Research on Fatigue Driving Detection Based on Deep Learning

Huinan Zhou, Hong Dai*

Abstract—**Fatigue driving is one of the main causes of traffic accidents. Effective fatigue driving detection technology can reduce traffic accidents caused by fatigue driving. Traditional fatigue driving detection methods usually use a two-stage detection method, which has the problems of low efficiency and poor accuracy. To solve these problems, this paper designs a single-stage fatigue driving detection model based on separable convolution Road Safety Net (RSNet), which extracts the features in the driver's image or video and classifies these features to determine whether the driver is in an abnormal driving state. RSNet borrows the structure of Swin Transformer and adopts a hierarchical design, which makes the model better able to extract features and classify. Each level is composed of a GDS Layer and multiple RS Blocks. To solve the problem of low efficiency of traditional methods, we use separable convolution in RS Block to replace the traditional convolutional neural network, which reduces the complexity of the model and effectively improves the computational efficiency of the model. Experimental results show that the Top-1 and Top-5 Accuracy of RSNet on the HMDB51 dataset reaches 85 % and 97.35 %, respectively, and the delay is 5.97 ms. Compared with ResNet, EfficientNet, and Swin Transformer, RSNet has higher accuracy and efficiency. In addition, this paper takes the YOLACT algorithm as an example, selects different backbone networks for object detection, and compares the indicators of mean Average Precision (*mAP*) and Frames Per Second (FPS). Experiments show that RSNet performs better when used as the backbone network for object detection, with *mAP* reaching 32.4 % and FPS of 39.6 ms. It indicates that RSNet performs well in action recognition tasks.**

Index Terms—**Fatigue driving detection, Convolutional neural network, Single stage detection, RSNet**

I. Introduction

Fatigued driving has significantly contributed to the rise in traffic accidents in recent years. They were operating their vehicles while fatigued, characterized by prolonged driving without sufficient rest or sleep. This leads to diminished focus, a longer reaction time, and an increased risk of making errors in judgment. Thus, early diagnosis and warning of drowsy driving are critical for reducing accidents. The detection based on physiological signals usually monitors and analyzes the driver's physiological indicators, and the driver's fatigue state can be evaluated by analyzing these physiological signals. Mao et al. identified fatigue levels by analyzing the driver's physiological signals and obtaining feature vectors using wavelet processing. Then, they used a two-step fuzzy cluster analysis to categorize the various weariness levels [1]. Artanto et al. developed a driver sleepiness detection system based on cheaper electromyography and the ESP8266 WIFI module. The technology measures the duration of the driver's eyelid closure by attaching EMG electrodes to the frames of their spectacles. The system alerts drivers when their eyelid closure exceeds a predetermined limit [2]. Sangeetha et al. detected driver weariness using electrocardiogram signals [3]. Physiological signal-based detection approaches often require many sensing devices, which might make drivers uncomfortable. As a result, fatigue detection approaches based on physiological signals have difficulties, such as being uncomfortable in real-world driving situations. Vehicle condition-based detection is another common approach that relies primarily on in-vehicle sensors or monitoring to monitor and analyze the vehicle's behaviors and assess whether the driver is tired. Zhang et al. investigated driving tiredness by examining fatigued drivers' steering wheel operation characteristics. However, using this strategy, the identification rate for fatigued driving was just 82 %, showing a low degree of accuracy [4]. In 2020, Liu L et al. adopted a neural network algorithm to analyze the rotation information of the vehicle steering wheel and the characteristics of vehicle deviation from the road to identify and detect the abnormal driving state of the driver [5]. Li et al. examined the vehicle's deviation from the lane's center line to determine if a driver was fatigued. These methods all have an unavoidable delay, and factors such as personal driving habits and road environment will also affect the detection accuracy [6]. Detection based on the driver's aberrant driving behavior is carried out using monitoring devices fitted in the automobile that record the driver's posture in real time and recognize the driver's abnormal driving behaviors. Hu et al. developed a multi-column convolutional neural network to detect aberrant driving in 2019 [7]. This method performs well in image-based recognition but needs to improve in video-based recognition. Rao et al. proposed a distracted driving recognition approach in 2021 that analyses image data from a camera using a deep convolutional neural network [8]. However, the trials revealed increased the recognition model's training time. Yan C [9] et al. proposed a system that applies a Convolutional Neural Network (CNN) to learn and predict the preset driving posture automatically. By monitoring the position of the driver's hand, safe and unsafe driving postures are predicted. However, the driver's hand is not always on the steering wheel in real-time during driving, and there is a high false detection rate. H. R. Qu [10]

Manuscript received August 17, 2024; revised December 18, 2024.

Huinan Zhou is a postgraduate student of University of Science and Technology Liaoning, Anshan, Liaoning, CO 114051 China (email: 2473767780@qq.com).

