Train Driver Fatigue Detection Using Eye Feature Vector and Support Vector Machine

Taiguo Li[1]*, Tiance Zhang[1], Quanqin Li[2]
[1]School of Automation & Electrical Engineering, Lanzhou Jiaotong University, Lanzhou 730070, Gansu, China
[2]Children's rehabilitation department, Shanxi provincial rehabilitation hospital, Xi'an 710065, Shanxi, China

Received: September 25, 2021. Revised: March 17, 2022. Accepted: April 21, 2022. Published: June 1, 2022.

Abstract—Fatigue driving is one of the main causes of traffic accidents. The eye features are the important cues of fatigue detection. In order to improve the accuracy and robustness of detection based on a single eye feature, we propose a fatigue detection algorithm based on the eye feature (EFV) vector. Firstly, the coordinates of the eye region were localized with facial landmarks detector and the landmarks geometric relation (LGR) was calculated as a feature value. Secondly, a deep transfer learning network was designed to classify the driver eye state on a small dataset. The probability value of the eyes being open state was calculated. Then an eye feature vector was constructed to overcome the limitations of a single fixed threshold and a support vector machine (SVM) model was trained for eye state classification recognition. Finally, the performance of the proposed detection model was evaluated by the percentage of eyelid closure over time (PERCLOS) criterion. The results show that the accuracy of this model can reach 91.67% on the test database, which is higher than the single-feature-based method. This work lays a foundation for the online fatigue detection of train drivers and the deployment of the train driving monitoring system.

Keywords—Fatigue detection, face detection, eye feature vector, transfer learning, PERCLOS

I. Introduction

IN recent decades, as the China Railway Industry development, train drivers will be in a state of heavy load, high tension, and continuous work for a long time, so that their bodies are very easy to be in a state of fatigue [1]. According to the accident statistics of the Chinese railway system, fatigue driving is an important factor causing train traffic accidents [2]. Therefore, it is necessary to monitor the fatigue state of the train driver in real-time. When the driver is fatigued, give a timely reminder, which can effectively protect the safety of the drivers and passengers. At present, most driver fatigue detection mainly adopts objective detection methods, mainly including vehicle-information-based technology [3,4], driver-physiological-signal-based algorithms [5,6], and driver-behavior-based technology [7,8].

Vehicle-information-based technology mainly uses image recognition and various vehicle sensors to extract some key parameters of the vehicle, including steering wheel angle, driving speed, the direction of vehicle movement, lane information, etc. Then the fatigue state of the driver can be comprehensively inferred by these parameters [9, 10, 11]. One characteristic of Vehicle-information-based technology is that most of the vehicle state can be obtained by detecting the lane line of the vehicle and then judging the driver fatigue state. This detection method is not suitable for trains because they use a track.

Driver-physiological-signal-based algorithm is to collect signals such as EEG (electroencephalogram) [12, 13], HRV (heart rate variability) [14], ECG (electrocardiogram) [15], and EOG (electrooculogram) [16], and then explore the physiological relationship between these signals and driver fatigue. This method can identify the driver fatigue in the early stage, and the probability of error is small. However, the main defect is that it is an invasive detection method, which requires the driver to connect some electrodes to his body. Thus, this method may affect the driver's normal driving, and they will feel unpleasant if used for a long time.

Driver-behavior-based technology is to identify driver fatigue changes by detecting the driver's behavioral reaction. Some important facial information can be used to interpret levels of fatigue. It has become a more popular detection method because it directly focuses on the visual features of the drivers rather than external devices. Furthermore, it is non-invasive and more practical than the other two methods mentioned above. This method usually extracts facial features from following sources: eyes [17, 18, 19, 20], mouth [21, 22, 23], and head [7, 24]. According to relevant research, features exhibited in the eye region are the most obvious symptoms of fatigue [25]. The fatigue detection model through the extraction of driver eye features has many advantages, such as being close to people's cognition, non-interference with driving, strong real-time performance, and relatively low cost. However, there are still some challenges of extracting eye features, mainly including different illumination conditions and large head pose variations. To overcome these challenges and improve the detection performance, we propose a model using the eye feature vector and support vector machine.

II. System Architecture

The overall architecture of the fatigue detection system based on eye feature vector visual analysis, as illustrated in Figure 1, has four major steps. (1) The deep learning method with strong feature learning ability is used to detect the train driver's face image from a video and localize the train driver's facial landmarks. (2) Two types of eye state recognition algorithms are designed to extract the driver eye state features. (3) A SVM information fusion strategy based on eye feature vector EFV is proposed. The robust estimation of the open and closed state of the driver's eyes is obtained by using the SVM classifier. (4) An experimental method using PERCLOS is designed to judge the driver fatigue state.

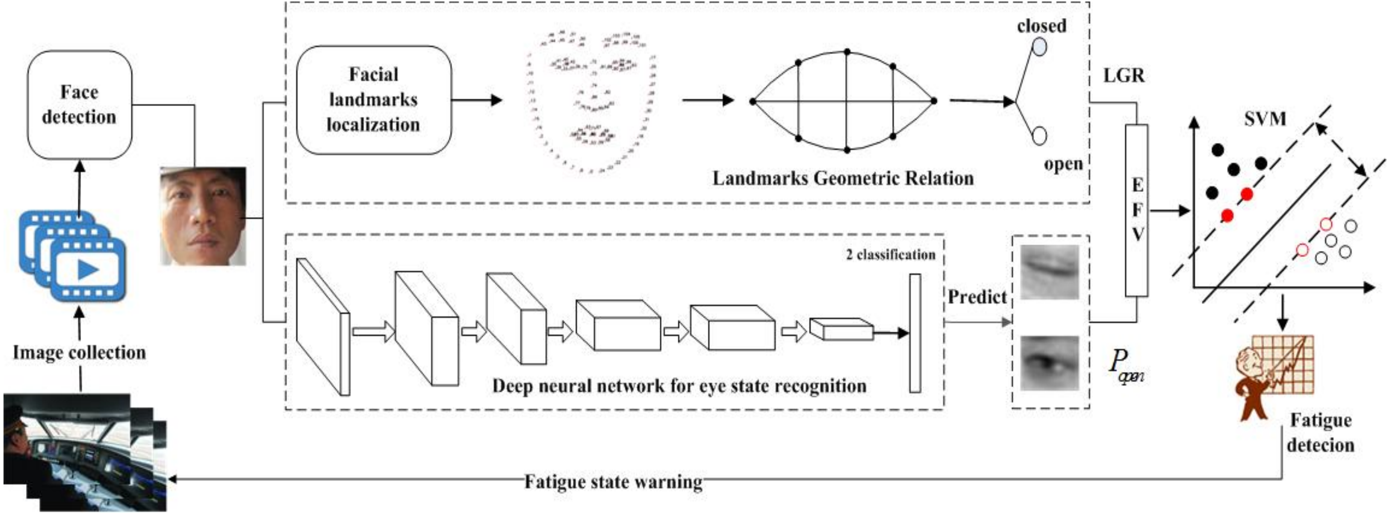


Figure 1. The overall architecture of train driver fatigue detection

III. Methods

A. Face detection and facial landmarks localization

At present, the majority of fatigue detection methods are based on Viola-Jones [26] algorithm to locate the face region. Viola-Jones is a fast face detection algorithm based on a simple Haar-like feature cascaded Adaboost. Viola-Jones face detection algorithm has a better detection effect on the frontal face image, but the detection accuracy for the profile face image is not very high. In the application scenario of the railway, although the position between the seat and the driver is constrained, the driver's head posture will change at any time to complete the driving task. The dlib face detection algorithm based on HOG feature descriptor can quickly complete face region search and can achieve better detection performance for both front and profile face. Then a Practical Facial Landmark Detector (PFLD) [30] is employed to extract the fine features of facial landmarks from the detected face images, which contains the 106 facial landmarks coordinate information. These facial landmarks are used as the key basis for the extraction of driver eye fatigue features.

