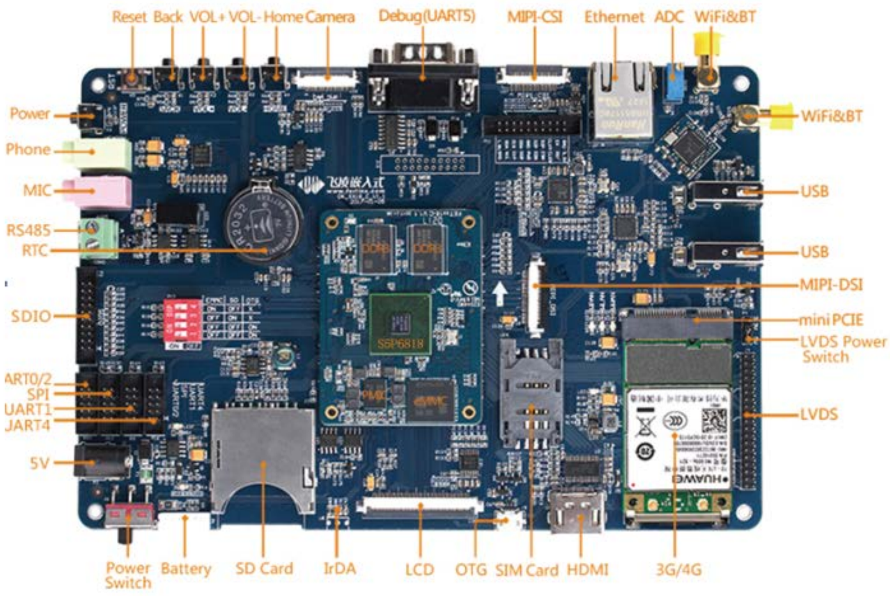

Figure 1: Ok6818-C Development board

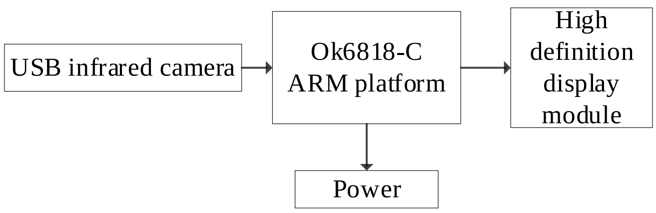

Figure 2: The hardware structure of fatigue driving system

2.2 Extracting the Region of Interest

Currently, detecting fatigue relying on driver's face images has become the mainstream. The location of driver's eyes is the key in drowsiness recognition. Due to various postures, illuminations and occlusions, it's challenging to detect and align face in an unconstrained environment. A deep-cascaded multi-task framework that utilizes their intrinsic connections to improve performance is proposed by Zhang, K and his colleagues (2016) [22]. Particularly, this framework adopts a cascaded structure with a deep three-stage convolution network carefully designed to predict face and landmark location in a coarse-to-fine manner, which also yields good results when detecting side faces.

As shown in Figure 3, we apply Multi-task cascaded convolutional network (MTCNN) to calibrate feature points of eyes, and then the corresponding image sequences of eyes are extracted from each frame according to the geometric relations between the feature points, which are shown as follows:

$$\begin{cases} d = x_B - x_A \\ w = d \times 0.6 \\ h = w \times 0.8 \\ x_C = x_A - w \times 0.5 \\ y_C = y_A - h \times 0.5 \end{cases} \tag{1}$$

Where (x_A, y_A) and (x_B, y_B) denote the pixel coordinates of eye feature point A, B, respectively. Point C, whose pixel coordinate is (x_C, y_C) , refers to the upper-right vertex of right eye area. And d, w as well as h are respectively the horizontal distance between point A and B, the width and height of the right eye area. See Figure 3 for more details.