Hong Dai* is a professor of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, Liaoning, CO 114051 China (corresponding author to provide phone: +086-186-4226-8599; fax: 0412-5929818; e-mail: dear_red9@163.com).

proposed a method based on machine vision to recognize drivers' fatigued driving and telephoning behaviors. The telephoning behavior detection uses the facial feature point alignment detection. However, during the driving process, the light changes greatly, mainly impacting face detection. As a result, most current approaches focus on identifying abnormal driving conduct to detect driver weariness. This fatigue monitoring approach does not involve physical touch with the driver and does not interfere with their usual driving. Deep-learning target detection algorithms mostly use single-stage and two-stage detection methods to detect abnormal driving behaviors. Single-stage detection directly detects the target from the input image without generating a candidate box, representing algorithms such as the YOLO algorithm [11] and the SSD algorithm [12]. It's simple, effective, and ideal for real-time apps and mobile devices. Two-stage detection requires the formation of candidate boxes, followed by classification and positioning. Common algorithms include Faster R-CNN [13] and Mask R-CNN [14]. The two-stage detection method is more complex and time-consuming than the single-stage method. The traditional fatigue driving detection method usually adopts the two-stage detection method, which has some problems, such as low efficiency and poor accuracy. In their study, He et al. proposed a real-time detection method for driver fatigue based on convolutional neural networks, including a detection and classification stage [15]. The two-stage detection method leads to a long detection time. Yang et al. suggested a method for detecting unsafe driving behaviors using enhanced YOLO V5 and OpenPose [16]. The method used a two-stage approach to recognize and locate the target and predict the emergence of risks. Although this strategy enhances target detection, it has a higher processing cost. The single-stage detection algorithm differs from the two-stage method in recognizing targets directly from the input image. Shakeel et al. used a MobileNet CNN structure with a Single Shot Multibox Detector (SSD) to identify driving weariness [17]. Ma et al. rebuilt the SSD network structure using the inverse residual notion from the SSD algorithm [18]. This change enables real-time monitoring of the driver's face. The single-stage detection method saves time compared to the two-stage method, increasing detection efficiency and accuracy. Thus, this paper adopts a single-stage detection method. The detection approach based on physiological signals requires employing many wearable sensing devices to record the driver's physiological parameters, which may cause discomfort. The detection approach based on vehicle status involves a time delay, and factors such as personal driving behaviors and road conditions will also influence detection accuracy. Furthermore, the two-stage fatigue detection approach requires the generation of candidate frames, followed by their classification and location, which has issues such as low efficiency and accuracy. To solve these problems, the paper introduces a single-stage fatigue driving detection model based on separable convolution, RSNet, that extracts features from driver photos or videos and classifies them to determine whether the driver is in an abnormal driving state.

II. Related Work

A. Swin Transformer

Swin Transformer [21] is a deep learning model based on an attention mechanism commonly used for image classification and object detection applications. Swin Transformer addresses the memory and computational complexity problems faced by traditional Transformer [22] when processing large-size images by introducing a hierarchical attention mechanism. It divides the input image into multiple small blocks and performs hierarchical attention operations on these blocks to process large-scale image data. The primary idea of this model is to capture the link between global and local variables using a hierarchical attention mechanism. It shifts block-level and window-level attention operations in a way known as a shifted window. This makes the model lower computational and memory requirements while maintaining global awareness. In addition, Swin Transformer introduces a cross-stage connection that enhances the expression of features by transferring information between different layers. This cross-stage connection helps the model capture multi-scale feature information better and improve its performance and generalization ability. Swin Transformer is more efficient in computing resources and memory consumption than traditional convolutional neural network models. RSNet uses the network structure of Swin Transformer for reference and adopts a hierarchical design to improve model accuracy and performance and reduce computing costs.

