



HAL
open science

Driver Fatigue Detection Using Multitask Cascaded Convolutional Networks

Xiaoshuang Liu, Zhijun Fang, Xiang Liu, Xiangxiang Zhang, Jianrong Gu, Qi Xu

► **To cite this version:**

Xiaoshuang Liu, Zhijun Fang, Xiang Liu, Xiangxiang Zhang, Jianrong Gu, et al.. Driver Fatigue Detection Using Multitask Cascaded Convolutional Networks. 2nd International Conference on Intelligence Science (ICIS), Oct 2017, Shanghai, China. pp.143-152, 10.1007/978-3-319-68121-4_15 . hal-01820907

HAL Id: hal-01820907

<https://inria.hal.science/hal-01820907>

Submitted on 22 Jun 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Driver Fatigue Detection Using Multitask Cascaded Convolutional Networks

Xiaoshuang Liu¹, Zhijun Fang¹, Xiang Liu¹, Xiangxiang Zhang¹, Jianrong Gu², Qi Xu³

¹ School of electronic & electric engineering, Shanghai University of Engineering Science, Shanghai, China

² Information Center, Shanghai University of Engineering Science, Shanghai, China

³ College of Information Engineering, Shanghai Maritime University, Shanghai, China
{xslu, zjfang, xliu, M020215124, rong154}@sues.edu.cn,
qixu@shmtu.edu.cn

Abstract. Driving fatigue is one of the main reasons of traffic accidents. In this paper, we apply the multitask cascaded convolutional networks to face detection and alignment in order to ensure the accuracy and real-time of the algorithm. Afterwards another convolution neural network (CNN) is used for eye state recognition. Finally, we calculate the percentage of eyelid closure (PERCLOS) to detect the fatigue. The experimental results show that the proposed method has high recognition accuracy of eye state and can detect the fatigue effectively in real-time.

Keywords: Fatigue detection, Cascaded convolutional neural network, Face alignment, PERCLOS.

1 Introduction

Along with the development of the auto industry and the transportation industry, traffic accidents have caused great loss in the property and damage to the society. Amongst these traffic accidents more than 20% of these traffic accidents are caused by fatigue driving. Safe driving has become a hot issue in today's society, Therefore, it is of great significance to develop a real-time and accurate fatigue detection system to send fatigue warning information when the driver is tired, which can effectively reduce the occurrence of traffic accidents.

At present, fatigue testing contains three main directions. First, fatigue detection based on the vehicle state detection method, mainly through the turning angle, vehicle driving speed to detect whether the driver fatigue, this method is subject to external interference, the detection accuracy has a greater impact. Second, based on driver's physiological information [7], mainly by detecting the driver's heart rate, pulse and other physiological signals to determine whether the driver is in a state of fatigue, This method requires the driver to carry a lot of testing equipment, very cumbersome, and the driver has a great interference. Third, fatigue detection methods based on computer vision [6][8-10], this method is a non-intrusive way, the facial features can be calculated

by analyzing the changes of facial expression, such as eye closure duration, yawning and so on.

In the fatigue detection, driver face detection and alignment are important. The multitask cascaded convolutional networks to face detection and alignment [1] has proven to be an effective method. Another very important step is the detection of human eye state. Compared to the traditional active infrared radiation method [2], normal camera image employs a safer passive way. To detect the state of eyes, There are many methods, such as AdaBoost classifier [3], SVM classifier [4] and so on. However, their ability of expressing features is limited. Recently, convolutional neural network (CNN) achieve remarkable progresses in a variety of computer vision tasks. In our paper, we design a driver fatigue detection system using multitask cascaded convolutional networks. As shown in Figure 1, the method mainly includes five parts: Joint face detection and alignment using multitask cascaded convolutional networks, normalize the current image and ground truth shape according to the scaled mean shape, extract the area of eye, state of eye recognition, fatigue detection.

2 Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks

Fatigue detection system should have high recognition accuracy and can detect the fatigue effectively in real-time. How to quickly and accurately detect the face of the driver and the eye alignment and overcome the impact of a certain light are the difficulties of fatigue detection system. Kaipeng et al [1]. propose a new cascaded CNNs-based framework for joint face detection and alignment, and carefully design lightweight CNN architecture for real-time performance. The overall pipeline is shown in Figure 2, which is the input of the following three-stage cascaded framework.

