Heliyon 10 (2024) e26182

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

A lightweight network for traffic sign recognition based on multi-scale feature and attention mechanism

Wei Wei [a], Lili Zhang [a,*], Kang Yang [a], Jing Li [a], Ning Cui [a], Yucheng Han [a], Ning Zhang [a], Xudong Yang [a], Hongxin Tan [b], Kai Wang [c]

[a] Beijing Institute of Petrochemical Technology, Beijing, 102617, China
[b] Science and Technology on Complex Aviation Systems Simulation Laboratory, Beijing, 100076, China
[c] Institute of National Defense Science and Technology Innovation, Academy of Military Sciences, Beijing, 100036, China

ARTICLE INFO

Keywords:
Traffic sign recognition
ConvNeSe
Lightweight
Multi-scale feature
Attention mechanism

ABSTRACT

Traffic sign recognition is an important part of intelligent transportation system. It uses computer vision and traffic sign recognition technology to detect and recognize traffic signs on the road automatically. In this paper, we propose a lightweight model for traffic sign recognition based on convolutional neural networks called ConvNeSe. Firstly, the feature extraction module of the model is constructed using the Depthwise Separable Convolution and Inverted Residuals structures. The model extracts multi-scale features with strong representation ability by optimizing the structure of convolutional neural networks and fusing of features. Then, the model introduces Squeeze and Excitation Block (SE Block) to improve the attention to important features, which can capture key information of traffic sign images. Finally, the accuracy of the model in the German Traffic Sign Recognition Benchmark Database (GTSRB) is 99.85%. At the same time, the model has good robustness according to the results of ablation experiments.

1. Introduction

With the development of intelligent transportation systems, traffic sign recognition technology has been concerned widely [1]. The traditional method of traffic sign recognition mainly depends on the feature extractor by hand. However, there are some problems in the practical application of these methods. For example, they are sensitive to environmental factors such as light, noise and so on [2]. Besides, different kinds of traffic signs need different feature extractors [3,4]. Deep neural networks (DNN) have achieved great success in the field of image recognition with the development of deep learning technology. The structure of DNN simulates the neural structure of the human brain when processing information. The main advantage is that it can learn features from a large amount of data automatically and perform good in complex classification tasks. Therefore, the feature extracted from the image using the DNN model is much better than the traditional algorithm. It helps to improve the robustness and stability of the model [5].

Convolutional neural networks (CNN) have the advantages of local perception and parameter sharing [6,7]. It is an ideal choice to solve the problem of traffic sign recognition. CNN reduce model complexity through parameter sharing and local perceptibility, which can capture the spatial structure in images effectively. CNN has certain robustness and generalization ability through large-scale data training. It can maintain stable performance in the face of interference such as light or noise in the image. CNN consists of several

* Corresponding author.
E-mail address: zhanglili@bipt.edu.cn (L. Zhang).

https://doi.org/10.1016/j.heliyon.2024.e26182
Received 24 September 2023; Received in revised form 29 January 2024; Accepted 8 February 2024
Available online 15 February 2024

convolutional layers, pool layers and fully connected layers. It can extract features from simple to complex due to its special convolution architecture [8], which makes it a great success in the field of object detection and image recognition [9,10].

We propose a lightweight model for traffic sign recognition based on CNNs called ConvNeSe. ConvNeSe adopts SE Block, ConvNeSe Block and Sample Block to improve recognition accuracy while maintaining fewer parameters. It can be used not only as image recognition models, but also as backbone to other object detection models. For example, we can build a object detection model to detect object in the image by using ConvNeSe network as the backbone. This method can improve the performance of the detection model, while keeping less parameters and computation effectively.

The main contributions of this paper can be summarized as follows：

- A lightweight traffic sign recognition model ConvNeSe is designed by using Depth Separable Convolution and reducing the number of convolution layers.
- A multi-scale feature extraction module ConvNeSe Block is designed by using Inverted Residuals structure and multiple types of convolutional modules. The model avoids the problems of gradient disappearance and information loss in deep structures, and thus improves the ability of multi-scale feature extraction and fusion.
- SE Block is applied to ConvNeSe Block to achieve adaptive adjustment of feature channel weights. The model enhances the attention to important features by strengthening the influence on important features.