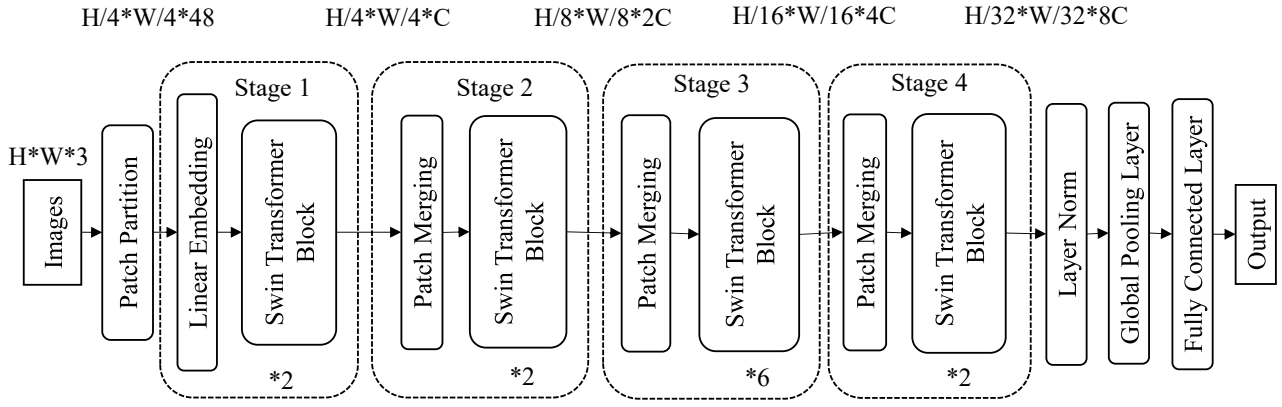

Fig. 1. Structure of Swin Transformer

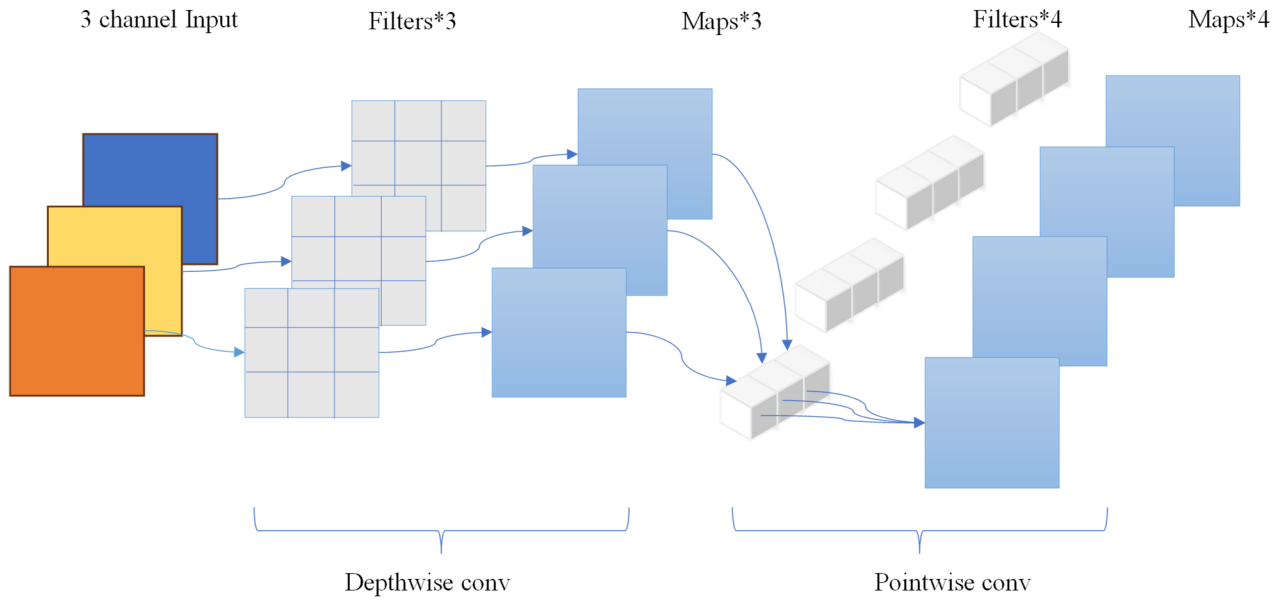

Fig. 2. Structure of separable convolution

B. Separable Convolution

Separable convolution [20] is a convolution operation commonly used in convolutional neural networks, which reduces the number of parameters and computational complexity to some extent while maintaining the expressive ability of the model. The traditional convolution operation consists of two steps: first, the convolution operation is performed on the input feature graph, and then the convolution result is linearly combined by channel. Separable convolution, on the other hand, separates the two steps and processes them separately. Firstly, separable convolution employs deep convolution to convolve the input channel independently, then uses a convolution check to generate the same number of feature graphs for each number of parameters since there is only one convolution kernel per channel. Next, separable convolution uses a point-by-point algorithm to perform a channel-by-channel linear combination of the results of the deep convolution. The 1 1 convolution kernel is used for point-by-point convolution, which operates on channel dimension, linear combination, and feature fusion of deep convolution results. Point-by-point convolution has a relatively small number of parameters, but it can provide a higher-dimensional feature representation. By combining deep convolution and point-by-point convolution, separable convolution can effectively reduce the number of parameters and improve the computational efficiency to some extent. At the same time, it also has a regularization effect, which can reduce the risk of overfitting.

C. Single-Stage Fatigue Detection Methods

The single-stage fatigue detection method detects the fatigue state directly in one stage. In single-stage fatigue detection, convolutional neural networks [19] collect information from photos or videos and input them into a classifier to determine the fatigue state. Convolutional neural networks are a deep learning model commonly utilized in image processing and computer vision applications. The method first provides the input image to the convolutional neural network. Basic layer architectures, such as convolution and pooling, extract local and high-level semantic characteristics from images. The convolution layer extracts visual features using a set of learnable convolution kernels on the local receptive field. The pooling layer minimizes the feature map's size while preserving its most significant features. The extracted features are then sent into the fully connected layer, which performs more complex feature representation and fatigue state classification or regression. The fully connected layer will learn to map image features to the associated fatigue state outputs. By using a convolutional neural network for fatigue detection, the method can automatically learn and extract the relevant features in the image to accurately determine the fatigue state.

III. Proposed Method

A. Single-Stage Fatigue Driving Detection Model Based On Separable Convolution: RSNet

RSNet is a single-stage fatigue detection model based on separable convolution, which borrows the network structure of Swin Transformer. The whole model adopts a hierarchical design and contains four stages. Each stage reduces the resolution of the input feature map and expands the receptive field layer by layer, like CNN. First, the input image passes through the Stem layer, whose primary function is to extract the features of the input data and provide a more abstract and informative representation for the following network layer. Then, it passes through various stages, each comprising a GDS Layer and several RS Blocks. The GDS Layer is a network layer designed to reduce the spatial dimension of the input data while preserving as much important information as possible. The network structure is down-sampled through a packet convolution layer, followed by channel mixing through a point convolution layer. Using separable convolution in RS Block instead of typical convolutional neural networks, the complexity of the model reduces the danger of overfitting and improves computational performance. Finally, it will go through Global Average Pooling (GAP), a 1×1 convolution, and a fully connected layer. The main function of GAP is to down

