

Article

ICONet: A Lightweight Network with Greater Environmental Adaptivity

Wei He ¹, Yanmei Huang ¹ , Zanhao Fu ²  and Yingcheng Lin ^{1,*} 

¹ Chongqing Key Laboratory of Space Information Network and Intelligent Information Fusion, School of Microelectronics and Communication Engineering, Chongqing University, Chongqing 400030, China; hewei007@cqu.edu.cn (W.H.); huangyanmei@cqu.edu.cn (Y.H.)

² Cincinnati Joint Co-op Institute, Chongqing University, Chongqing 400030, China; fuzo@mail.uc.edu

* Correspondence: linye@cqu.edu.cn

Received: 17 November 2020; Accepted: 19 December 2020; Published: 21 December 2020



Abstract: With the increasing popularity of artificial intelligence, deep learning has been applied to various fields, especially in computer vision. Since artificial intelligence is migrating from cloud to edge, deep learning nowadays should be edge-oriented and adaptive to complex environments. Aiming at these goals, this paper proposes an ICONet (illumination condition optimized network). Based on OTSU segmentation algorithm and fuzzy c-means clustering algorithm, the illumination condition classification subnet increases the environmental adaptivity of our network. The reduced time complexity and optimized size of our convolutional neural network (CNN) model enables the implementation of ICONet on edge devices. In the field of fatigue driving, we test the performance of ICONet on YawDD and self-collected datasets. Our network achieves a general accuracy of 98.56% and our models are about 590 kilobytes. Compared to other proposed networks, the ICONet shows significant success and superiority. Applying ICONet to fatigue driving detection is helpful to solve the symmetry of the needs of edge-oriented detection under complex illumination condition environments and the scarcity of related approaches.

Keywords: edge-oriented; illumination condition classification; fatigue driving detection; convolutional neural network

1. Introduction

Considering that 14–20% of traffic accidents are caused by fatigue driving [1], fatigue driving detection is facing urgent needs and with high research significance. An effective fatigue driving detection approach will significantly reduce the consequent traffic accidents.

At present, there are mainly three types of methods to monitor fatigue driving. The first type of method is based on vehicle parameters. These methods focus on the rotation speed of the steering wheel, the change of the offset angle, and the changing frequency of the pedal [2]. The second type is based on physiological characteristics. These methods distinguish the driver's mental state based on the driver's multiple physiological characteristics [3], including blood pressure, pulse, heart rate, EMG (electromyography) signal, and EEG (electroencephalogram) signal. The involved algorithms include logistic regression, support vector machine, the k-nearest neighbor classifier [4,5], and artificial neural network [6]. The third type is based on computer vision. The driver's driving behavior data is collected by a camera installed in the vehicle. These methods are based on relevant algorithms to comprehensively analyze the video data. They mainly focus on the eyes and mouth of the driver [7–9].

There are some noteworthy drawbacks to the first two types of methods. The methods based on vehicle parameters require a lot of installed sensors, which increases the cost. Meanwhile, the sensing and processing lag of related sensors may not provide a real-time result. The methods based on

physiological characteristics require physical contact with the driver and the collected information is private, which may be difficult to be widely utilized due to various individual reasons. Considering the drawbacks of these two types of methods, most research nowadays focuses on the proper utilization of computer vision.

Deep learning has recognized many advances in the past several decades. Artificial intelligence is migrating from the cloud to the edge. Based on deep learning, future fatigue driving detection methods should have high accuracy, great environmental adaptivity, and orient to edge devices.

The motivation of this paper is twofold. First, compared to fatigue detection approaches based on vehicle parameters and physiological characteristics, as well as considering the recognized success of deep learning, the deep learning-based approaches are with less cost and able to be widely spread. Second, the current deep learning-based approaches do not consider both great environmental adaptivity and edge-orientation. This is because most deep learning frameworks are too complicated to be loaded on resource-limited edge devices [10].

In this paper, we propose a novel framework named ICONet (illumination condition optimized network). The main contributions of our work are shown as follows:

1. We propose an illumination condition classification subnet based on the OTSU segmentation algorithm and fuzzy *c*-means clustering algorithm. This subnet focuses on classifying input pictures into three types, including normal daylight, weak daylight, and night.
2. We propose a convolutional neural network subnet based on Haar-like features, AdaBoost algorithm, and the modified LeNet-5 network model. This subnet focuses on drivers' face extraction and behavior classification.
3. We design the two subnets in the ICONet with high modularity. Not limited to fatigue driving detection, ICONet can be applied to other classification problems under various illumination conditions.

The rest of this paper is organized as follows. Section 2 describes the related work. Section 3 introduces the methods used in the ICONet. Section 4 describes the experimental results. Section 5 provides a discussion. Section 6 concludes the whole paper.

2. Related Work

In this section, we explore the current research of fatigue driving detection in the field of deep learning, with regard to the aspect of environmental adaptivity and edge-orientation.

2.1. Deep Learning Approaches Considering Environmental Adaptivity

In the aspect of environmental adaptivity, the interference of changing illumination conditions must be considered. Fatigue driving usually occurs at night or dusk when the illumination condition is weak daylight or night.

Ma, Chau et al. [11] presented a convolutional three-stream network architecture, which integrated current-infrared-frame-based spatial information and achieved an accuracy of 94.68%. Hao et al. [12] presented a parallel convolutional neural network (CNN). The proposed method was based on the different detection characteristics of the same image. CNN was used to automatically complete the feature learning. It was claimed with high robustness to a complex environment. Villanueva et al. [13] used a deep learning algorithm. The proposed system used images captured by the camera to detect patterns in the driver's facial features (eyes closed, nodding/head tilt, and yawning). A deep neural network named SqueezeNet was used for faster model development and retraining. There existed an alarm when drowsy driving was detected. Garcia et al. [14] presented a non-intrusive approach including three stages. The first stage was face detection and normalization. The second stage performed pupil position detection and characterization. The final stage calculated the percentage of eyelid closure (PERCLOS) based on closed eyes information. Songkroh et al. [15] used both vehicle speed and driver behavior to analyze and to determine the risk level of the driver. The proposed system

included facial image preprocessing, facial feature detection, feature classification, and an analysis module. The risk alert yielded an accuracy of 86.30% at any vehicle speed. Yu et al. [16] proposed a condition-adaptive representation learning framework. The framework combined spatio-temporal representation and estimated scene conditions were merged to enhance the discriminative power. Spatio-temporal representation learning extracted features that could simultaneously describe motions and appearances. Scene condition understanding classified the scene conditions related to various situations. Memon et al. [17] built a non-intrusive constant checking framework based on OpenCV. Ma et al. [18] presented a two-stream CNN structure focusing on the night situation and achieved an accuracy of 91.57%.

Despite some good results in methods considering environmental adaptivity, these networks have high computational complexity and cannot be applied to resource-limited edge devices.

2.2. Deep Learning Approaches Considering Edge-Orientation

In the aspect of edge-oriented methods, most previous works faced embedded systems including Raspberry Pi and Android devices including smartphones.

On embedded systems, Gu et al. [19] proposed a convolutional neural network model with multi-scale pooling (MSP-Net) and implemented it on an NVIDIA JETSON TX2 development board. On Raspberry Pi, Sharan et al. [20] proposed a driver fatigue system based on eye states using a convolutional neural network. Ghazal et al. [21] used CNN to perform embedded fatigue detection and achieved an accuracy of 95%. Their proposed approach included video signal spatial processing and deep convolutional neural network classification.

On Android devices including smartphones, Xu et al. [22] presented the Sober-Drive system based on a neural network and achieved an accuracy of 90%. Dasgupta et al. [23] proposed a three-stage drowsiness detection with an accuracy of 93.33%. The three stages included PERCLOS calculation, the voiced to the unvoiced ratio, and touch response, which could generally detect drowsy driving and subsequently raised an alarm. Galarza et al. [24] proposed a surveillance system for real-time driver drowsiness detection and with an accuracy of 93.37%. Jabbar et al. [25] proposed a real-time driver drowsiness detection system. Their saved CNN model was within 75 kilobytes and had an accuracy of 83%.