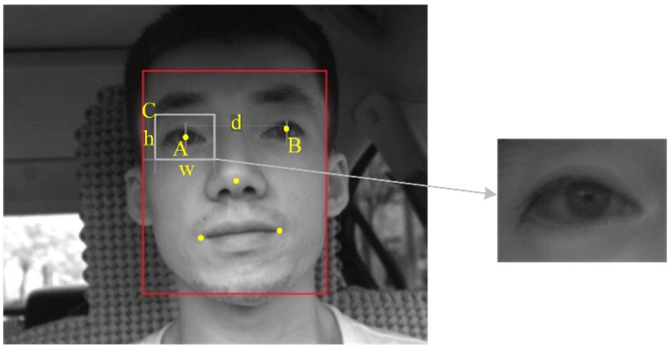

Figure 3: Extracting the region of interest by MTCNN

2.3 Eye state recognition network

CNN is generally composed of convolution layer, activation layer, pooling layer and fully connected layer. After the data are input into CNN, the features are extracted through the convolution layer and the pooling layer. The extracted features are activated by the activation layer. Finally, the full connection layer performs dimension transformation, feature combination and semantic mapping of the highly abstract features. The error is calculated by loss function. Then the gradient is propagated by the back propagation algorithm and the gradient is updated by the gradient updating algorithm to achieve the optimal solution. The convolution layer is the core of CNN. Different features of input data are extracted by convolution kernels with different sizes. These convolution kernels are trainable weight matrix.

Generally speaking, when the convolution layer extracts a feature of the target, it is necessary to arrange a pooling layer between two adjacent convolution layers. The pooling layer is actually a statistical function. There are two functions of the pooling layer: one is to reduce the amount of data to be processed in the next layer, that is, dimension reduction. The other is to prevent over-fitting of the network; At the end of CNN, there is generally a Fully Connected Layer. Full connection means that all neurons in the front layer network are connected to all neurons in the next layer. The purpose of fully connected layer is to map the distributed feature representations learnt from the previous layers to the sample label space; Then the loss function is used to regulate the learning process; Finally, the classification prediction of the object is given. In fact, the fully connected layer is equivalent to the traditional multi-layer perceptron; in CNN, in order to achieve multi-classification, it is necessary to introduce a non-linear structure. Activation layer maps the output of convolution layer nonlinearly. It is more likely that the sparse feature can be linearly separable; Besides, the network can better fit more complex polynomial functions.

In this paper, CNN is used to identify the two states of the driver's eye. The recognition network is named ESRNet (Eye State Recognition Network).

The network consists of five layers: two convolution layers, two pooling layers and one fully connected layer. A dropout layer is added at the end of ESRNet to avoid over-fitting in the training process. The structure of ESRNet is shown in Figure 4. C1 and C2 represent convolution layer. S1 and S2 represent pooling layer and F1 represents fully connected layer. The specific parameters of each layer of ESRNet and the size of the feature map output from each layer are shown in Table 1.

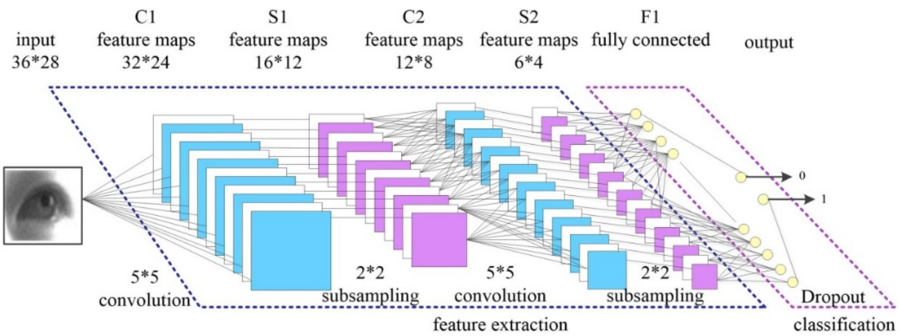

Figure 4: Network architecture of ESRNet

Table 1: Layer parameters of ESRNet

Layer name	Layer type	Related parameters	Feature-map size
C1	Convolution	5×5, 32, stride 1	32×24
S1	Pooling	2×2, max pooling, stride 2	16×12
C2	Convolution	5×5, 64, stride 1	12×8
S2	Pooling	2×2, max pooling, stride 2	6×4
F1	Fully connected	2-d	\
Dropout	Dropout	dropout-ratio 0.5	\

2.4 Flow of fatigue detection software

The software implementation flow of fatigue detection algorithm is as follows:

- Cross-compile Opencv and transplant it to the / lib directory of ARM development board root. Cross-compile NCNN source code and generate NCNN library: libncnn.a, which is placed in the project directory.
- Create QT widget application based on Qwidget class; Video_device class manages cameras; Eye_states class realizes eye state recognition; Mouth_states class realizes mouth state recognition; Mtcnn class realizes face and feature detection; Widget class realizes fatigue detection and nodding warning based on PERCLOS standard.
- The yuv-type video frames captured by USB cameras are converted to uchar* type and then to cv::Mat type. Opencv is used for image processing. After processing, turn to Qimage type and display it on the label control of QT.
- The fatigue detection delay is 5 seconds and the fatigue threshold is set. If the number of closed eye frames in 5 seconds is larger than the threshold, the fatigue will be judged and the warning will be given in time. In addition, the head height average is counted when driving normally. If the ratio of the current head height to the average value is larger than the threshold, the head is regarded as nodding and an alarm is issued.

3. Experiment and results analysis

In this section, experiments will be designed to verify and evaluate the effectiveness of the fatigue driving recognition algorithm. The training platform of CNN is GTX1080Ti series GPU + Ubuntu 16.04 operating system + Intel Corei7 processor + 12G memory. By fine-tuning the network super-parameters, the network model with high accuracy can be obtained efficiently and quickly.

3.1 TJPU-FDD dataset

The success of deep learning in the field of object recognition is based on abundant experimental data distributed independently. Due to the lack of public datasets, twenty-six subjects, which all had a current driver’s license, participated in the driving simulations experiment. Under the condition of wearing myopia glasses, wearing sunglasses, and not wearing glasses, respectively, the subjects simulate two kinds of driving states (fatigue and normal) on the passenger seat because fatigue driving behaviour is dangerous and illegal. In order to reduce the impact of illumination changes and get clear eye images when drivers wear sunglasses, we use infrared camera with filters to capture face videos at 30 fps with a resolution of 1920×1080 as experimental dataset named “TJPU-FDD”, which is divided into two classes: fatigue driving and normal driving. All samples from TJPU-FDD consist of 500 video clips, each lasting about 6 seconds. In our samples labeled fatigue, the driver blinks more slowly and eyes close for a long time. In addition, the driver cannot keep eye open like normal. And the phenomena such as slow eyeball movement, yawning, eyelid closure occur.

3.2 Face Detection and Alignment Experiment

A total of 31501 frames were randomly selected from 170 videos (30 wearing sunglasses, 68 wearing myopic glasses and 72 wearing no glasses) in TJPU-FDD, 20 videos (10 wearing myopic glasses and 10 wearing no glasses) in ZJU data set, 100 videos (47 wearing myopic glasses and 53 wearing no glasses) in Brain4Cars data set for face detection and alignment. In the experiment, the size of all frames is adjusted to 640×480.

Table 2 shows the results of locating eye region by MTCNN on the test set. The average accuracy of extracting eye region can reach 98.36%. The experimental results show that when the driver wears sunglasses and the side face angle is not too big, it can also achieve good detection results. Moreover, in a more stable light environment, face detection and alignment based on MTCNN for color images can achieve higher detection accuracy, but it is difficult to obtain clear eye regions. Light changes will increase the difficulty of face detection and alignment; Capturing face images by infrared camera can improve the robustness of face detection and alignment algorithm and solve the problems caused by light changes and sunglasses occlusion. Fig. 5 shows the detection results of some samples.

Table 2: The results of locating the eye region by MTCNN on the test set.

Dataset	Eye shelter type	Area	Number of video frames	Number of video frames located accurately	Accuracy/%
TJPU-FDD	No glasses	Left eye	13858	13654	98.53
		Right eye	13858	13653	98.52
	Myopia glasses	Left eye	12240	11889	97.13
		Right eye	12240	11900	97.22
	sunglasses	Left eye	5403	5064	96.73
		Right eye	5403	5224	96.69
ZJU	No glasses	Left eye	2064	2051	99.35
		Right eye	2064	2052	99.41
	Myopia glasses	Left eye	2037	2011	98.74
		Right eye	2037	2013	98.82
Brain4Cars	No glasses	Left eye	7506	7441	99.13
		Right eye	7506	7444	99.18
	Myopia glasses	Left eye	7524	7437	98.85
		Right eye	7524	7439	98.87