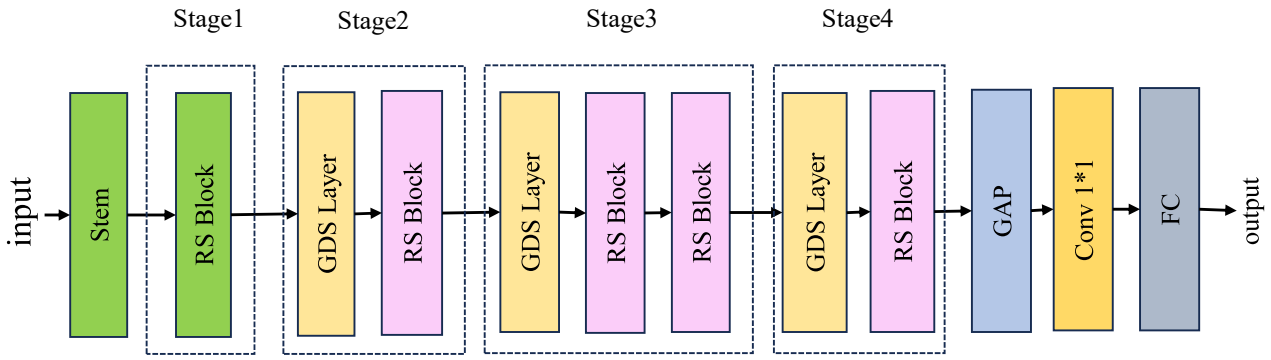

Fig. 3. Structure of the RSNet

sampling the feature map and simplify the spatial dimension of each feature map to a single average value, which significantly reduces the dimension of the feature map. It helps reduce the number of parameters and calculation amount of the model while reducing the risk of overfitting. The 1×1 convolution is used to reduce or increase the number of feature map channels while keeping the feature graph's spatial dimension constant. The fully connected layer is responsible for feature synthesis and classification at the final stage. It transfers the preceding level's eigenvectors to a space with the same dimensions as the number of categories, each representing a category's anticipated score.

i. Stem

As the first level of the RSNet network structure, the Stem layer mainly aims to extract key features from the input data to provide a more abstract and rich information representation for subsequent network layers. The network structure is shown in Fig. 4.

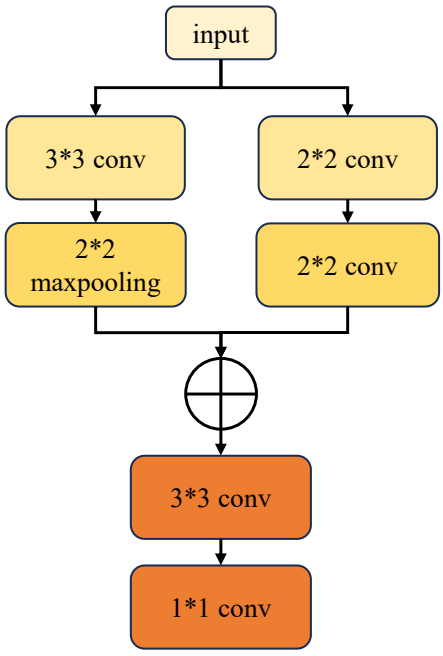

Fig. 4. Structure of the Stem

The Stem layer performs feature extraction and down sampling in the network structure with the convolutional and max pooling layers. Raw data contains a large amount of redundant information, making it inefficient to process directly. As a result, it is important to preprocess the input data through the Stem layer. The step helps transform the input data into a more compact and meaningful representation from which key task-relevant features are extracted. The Stem layer can extract spatial features and patterns from the input data through convolution, while the max pooling layer helps down sampling and retaining the most relevant information. The preliminary feature extraction and dimensionality reduction operations in the Stem layer lay the groundwork for the network's subsequent layers, which can refine the extracted features for more accurate and efficient processing.

ii. RS Block

RS Block comprises two structures: the upper part contains several 3×3 convolution kernel separable convolution blocks, a 1×1 convolution block, and Group Normalization three parts in parallel. The structure's lower part comprises many separable convolution blocks with a 3×3 convolution kernel and Group Normalization. The swish activation function connects the upper and lower structures, as shown in Fig. 5. The separable convolution block in the upper part helps to effectively capture the details and structural information in the image by separating the processing of spatial features and channel features. The 1×1 convolution block is used to reduce the dimension or expand the feature space to improve the expression ability of the network. As a normalization approach, Group Normalization accelerates the network's training process while enhancing the model's generalization capabilities. The lower part of the structure continues to utilize separable convolutional blocks and Group Normalization to extract further and integrate feature information. The swish activation function connects the two structures, has a smooth curve property, helps to ease the vanishing gradient problem, and improves the network's nonlinear modelling capacity.

This study uses separable convolution instead of ordinary convolution in RS Block. The separable convolution block first performs depth convolution on the input feature map, then point-by-point convolution and finally, a channel-by-channel linear combination of the depth convolution outputs. Point-by-point convolution uses a 1×1 convolution kernel to perform convolution operations in the channel dimension, whereas depth convolution results are linearly combined and fused. Each channel is convolved separately, and the results are subsequently merged to extract spatial information across several channels. This reduces the model's complexity while improving its computing efficiency.