These edge-oriented methods and systems can effectively perform fatigue driving detection and have been realized on edge devices. However, they are not adaptive to various environmental conditions and have relatively low accuracy. Several seconds of fatigue driving may directly cause a fatal traffic accident. An optimized method must detect fatigue driving with high accuracy in whatever environmental condition.

3. Methods

The structure of the illumination condition optimized network is shown in Figure 1. ICONet includes two subnets. The first subnet classifies illumination conditions, and the second subnet classifies related behaviors based on a modified LeNet-5 CNN model. The final stage of ICONet is a comprehensive judgement based on the output results of the two subnets. The whole network is designed to aim at greater environmental adaptivity and implementation on edge devices. For the first goal, our idea focuses on an effective classification. An optimized network is expected to accurately classify images under different illumination conditions, then accordingly call the specific pre-trained model. Each illumination condition type corresponds to a CNN model, which correlates with the symmetry concept. Instead of involving image correction, directly calling the pre-trained models under various conditions will reduce the real-time computing power, which is suitable for edge devices. For the second goal, considering the limited resources on edge devices, we optimize the CNN network structure, remove unnecessary layers, and reduce the convolutional kernel size. The network parameters are carefully adjusted to guarantee high accuracy when applied to fatigue driving detection.

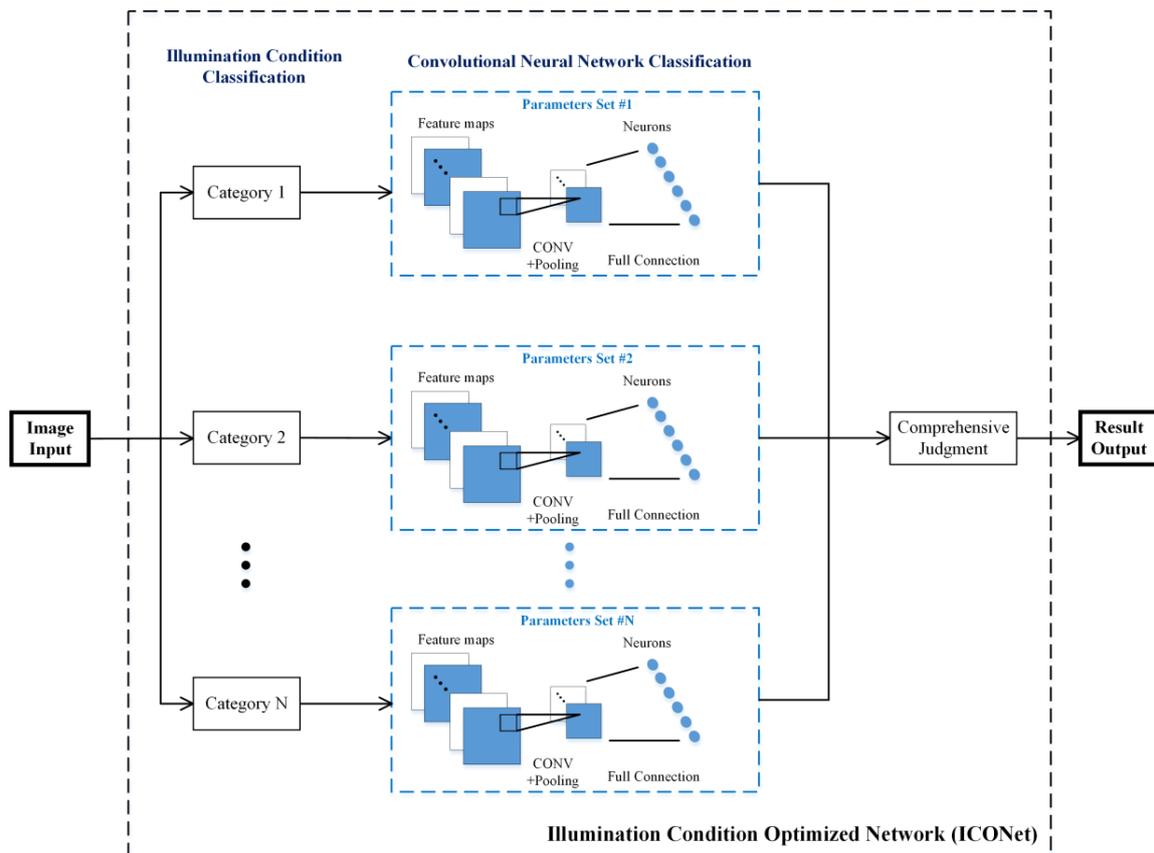


Figure 1. The schematic of the illumination condition optimized network (ICONet).

3.1. Illumination Condition Classification Subnet

This subnet is based on OTSU segmentation algorithm and fuzzy c-means clustering algorithm.

3.1.1. OTSU Segmentation Algorithm

The OTSU segmentation algorithm [26] is used to determine the image binary segmentation threshold value. Based on the one-dimensional histogram of the gray image, it selects the best threshold value and uses this threshold value to divide the entire image into two parts, including the target and the background. The optimal threshold value can maximize the variance between the two parts of the image. The detail of the OTSU segmentation algorithm is described as follows.

Consider a threshold gray value $Thre$ splits the pixels in an image into two groups: C_1 (with pixel gray value $< Thre$) and C_2 (with pixel gray value $> Thre$). The average gray values of C_1 , C_2 pixels are m_1 , m_2 . The average global gray value of the image is m_G . The probabilities of a pixel classified as C_1 or C_2 are p_1 and p_2 .

$$p_1 = \sum_{i=0}^{Thre} p_i \quad (1)$$

$$p_2 = \sum_{i=Thre+1}^{255} p_i \quad (2)$$

$$p_i = \frac{n_i}{n} \quad (3)$$

where n_i is the number of pixels with gray value i .

$$m_1 = \frac{1}{p_1} \sum_{i=0}^{Thre} ip_i \quad (4)$$

$$m_2 = \frac{1}{p_2} \sum_{i=Thre+1}^{255} ip_i \quad (5)$$

$$m_G = \sum_{i=0}^{255} ip_i \quad (6)$$

with constraints in Equations (7) and (8).

$$p_1 m_1 + p_2 m_2 = m_G \quad (7)$$

$$p_1 + p_2 = 1 \quad (8)$$

The variance between these two classes is

$$\sigma^2 = p_1(m_1 - m_G)^2 + p_2(m_2 - m_G)^2 \quad (9)$$

By substituting, we have

$$\sigma^2 = p_1 p_2 (m_1 - m_2)^2 \quad (10)$$

The optimized threshold value will obtain a maximum variance σ^2 among all the 256 gray values. The whole process of the OTSU segmentation algorithm is described in Algorithm 1.