Compared with the existing state-of-the-art model, the ConvNeSe network has better robustness, higher recognition accuracy lower Floating Point Operations (FLOPs) and fewer parameters.

The paper is organized as follows. Section 2 reviews the literature. Section 3 describes the proposed approach. Section 4 shows the experimental results. Section 5 summarizes the conclusions.

2. Related works

In recent years, the deep learning model based on CNN has been used in the research of traffic sign recognition widely. The model needs to extract as many features as possible to improve the accuracy of traffic sign detection. The CNN model has powerful feature extraction capabilities, allowing problem solving through parallel processing capabilities and self-learning capabilities [11]. The VGG network proposed after Alexnet deepens the depth of the network and improves the accuracy [12,13]. GoogLeNet was proposed by the Google team [14]. This model combines convolution cores of different sizes and pool layers to build the Inception module, improving the performance of the model without increasing the number of parameters. ResNet is a model proposed by Kaiming He et al. [15]. It avoids the problems of gradient disappearance and gradient explosion by connecting the original matrix input to the matrix output to construct the residual connection directly. MobileNet is a lightweight CNN [16–18]. It adopts Depth Separable Convolution to reduce computation and number of parameters while maintaining high classification performance. MobileNet is widely used on mobile devices, and it is suitable for scenarios that require fast response and low power consumption. EfficientNet is an efficient and accurate architecture of CNN [19,20]. It achieves a balance between better performance and computing efficiency by optimizing network parameters such as depth, width and resolution. Bangquan Xiong proposed a new and efficient network for traffic sign classification called ENet. It has few parameters and computation relatively, so that it is suitable for real-time operation on embedded devices [21]. The experimental results show that the accuracy of ENet achieves 98.6% on the GTSRB dataset. ENet improves the model recognition speed by reducing the parameters, but reduces the model accuracy. We address the balance of depth and precision of the model by using Depthwise Separable Convolution and using less Layer Normalization, Drop Path, and Inverted Residuals structures.

Multi-scale feature fusion is an indispensable part of deep learning. It can combine shallow features and deep features to express more complex semantic information, thus improving the performance and robustness of the network. At present, many advanced models have adopted techniques of multi-scale feature fusion. A recognition algorithm of multi-layer and multi-scale CNN is proposed by Cai Zhao [22]. Firstly, the feature extraction method of single-scale CNN is improved to extract the global and local features of traffic sign images and to fuse the features generated from multiple levels into multi-scale features. The recognition accuracy of the model is 98.62% on the GTSRB dataset. However, there is a problem of high computational cost in improving feature extraction. Song proposed a two-stage binarized multi-scale neural network framework (B-MNN) [23]. Experimental results show that the recognition accuracy achieves 91.34% on GTSRB dataset. B-MNN improves the recognition speed of the model by reducing the number of parameters, thus reducing the accuracy of the model. Lan proposed a feature extraction method for multi-scale asymmetric convolution blocks and weighted mixed loss functions [24]. Multi-scale asymmetric convolution blocks are used to extract multi-scale features. At the same time, the authors propose a weighted mixed loss function to make the model pay more attention to the characteristics of hard-to-classify samples. The recognition accuracy of the model is 98.92% on GTSRB dataset. Using a weighted mixed loss function may require manual selection of weights, which may result in poor model performance if the weights are not selected well. Chen proposed a Multi-Scale Capsule Network (Multi-Scale CapsNet) [25]. Image features are extracted by multi-channel convolution of multi-convolution kernel, which makes the extracted features more diversified. The recognition accuracy of the model is 99.40% on the GTSRB dataset. However, Multi-Scale CapsNet needs to adjust the parameters of multiple CapsNet with different scales, which increases the complexity of parameter adjustment. Liang proposed a method of traffic sign recognition based on multi-scale features and attention mechanism [26]. Multi-scale features are used to fuse different levels of feature information, enrich feature semantic information, and enhance feature extraction ability. The recognition accuracy of the model in GTSRB dataset reached 98.96%. However, the improved ResNet network has the problem of high computing cost. Zhonghua Wei proposed a method of multi-scale feature extraction based on CNN [27]. By introducing multiple branches between convolution layers, image information of