1) Face detection

The dlib face detection algorithm extracts the HOG feature from the positive sample (including the image of the face) dataset and obtains the HOG feature descriptor. HOG feature is extracted from negative sample (image without face) dataset to obtain HOG descriptor. The model is obtained by training positive and negative samples with an SVM algorithm. Finally, the SVM model is used to mine the difficult samples to improve the model classification ability.

2) Facial landmarks localization based on PFLD

The facial landmarks localization task feeds the face images to the detector, and outputs feature points such as nose tip, mouth, eyes, and eyebrows through the localization detector. At present, most of the facial landmarks localization algorithms used for fatigue detection are based on dlib toolkits. However, this method can only localize 68 facial landmarks, and the accuracy and robustness of the detection model can be further improved. PFLD can localize 106 facial landmarks, and obtain higher accuracy in complex scenes (including unconstrained posture, expression, illumination, occlusion, etc.). Considering the geometric constraints and data imbalance, PFLD designed a new loss function:

$$L \equiv \frac{1}{M}\sum_{m=1}^{M}\sum_{n=1}^{N}(\sum_{c=1}^{C}\omega_n^c\sum_{k=1}^{K}(1-\cos\theta_n^k))\left\|d_n^m\right\|_2^2 \quad (1)$$

Where $\gamma_n = \sum_{c=1}^{C}\omega_n^c\sum_{k=1}^{K}(1-\cos\theta_n^k)$ is the weight factor that plays a key role in network training, $\|g\|$ designates a certain metric to measure the distance of the n-th landmark of the m-th input.

A video image is fed into the face detection algorithm to obtain face image *I*. Where $f=[x,y,w,h]$ denotes the face image region, (x,y) represents the top-left coordinate of the face rectangular region, w,h represents the width and height of the face rectangular region respectively. The facial landmarks coordinate set $P = \{p_1, p_2, ..., p_N\}$ can be obtained using the facial landmarks localization algorithm. Where $N = 106$, $p_i \in \mathbb{R}^2$ is the (x, y) coordinate of the i-th facial landmark in the face image. The result of facial landmarks localization and coordinate index is shown in Figure 2.

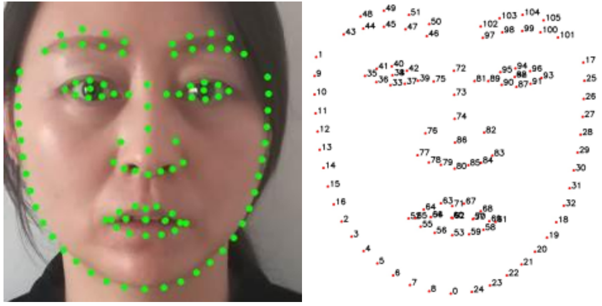


Figure 2. Result of facial landmarks localization

B. Eye state features extraction

The key to train driver fatigue detection using eye feature vector (EFV) is to design accurate and efficient eye feature extraction algorithms.

1) Eye state feature extraction using facial landmarks

According to the localization of face key points, LGR (Landmarks Geometric Relation) can describe the geometric relationship between eye region landmarks. Therefore, LGR can be regarded as an important factor in eye state recognition.

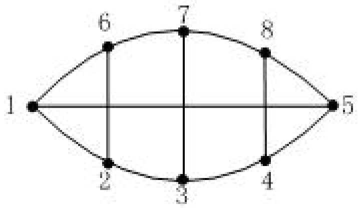


Fig. 3 Eye region landmarks

As shown in Figure 3, $e_1, ..., e_8$ are the eight feature points corresponding to the eye contour obtained through the facial landmarks detection. Therefore, eight (x, y) coordinates can delineate the contour of each eye. The eye region landmarks geometric relation (LGR) can be calculated in the following form:

$$A = \|e_6 - e_2\| \quad B = \|e_7 - e_3\| \quad C = \|e_8 - e_4\| \quad D = \|e_1 - e_5\| \tag{2}$$

$$LGR = \frac{A + B + C}{3 \times D} \tag{3}$$

Where *A*, *B*, and *C* are respectively the longitudinal Euclidean distance corresponding to eye contour landmarks. *D* denotes the transverse Euclidean distance between the two corners of the eye contour. LGR represents the aspect ratio of the eye contour.

The LGR changes very little when eyes are open and is a very small value while closing eyes.

2) Eye state feature extraction using deep transfer learning

Deep Convolutional neural networks (Resnet [28], VGG nets [29], etc.) have achieved excellent performance for visual recognition tasks. Evidence reveals that training data amount is of crucial importance for model performance. At present, gaining a rich supply of eye state datasets that directly meet a deep neural network training for driver fatigue detection is difficult. If we train the deep neural network model with a small sample dataset, it is likely to occur overfitting. Therefore, it will seriously affect the extraction result of eye state features. In contrast, deep transfer learning (DTL) has a strong ability to automatically extract features. It can overcome the disadvantage of overfitting and meet the end-to-end requirements in practical applications.

DTL takes advantage of a deep neural network to transfer the existing knowledge from the source domain to the target domain. In this study, we use the model transfer approach to seek the shared parameter information from the source domain and the target domain. A data domain is made up of feature space X and a probability distribution $P(X)$. For a given source domain D_s and target domain D_t, it can be expressed as $D_s = \{X_s, P(X_s)\}$ and $D_t = \{X_t, P(X_t)\}$ $T_s = \{Y_s, f_s\}$, $T_t = \{Y_t, f_t\}$ where T_s indicates domain task, Y_s represents source domain class space, and Y_t is target domain class space. f_s and f_t represent mapping functions for source and target domains, respectively. The objective of DTL is to utilize the tagged data D_s to learn a target prediction function $f_t : x_t \mapsto y_t$. Finally, we can predict the target domain D_t belongs to which class $y_t \in Y_t$.