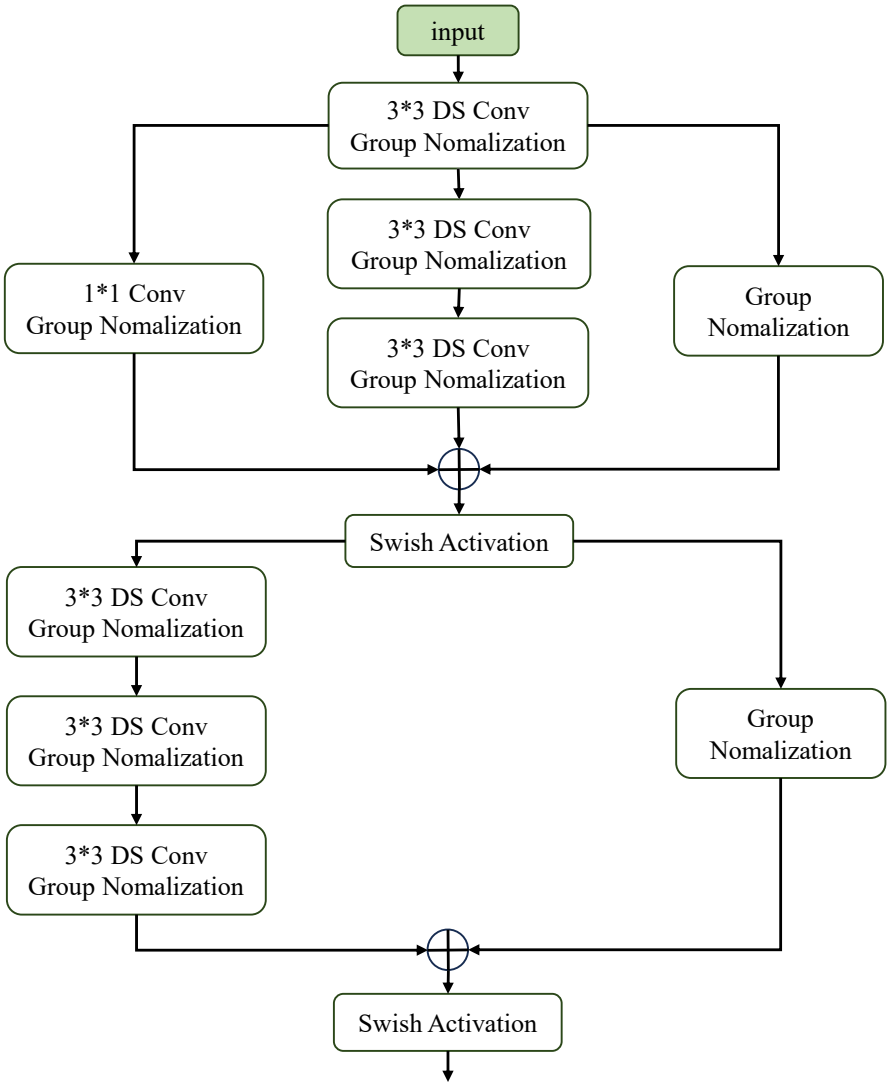

Fig. 5. Structure of the RS Block

In addition, we employ Progressive Learning to gradually raise the difficulty of training in the early stages of model training. By gradually raising the complexity of the learning challenge or adjusting the learning rate, we can improve the model's generalization ability. The Annealing Weight Decay approach reduces the weight decay rate, allowing the model to control the model's complexity in the later stages of training and avoid overfitting. Finally, in the last stages of model training, the Exponential Moving Average (EMA) method performs the exponential moving average of the model parameters, smoothing the training process, reducing parameter mutation and fluctuation, and improving the model's stability and robustness. During the model training process, we employ the Auto Augment approach to enhance the diversity of training data and the model's robustness and generalization capabilities. Combining these methods allows the model to analyze and classify the input data better, resulting in improved accuracy and robustness.

IV. Experiments

A. Dataset

The experiments in this paper are conducted using the HMDB51 dataset. The HMDB51 dataset is a standard video dataset for action recognition research, containing 51 video clips of various human behaviors. Each category includes an average of about 101 video clips, resulting in a total of about 6,766 video clips. The average time of each video clip is 3-4 seconds, and the resolution is 320 by 240 pixels. The dataset's video clips are in AVI format and annotated at the frame level. Each video clip features a label that aligns with the activity category it displays.

B. Evaluation Indicators

In this paper, the assessment metrics for experiments include Top-1 Accuracy, Top-5 Accuracy, and latency.

Top-1 Acc: "Top-1 Acc" represents the fraction of examples in the dataset whose predictions exactly match the actual label, that is, the percentage of accurate classification. The formula is shown as follows:

$$Top\text{-}1\ Acc = \frac{C_1}{N} * 100\% \tag{1}$$

Where C_1 is the number of samples for which the predicted result matches the label. N is the total number of predicted samples.

Top-5 Acc: "Top-5 Acc" is a popular metric for evaluating classification models, assessing how well the predicted five highest probability classes correspond to the actual labels. The formula is as follows:

$$Top\text{-}5\ Acc = \frac{C_5}{N} * 100\% \quad (2)$$

Where C_5 is the number of true label samples in the category with the top five highest probabilities in the prediction result. N is the total number of predicted samples. This metric considers the diversity of the prediction results; that is, if the true label is within the top five prediction categories, it is regarded as accurately predicted.

Latency is how long the model takes to infer or predict. It is commonly measured in milliseconds. Latency is an important parameter for determining the model's response speed and real-time performance; lower latency indicates that the model can accomplish the inference task faster.

The *mAP* is a commonly used evaluation metric for object detection algorithms. It measures the model's detection performance in multiple categories. In general, values range between 0 and 100, with higher values indicating better performance on the object-detecting task.

FPS measures how quickly a computer vision system or algorithm processes information. It denotes the number of picture frames the system can handle per second. The higher the number, the faster the system can process further pictures or video frames in real-time.