Algorithm 1 OTSU segmentation algorithm

4. **Initial:** $Thre = 0, \sigma_{Thre}^2 = 0$
 5. **for** $i = 0$ **to** 255
 6. Calculate p_1, p_2, m_1, m_2, m_G
 7. Calculate σ_i^2
 8. **if** $\sigma_i^2 > \sigma_{Thre}^2$
 9. $\sigma_{Thre}^2 = \sigma_i^2$
 10. $Thre = i$
 11. **end if**
 12. **end for**
 13. **return** $Thre, \sigma_{Thre}^2$
-

3.1.2. Fuzzy C-Means Clustering Algorithm

The fuzzy c-means clustering algorithm [27] introduces the fuzzy theory to provide more flexible clustering results than normal hard clustering. The algorithm assigns a weight to each object and each cluster. It indicates the degree to which the object belongs to the cluster. Pixels in the image are divided into many disjoint sets based on the characteristic distance of each pixel. This distance reflects the similarity among all the pixels. Consider $X = \{x_1, \dots, x_n\}$ as a set of n objects, and $V = \{v_1, \dots, v_k\}$ is the set of centers of the k clusters. U is a $k \times n$ partition matrix, where u_{ij} is the membership degree of a sample x_j to the cluster center v_i . All cluster centers and the membership of the pixels can be obtained by minimizing the target function. The target function is

$$J(U, V) = \sum_{i=1}^k \sum_{j=1}^n u_{ij}^m d_{ij}^2 = \sum_{i=1}^k \sum_{j=1}^n u_{ij}^m \|x_j - v_i\|^2 \quad (11)$$

$$\sum_{j=1}^k u_{ij} = 1, 0 \leq u_{ij} \leq 1 \quad (12)$$

where n represents the number of samples, k represents the number of clusters, fuzzy weight index $m \in (1, +\infty)$. m usually equals to 2. $d_{ij} = \|x_j - v_i\|$ is the spatial Euclidean distance from the pixel x_j to the cluster center v_i .

$$v_i = \frac{\sum_{j=1}^n u_{ij}^m x_j}{\sum_{j=1}^n u_{ij}^m}, 1 \leq i \leq k \quad (13)$$

$$u_{ij} = \frac{1}{\sum_{p=1}^k \left(\frac{d_{ij}}{d_{pj}}\right)^{\frac{2}{m-1}}}, 1 \leq i \leq k, 1 \leq j \leq n \quad (14)$$

The membership degrees are initially randomly assigned. The cluster center and membership in each following iteration are calculated based on Equations (13) and (14). The iterative process will stop if and only if

$$|U^{(t)} - U^{(t-1)}| = \max\left\{|u_{ij}^{(t)} - u_{ij}^{(t-1)}|\right\} > \varepsilon, 1 \leq i \leq k, 1 \leq j \leq n \quad (15)$$

where $\varepsilon > 0$ and is preassigned. The whole process of fuzzy c-means clustering is described in Algorithm 2.

Algorithm 2 Fuzzy C-Means Clustering algorithm

1. **Initial:** m, ε, k
 2. Randomly initialize the partition matrix U
 3. **while** $|U^{(t)} - U^{(t-1)}| < \varepsilon$ **do**
 4. Calculate $v_i, 1 \leq i \leq k$
 5. Calculate $u_{ij}, 1 \leq i \leq k, 1 \leq j \leq n$
 6. **end while**
-

After performing a lot of tests, we observed that the OTSU threshold value, the ratio of OTSU threshold and average gray value, and the minimum of the target function reflects the illumination condition of an image.

3.2. Convolutional Neural Network Classification Subnet

When applied in the field of fatigue driving detection, this subnet involves face detection and the extraction of the region of interest at the beginning. Face detection in our model is based on Haar-like features and the AdaBoost algorithm [28]. The Haar-like feature is a simple rectangular feature in the face detection system. It is defined as the difference of the pixel's global gray value in adjacent areas of an image. The rectangular feature can reflect the gray changes of the local features of the detected object. The introduction of the integral images accelerates the feature acquisition speed of the detector. The basic idea of the AdaBoost algorithm is to superpose lots of weak classifiers to be a strong classifier with strong classification ability. Then, several strong classifiers are connected in series to complete image retrieval.

We modified the traditional LeNet-5 model, and the structure of the proposed CNN is shown in Figure 2. For the input image, our model uses thirty-two convolution kernels with 5×5 size, one pooling layer with 2×2 size, sixteen convolution kernels with 5×5 size, one pooling layer with 2×2 size, and three full connection layers.

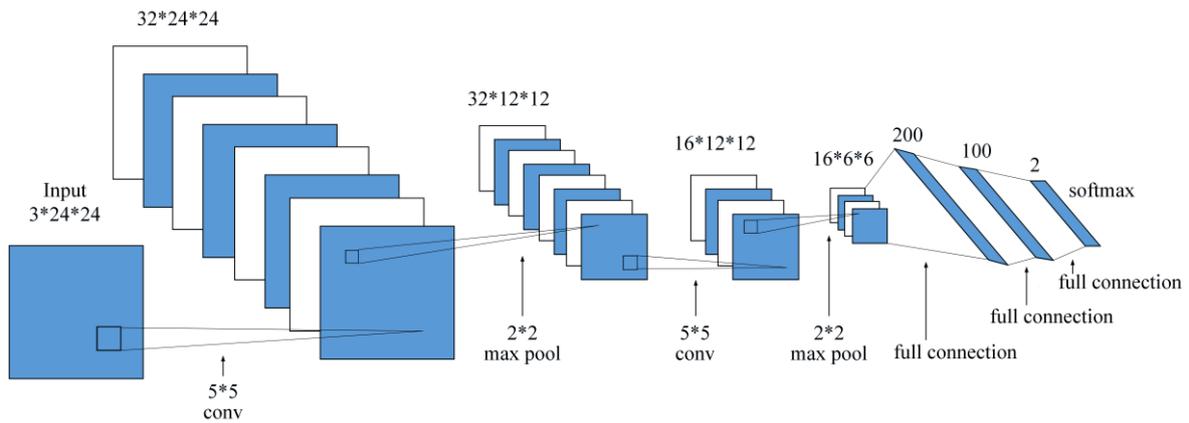


Figure 2. The structure of the proposed convolutional neural network.

There is a SoftMax layer after the last full connection layer. The SoftMax function is defined in Equation (16).

$$S(z)_i = \frac{e^{z_i}}{\sum_{j=1}^j e^{z_j}}, i = 1, 2, \dots, j \quad (16)$$

The loss function based on SoftMax cross-entropy is defined in Equation (17).

$$Loss = - \sum_i y_i \ln s_i \quad (17)$$

where y_i represents the label, and s_i represents the predicted probability.

We performed regularization in loss function to improve the generalization ability of the model and avoid over-fitting problems [29]. Indicators that can reflect the complexity of the model are added to the loss function. If the loss function used to describe the performance of the model on training data is $J(\theta)$, then the optimized target function is $J(\theta) + \lambda R(w)$. $R(w)$ represents the complexity of the model, and λ represents the proportion of the model's complexity loss in the total loss. θ represents all the parameters in a neural network, including the weight w and the bias term b . The L2-norm regularization formula [30] used in this paper is shown in Equation (18).

$$R(w) = \|w\|_2^2 = \sum_i |w_i^2| \quad (18)$$

The time complexity (floating-point operations per second) of convolutional layers in a CNN model is defined in Equation (19).

$$FLOPs_{CONV} = O\left(\sum_{l=1}^D M_l^2 K_l^2 C_{l-1} C_l\right) \quad (19)$$

where D is the depth of the network; l is the l^{th} convolution layer; M is the length of the feature map belonged to each convolution kernel; K is the size of convolution kernel; C_l is the number of output channels of the l^{th} convolution layer.

As for a fully connected layer, consider the dimension of input data is (N, D) , the weight dimension of a hidden layer is (D, out) , and the dimension of output data is (N, out) . Then, the time complexity (FLOPs) of a fully connected layer in a CNN model is defined in Equation (20). We compare the time complexity of ICONet with other approaches in Section 4.3.2.

$$FLOPs_{FC} = (2D - 1) \cdot out \quad (20)$$

3.3. Comprehensive Judgement

Applying in the field of fatigue driving detection, the comprehensive judgement stage in ICONet is about fatigue judgement. Fatigue is closely related to the frequency of eyes and mouth closing.

PERCLOS [31] calculates the ratio of the eye closure frames within a certain period and then infers the driver's eye closure frequency. PERCLOS can be calculated by:

$$f_{PERCLOS} = \frac{n_{close}}{n_{close} + n_{open}} \times 100\% \quad (21)$$

Similar to PERCLOS, FOM (frequency of mouth) [32] calculates the ratio of mouth open within a certain period and then infers the driver's yawn frequency. FOM is calculated by:

$$f_{FOM} = \frac{n_{open}}{n_{close} + n_{open}} \times 100\% \quad (22)$$

For an input video, a certain frequency range is required for accurately calculating fatigue parameters. After a certain frame is detected by ICONet, according to the first-in-first-out principle, the latest result is added to the first place of the queue and the last value of the queue is removed. The total number of frames in the queue remains constant. In Figure 3, "1" represents a closed mouth or eye, and "0" represents an opened mouth or eye.