We present comprehensive experiments on Closed Eyes in the Wild (CEW) dataset to evaluate the effectiveness of different deep convolutional neural networks. We obtain better results by the Resnet34 network compared to Resnet18, Resnet50, and VGG16 networks. Therefore, we select Resnet34 as the backbone network in this study. The feature knowledge learned by Resnet34 on the ImageNet dataset is used as the pre-training model. Finally, the trained deep transfer learning network will be utilized for eye state recognition. ResNet, the winner of ILSVRC 2015, has the advantage of shortening the network model training time while making the network deeper. As shown in table 1, it is the resnet34 network structure configuration.

Table 1. Network structure of ResNet34

layer name	**output size**	**parameter settings**
conv1	112×112	7×7, 64, stride 2
conv2.x	56×56	3×3, max pool, stride 2 $\begin{bmatrix} 3\times3 & 64 \\ 3\times3 & 64 \end{bmatrix} \times 3$
conv3.x	28×28	$\begin{bmatrix} 3\times3 & 128 \\ 3\times3 & 128 \end{bmatrix} \times 4$
conv4.x	14×14	$\begin{bmatrix} 3\times3 & 256 \\ 3\times3 & 256 \end{bmatrix} \times 6$
conv5.x	7×7	$\begin{bmatrix} 3\times3 & 512 \\ 3\times3 & 512 \end{bmatrix} \times 3$
FC	1×1	average pool, 1000-fc, softmax

Resnet34 is employed as the backbone of eye state feature extraction and pre-trained on the ImageNet-2012 dataset with more than 1.2 million images by the supervised learning method. Then, using the idea of transfer learning, the learned parameters of the pre-trained backbone network are transmitted to the target network for extracting eye state features.

After training the model on the CEW dataset, the transfer network model can be employed to extract the eye state features, and the DTL network framework is illustrated in Figure 4.

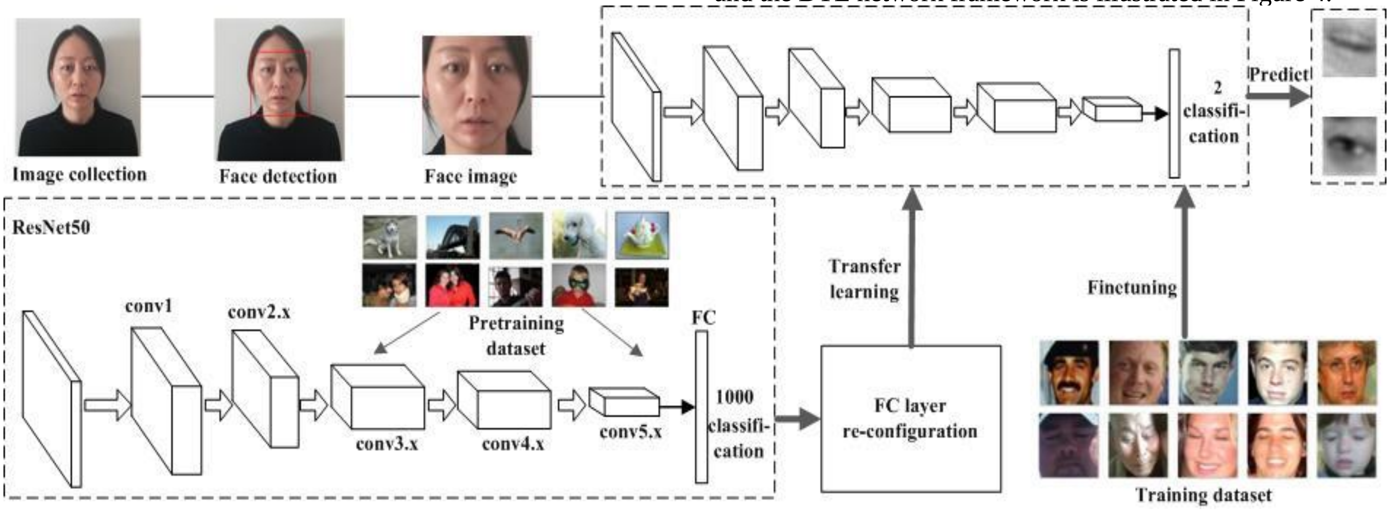


Figure 4. Deep transfer learning network framework for eye state feature extraction

After multiple convolution-pooling operations, the feature results learned are expanded into a one-dimensional array and feed to the FC layer. After the FC layer, the open or closed binary classification result of the eye images can be obtained through the softmax classifier. Finally, the P_{open} value is taken as a feature value to construct the eye feature vector. Where P_{open} is the probability that the eyes are open.

C. Eye state recognition based on EFV

1) Eye feature vector construction

LGR algorithm is to judge the eye-openness state through the geometric relationship between the coordinates of the eye contour landmarks. However, the deep transfer learning network extracts the facial image features to judge the eye-openness state through supervised learning. The common idea of these two methods is to use a single feature to judge the eye-openness state. However, in real driving scenes, the accuracy of a single feature recognition algorithm is more likely to be affected by the changes of head posture, light, and other conditions than that of multiple features. To achieve higher recognition accuracy, EFV (Eye Feature Vector) two-dimensional feature vector is constructed based on the LGR algorithm and deep transfer learning network model. Two features are fused to recognize the driver's final eye state. This approach can enhance the accuracy and robustness of detection based on a single eye feature. EFV is defined as:

$$EFV = (P_{open}, LGR) \qquad (4)$$

The P_{open} probability is taken as the X value and the LGR result as the Y value. As shown in Figure 5, black dots are the open-eye feature vector and red dots represent the closed-eye feature vector. Therefore, we can plot the mapping diagram between the eye feature values and EFV.

As shown in Figure 5, we can see that when the driver's eye features take different values, the coordinate positions of EFV are different (the red dot is the EFV coordinate when the eyes are closed, and the black dot is the EFV coordinate when the eyes are open). Consequently, EFV can be determined as an important parameter to represent the eye state in the driver fatigue detection model.

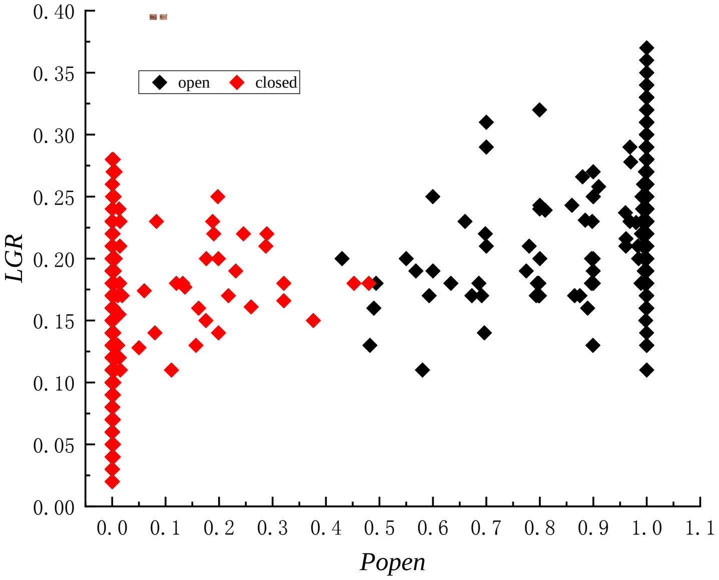


Figure 5. Eye feature values and EFV mapping diagram

2) Eye state recognition classifier

The traditional method is to judge eye-openness state according to a single feature and fixed threshold value, which will have the problems of weak generalization ability and weak adaptability to the environment. Therefore, to overcome these problems and improve performance, EFV values are calculated through the images provided by the self-built training dataset and fed to a machine learning (ML) classifier. In this study, we analyzed three types of classifiers: Random Forest (RF), Artificial Neural Network (ANN), and Support Vector Machine (SVM) [27].