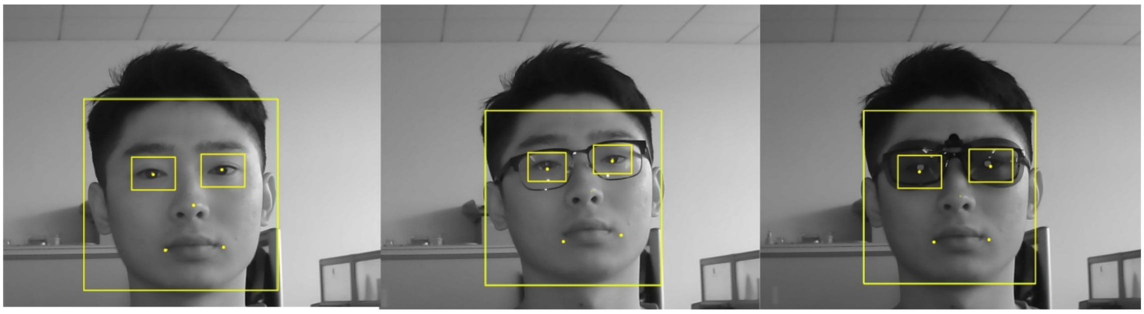

Figure 5: The diagram of face detection and alignment results

3.3 Experimental results and analysis of fatigue driving detection

Tests are conducted by using the ESRNet model generated after 50000 iterations. The experimental data sets are TJPU-FDD and Brain4Cars. The experimental results show that ESRNet model has a high recognition accuracy of eye state in different environments. But the accuracy when wearing glasses is lower than other situations. It is mainly due to the interference of optic reflection on the recognition of eye state. The average recognition accuracy of eye state is 98.9%. Test results are shown in Table 3.

Table 3: The eye state recognition results of ESRNet model on the test set

Dataset	Eye shelter type	Eye state	Number of test samples	Number of wrong recognition frames	Accuracy
TJPU-FDD	No glasses	Open	3251	8	99.75%
		Close	2138	6	99.71%
TJPU-FDD	Myopia glasses	Open	3403	37	98.91%
		Close	2755	79	97.15%
TJPU-FDD	sunglasses	Open	3369	66	98.04%
		Close	2840	113	96.02%
Brain4Cars	\	Open	3625	11	99.70%
		Close	2184	23	98.95%

Usually, the PERCLOS value is less than that of fatigue when the driver is in normal driving environment. During test, we analyze a certain length of image sequence and identify driver state as "clear" or "fatigue". The PERCLOS value is counted for a period of time. When the PERCLOS value is less than the pre-set threshold, it is judged to be normal. Otherwise, it is judged to be fatigue and the system issues a dangerous warning. In addition, when the driver's head height is less than the pre-set threshold, the system alerts the driver to look ahead. The experimental results show that the fatigue detection system designed in this paper has a stable early warning effect and has a high computing speed. On the embedded platform mentioned above, the processing speed of the algorithm can reach more than 20 FPS, which meets the real-time detection requirements. The test results on the TJPU-FDD test set are shown in Table 4, where N represents the number of error Recognition samples. Fig 6 shows the screenshots of monitoring the driving state on the embedded platform.

Table 4: The test results of fatigue driving recognition system on the TJPU-FDD test set

Data distribution of test set	Recognition results		N	Precision /%	Recall rate /%	F1 score /%	Average test time /ms
	Fatigue	Normal					
Fatigue (48)	44	4	7	95.56	89.58	92.47	41.58
Normal (55)	3	52					

Figure 6: The screenshots of real-time fatigue driving detection on embedded platform

4. Conclusion

In this paper, a real-time fatigue driving detection system based on deep learning and embedded platform is designed. Infrared camera is used to collect driver's face image to reduce the interference caused by light change and sunglasses occlusion. Firstly, MTCNN is used to detect face and locate eye feature points. Then, the geometric relationship between feature points is used to obtain eye region. Moreover, CNN is used to recognize driver's eye state. Based on the idea of model compression, we design a small recognition network, which accounts for only 1.3M of memory. Finally, the fatigue judgment is made based on PERCLOS standard. In order to apply AI algorithm to fatigue driving detection task in embedded and mobile platforms, NCNN is chosen as the deep learning framework of the recognition system. The system has been tested in the experimental simulation environment and the actual driving environment. The real-time detection speed of more than 20 FPS has been achieved, which shows the effectiveness and feasibility of the proposed method.