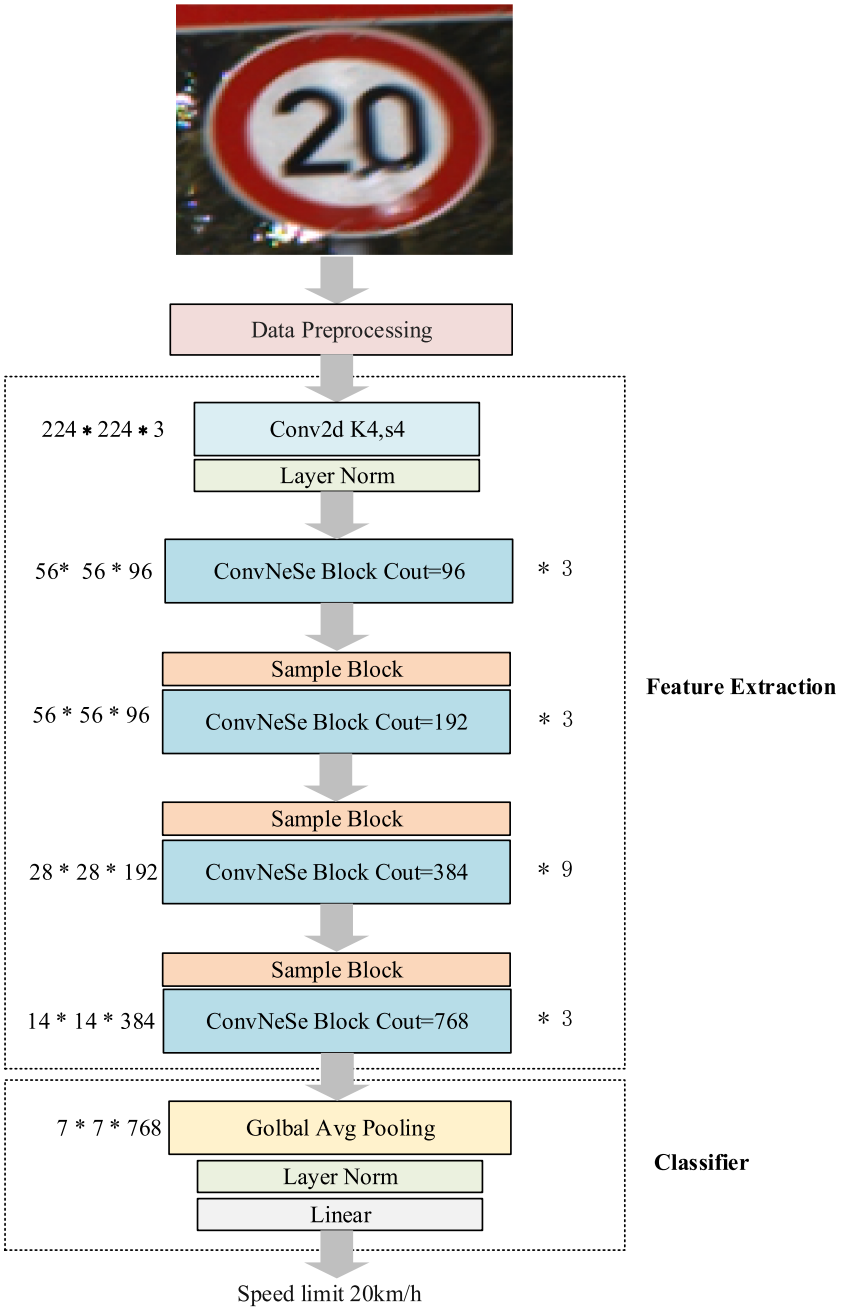

Fig. 1. ConvNeSe structure.

different scales is processed. Image information of different scales is processed by using multiple branches between convolution layers. Experimental results show that the recognition accuracy achieves 99.36% on GTSRB dataset. In general. The existing multi-scale feature fusion algorithms include improved feature extraction, pyramid pool, multi-scale convolutional neural network, etc. These algorithms have high accuracy in traffic sign recognition and other fields, but there are also problems such as high calculation cost, difficult parameter adjustment and poor effect in dealing with light intensity. We propose ConvNeSe Block to improve the multi-scale feature extraction and fusion capability of the model. Different types of convolutional modules and Inverted Residuals structure are introduced to optimize the model structure so that the model can extract more abundant features.