Among them, SVM is a machine-learning model that creates a mapping function between input vectors and output results based on the training dataset. It does not need to consider any previous knowledge about the mapping function relationship. The trained SVM model is used as a classifier for eye state recognition. The basic model of SVM is defined as a classifier with the largest spacing in the feature space. Assume that the training dataset in the feature space is T :

$$T = \{(\mathbf{x}_i, y_i), i = 1, 2, ..., l\} \qquad (5)$$

Where $x_i \in \mathbb{R}^n$ is the *i*-th EFV feature vector, $y_i \in \{+1, -1\}$ is the class labels. When $y_i = +1$, it is a positive example (representing the open state of the eye), and when $y_i = -1$ is a negative example (representing the closed state of the eye).

The SVM classifier model is to solve the maximum partition hyperplane, which can be expressed as a constrained optimization problem:

$$\begin{cases} \min\limits_{w,b,\xi} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{l}\xi_i \\ s.t. \quad y_i(\mathbf{w}^T\phi(\mathbf{x}_i) + b) \geq 1 - \xi_i \quad (\xi_i > 0) \end{cases} \qquad (6)$$

Where $\phi(\cdot)$ is the nonlinear mapping function. C > 0 is the penalty parameter of the error term. $K(\mathbf{x}_i, \mathbf{x}_j) \equiv \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$ is defined as the kernel function. There are four basic kernel functions in SVM, such as linear, polynomial, radial basis function (RBF), and sigmoid.

D. Driver fatigue detection algorithm

PERCLOS, proposed by CMU Research Institute, is widely considered the most effective fatigue evaluation rule at present. PERCLOS can be described by formula (7) as:

$$PERCLOS = \frac{T_{close}}{T} \times 100\% \qquad (7)$$

Where T is the detection period (unit: second), T_{close} represents the time when the driver's eyes are closed in the detection period.

In the video frame sequence of driving fatigue detection, although there is no time scale, the video frame speed is fixed and the image is continuous. Based on this feature, to simplify the computational complexity of PERCLOS, the calculation of Formula (7) is converted into Formula (8).

$$PERCLOS = \frac{F_{close}}{F} \times 100\% \qquad (8)$$

Where F is the total number of frames in a detection period and F_{close} is the number of frames with the eye closed in the detection period.

If $PERCLOS \leq PER_{\text{threshold}}$ (threshold of PERCLOS), it can be determined that the driver is in an awake and normal driving state during the detection period. If $PERCLOS > PER_{\text{threshold}}$, it indicates that the driver may have experienced fatigue driving during the detection period. The $PER_{\text{threshold}}$ will be determined by fatigue detection experiment.

IV. Experiment

A. Deep transfer learning network training

we select Resnet34 as the backbone network in this study. The feature knowledge learned by Resnet34 on the ImageNet dataset is used as the pre-training model. We present comprehensive experiments on Closed Eyes in the Wild (CEW) dataset to evaluate the effectiveness of different deep convolutional neural networks.

1) Datasets and environment

ImageNet-2012 dataset[31]: There are 1,000 image categories in ImageNet-2012, among which the training dataset contains 1,281,167 images, with data ranging from 732 to 1,300 for each category, while the verification dataset contains 50,000 images, with an average of 50 images for each category.

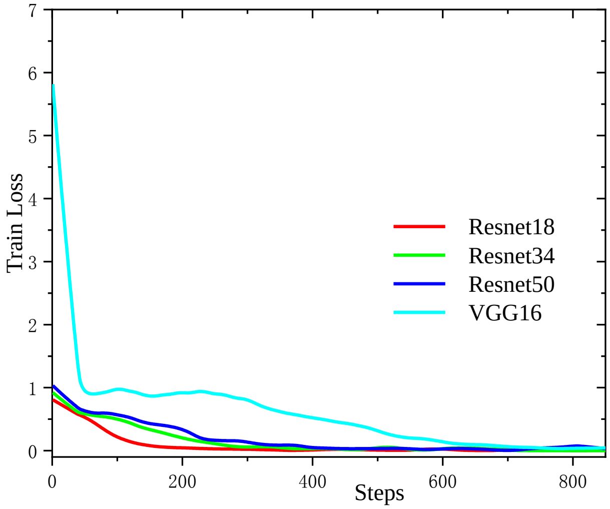
Closed eyes in the wild（CEW）dataset[32]: A dataset constructed by Song et al., Nanjing University of Aeronautics and Astronautics for eye state close-up detection in the field consisted of 2,423 participants, among which 1,192 participants with eyes closed are directly gathered from the Internet, and 1,231 participants with eyes open are selected from the LFW database.

The training environment of the network model is Baidu AIStudio platform. The hardware environment is CPU : 4Cores, RAM : 32GB, Disk : 100GB, GPU: Tesla V100, Video Mem 16GB. The software environment is Python 3.7, Baidu open-source deep learning framework PaddlePaddle 1.8.4. The end-to-end model is adopted in the model training process, batchsize=128, epoch_num=50, Adam optimizer, the global learning rate is 5×10^{-4}, learning rate scheduling method is linear_decay.

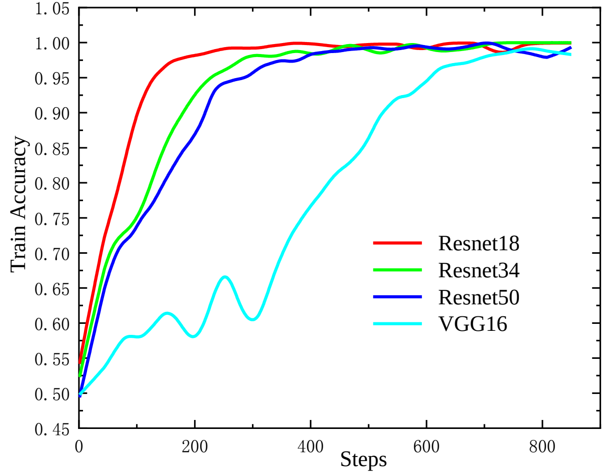
2) Network training and comparative experiment

The deep transfer learning (DTL) network for eye state feature extraction is trained on Baidu AIStudio platform. The CEW dataset used for finetuning has 2,423 images, which is divided into training and validation datasets according to the ratio of 9:1.