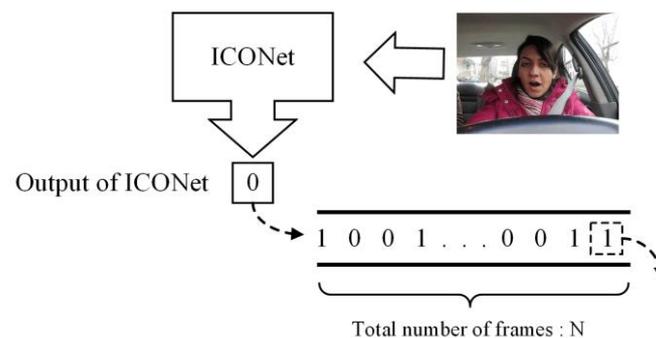


Figure 3. Work of frequency queue.

Combining the frequency queue with PERCLOS and FOM parameters, we can judge whether a driver is fatigued driving.

4. Experimental Results

All experiments were conducted on a computer with Intel(R) Core™ i7-10750H CPU @2.6GHz, 16.0GB RAM, NVIDIA GeForce RTX 2060, and Windows 10. The algorithms were developed in Python 3.6 via OpenCV 3.3.1 and TensorFlow 1.13.1.

4.1. Dataset

4.1.1. YawDD Dataset

YawDD dataset [33] includes two video datasets of car drivers' behaviors in the car. Fatigue driving behaviors are involved. These drivers include males and females, with and without glasses, and from different races. In the 322 videos of the first video set, the camera is installed under the front mirror of the car. In the 29 videos of the second video set, the camera is installed on the dashboard.

4.1.2. CEW (Closed Eyes in the Wild) Dataset

The CEW dataset [34] is about eye state detection under normal daylight. It includes 2423 volunteers and 1192 of them are with eyes closed. There are 2462 images with open eyes and 2384 images with closed eyes.

4.1.3. Self-Collected Dataset

Our network focuses on environments with complex illumination conditions, however, YawDD and CEW datasets are about various driving behaviors under normal daylight. There are currently no open-access datasets focusing on environments with different illumination conditions. Thus, we collected our dataset by an infrared camera including fatigue driving behaviors under weak daylight and night.

The self-collected data includes 15 drivers. Each driver has 1 or 2 videos, depending on whether they wear glasses or not. Normal driving and yawning behaviors are involved in each video. The video is captured at 30 frames per second, and the frame height and width are 1920×1080 .

4.2. Illumination Condition Classification Subnet

We have mentioned in Section 3.1 that four threshold values can represent the illumination condition of a picture. The four threshold values include the OTSU threshold value, average gray value, the ratio of OTSU threshold value and average gray value, and a minimum of target function in fuzzy c-means clustering. We classified the illumination condition types of pictures into normal daylight, weak daylight, and night. Based on YawDD and the self-collected dataset, we captured the videos according to a certain frame rate. We randomly picked 70% of the pictures to calculate the threshold ranges. The remaining 30% were used to verify the accuracy. The distribution of related parameters in the set of pictures used for calculation is shown in Figure 4.

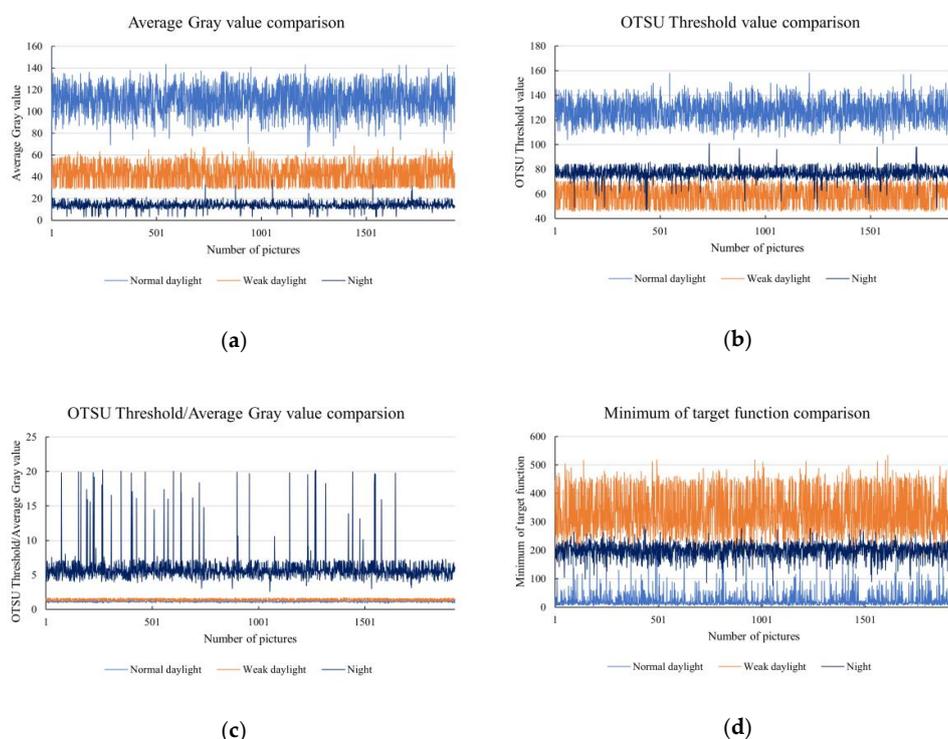


Figure 4. The distribution of related parameters used for illumination condition classification. (a) Average gray value; (b) OTSU threshold value; (c) the ratio of OTSU threshold value and average gray value; (d) minimum of the fuzzy c-means clustering algorithm target function.

Figure 4 indicates that under three types of illumination condition, the related parameters are within a certain range. After comparing the distribution of related parameters, we can obtain that the condition for normal daylight is *Average Gray value* $\in [70, 150]$ && *OTSU Threshold value* $\in [100, 160]$ && *ratio* $\in [0.8, 1.6]$ && *Minimum of target function* $\in [0, 150]$, the condition for weak daylight is *Average Gray value* $\in [25, 70]$ && *Minimum of target function* $\in [200, 550]$, the condition for the night is *Average Gray value* $\in [0, 25]$ && *OTSU Threshold value* $\in [40, 90]$ && *ratio* $\in [2, 20]$. To test the accuracy of the method, we performed this method on the test datasets mentioned in Section 4.1. The results are shown in Table 1.

Table 1. The accuracy of illumination condition classification.

Illumination Condition	Accuracy	General Accuracy
Normal daylight	98.55%	
Weak daylight	96.61%	98.31%
Night	99.76%	

According to Table 1, the proposed illumination condition classification subnet achieves a general accuracy of 98.31% when classifying the various illumination conditions. This subnet works as a pre-classification stage and leads to a well-directed behavior classification in the next subnet.

4.3. Convolutional Neural Network Classification Subnet

4.3.1. Experimental Results

In the CNN classification subnet, we first performed face detection and region of interest extraction based on the AdaBoost algorithm and Haar-like feature. The process is shown in Figure 5. In the classification process, the related CNN models are pre-trained. We used CEW and self-collected datasets to train eye and mouth models based on the proposed CNN model. The ratio of the training set and testing set was 70% to 30%.

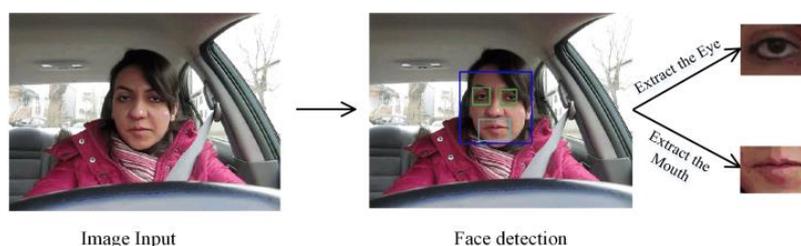


Figure 5. Face detection and extraction of the region of interest.