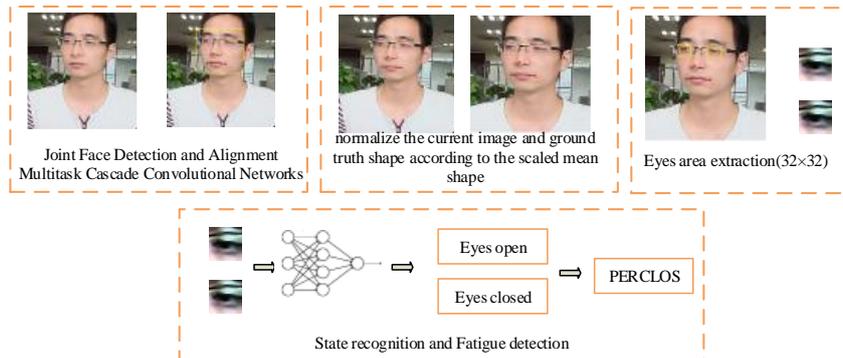


Fig. 1. Algorithm block diagrams.



Fig. 2. Joint face detection and alignment using multitask cascaded convolutional networks.

Stage 1: Exploit a fully convolutional network, called proposal network (P-Net), to obtain the candidate facial windows and their bounding box regression vectors. Then candidates are calibrated based on the estimated bounding box regression vectors. After that, employ nonmaximum suppression (NMS) to merge highly overlapped candidates.

Stage 2: All candidates are fed to another CNN, called refine network (R-Net), which further rejects a large number of false candidates, performs calibration with bounding box regression, and conducts NMS.

Stage 3: This stage is similar to the second stage, but in this stage we aim to identify face regions with more supervision. In particular, the network will output five facial landmarks' positions.

3 Extraction Area Eye

3.1 Face normalization

In order to accurately extract the eye areas, we need to calculate the average face. Then normalize the current image and ground truth shape according to the scaled mean shape, this process is 2D affine transformation. The 2D affine transformation is a method used to change the rotation angle, the scale, and the location of a shape. The transformation can be represented as equation (1).

$$\begin{cases} x = ax' + by' + c \\ y = dx' + ey' + f \end{cases} \quad (1)$$

Where $(x_i, y_i)^T$ is the coordinate of the i th feature point on the average face, $(x'_i, y'_i)^T$ is the coordinate of the i th feature point on the detected face. It has a matrix representation shown as equation (2).

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x'_i \\ y'_i \\ 1 \end{bmatrix} = M \begin{bmatrix} x'_i \\ y'_i \\ 1 \end{bmatrix} \quad (2)$$

For convenience, equation (2) can be rewritten as equation (3).

$$U = Kh \quad (3)$$

Where U is the feature point matrix of the average face, K is the feature point matrix of the detected face. h is affine transformation matrix. It can be Calculated with least squares solution. Then, the solution of h can be obtained as equation (4).

$$h = (K^T K)^{-1} K^T U \quad (4)$$

Normalize the current image and ground truth shape.



Fig. 3. Normalize the current image and ground truth shape according to the scaled mean

According to the scaled mean shape aimed at change the detected faces' rotation angle, the scale, and the location of a shape. As shown in Figure 3.

3.2 Eye area extraction

In this paper, we extract the area of eyes based on the facial landmarks after normalization as shown in Figure 4 .The eye area has a size of 32×32 .

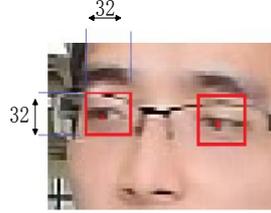


Fig. 4. The extraction result of the eye area.

4 Eye state recognition

CNN expresses features more better, avoiding the manual feature selection. So we used convolutional neural network to detect the state of eyes.

4.1 Convolutional neural network

To have high recognition accuracy of state of eyes and can detect the fatigue effectively real-time, three convolutional layers are used in our proposed network as shown in Figure 5. Each convolution layer connects a pooling layer, the first convolution layer is connected with a max pooling, the last two convolutions are connected with average pooling. The ReLU layers add non-linear constraints and the the Dropout layers prevents overfitting in the networks.

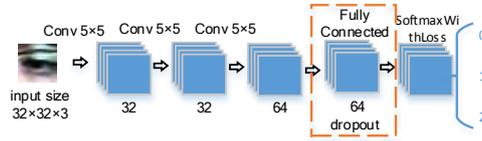


Fig. 5. Structure of the convolution neural network.

4.2 Activation Functions

Sigmoid function and tanh function are commonly used non-linear activation functions, but these functions exist the gradient vanishing, So we use the ReLU function (Rectified linear unit) which is defined as equation (5).

$$f(x) = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \quad (5)$$

ReLU can effectively alleviate the problem of gradient vanishing, So as to train the deep neural network directly in a supervised manner. The network can get sparse expression after the ReLU function, with the advantage of unilateral suppression.