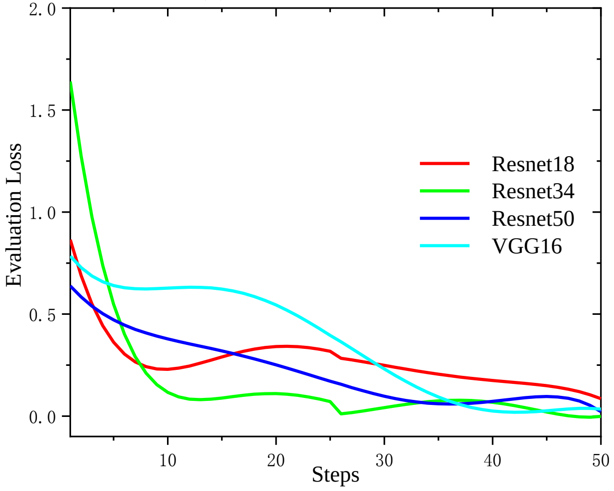
To analysis the possible advantage of a deep learning network for eye state feature extraction, under the same experimental environment, we conducted a group of experiments on the CEW dataset by using different deep learning networks. First, we make a comparison between VGG and Resnet series networks. One representative network of the former is the VGG16. About Resnet, according to different network depths, we select Resnet18, Resnet 34, and Resnet50 to test the performance of the eye state feature extraction. Figure 6 gives the training and validation results.



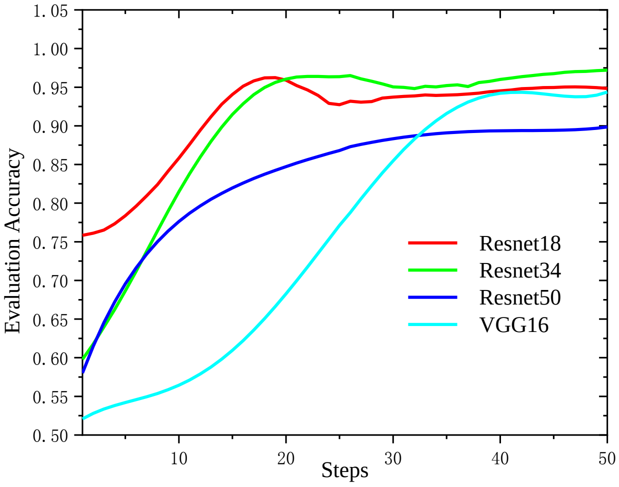
(a) Training Loss



(b) Training Accuracy



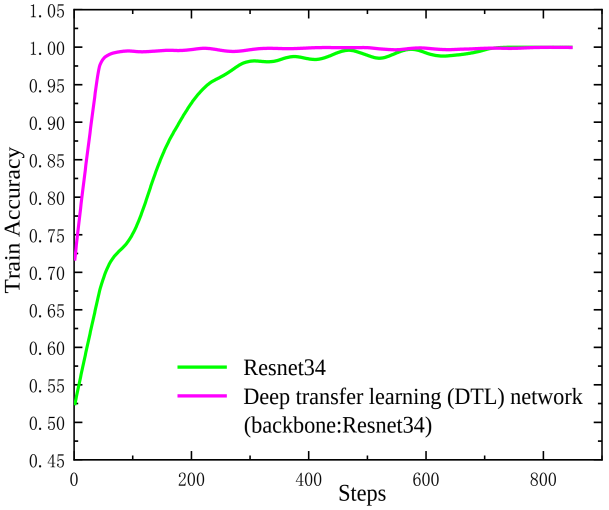
(c) Evaluation Loss



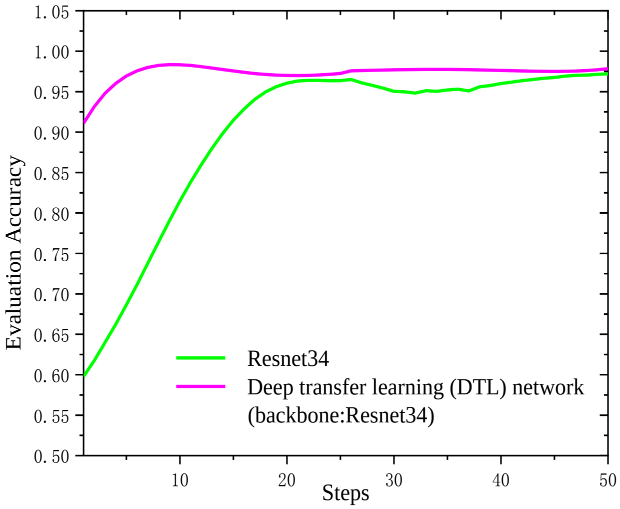
(d) Evaluation Accuracy

Figure 6. Loss and accuracy curves of various deep learning networks

We can see that the training loss value becomes very small, near-zero, at the end of training. Correspondingly, the training accuracy also reaches a stable value. It can also be observed the evaluation loss and accuracy on the CEW dataset are consistent with the changing trends in training loss and accuracy. Through statistics, the evaluation loss: Resnet34 (0.0019)<Resnet50(0.0216)<VGG16(0.0345)<Resnet18 (0.0852), the evaluation accuracy: Resnet34(97.2%)> Resnet18 (96.2%)>VGG16(94.4%)>Resnet50(89.9%). The Discovery of resnet34 has better performance than the other three models by experimental results analysis. Therefore, a DTL network for eye state feature extraction selected Resnet34 as the backbone network. We further train the DTL network on the CEW dataset and validate its effectiveness.



(a) Train Accuracy



(b) Evaluation Accuracy

Figure 7. Accuracy curves of the DTL network and Resnet34

As shown in Figure 7, experimental results suggest that the proposed DTL network provides superior performance and achieves higher accuracy than Resnet34. The DTL network reaches an evaluation accuracy of 98.3% on the CEW dataset.

B. *Eye state recognition classifier training and test*

In this experiment, we use the extracted eye features, P_{open} and LGR, to construct EFV as the input of the training eye state recognition classifier.

1) classifier training dataset

Due to the limited resources provided by the public eye state recognition dataset, images are manually screened from the Dynamic Facial Expression in the Wild (DFEW)[33] database to establish the training dataset of the classifier. The training dataset divides into two parts, 500 open eye images and 500 closed respectively. The dataset includes various challenging interferences in practical scenarios such as extreme illumination, occlusions, and head posture changes. It is beneficial to boost the generalization ability of the eye state recognition classifier. Using the same method, a test dataset is established to test the effectiveness of the trained classifiers when predicting unknown images. The sample images are shown in Figure. 8.