In the training process, our model first resizes the input picture to 24×24 , then optimizes the model based on the stochastic gradient descent (SGD) method and updates the parameters of the neural network. The batch size of the neural network is 120 and the learning rate is 0.001. Based on the CEW dataset, we trained the eye models under different illumination conditions. The mouth models were trained based on YawDD and self-collected datasets. The models' loss and accuracy of training and testing are shown in Figures 6 and 7.

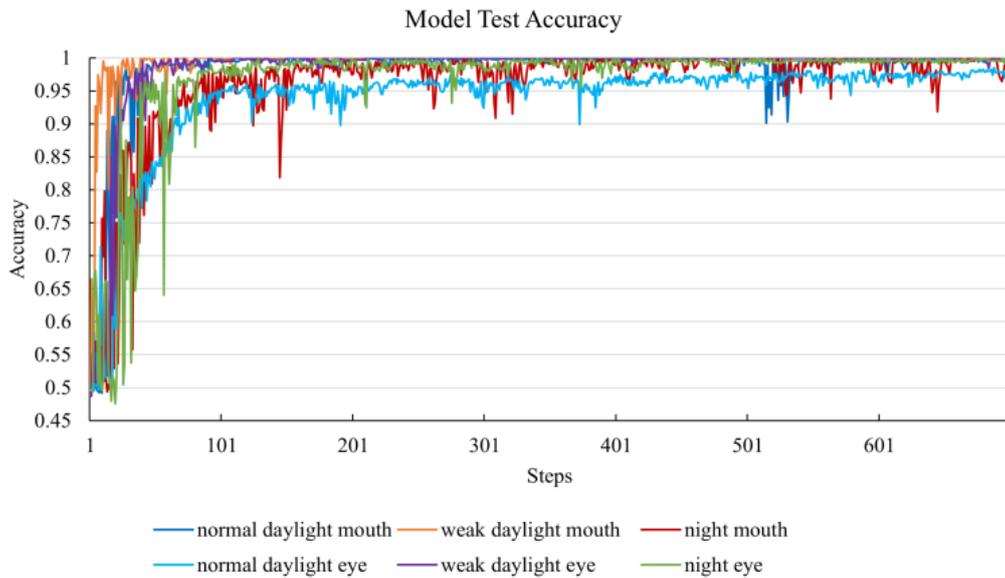


Figure 6. The test accuracies of all the involved convolutional neural network (CNN) models in ICONet.

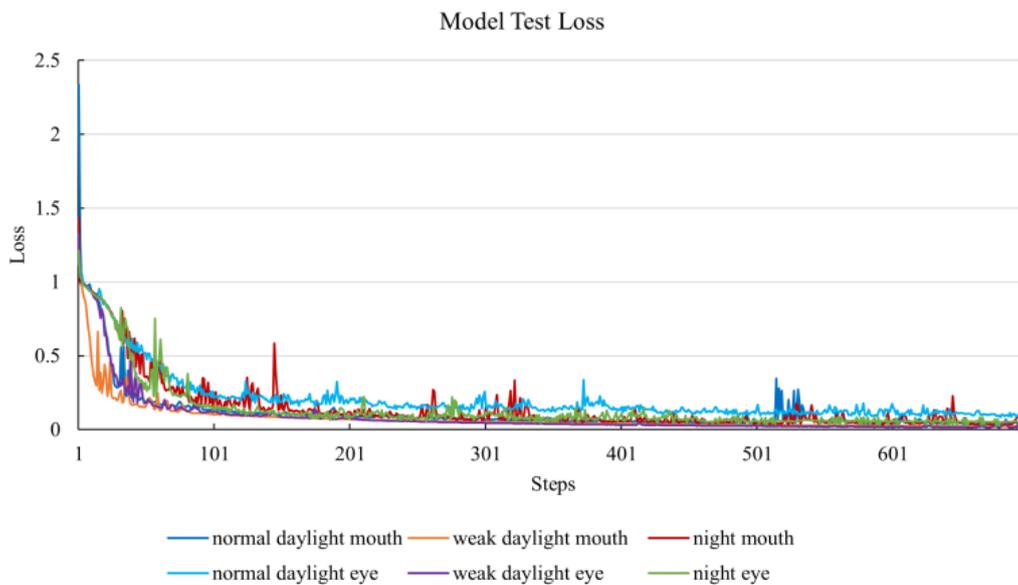


Figure 7. The test loss of all the involved CNN models in ICONet.

Besides, we mixed the dataset under different illumination conditions and trained a hybrid model. The involvement of this hybrid model works as an ablation study. We expected to earn a higher accuracy of our subnet than the hybrid model. It will demonstrate the necessity and effectiveness of the involvement of the illumination condition classification subnet.

Under the circumstances of inputting pictures captured from a video, we compared the accuracy of this hybrid model with models involving illumination condition classification. The result is shown in Table 2.

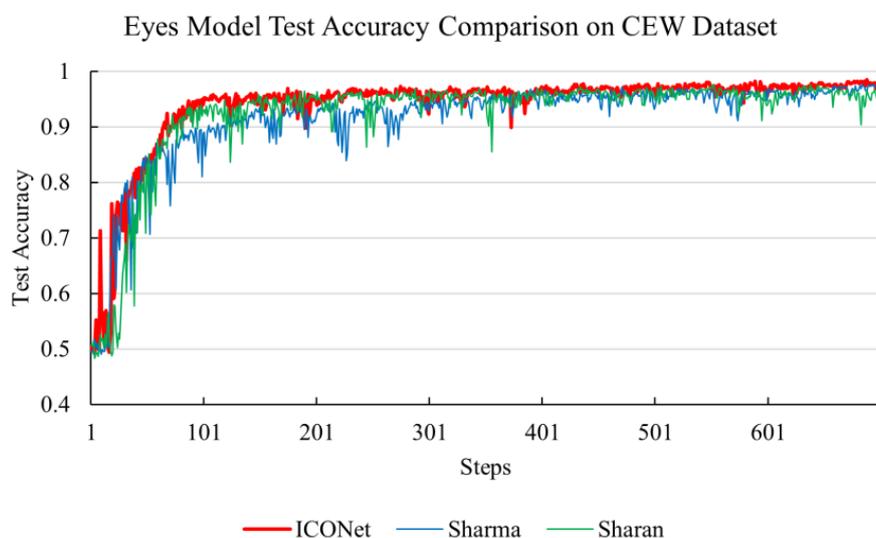
Table 2. Accuracy comparison of the hybrid model and models under different illumination conditions.

Illumination Condition	Eyes Classification Accuracy	Mouth Classification Accuracy	General Accuracy
Normal daylight	97.43%	99.65%	98.56%
Weak daylight	98.21%	98.57%	
Night	98.93%	98.67%	
Hybrid	94.24%	92.45%	93.33%

According to Table 2, the classification of illumination conditions earns a 5.23% superiority of accuracy. In other words, the involvement of the first subnet can guarantee at least one more correct behavior classification in every twenty detections. It should be noted that an accident may occur only after several seconds of fatigue driving. A more accurate driving behavior classification will result in an earlier notification if the proposed model is implemented on the vehicular devices, which will reduce the possibility of a tragic accident.

4.3.2. Comparison with Other Approaches

Additional to the ablation study, we performed comparisons between the proposed ICONet and other approaches. Since the training of all the models involves a self-collected dataset, we first performed a comparison of the eye models on the public CEW dataset, in order to strengthen the persuasiveness and demonstrate the superior ability of our proposed network. As is introduced in Section 4.1.2, the CEW dataset only includes eye images under normal daylight. The comparative results are shown in Figure 8 and Table 3.