The attention mechanism makes the model pay more attention to important features and suppress redundant features, thus improving the accuracy, stability and generalization ability of the model. Wang uses the attention mechanism to improve the neural network based on the existing VGG16 network architecture [28]. The accuracy of the model is 99.34% on the GTSRB dataset. Daihui Li proposed a lightweight and efficient Cyclic Spatial Attention Module (CSAM) for convolutional neural networks [29]. CSAM generates

Table 1
Data volume of training set, verification set, and test set in DTSRB.

Traffic sign label	Totality	Training set	Validation set	Test set
Speed limit 20 km/h	210	147	63	60
Speed limit 30 km/h	2220	1554	666	720
Speed limit 50 km/h	2250	1575	675	750
Speed limit 60 km/h	1410	987	423	450
Speed limit 70 km/h	1980	1386	594	660
Speed limit 80 km/h	1860	1302	558	630
Speed limit 100 km/h	1440	1008	432	450
Speed limit 120 km/h	1410	987	423	450
End of speed limit 80 km/h	420	294	126	150
No passing	1470	1029	441	480
Right-of-way at the next intersection	1320	924	396	420
No passing for vehicles over 3.5 metric tons	2010	1407	603	660
Priority road	2100	1470	630	690
Yield	2160	1512	648	720
Stop	780	546	234	270
No vehicles	630	441	189	210
Vehicles over 3.5 metric tons prohibited	420	294	126	150
No entry	1110	777	333	360
General caution	1200	840	360	390
Dangerous curve to the left	210	147	63	60
Dangerous curve to the right	360	252	108	90
Double curve	330	231	99	90
Bumpy road	390	273	117	120
Slippery road	510	357	153	150
Road narrows on the right	270	189	81	90
Road work	1500	1050	450	480
Traffic signals	600	420	180	180
Pedestrians	240	168	72	60
Children crossing	540	378	162	150
Bicycles crossing	270	189	81	90
Beware of ice/snow	450	315	135	150
Wild animals crossing	780	546	234	270
End of all speed and passing limits	240	168	72	60
Turn right ahead	689	485	204	210
Turn left ahead	420	294	126	120
Ahead only	1200	840	360	390
Go straight or right	390	273	117	120
Go straight or left	210	147	63	60
Keep right	2070	1449	621	690
Keep left	300	210	90	90
Roundabout mandatory	360	252	108	90
End of no passing	240	168	72	60
End of no passing by vehicles over 3.5 metric tons	240	168	72	90

attention maps based on the output features. Attention diagram to adjust the weight of different positions in the feature map. CSAM applied to the ResNet network achieved 97.73% recognition accuracy on the GTSRB dataset. The introduction of cycle structure will increase the computation and storage capacity. At the same time, CSAM also needs a lot of parameter tuning. Ke Zhou proposed a method of traffic sign recognition based on regional attention network [30]. The most important areas are selected through regional attention networks adaptively. The accuracy of the model is 97.21% in the GTSRB dataset. This method requires the degree of attention to be calculated for each area, increasing the amount of computation. Yash Garg proposed an innovative feature learning framework: scale-invariant attention networks (SAN) [31]. SAN uses the information and attention mechanism of multi-scale space to capture local and global features in images effectively. Unlike existing attention networks, SAN focuses attention on parts that change across Spaces and scales significantly. SAN was applied to the ResNet network and achieved 99.76% recognition accuracy on the GTSRB dataset. This method needs to fuse and balance the feature maps of different scales, which increases the of parameter adjustment difficulty. Chung proposed a network of convolutional pooling neural based on attention [32]. The attention mechanism is applied in feature mapping to obtain key features, and convolution pooling is used to improve recognition accuracy in harsh environments. The accuracy of the model is 97.10% on the GTSRB dataset. However, using this method for recognition in harsh environments requires large memory relatively. Chung proposed an attentional deconvolution module (ADM) based network (ADM-Net) [33]. The network uses ADM, convolutional pools and full convolutional networks to improve classification under such harsh conditions. ADM-Net validates ADM-Net with different noise cases at GTSRB. The accuracy rate was 92.329% in the absence of information and lighting. ADM-Net requires extensive parameter adjustments to achieve optimal performance. In summary, attention mechanism is used in traffic sign recognition tasks widely and has become one of the key technologies to improve the accuracy and interpretability of the model. However, different attention mechanisms have different problems and challenges in specific implementation and application. For example, there is a high computational cost to calculate attention weights and a large amount of parameter tuning to achieve optimal