5 Fatigue detection based on PERCLOS

After eye area extraction, the next step is to detect driver fatigue based on PERCLOS (percentage of eyelid closure over the pupil over time). PERCLOS is an established parameter to detect the level of drowsiness. Level of drowsiness can be judged based on the PERCLOS threshold value, PERCLOS is a parameter that is used to detect driver fatigue [5]. It is calculated as (6).

$$f_{PERCLOS} = \frac{n_{close}}{N_{total}} \times 100\% \quad (6)$$

Let n_{close} be the number of eye-close frames over a period time. N_{total} is the total number of frames over a period time. When the driver is in a state of fatigue, the driver's PERCLOS value will be higher than normal. We set the PERCLOS threshold, when the driver's PERCLOS value is higher than this threshold, then the current driver is considered fatigue.

6 Experiment and results

VS2013, running on a Win7 system with Intel (R) Core(TM) i7-6700HQ, CPUs (3.40GHz), 32GB memory, GPU NVNID GeForce GTX 1070

6.1 Train

In order to overcome the influence of light on image, the training data must contain data for different light intensities to enhance the robustness of the network, as shown in Figure 6.



Fig. 6. Different light intensities of Parts of the training samples.

Since we perform eye state recognition, here we use the following two different kinds of data annotation in our training process:

(1) negatives: 36×36 sample area was randomly intercepted near the eye area, regions whose the intersection-over-union (IoU) ratio is less than 0.4 to any ground-truth eyes as shown in Figure 7.



Fig. 7. Negatives training samples.

(2) positives: Positive samples are divided into two types, open eyes samples and closed eyes samples, their IoU above 0.6 to a ground truth face, as shown in Figure 8.



Fig. 8. Positives training samples.

6.2 Training results

We select images including eye images of open and closed as positives samples, and randomly crop several patches to collect negatives samples. We select 120000 images as training samples. The eye state recognition rate of the network has an increase in the number of iterations when training the samples, the result is shown in Figure 9.

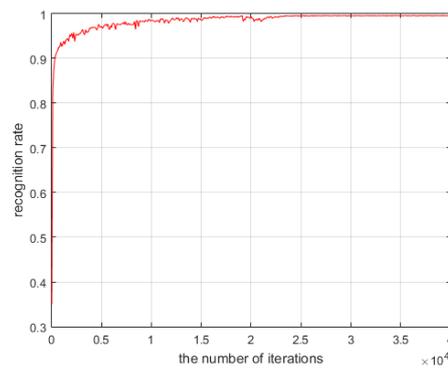


Fig. 9. The result of recognition rate.

With the increase of the iteration number, the accuracy rate gradually increased, the final accuracy rate between 0.995 to 0.996 fluctuations. In order to test the performance of the network, we collected three sections of video data, respectively, the accuracy rate

shown in the Table 1.

Table 1. The test result of eye state.

Video	Number of frames	Accuracy
Test1	243	97.11%
Test2	239	95.39%
Test3	249	98.39%
Test1	243	97.11%
Test2	239	95.39%

Through statistical 5 test videos includes 1239 frames of 320 * 240 images, computing the average time-consuming of the method include each module and overall time. Table 2 is the time-consuming result. The method complies with the requirement of real-time.

Table 2. The test of Time consuming.

Video	Face detection , alignment and Eye area extraction (ms)	State recognition (ms)	Total (ms)
Test1	23.41	6.57	34.92
Test2	24.50	7.10	36.63
Test3	23.12	7.09	34.76
Test4	23.42	6.51	34.09
Test5	22.13	6.98	33.46

6.3 Fatigue detection based on PERCLOS

When the driver is in a state of fatigue, the driver's PERCLOS value will be higher than normal, by setting the PERCLOS threshold, when the driver's PERCLOS value is higher than this threshold, then the current driver is considered fatigue. In this paper, the PERCLOS threshold is set to 0.30, when the driver is fatigue, the PERCLOS value is bigger than 0.30, Figure 10 shows PERCLOS result.