**Figure 8.** The model test accuracy comparison on the CEW (Closed Eyes in the Wild) dataset.**Table 3.** Accuracy comparison of testing on the CEW dataset.

Author or Network	Year	Accuracy
Sharma [35]	2018	97.80%
Sharan [20]	2019	96.56%
ICONet	2020	98.30%

We reproduced the network proposed by Sharma [35] and Sharan [20]. According to Figure 8 and Table 3, our network earns a superior accuracy of at least 0.5%. The results indicate that regardless of the illumination condition classification, simply utilizing the second subnet of our network can

comparatively obtain a better result. This proves that our improvements to the traditional LeNet-5 framework are effective and essential.

In the literature [12,13], the authors noted that their networks could be applied under different illumination conditions. To prove the superiority of the proposed ICONet, we reproduced several models in the two pieces of literature and compared them with ICONet. The results are shown in Figure 9.

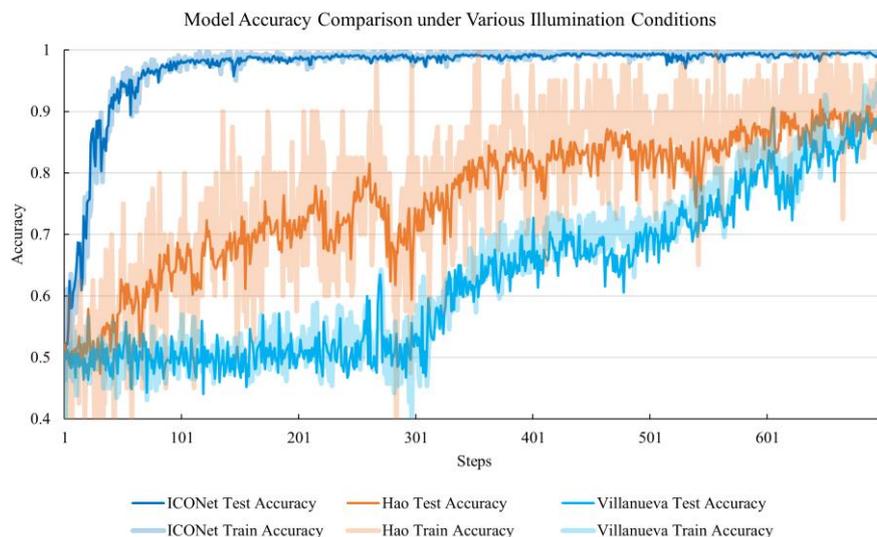


Figure 9. Model accuracies comparison among ICONet and the other two approaches.

Figure 9 compares the changing of test and train accuracies among the three networks. It can be observed that previous approaches are with high training cost, while ICONet requires approximately 100 steps to reach a stable accuracy. When comparing the final stable accuracy, the results demonstrate that although previous approaches can be utilized in various illumination conditions, they are not designed with superior environmental adaptivity compared to ICONet.

Verification of the proposed network with high accuracy on a computer or a server is not the final step. Instead, the resources-limited vehicular devices are closer to a driver in a real scenario. Aiming at the network implementation on edge devices, we compare the model size and time complexity (FLOPs) of ICONet with other approaches. A network with lower time complexity and smaller model size can perform better on edge devices and comparatively without serious lag. The time complexity (FLOPs) is calculated based on Equations (19) and (20). The results are shown in Table 4.

Table 4. Time complexity and model size comparison.

Approach or Network	Involved Dataset		Time Complexity $FLOPs_{CONV}$ (Convolutional Layers)	Time Complexity $FLOPs_{FC}$ (Fully Connected Layers)	Model Size (Kilobytes)
	CEW	YawDD and Self-Collected			
Sharma [35]	✓		3,888,000	1,801,498	3624.96
Sharan [20]	✓		15,206,400	1,704,958	14,725.12
Hao [12]		✓	1,525,678,080	26,210,044	65,873.92
Villanueva [13]		✓	835,794,912	/	4915.20
ICONet	✓	✓	3,225,600	270,498	592.72

According to Table 4, ICONet has superiority in convolutional layer time complexity of at least 17.04%, and a fully connected layer time complexity of at least 84.13%. Working as a lightweight network, ICONet is capable to be loaded on edge devices.

4.4. Comprehensive Judgement

The illumination condition classification subnet and the convolutional neural network subnet can only provide the classification results of mouth and eyes behavior under various illuminations. The comprehensive judgement stage is designed to combine these classification results and determine whether a driver is in fatigue driving. This determination is based on the PERCLOS and FOM threshold values. The process is shown in Figure 10.

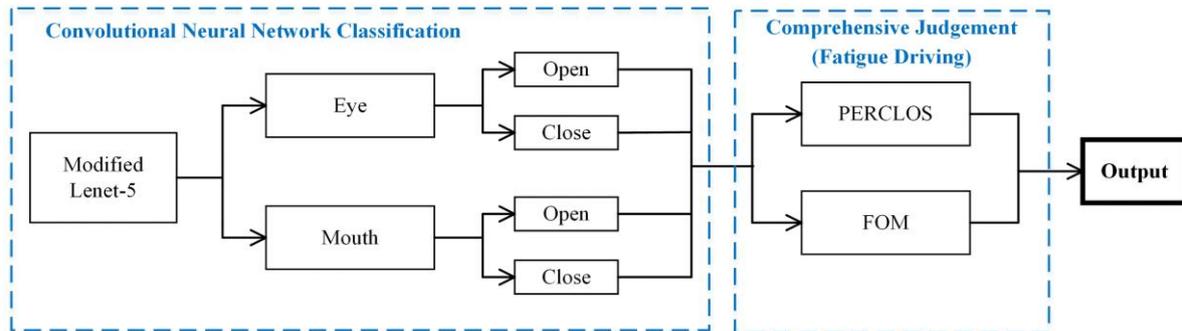


Figure 10. The comprehensive judgement of fatigue driving.

In the process of fatigue driving detection, only focusing on one characteristic is inaccurate. For example, a driver may blink with high frequency or close eyes for a long time under intense illumination. Our approach combines the characteristics of both eyes and mouth.

Figure 11 shows the eye and mouth results during yawning, where “1” represents eye or mouth closing, and “0” represents eye or mouth opening. From the 30th captured frame to the 50th captured frame, the driver can be considered to be fatigue driving. According to related literature [23] and experiments, we set the threshold value as $f_{PERCLOS_Threshold} = 0.25$, $f_{FOM_Threshold} = 0.2$. A driver can be considered fatigue driving if $f_{PERCLOS} \geq 0.25$ or $f_{FOM} \geq 0.2$, i.e., in 100 continuously captured frames, there are at least 25 frames closing eyes or 20 frames opening mouth. Based on the threshold value range, we performed tests on YawDD and self-collected datasets. The result is shown in Figure 12.

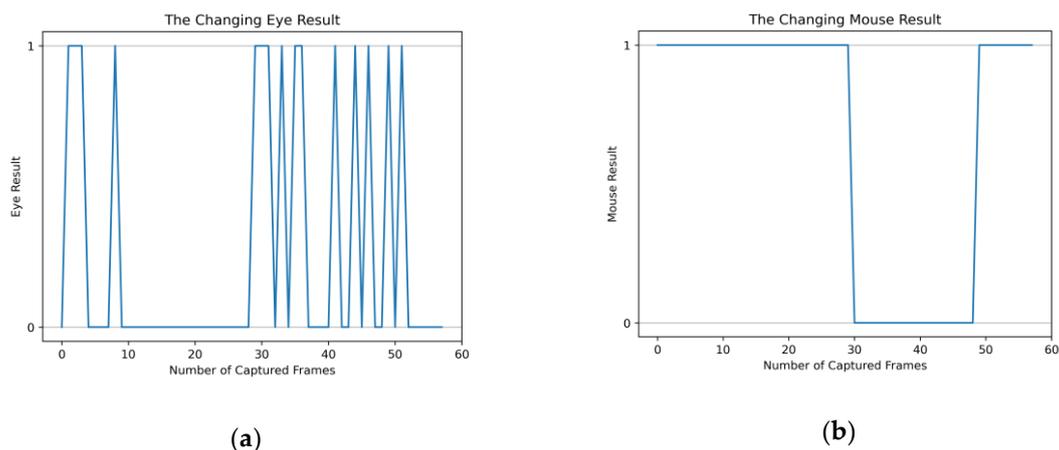


Figure 11. The changing eye and mouth results when a yawn occurs. (a) The changing eye result; (b) the changing mouth result.