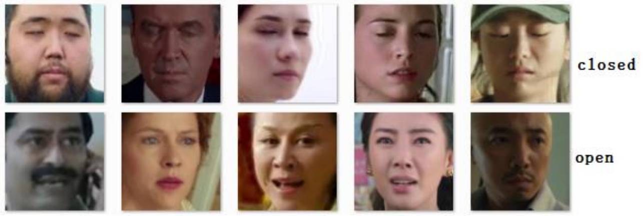


Figure 8. Sample of the classifier training dataset

2) classifier training and test

Subsequently, we will compare the performance among the three classifiers mentioned in section 3.3.2 with the same dataset and input EFV. All codes related to the classifier model are developed under the python programming environment. To evaluate the performance of the various eye state recognition classifiers, we calculate the precision, recall, and accuracy according to formulas (9)-(11).

$$\Pr ecision = \frac{TP}{TP+FP} \times 100\% \tag{9}$$

$$\operatorname{Re} call = \frac{TP}{TP+FN} \times 100\% \tag{10}$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \tag{11}$$

where

TP = the state correctly identified as the closed state while eyes were actually in the closed state.

TN = the state correctly identified as the open state while eyes were actually in the open state.

FP = the number of incorrectly identified as the open state while eyes were actually in the closed state.

FN = the number of incorrectly identified as the closed state while eyes were actually in the open state.

Table 2. Recognition performance of various classifiers

			Training				Test			
			Precision (%)	Recall (%)	Accuracy (%)	Time (ms)	Precision (%)	Recall (%)	Accuracy (%)	Time (ms)
RF	optimized		99.60	99.40	99.50	106	93.27	92.33	92.83	0.017
ANN	2 layers	(32,32)	100	99.80	99.90	624	99.65	96.00	97.83	1.17
		(64,64)	100	99.80	99.90	642	99.66	96.33	98.00	1.19
		(128,128)	100	99.80	99.90	710	99.65	95.33	97.50	1.26
	3 layers	(32,32,32)	100	99.80	99.90	701	99.65	95.33	97.50	1.31
		(64,64,64)	99.80	99.80	99.80	735	99.32	96.69	98.01	1.32
		(128,128,128)	99.80	99.80	99.80	777	99.65	94.00	96.83	1.37
SVM	linear		99.60	99.80	99.70	4.0	99.31	96.00	97.67	0.002
	poly		99.80	99.80	99.80	3.8	99.31	95.67	97.50	0.003
	sigmoid		100	99.60	99.80	5.8	98.21	91.67	95.00	0.003
	rbf		99.80	99.80	99.80	4.6	99.66	98.00	**98.83**	**0.002**

Table 2 shows that the RF classifier has an excellent performance in training. However, accuracy is the worst in testing among three classifiers which implies overfitting on the training dataset. ANN classifiers with two or three layers have better performance both in training and testing. At the same time, we also can see that the time cost of the ANN classifier is hundreds of times longer than the SVM classifier, which has a great impact on real-time embedded devices. In the SVM classifiers, there are four different kernel functions to choose from parameter c in the training. Through comparison, it is found that there is little difference in the training and testing accuracy of the three SVM classifiers. This indicates that the SVM classifier has good generalization ability. However, the RBF-based SVM model is superior to the other three SVM classifiers in terms of precision, recall rate, and accuracy, which has an accuracy rate of 98.8% for driver eye state recognition on the test dataset.

C. Driver fatigue detection

1) DROZY database

The public “Ulg multimodality drowsiness database” (short name “DROZY”) provided by Massoz [34] et al. contains multiple modalities of drowsiness-related data (videos, polysomnography signals, etc.). It can help researchers to carry out various types of drowsiness-related developments and experiments (fatigue, cognitive distraction detection, etc.). Therefore, the test and performance validation of fatigue detection was performed with the DROZY database in this study.

In the DROZY database, 14 healthy subjects performed three psychomotor vigilance tests (PVTs) over two consecutive days. The data collection schedule is shown in Figure 9.**DAY 1.** It is required to install test equipment after arriving at the laboratory at 8:30. At the beginning of PVT data collection, subjects were asked to evaluate his/her level of fatigue according to the Karolinska Sleepiness Scale (KSS)[35] and fulfill a form. Between 10:00 and 11:00, subject completed the PVT1 test. Once a subject started the first PVT, he/she can not rest until the third PVT was finished, the total sleep deprivation was about 28-30 hours. **DAY 2.** Subject completed PVT2 and PVT3 tests 3:30 - 4:00 and 12:00 - 12:30, respectively.

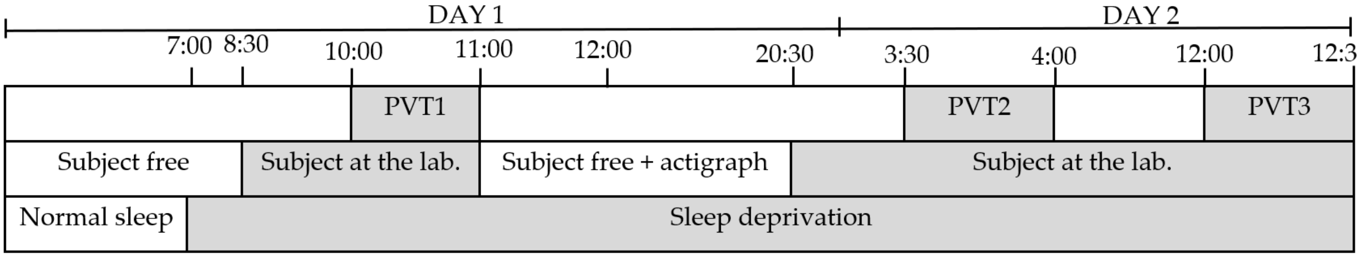


Figure 9. PVT data collection schedule

In traditional mental fatigue detection methods, KSS is a subjective means to evaluate one’s level of fatigue, which is sensitive and usually used for estimating the overall fatigue. The KSS level is shown in Table 3

Table 3. KSS level for self-estimation of fatigue

Karolinska Sleepiness Scale (KSS)	
1	Extremely Alert
2	Very Alert
3	Alert
4	Rather Alert
5	Neither Alert nor Sleepy
6	Some Signs of Sleepiness
7	Sleepy, But No Effort to Keep Awake
8	Sleepy, some Effort to Keep Awake
9	Very Sleepy, Fighting Sleep

2) Fatigue detection subjects and parameter setting

Since KSS is a subjective self-estimation method, it is also sensitive. According to Table 3, this study divides KSS levels into two categories – Normal ($KSS \le 3$) and Fatigue ($KSS \ge 7$)- to reduce possible inconsistency in consideration of the subject is difficult to determine the thresholds between adjacent KSS levels. The literature [15] and [36] also adopt $KSS \le 3$ for alertness and $KSS \ge 7$ drowsiness. According to this classification method and the PVT data provided by the DROZY database, eight subjects experienced both alert and drowsy states during the experiment. At the same time, eye state was recognized for these subjects. Generally speaking, higher KSS levels represent a greater percentage of eye closure. Finally, according to this principle, three subjects for which the fatigue level was detected. Table 4 summarizes the KSS levels of each subject and the percentage of eyelid closure (PEC).