5. Recommendations

Our work has yielded good results for fatigue driving detection. However, there are still some shortcomings in this system, which make it difficult to apply the system in practice. In the future work, the following aspects can be further studied and improved:

- To improve the image acquisition system aimed to reduce the interference of high-brightness reflecting points on eye region extraction.
- To combine with more visual fatigue parameters, such as driver's gaze, yawning and face orientation so that the driving state can be evaluated more comprehensively.
- To develop a face detection and alignment algorithm with stronger robustness to face posture.
- To realize real-time detection of driver's driving state by means of hardware and software, such as constructing heterogeneous chips, fixing point and pruning network.

References

[1] Pratama B G, Ardiyanto I, Adji T B. A review on driver drowsiness based on image, bio-signal, and driver behavior[C]//2017 3rd International Conference on Science and Technology-Computer (ICST). IEEE, 2017: 70-75.

[2] Triyanti V, Iridiastadi H. Challenges in detecting drowsiness based on driver’s behavior[C]//IOP Conference Series: Materials Science and Engineering. IOP Publishing, 2017, 277(1): 012042.

[3] Mardi Z, Ashtiani S N M, Mikaili M. EEG-based drowsiness detection for safe driving using chaotic features and statistical tests[J]. Journal of medical signals and sensors, 2011, 1(2): 130.

[4] Jung S J, Shin H S, Chung W Y. Driver fatigue and drowsiness monitoring system with embedded electrocardiogram sensor on steering wheel[J]. IET Intelligent Transport Systems, 2014, 8(1): 43-50.

[5] Fatourechi M, Bashashati A, Ward R K, et al. EMG and EOG artifacts in brain computer interface systems: A survey[J]. Clinical neurophysiology, 2007, 118(3): 480-494.

[6] Gasser T, Sroka L, Möcks J. The transfer of EOG activity into the EEG for eyes open and closed[J]. Electroencephalography and clinical neurophysiology, 1985, 61(2): 181-193.

[7] Li Z, Li S, Li R, et al. Online detection of driver fatigue using steering wheel angles for real driving conditions[J]. Sensors, 2017, 17(3): 495.

[8] Ma J, Murphey Y L, Zhao H. Real time drowsiness detection based on lateral distance using wavelet transform and neural network[C]//2015 IEEE symposium series on computational intelligence. IEEE, 2015: 411-418.

[9] Alioua N, Amine A, Rziza M. Driver’s fatigue detection based on yawning extraction[J]. International journal of vehicular technology, 2014, 2014.

[10] Choi I H, Kim Y G. Head pose and gaze direction tracking for detecting a drowsy driver[C]//2014 international conference on big data and smart computing (BIGCOMP). IEEE, 2014: 241-244.

[11] Karchani M, Mazloumi A, Saraji G N, et al. Presenting a model for dynamic facial expression changes in detecting drivers' drowsiness[J]. Electronic physician, 2015, 7(2): 1073.

[12] Dinges D F, Grace R. PERCLOS: A valid psychophysiological measure of alertness as assessed by psychomotor vigilance[J]. US Department of Transportation, Federal Highway Administration, Publication Number FHWA-MCRT-98-006, 1998.

[13] LeCun Y, Bengio Y, Hinton G. Deep learning[J]. nature, 2015, 521(7553): 436.

[14] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[C]//Advances in neural information processing systems. 2012: 1097-1105.

[15] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.

[16] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[17] Huang G, Liu Z, Van Der Maaten L, et al. Densely connected convolutional networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 4700-4708.

[18] Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7132-7141.

[19] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 1-9.

[20] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[J]. arXiv preprint arXiv:1502.03167, 2015.

[21] Oquab M, Bottou L, Laptev I, et al. Learning and transferring mid-level image representations using convolutional neural networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 1717-1724.

[22] Zhang K, Zhang Z, Li Z, et al. Joint face detection and alignment using multitask cascaded convolutional networks[J]. IEEE Signal Processing Letters, 2016, 23(10): 1499-1503.