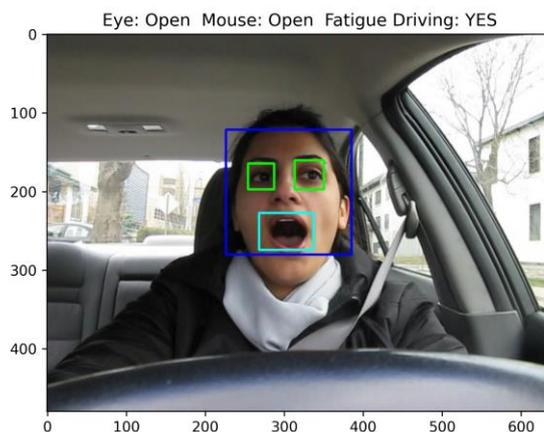


Figure 12. Fatigue driving detection under normal daylight.

Figure 12 demonstrates that under the high accuracy of the previous two subnets, ICONet can effectively judge fatigue driving.

5. Discussion

In the field of deep learning-based fatigue driving detection, previous works have achieved some significant success. However, with the migration of artificial intelligence from the cloud to the edge, fatigue driving detection is required to be effectively loaded on edge devices and have high accuracy under various environments. Most previous works mainly focus on one aspect and perform undesirably when being applied to other aspects.

Aiming at a greater environmental adaptivity, some pieces of literature design complicated network structures [12,13], while other approaches attempt to obtain a desirable result based on a large training set. The proposed illumination condition classification subnet in this paper is based on traditional image processing algorithms. The experimental results prove the effectiveness of our framework.

Aiming at the implementation on edge devices, we modified the LeNet-5 model, which is the most lightweight framework among all the CNN frameworks. The convolutional layers and kernels are optimized to obtain a lower time complexity. The experimental results comparatively demonstrate the compactness of ICONet.

Additional to the presented results, it should be noted that ICONet is designed to be a universal network, which works as a general solution towards classification problems in various illumination condition environments. Besides fatigue driving detection, it has the potential to be applied to other fields including but not limited to traffic classification, human activity classification, and classification problems in medical science and agriculture.

By comparing with other proposed networks [14,36–38], ICONet has several limitations, which guides the direction of our future study.

In the illumination condition classification subnet, we focused on the natural illumination conditions and classified the input pictures into normal daylight, weak daylight, and night. However, the complex illumination conditions may also include different luminance and location of the light source. Especially at night, the suddenly exerted intense light may affect the classification result. When considering these factors, the subnet may fail to classify well. In future work, we will focus on the classification of unnatural illumination conditions.

In the convolutional neural network classification subnet, the model focuses on a single person, which is the driver. However, there may exist more than one person in real scenarios, including the copilot, passengers, and people outside the vehicle. The subnet may fail to correctly detect the driver's face under these circumstances. We will involve additional image processing algorithms in our future work.

Despite the mentioned limitations, ICONet provides a reference for designing a multi-subnet framework. With the sharp increase of edge devices, ICONet works as an attempt towards the future development of edge-oriented deep learning.

6. Conclusions

Artificial intelligence is migrating from cloud to edge, and deep learning is required to be edge-oriented and adaptive to complex environments. In this paper, we proposed an illumination condition optimized network (ICONet) and applied it to fatigue driving detection. Based on the OTSU segmentation algorithm and fuzzy c-means clustering algorithm, the illumination condition classification subnet classifies pictures under normal daylight, weak daylight, and night. After the face detection and the extraction of the region of interest, the CNN classification subnet provides the classification results of eyes and mouth based on the modified LeNet-5 model. According to indicators including PERCLOS and FOM, ICONet can comprehensively judge fatigue driving. ICONet achieves a general accuracy of 98.56% and time complexity is reduced by at least 17.04% compared to the previous work. The size of all the CNN models is about 590 kilobytes. Experimental results demonstrate the feasibility of applying ICONet on edge devices under various illumination conditions in fatigue driving detection.

In our future work, besides solving the mentioned limitations, we will transplant the ICONet to the Android platform and test it on onboard devices. Additionally, we will add more driving behaviors and further optimize our model to improve its environmental adaptivity.

Author Contributions: Conceptualization, methodology, software, validation, formal analysis, investigation, data curation, visualization, writing—original draft preparation, Y.H. and Z.F.; resources, writing—review and editing, supervision, project administration, funding acquisition, W.H. and Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key Research and Development Program of China, grant number 2020YFC0832700.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Amodio, A.; Ermidoro, M.; Maggi, D.; Formentin, S.; Savaresi, S.M. Automatic detection of driver impairment based on pupillary light reflex. *IEEE Trans. Intell. Transp. Syst.* **2018**, *20*, 3038–3048. [[CrossRef](#)]
2. Shahverdy, M.; Fathy, M.; Berangi, R.; Sabokrou, M. Driver behavior detection and classification using deep convolutional neural networks. *Expert Syst. Appl.* **2020**, *149*, 113240. [[CrossRef](#)]
3. Chen, L.-L.; Zhao, Y.; Zhang, J.; Zou, J.-Z. Automatic detection of alertness/drowsiness from physiological signals using wavelet-based nonlinear features and machine learning. *Expert Syst. Appl.* **2015**, *42*, 7344–7355. [[CrossRef](#)]
4. Gwak, J.; Shino, M.; Hirao, A. Early detection of driver drowsiness utilizing machine learning based on physiological signals, behavioral measures, and driving performance. In Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, Hawaii, 4–7 November 2018; pp. 1794–1800.
5. Murugan, S.; Selvaraj, J.; Sahayadhas, A. Detection and analysis: Driver state with electrocardiogram (ECG). *Phys. Eng. Sci. Med.* **2020**, *43*, 525–537. [[CrossRef](#)] [[PubMed](#)]
6. Mardi, Z.; Ashtiani, S.N.M.; Mikaili, M. EEG-based drowsiness detection for safe driving using chaotic features and statistical tests. *J. Med. Signals Sens.* **2011**, *1*, 130. [[PubMed](#)]
7. Chaisiriprasert, P.; Yongsiriwit, K. Surveillance System for Abnormal Driving Behavior Detection. In Proceedings of the 2019 4th International Conference on Information Technology (InCIT), Bangkok, Thailand, 24–25 October 2019; pp. 155–158.
8. Zhang, F.; Su, J.; Geng, L.; Xiao, Z. Driver Fatigue Detection Based on Eye State Recognition. In Proceedings of the 2017 International Conference on Machine Vision and Information Technology (CMVIT), Singapore, 17–19 February 2017; pp. 105–110.