$$PEC = \frac{E_{closed}}{E_{closed} + E_{open}} \times 100\% \qquad (10)$$

Table 4.. Summary of three subjects

Subject	PVT	KSS level	Eye-open image	Eye-closed image	PEC(%)
1	PVT1	3	16,996	869	4.86
	PVT2	6	8,806	691	7.28
	PVT3	7	7,957	909	10.25
2	PVT1	3	16,824	1,075	6.01
	PVT3	6	6,554	1,592	19.54
	PVT2	7	3,891	5,510	58.61
8	PVT1	2	16,806	1,062	5.94
	PVT2	6	7,646	1,228	13.83

PVT3	8	7,697	1,328	14.71

After the subjects are selected, MissRate and ErrorRate are employed to determine the appropriate threshold value $PER_{\text{threshold}}$. MissRate refers to the rate at which the model is unsuccessfully recognized as a fatigued state while the driver is actually in a fatigue state, and ErrorRate refers to the rate at which the model is misjudged as a fatigued state while the driver is actually in non-fatigue state. The experimental results of MissRate and ErrorRate are presented in Figure 10. We can see that the MissRate value decreases with the increase in PERCLOS value, and the ErrorRate has the opposite trend. When the threshold is 0.20 ($PER_{\text{threshold}} = 0.20$), the model achieves the best performance.

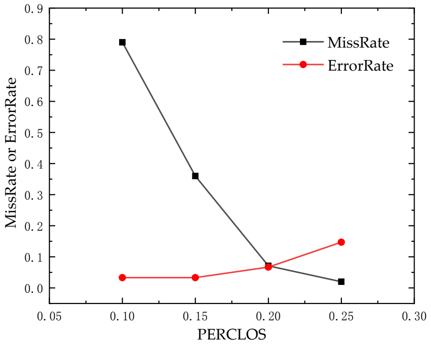


Figure 10/ Relationship between PERCLOS and MissRate and ErrorRate

In addition, considering the fatigue does not fluctuate rapidly, only calculating the current output of PERCLOS value is not enough to predict fatigue. Hence, the final result is extracted from six pre-set slide windows which have lengths of {10,20,30,40,50,60} seconds, respectively. If the PERCLOS value exceeds the threshold value (0.20) in three or more windows, it is judged that the current state is fatigue.

3) Results

We test the performance of our model for driver fatigue detection on three subjects (i.e., subject ‘1’, ‘2’, and ‘8’) of DROZY database. The test data consists of six videos performed by three subjects. The final detection results are shown in Table 5.

Table 5. Experimental results

Subject	PVT	KSS level	Video length	Driver state	Accuracy (%)	
					Soukupova et al. [33] (EAR-based)	Ours (EFV + SVM)
1	PVT1	3	00:09:55	Normal	76.36	92.73
	PVT3	7	00:09:51	Fatigue	83.64	96.36
2	PVT1	3	00:09:56	Normal	89.09	89.09
	PVT2	7	00:10:26	Fatigue	98.28	98.28
8	PVT1	2	00:09:55	Normal	76.36	80.00
	PVT3	8	00:10:01	Fatigue	80.39	94.12
Average					84.02	**91.76**

It can be observed from tables 4 and 5 that different people have unique understanding and performances related to normal and fatigue states. Despite the same KSS level, eye performance varies among different drivers. The eye aspect ratio (EAR) proposed by Soukupova et al. [37] is a traditional and common fatigue detection method, which only extracts a single feature of the eye. The detection accuracy of fatigue state is 84.02%. The comparison results in Table 5 show that, once EFV and SVM are integrated, the detection accuracy significantly improves and becomes higher than the EAR-based method. The accuracy is up to 91.76%.

V. Conclusion

This study selects the eye state which can directly reflect the fatigue level of the train driver as the feature and uses the method of machine learning to detect and analyze the fatigue state.

Two algorithm models of eye state feature extraction are designed. The eye state feature vector EFV is constructed to represent the fatigue characteristics of eyes from a higher dimension, which improves the performance of eye state recognition.

An information fusion strategy based on SVM is proposed with EFV vector as input, and the knowledge independently learned by SVM on the training dataset is used to guide the classification of driver eye state, which overcomes the limitation of a single fixed threshold and improves the adaptability and robustness of the algorithm model. Finally, We carried out the driver fatigue detection experiment based on PERCLOS criterion, and the comparison experiments indicate that our method has a detection accuracy rate of 91.76%, which is higher than the EAR-based method.

Because driving fatigue is the result of physiological and psychological factors, fatigue state will be reflected in many visual features such as eyes, mouth, and facial expressions. Based on the visual analysis of the state of the eyes, the influence of the change of the state of the mouth and the change of facial expression will be further considered.

Acknowledgment

This research is supported by the science and technology program of Gansu Province (21JR7RA303, 21JR1RA235);

Lanzhou Jiaotong University Youth Science Foundation Project (2020002).

References

[1] Yang, H.; Chen, X.; Lei, J.; Wang, Y. A Fatigue Detection Method for Train Drivers Based on Inverse Projection Correction and Eye Gaze Correction. *Tiedao XuebaoJournal China Railw. Soc.* **2018**, *v 40, n 4*, p 83-89.

[2] Zhang, X.; Li, J.; Liu, Y.; Zhang, Z.; Wang, Z.; Luo, D.; Zhou, X.; Zhu, M.; Salman, W.; Hu, G.; et al. Design of a Fatigue Detection System for High-Speed Trains Based on Driver Vigilance Using a Wireless Wearable EEG. *Sensors* **2017**, *17*, 486.

[3] Arefnezhad, S.; Samiee, S.; Eichberger, A.; Frühwirth, M.; Kaufmann, C.; Klotz, E. Applying Deep Neural Networks for Multi-Level Classification of Driver Drowsiness Using Vehicle-Based Measures. *Expert Systems with Applications* **2020**, *162*, 113778.

[4] Wang, F.; Wu, S.; Ping, J.; Xu, Z.; Chu, H. EEG Driving Fatigue Detection With PDC-Based Brain Functional Network. *IEEE Sensors J.* **2021**, *21*, 10811–10823.

[5] Chaudhuri, A.; Routray, A. Driver Fatigue Detection through Chaotic Entropy Analysis of Cortical Sources Obtained From Scalp EEG Signals. *IEEE Trans. Intell. Transport. Syst.* **2020**, *21*, 185–198.

[6] Doudou, M.; Bouabdallah, A.; Berge-Cherfaoui, V. Driver Drowsiness Measurement Technologies: Current Research, Market Solutions, and Challenges. *Int. J. ITS Res.* **2020**, *18*, 297–319.

[7] Zhao, G.; He, Y.; Yang, H.; Tao, Y. Research on Fatigue Detection Based on Visual Features. *IET image process* **2021**, ipr2.12207.

[8] Sikander, G.; Anwar, S. Driver Fatigue Detection Systems: A Review. *IEEE Trans. Intell. Transport. Syst.* **2019**, *20*,2339–2352.