C. Training of Model

During the model training phase, the PyTorch library is trained on a system with two NVIDIA Tesla V100 GPUs. During training, a Stochastic Gradient Descent optimizer with momentum is used. The SGD optimizer can adjust the model's weights depending on the sample gradient to minimize the loss function. The experiment was trained for 300 iterations, with each iteration representing a traverse through the whole training dataset. In order to improve the generalization ability of the model, the label smoothing regularization technique is used. By incorporating a smoothing factor into the cross-entropy loss function, the smoothing factor is set to 0.1, which reduces overfitting and allows the model to better adapt to the input data. In terms of learning rate and weight decay, the cosine schedule is utilized, with an initial learning rate of 0.1. It means that as training advances, the learning rate is gradually reduced to optimize the model's parameters better. The initial weight decay coefficient is set at 10^{-4} and gradually reduced to 10^{-5} with the same cosine schedule.

Table I shows the Top-1 and Top-5 Accuracy of the RSNet model after using various performance enhancement approaches during model training. When Progressive Learning was introduced early in model training, both Top-1 and Top-5 Accuracy improved. This method improves the model's generalizability by gradually raising the training difficulty. We use the Annealing Weight Decay approach in model training to gradually reduce the weight decay rate, enabling the model to regulate its complexity later in the training process and prevent overfitting. Using EMA at a later stage of model training increases the model's performance even further. EMA improves the training process by smoothing the parameter update, lowering mutation and volatility, and increasing the model's stability and robustness. Auto Augment increases the diversity of training data and improves Top-5 Accuracy, but the improvement in Top-1 Accuracy is minimal. Different performance enhancement methods have a substantial impact on improving the RSNet model. Combining these methods improves the model's understanding and classification of incoming data, increasing its accuracy and robustness.

TABLE I
TECHNOLOGIES TO IMPROVE THE PERFORMANCE OF RSNET

Model optimization	Top-1 Accuracy (%)	Top-5 Accuracy (%)
RSNet	83.9	95.8
+Progress Learning	84.3(+0.4)	96.1(+0.3)
+EMA	84.7(+0.4)	96.4(+0.3)
+Auto Augment	84.8(+0.1)	96.8(+0.4)
+Annealing Weight Decay	85.0(+0.2)	97.0(+0.2)

D. Experimental Results

RSNet borrows the network structure of Swin Transformer and adopts a hierarchical design, which improves the model's accuracy and performance and has a low computational cost. Table II shows how RSNet balances accuracy and inference delay by adjusting network configurations such as the number of channels, RS Blocks, and depthwise separable convolutional layers to adapt to different application scenarios.

TABLE II
CONFIGURATION AND PERFORMANCE OF THE RSNET

Parameters	RSNet_small	RSNet_middle	RSNet_large
in-channel	[96,224,448,512]	[128,256,512,768]	[160,320,640,960]
out-channel	[224,448,512,768]	[256,512,768,1024]	[320,640,960,1280]
RS block	[1,1,2,1]	[1,2,3,2]	[1,2,3,2]
D.S. Conv	2	3	3
Top-1 Accuracy (%)	81.95	83.82	85.00
Top-5 Accuracy (%)	96.12	96.81	97.35
Latency (ms)	1.77	2.52	5.97

As shown in Table II, RSNet offers three configuration sizes, RSNet_small, RSNet_middle, and RSNet_large, to show the relationship between model configuration and performance. RSNet_small has a small number of channels and 2-layer depthwise separable convolutions. Even with this tiny configuration, it achieves 81.95 % Top-1 Accuracy and 1.77 ms latency. RSNet_middle has three layers of depthwise separable convolutions and more channels than RSNet_small. This medium-scale model configuration has a Top-1 Accuracy of 83.82 %, but the latency rises to 2.52 ms. RSNet_large is the most extensive configuration with the most channels. This configuration achieves the highest level of accuracy, 85.00 %, but the latency increases to 5.97 ms. Increasing the model's configuration improves RSNet's accuracy and latency, making it available to various scenarios

In this experiment, we compare the performance of RSNet to other models in Top-1 Accuracy, Top-5 Accuracy, and inference delay on image recognition tasks. As shown in Table III.

TABLE III
COMPARISON OF MODEL PERFORMANCE

Model	Top-1 Accuracy (%)	Top-5 Accuracy (%)	Latency (ms)
ResNet34	74.57	92.14	1.97
EfficientNetB0	77.38	93.31	1.96
RSNet_small	81.95	96.12	1.77
ResNet50	76.50	93.00	2.54
EfficientNetB1	79.15	94.41	2.88
Swin Transformer_tiny	81.20	95.50	6.59
RSNet_middle	83.82	96.81	2.52
Res2Net200	85.13	97.42	11.45
ResNeXt101	85.37	97.69	55.07
Swin Transformer_base	85.2	97.50	13.53
RSNet_large	85.00	97.35	5.97

First, ResNet34 and ResNet50 exhibit high Top-1 and Top-5 Accuracy while maintaining low latency. It shows that the ResNet model delivers excellent real-time performance while maintaining high accuracy. Second, Res2Net200 has a much higher latency than ResNet34 and ResNet50, measuring 11.45 ms. However, the Top-1 and Top-5 Accuracy are 85.13 % and 97.42 %, respectively, indicating that the model's accuracy can be significantly increased by enhancing the ResNet design. However, the delay will grow. The EfficientNet and Swin Transformer series perform well in terms of accuracy, although their latency is higher than that of the ResNet series. In particular, Swin Transformer_base is delayed by 13.53 ms. The latency of Swin Transformer_base surpasses that of most ResNet models. Finally, the RSNet series, RSNet_small and RSNet_middle, have higher accuracy and lower latency. RSNet_large has slightly lesser accuracy than ResNeXt101 and Swin Transformer_base, but its latency is lower at 5.97 ms. Compared to the ResNet series, it has higher latency but higher accuracy. In summary, the RSNet has advantages over many models in terms of accuracy and latency.