9. Maior, C.B.S.; Moura, M.J.d.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. [[CrossRef](#)]
10. Elhassouny, A.; Smarandache, F. Trends in deep convolutional neural Networks architectures: A review. In Proceedings of the 2019 International Conference of Computer Science and Renewable Energies (ICCSRE), Agadir, Morocco, 22–24 July 2019; pp. 1–8.
11. Ma, X.; Chau, L.-P.; Yap, K.-H.; Ping, G. Convolutional three-stream network fusion for driver fatigue detection from infrared videos. In Proceedings of the 2019 IEEE International Symposium of Circuits and Systems (ISCAS), Sapporo, Japan, 26–29 May 2019; pp. 1–5.
12. Hao, Z.; Wan, G.; Tian, Y.; Tang, Y.; Dai, T.; Liu, M.; Wei, R. Research on Driver Fatigue Detection Method Based on Parallel Convolution Neural Network. In Proceedings of the 2019 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS), Shenyang, China, 30–31 May 2019; pp. 164–168.
13. Villanueva, A.; Benemerito, R.L.L.; Cabug-Os, M.J.M.; Chua, R.B.; Rebeca, C.K.D.C.; Miranda, M. Somnolence Detection System Utilizing Deep Neural Network. In Proceedings of the 2019 International Conference on Information and Communications Technology (ICOIACT), Yogyakarta, Indonesia, 24–25 July 2019; pp. 602–607.
14. Garcia, I.; Bronte, S.; Bergasa, L.M.; Almazán, J.; Yebes, J. Vision-based drowsiness detector for real driving conditions. In Proceedings of the 2012 IEEE Intelligent Vehicles Symposium (IV), Alcal de Henares, Madrid, Spain, 3–7 June 2012; pp. 618–623.
15. Songkroh, A.; Kurutach, W. An intelligent hybrid approach for detection of drowsy driving risk in real environments. In Proceedings of the 2017 14th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Phuket, Thailand, 27–30 June 2017; pp. 876–881.
16. Yu, J.; Park, S.; Lee, S.; Jeon, M. Driver Drowsiness Detection Using Condition-Adaptive Representation Learning Framework. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 4206–4218. [[CrossRef](#)]
17. Memon, S.; Memon, M.; Bhatti, S.; Khanzada, T.J.; Memon, A.A. Tracker for sleepy drivers at the wheel. In Proceedings of the 2017 11th International Conference on Signal Processing and Communication Systems (ICSPCS), Surfers Paradise, Australia, 13–15 December 2017; pp. 1–8.
18. Ma, X.; Chau, L.-P.; Yap, K.-H. Depth video-based two-stream convolutional neural networks for driver fatigue detection. In Proceedings of the 2017 International Conference on Orange Technologies (ICOT), Singapore, 8–10 December 2017; pp. 155–158.
19. Gu, W.H.; Zhu, Y.; Chen, X.D.; He, L.F.; Zheng, B.B. Hierarchical CNN-based real-time fatigue detection system by visual-based technologies using MSP model. *Int Image Process.* **2018**, *12*, 2319–2329. [[CrossRef](#)]
20. Sharan, S.S.; Viji, R.; Pradeep, R.; Sajith, V. Driver Fatigue Detection Based On Eye State Recognition Using Convolutional Neural Network. In Proceedings of the 2019 International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 15–16 November 2019; pp. 2057–2063.
21. Ghazal, M.; Abu Haeyeh, Y.; Abed, A.; Ghazal, S. Embedded Fatigue Detection Using Convolutional Neural Networks with Mobile Integration. In Proceedings of the 2018 6th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW), Barcelona, Spain, 6–8 August 2018; pp. 129–133.
22. Xu, L.; Li, S.; Bian, K.; Zhao, T.; Yan, W. Sober-Drive: A smartphone-assisted drowsy driving detection system. In Proceedings of the 2014 International Conference on Computing, Networking and Communications (ICNC), Honolulu, HI, USA, 3–6 February 2014; pp. 398–402.
23. Dasgupta, A.; Rahman, D.; Routray, A. A Smartphone-Based Drowsiness Detection and Warning System for Automotive Drivers. *IEEE Trans. On Intell. Transp. Syst.* **2019**, *20*, 4045–4054. [[CrossRef](#)]
24. Galarza, E.E.; Egas, F.D.; Silva, F.M.; Velasco, P.M.; Galarza, E.D. Real Time Driver Drowsiness Detection Based on Driver's Face Image Behavior Using a System of Human Computer Interaction Implemented in a Smartphone. In Proceedings of the International Conference on Information Technology & Systems (ICITS 2018), Libertad City, Ecuador, 10–12 January 2018; pp. 563–572.
25. Jabbar, R.; Shinoy, M.; Kharbeche, M.; Al-Khalifa, K.; Krichen, M.; Barkaoui, K. Driver drowsiness detection model using convolutional neural networks techniques for android application. In Proceedings of the 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIOT), Doha, Qatar, 2–5 February 2020; pp. 237–242.
26. Chen, Q.; Zhao, L.; Lu, J.; Kuang, G.; Wang, N.; Jiang, Y. Modified two-dimensional Otsu image segmentation algorithm and fast realisation. *Int Image Process.* **2012**, *6*, 426–433. [[CrossRef](#)]

27. Yeom, C.-U.; Kwak, K.-C. Performance Evaluation of Automobile Fuel Consumption Using a Fuzzy-Based Granular Model with Coverage and Specificity. *Symmetry* **2019**, *11*, 1480. [[CrossRef](#)]
28. Zhang, Y.; Jia, Y.; Wu, W.; Cheng, Z.; Su, X.; Lin, A. A Diagnosis Method for the Compound Fault of Gearboxes Based on Multi-Feature and BP-AdaBoost. *Symmetry* **2020**, *12*, 461. [[CrossRef](#)]
29. Nusrat, I.; Jang, S.-B. A Comparison of Regularization Techniques in Deep Neural Networks. *Symmetry* **2018**, *10*, 648. [[CrossRef](#)]
30. Zhang, K.; Su, H.; Dou, Y.; Shen, S. Evaluation of the Influences of Hyper-Parameters and L2-Norm Regularization on ANN Model for MNIST Recognition. In Proceedings of the 2019 International Conference on Intelligent Computing, Automation and Systems (ICICAS), Chongqing, China, 6–8 December 2019; pp. 379–386.
31. Yan, J.-J.; Kuo, H.-H.; Lin, Y.-F.; Liao, T.-L. Real-Time Driver Drowsiness Detection System Based on PERCLOS and Grayscale Image Processing. In Proceedings of the 2016 International Symposium on Computer, Consumer and Control (IS3C), Xi'an, China, 4–6 July 2016; pp. 243–246.
32. Savas, B.K.; Becerikli, Y. Real Time Driver Fatigue Detection System Based on Multi-Task ConNN. *IEEE Access* **2020**, *8*, 12491–12498. [[CrossRef](#)]
33. Abtahi, S.; Omidyeganeh, M.; Shirmohammadi, S.; Hariri, B. YawDD: A yawning detection dataset. In Proceedings of the 5th ACM Multimedia Systems Conference, Singapore, 19–21 March 2014; pp. 24–28.
34. Song, F.; Tan, X.; Liu, X.; Chen, S. Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients. *Pattern Recognit.* **2014**, *47*, 2825–2838. [[CrossRef](#)]
35. Sharma, S.; Negi, A.; Singh, S.; Raj, D.S.S.; Graceline, J.S.; Vaidehi, V.; Ganesan, S. Eye state detection for use in advanced driver assistance systems. In Proceedings of the 2018 International Conference on Recent Trends in Advance Computing (ICRTAC), VIT, Chennai, India, 10–11 September 2018; pp. 155–161.
36. He, L.; Liu, G.; Tian, G.; Zhang, J.; Ji, Z. Efficient Multi-View Multi-Target Tracking Using a Distributed Camera Network. *IEEE Sens. J.* **2020**, *20*, 2056–2063. [[CrossRef](#)]
37. Ranjan, R.; Bansal, A.; Zheng, J.; Xu, H.; Gleason, J.; Lu, B.; Nanduri, A.; Chen, J.-C.; Castillo, C.; Chellappa, R. A Fast and Accurate System for Face Detection, Identification, and Verification. *IEEE Trans. Biom. Behav. Identity Sci.* **2019**, *1*, 82–96. [[CrossRef](#)]
38. Ranjan, R.; Patel, V.M.; Chellappa, R. HyperFace: A Deep Multi-Task Learning Framework for Face Detection, Landmark Localization, Pose Estimation, and Gender Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 121–135. [[CrossRef](#)] [[PubMed](#)]

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).