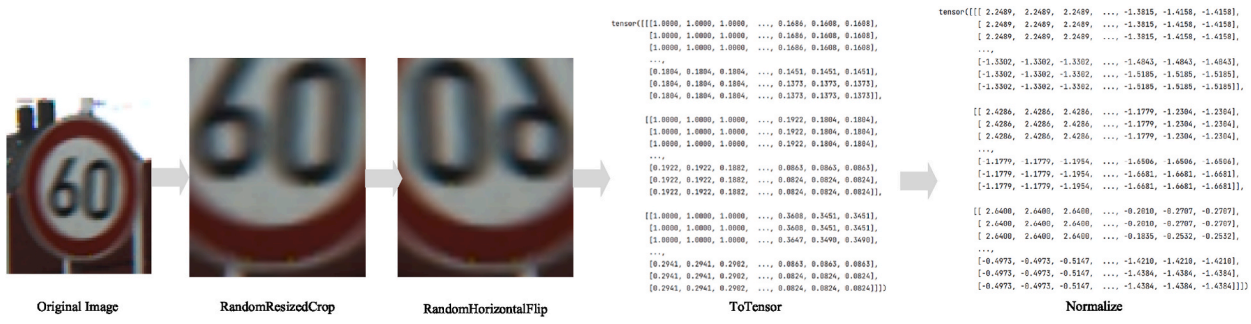

Fig. 2. Training process preprocessing operations.

performance. We apply SE Block to ConvNeSe Block to achieve adaptive adjustment of feature channel weights. The model improves the attention to important features by strengthening the influence on important features.

3. Proposed methods

Deep learning has driven the development of computer vision in image classification tasks. Previous image classification models only improve the accuracy of the model by increasing the depth of the network. However, increasing the depth of the network cannot guarantee the improvement of the model accuracy simply, and even leads to the aggravation of the high variance-high bias problem. In this paper, we propose an image classification model ConvNeSe for traffic sign recognition as shown in Fig. 1. ConvNeSe is based on convolutional neural networks and solves the problem of balancing the depth and precision of the model by using less Layer Normalization, Drop Path, and Inverted Residuals structure [34,35]. ConvNeSe enables efficient feature learning and image processing through advantages such as translation invariance, local feature extraction, and preservation of spatial structure. At the same time, it is able to better than Transformer Network in terms of accuracy, scalability, and robustness.

Firstly, the traffic sign image is preprocessed. Secondly, the processed data enters the feature extraction module. Downsampling is performed by a convolution layer with a convolution kernel size of 4*4. Data normalization is performed by Layer Normalization. The ConvNeSe Block is then cross-stacked with the Sample Block. The stacking times of ConvNeSe Blocks are (3,3,9,3) and the number of channels output by each ConvNeSe Block is (96, 192, 384, 768). Finally, the recognition results of traffic sign are output through the classifier.

3.1. Data

3.1.1. Dataset

The German Traffic Sign Recognition Benchmark dataset [36] is a publicly available dataset for traffic sign detection and recognition. The GTSRB dataset contains 51839 images covering 43 types of traffic signs. Such as speed limits, no parking, danger signs, etc., each image is annotated with a corresponding category label. Each traffic sign contains 200 to 2000 images. Image resolution ranges from 15 * 15 to 250 * 250. In order to better predict and evaluate the model, 39209 images from the dataset are used to build the training set and the verification set, and the ratio of the training set to the verification set is 7:3. 12630 images are used to build the test set. The division results of each traffic sign in the data set as shown in Table 1. In this experiment, training, validation and testing are performed on the GTSRB dataset, and the proposed model is also applicable to other standard datasets. Examples include Belgium Traffic Sign Dataset (BTSD), Changsha University of Science and Technology Remote Sensing Image Dataset (CSUST), Chinese Traffic Sign Detection Bench-mark (CCTSDB) and German Traffic Sign Detection Benchmark (GTSDB) [37–40].

3.1.2. Data preprocessing

The narrow data coverage in GTSRB makes the model can not deal with the image of illumination change, scale difference, rotation, local occlusion and so on effectively. The application of data enhancement techniques can increase the diversity of images, thus improving the performance and robustness of the model. The model can recognize traffic signs in various complex environments. At the same time, some preprocessing operations can filter out redundant information and save storage and computing resources.