1) ResNet: ResNet34 [23], a variant of ResNet, is an architecture that solves the difficulties of deep network training by introducing residual connections with a relatively small number of parameters. ResNet50 [24] is an important variant in the ResNet series with a depth of 50 layers. ResNet50 has significantly higher expressive power than ResNet34.

2) Res2Net200: Res2Net [25] is a modified residual network with 200 layers that improves its feature representation ability. The introduction of multi-scale feature representation allows the network to learn features at various scales, boosting classification and detection performance.

3) ResNeXt101: ResNeXt [26] is a ResNet-based network design that utilizes grouped convolution to enhance parameter efficiency and performance. ResNeXt101, a version of this architecture, has 101 layers.

4) EfficientNet: EfficientNet [27] is a convolutional neural network architecture with EfficientNetB0 as its basis model. EfficientNetB1 is the second model in the EfficientNet series; it has been optimized and extended from EfficientNetB0. EfficientNetB1 performs better by increasing the depth and width.

5) Swin Transformer: Swin Transformer is a novel vision Transformer architecture that aims to solve the problems faced by traditional transformers in computer vision tasks. Swin Transformer adopts hierarchical feature representation and a local windowed self-attention mechanism to improve computational efficiency and performance. Swin Transformer_base is the basic version, which is more efficient and performant, and Swin Transformer_tiny has a relatively small number of parameters.

This paper uses the YOLACT [28] algorithm as an example to compare the impact on *mAP* and FPS when choosing different backbone networks for object detection. Resnet is faster but needs to be more accurate. Swin Transformer has significantly higher accuracy but is slower than Resnet. EfficientNet achieves excellent accuracy while being fast, and its performance is balanced. RSNet exceeds other networks in accuracy and processing speed, with a mAP of 32.4 % and an FPS of 39.6 ms, indicating good performance.

To summarize, when using the YOLACT method for object detection, the choice of backbone networks substantially impacts detection accuracy and processing performance. RSNet maintains rapid processing speed while assuring high accuracy, demonstrating its exceptional performance as a backbone network in object detection tasks.

TABLE IV
PERFORMANCE COMPARISON OF TARGET DETECTION MODELS

Model	*mAP* (%)	FPS (ms)
ResNet	28.2	42.5
Swin transformer	30.5	35.5
EfficientNet	31.6	38.6
RSNet	32.4	39.6

The confusion matrix of Fig. 6 presents the classification results for the six driving behavior categories: "smoking," "cellphone," "safety_belt," "yawn," "closed-eye," and "look_ahead." The numbers on the diagonal of the confusion matrix represent the number of times the model correctly identified that class, whereas the values on the off-diagonal show the number of times it misclassified. The confusion matrix reveals that RSNet has excellent recognition accuracy and less misclassification.

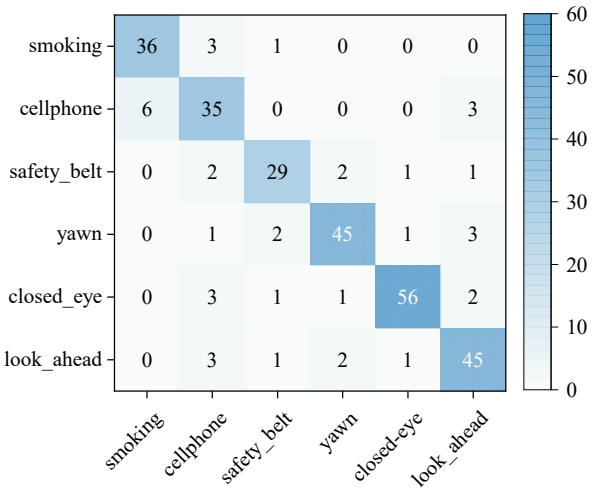

Fig. 6. Confusion matrix

Fig. 7 shows the detection results of driving behavior, and the RSNet model correctly identifies the abnormal driving activity in the image.

Fig. 7. Driving behavior detection results

V. Conclusion

The traditional two-stage fatigue driving detection method has problems such as low accuracy and poor efficiency. In order to solve these problems, this paper proposes a fatigue driving detection model called RSNet. The model extracts and classifies the features in the image or video to determine whether the driver is in an abnormal driving state. RSNet is a single-stage fatigue detection method based on separable convolution, which combines image classification techniques and hierarchical design. The hierarchical design enables the model to extract features and classify them more effectively, thus overcoming the shortcomings of traditional methods in efficiency and accuracy. By introducing separable convolutions into RS Block, the model complexity is reduced, the risk of overfitting is reduced, and the computational efficiency is improved. Separable convolutions can better capture spatial features when processing images or feature maps, enhancing feature extraction capabilities. In the training process, Progress Learning, EMA, Auto Augment, and Annealing Weight Decay are used to improve the performance and generalization ability of the model. Experiments show that the Top-1 Accuracy of RSNet on the HMDB51 dataset reaches 85 %, the Top-5 Accuracy is 97.35 %, and the delay time is 5.97 ms, which overcomes traditional methods' accuracy and efficiency problems. In addition, the YOLACT algorithm is used as an example to compare the impact of different backbone networks on object detection accuracy *mAP* and FPS. Experimental results show that using RSNet as the backbone network of YOLACT can obtain better performance, with *mAP* reaching 32.4 % and FPS reaching 39.6 ms.