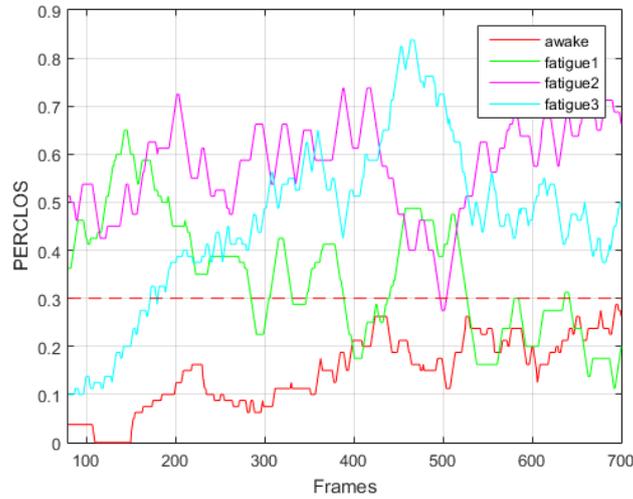


Fig. 10. The PERCLOS value.

Figure 11 shows the Sample images of detection results.

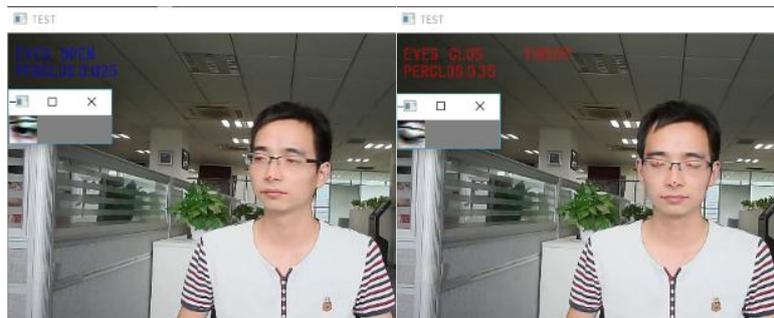


Fig. 11. Driver Fatigue Detection system.

7 Conclusion

In this paper we propose a driver fatigue detection system. This system uses the multitask cascaded convolutional networks to face detection and alignment. And then use another convolution neural network (CNN) for eye state recognition. Finally we calculate the percentage of eyelid closure (PERCLOS) to detect the fatigue. The method of eye state recognition provides high accuracy and can detect the fatigue effectively in real-time. Tests show that the system implementation is successful and the system does indeed infer fatigue reliably.

Acknowledgments. This research was partially supported by the National Nature Science Foundation of China (No.61461021) and Local Colleges Faculty Construction of Shanghai MSTC (No. 15590501300).

References

1. Zhang, K., Zhang, Z., Li, Z., Qiao, Y.: Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks. *IEEE Signal Processing Letters*. 10, 1499-1503 (2016).
2. Zhang, F., Su, J., Geng, L., Xiao, Z.: Driver Fatigue Detection Based on Eye State Recognition. In: 2017 International Conference on Machine Vision and Information Technology (CMVIT), pp. 105-110. Singapore (2017).
3. Shi-Feng, X.U., Zeng, Y.: Eyes state detection based on adaboost algorithm. *Computer Simulation*. 7, 214-217 (2007).
4. Li, Y.S., Zhang, X.H., Zhou, W.H., Liu, J.: Eyes state detection method based on LBP. *Application Research of Computers*. 6, 1897-1901(2015).
5. Wierwille, W.W., Ellsworth, L. A., Wreggit, S. S., Fairbanks, R. J., & Kim, C. L.: Research on vehicle-based driver status performance monitoring: development, validation, and refinement of algorithms for detection of driver drowsiness. National Highway Traffic Safety Administration (1994).
6. Liu, A., Li, Z., Wang, L., and Zhao, Y.: A practical driver fatigue detection algorithm based on eye state. In: *2010 Asia Pacific Conference on Postgraduate Research in Microelectronics and Electronics (PrimeAsia)*, 235-238. Shanghai (2010).
7. Wang, Y., Liu, X., Zhang, Y., Zhu, Z., Liu, D., and Sun, J.: Driving Fatigue Detection Based on EEG Signal. In: 2015 Fifth International Conference on Instrumentation and Measurement, Computer, Communication and Control (IMCCC), 715-718. Qinhuangdao (2015).
8. Zhao, G. F., and Han, A. X.: Method of Detecting Logistics Driver's Fatigue State Based on Computer Vision. In: *2015 International Conference on Computer Science and Applications (CSA)*, 60-63. Wuhan (2015).
9. Li, X., Wu, Q., Kou, Y., Hou, L., and Xie, H.: Driver's Eyes State Detection Based on Adaboost Algorithm and Image Complexity. In: *2015 Sixth International Conference on Intelligent Systems Design and Engineering Applications (ISDEA)*, 349-352. Guiyang (2015).
10. Guohui, H., Wanying, W.: An Algorithm for Fatigue Driving Face Detection and Location. In: *2015 8th International Conference on Intelligent Computation Technology and Automation (ICICTA)*, 130-132. Nanchang (2015).