[9] Narote, S.P.; Bhujbal, P.N.; Narote, A.S.; Dhane, D.M. A Review of Recent Advances in Lane Detection and Departure Warning System. *Pattern Recognit.* **2018**, *73*, 216–234.

[10] Arefnezhad, S.; Samiee, S.; Eichberger, A.; Frühwirth, M.; Kaufmann, C.; Klotz, E. Applying Deep Neural Networks for Multi-Level Classification of Driver Drowsiness Using Vehicle-Based Measures. *Expert Syst. Appl.* **2020**, *162*, 113778.

[11] Forsman, P.M.; Vila, B.J.; Short, R.A.; Mott, C.G.; Van Dongen, H.P.A. Efficient Driver Drowsiness Detection at Moderate Levels of Drowsiness. *Accid. Anal. Prev.* **2013**, *50*, 341–350.

[12] Khushaba, R.N.; Kodagoda, S.; Lal, S.; Dissanayake, G. Driver Drowsiness Classification Using Fuzzy Wavelet-Packet-Based Feature-Extraction Algorithm. *IEEE Trans. Biomed. Eng.* **2011**, *58*, 121–131.

[13] Jiang, Y.; Zhang, Y.; Lin, C.; Wu, D.; Lin, C.-T. EEG-Based Driver Drowsiness Estimation Using an Online Multi-View and Transfer TSK Fuzzy System. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 1752–1764.

[14] Fujiwara, K.; Abe, E.; Kamata, K.; Nakayama, C.; Suzuki, Y.; Yamakawa, T.; Hiraoka, T.; Kano, M.; Sumi, Y.; Masuda, F.; et al. Heart Rate Variability-Based Driver Drowsiness Detection and Its Validation With EEG. *IEEE Trans. Biomed. Eng.* **2019**, *66*, 1769–1778.

[15] Chui, K.T.; Tsang, K.F.; Chi, H.R.; Ling, B.W.K.; Wu, C.K. An Accurate ECG-Based Transportation Safety Drowsiness Detection Scheme. *IEEE Trans. Ind. Inform.* **2016**, *12*, 1438–1452.

[16] Jiao, Y.; Deng, Y.; Luo, Y.; Lu, B.-L. Driver Sleepiness Detection from EEG and EOG Signals Using GAN and LSTM Networks. *Neurocomputing* **2020**, *408*, 100–111.

[17] Mandal, B.; Li, L.; Wang, G.S.; Lin, J. Towards Detection of Bus Driver Fatigue Based on Robust Visual Analysis of Eye State. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 545–557.

[18] Maior, C.B.S.; Moura, M.J. das C.; Santana, J.M.M.; Lins, I.D. Real-Time Classification for Autonomous Drowsiness Detection Using Eye Aspect Ratio. *Expert Syst. Appl.* **2020**, *158*, 113505.

[19] Jo, J.; Lee, S.J.; Park, K.R.; Kim, I.-J.; Kim, J. Detecting Driver Drowsiness Using Feature-Level Fusion and User-Specific Classification. *Expert Syst. Appl.* **2014**, *41*, 1139–1152.

[20] Dong, Y.; Zhang, Y.; Yue, J.; Hu, Z. Comparison of Random Forest, Random Ferns and Support Vector Machine for Eye State Classification. *Multimed. Tools Appl.* **2016**, *75*, 11763–11783.

[21] Alioua, N.; Amine, A.; Rziza, M. Driver's Fatigue Detection Based on Yawning Extraction. *Int. J. Veh. Technol.* **2014**, *2014*.

[22] Ibrahim, M.M.; Soraghan, J.J.; Petropoulakis, L.; Di Caterina, G. Yawn Analysis with Mouth Occlusion Detection. *Biomedical Signal Processing and Control* **2015**, *18*, 360–369.

[23] Omidyeganeh, M.; Shirmohammadi, S.; Abtahi, S.; Khurshid, A.; Farhan, M.; Scharcanski, J.; Hariri, B.; Laroche, D.; Martel, L. Yawning Detection Using Embedded Smart Cameras. *IEEE Trans. Instrum. Meas.* **2016**, *65*, 570–582.

[24] Mittal, A.; Kumar, K.; Dhamija, S.; Kaur, M. Head Movement-Based Driver Drowsiness Detection: A Review of State-of-Art Techniques. In Proceedings of the 2016 IEEE International Conference on Engineering and Technology (ICETECH); IEEE: Coimbatore, India, March 2016; pp. 903–908.

[25] Jamshidi, S.; Azmi, R.; Sharghi, M.; Soryani, M. Hierarchical Deep Neural Networks to Detect Driver Drowsiness. *Multimed Tools Appl* **2021**, *80*, 16045–16058,

[26] Viola, P.; Jones, M.J. Robust Real-Time Face Detection. *Int. J. Comput. Vision.* ***2004***, *57*, 137-154.

[27] Guo, X.; Li, S.; Yu, J.; Zhang, J.; Ma, J.; Ma, L.; Liu, W.; Ling, H. PFLD: A Practical Facial Landmark Detector. *ArXiv190210859 Cs* **2019**.

[28] He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); IEEE: Las Vegas, NV, USA, June 2016; pp. 770–778.

[29] Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *ArXiv14091556 Cs* **2015**.

[30] Chang, C.-C.; Lin, C.-J. LIBSVM: A Library for Support Vector Machines. *ACM Trans. Intell. Syst. Technol.* **2011**, *2*, 1–27.

[31] Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. ImageNet Large Scale Visual Recognition Challenge. *Int J Comput Vis* **2015.**

[32] Song, F.; Tan, X.; Liu, X.; Chen, S. Eyes Closeness Detection from Still Images with Multi-Scale Histograms of Principal Oriented Gradients. *Pattern Recognition* **2014**, *47*, 2825-2838.

[33] Jiang, X.; Zong, Y.; Zheng, W.; Tang, C.; Xia, W.; Lu, C.; Liu, J. DFEW: A Large-Scale Database for Recognizing Dynamic Facial Expressions in the Wild. *ArXiv200805924 Cs* **2020**.

[34] Massoz, Q.; Langohr, T.; Francois, C.; Verly, J.G. The ULg Multimodality Drowsiness Database (Called DROZY) and Examples of Use. In Proceedings of the 2016 IEEE Winter Conference on Applications of Computer Vision (WACV); IEEE: Lake Placid, NY, USA, March 2016; pp. 1–7.

[35] Åkerstedt, T.; Gillberg, M. Subjective and Objective Sleepiness in the Active Individual. *Int. J. Neurosci.* **1990**, *52*, 29–37.

[36] Ogino, M.; Mitsukura, Y. Portable Drowsiness Detection through Use of a Prefrontal Single-Channel Electroencephalogram. *Sensors* **2018**, *18*, 4477.

[37] Cech, J.; Soukupova, T. Real-time eye blink detection using facial landmarks. Cent. Mach. Perception, Dep. Cybern. Fac. Electr. Eng. Czech Tech. Univ. Prague 2016; pp. 1-8.