Since the fatigue determination in this paper is based on analyzing drivers' abnormal driving behavior, factors such as personal driving habits, environmental conditions, postural changes, and lighting conditions may affect the accuracy of abnormal driving behavior detection. Future research will combine multiple factors to detect driver fatigue, further improving the accuracy of fatigue driving detection.

References

[1] Z. Mao, X. Yan, and C. Wu, "Driving fatigue identification method based on physiological signals", *Plan, Build, and Manage Transportation Infrastructure in China*, pp. 341-352, 2008.

[2] D. Artanto, M. P. Sulistyanto, I. D. Pranowo, and E. E. Pramesta, "Drowsiness detection system based on eye-closure using a low-cost EMG and ESP8266", *2017 2nd International Conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE). IEEE*, pp. 235-238, 2017.

[3] M. Sangeetha, S. Kalpanadevi, M. Rajendiran, and G. Malathi, "Embedded ECG based real time monitoring and control of driver drowsiness condition", *International Journal of Science, Technology and Society*, vol.3, no.4, pp. 176, 2015.

[4] X. B. Zhang, B. Cheng, and R. J. Feng, "A real-time driver fatigue detection method based on steering wheel operation", *Journal of Tsinghua University: Natural Science Edition*, vol.50, no.7, pp. 1072-1076, 2010.

[5] L. Liu, Z. Wang, and S. Qiu, "Driving behavior tracking and recognition based on multisensors data fusion", *IEEE Sensors Journal*, vol.20, no.18, pp. 10811-10823, 2020.

[6] X. Li, and E. Seignez, "Driver inattention monitoring system based on multimodal fusion with visual cues to improve driving safety", *Transactions of the Institute of Measurement and Control*, vol.40, no.3, pp. 885-895, 2018.

[7] Y. Hu, M. Lu, and X. Lu, "Driving behavior recognition from still images by using multi-stream fusion CNN", *Machine Vision and Applications*, vol.30, pp. 851-865, 2019.

[8] X. Rao, F. Lin, Z. Chen, and J. Zhao, "Distracted driving recognition method based on deep convolutional neural network", *Journal of*

Ambient Intelligence and Humanized Computing, vol.12, no.1, pp. 193-200, 2021.
[9] C. Yan, F. Coenen, and B. Zhang, "Driving posture recognition by convolutional neural networks", *IET Computer Vision*, vol.10, no.2, pp. 103-114, 2016.
[10] H. R. Qu, "Detection of abnormal driver behavior based on machine vision", *Tianjin Polytechnic University*, 2020.
[11] Y. Q. Zhao, Y. Rao, S. P. Dong, and J. Y. Zhang, "Survey on deep learning object detection", *Journal of Image and Graphics*, vol.25, no.4, pp. 629-654, 2020.
[12] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, et al., "Ssd: Single shot multibox detector", *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer International Publishing*, pp. 21-37, 2016.
[13] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks", *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol.39, no.6, pp. 1137-1149, 2017.
[14] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN", *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2961-2969, 2017.
[15] H. He, X. Zhang, F. Jiang, C. Wang, Y. Yang, et al., "A real-time driver fatigue detection method based on two-stage convolutional neural network", *IFAC-PapersOnLine*, vol.53, no.2, pp. 15374-15379, 2020.
[16] N. Yang, and J. Zhao, "Dangerous Driving Behavior Recognition Based on Improved YoloV5 and OpenPose", *IAENG International Journal of Computer Science*, vol.49, no.4, pp. 1112-1122, 2022.
[17] M. F. Shakeel, N. A. Bajwa, A. M. Anwaar, A. Sohail, A. Khan, et al., "Detecting driver drowsiness in real time through deep learning based object detection", *International Work-conference on Artificial Neural Networks. Cham: Springer International Publishing*, pp. 283-296, 2019.
[18] Y. Ma, Y. Tao, Y. Gong, W. Cui, and B. Wang, "Driver identification and fatigue detection algorithm based on deep learning", *Mathematical Biosciences and Engineering*, vol.20, no.5, pp. 8162-8189, 2023.
[19] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. AI-Dujaili, Y. Duan, et al., "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions", *Journal of Big Data*, vol.8, pp. 1-74, 2021.
[20] F. Chollet, "Xception: Deep learning with depthwise separable convolutions", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1251-1258, 2017.
[21] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, et al., "Swin transformer: Hierarchical vision transformer using shifted windows", *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10012-10022, 2021.
[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, et al., "Attention is all you need", *Advances in Neural Information Processing Systems*, pp. 30, 2017.
[23] B. Koonce, and B. E. Koonce, "ResNet 34", *Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization*, pp. 51-61, 2021.
[24] B. Koonce, and B. E. Koonce, "ResNet 50", *Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization*, pp. 63-72, 2021.
[25] S. H. Gao, M. M. Cheng, K. Zhao, X. Y. Zhang, M. H. Yang, et al., "Res2net: A new multi-scale backbone architecture", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.43, no.2, pp. 652-662, 2019.
[26] T. Zhou, Y. Zhao, and J. Wu, "ResNeXt and res2net structures for speaker verification", *2021 IEEE Spoken Language Technology Workshop (SLT). IEEE*, pp. 301-307, 2021.
[27] B. Koonce, and B. E. Koonce, "EfficientNet", *Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization*, pp. 109-123, 2021.
[28] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "YOLACT: Real-time instance segmentation", *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9157-9166, 2019.