In this experiment, the pre-processing operations on the training set data include RandomResizedCrop, RandomHorizontalFilp, ToTensor and Normalize as shown in Fig. 2. RandomResizedCrop crops the image to different sizes and aspect ratios randomly, and then scales the cropped image to the specified size 224*224. This method can help the model adapt to images of different sizes and scales, thus improving the generalization ability and performance of the model. RandomHorizontalFilp flips the image horizontally which can increase the diversity of images in the data set. The probability of RandomHorizontalFilp preprocessing for each photo is 50%. ToTensor converts the image into Tensor format. Normalize standardizes Tensor data before training which can improve the accuracy of model training effectively and reduce the deviation of training process in deep learning models. The mean and standard deviation of the red, green and blue channels on the ImageNet dataset are calculated to obtain the Normalized parameters which is [0.485, 0.456, 0.406] and [0.229, 0.224, 0.225].

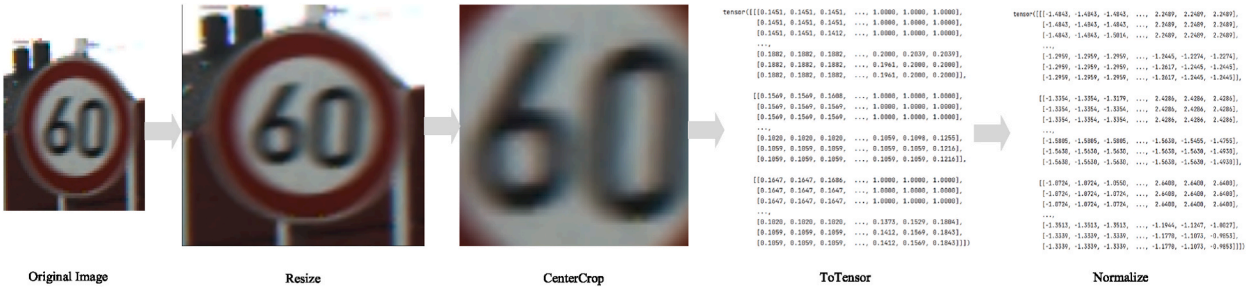

Fig. 3. Verification process preprocessing operations.

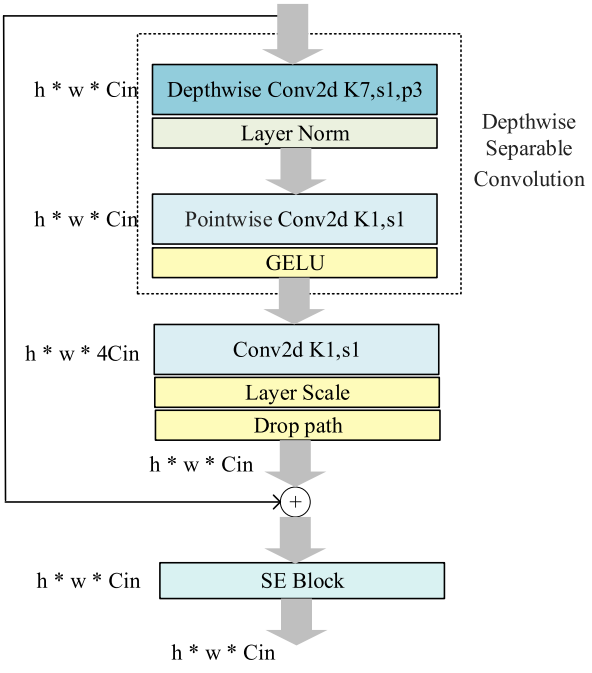

Fig. 4. ConvNeSe Block structure.

In this experiment, the pre-processing operations on the validation set data include Resize, CenterCrop, ToTensor, and Normalize as shown in Fig. 3. Resize scales the to the specified size 256*256, which can ensures the consistency of the input data and avoids model validation errors due to images of different sizes. CenterCrop crops the image from the center pixel to the specified size 224*224, which can reduce the interference of image edge information to the accuracy of the model.

3.2. *ConvNeSe block*

ConvNeSe Block is a network structure capable of extracting multi-scale features as shown in Fig. 4. Different types of convolutional modules and Inverted Residuals structure are introduced to optimize the model structure. The network can be extended to a deeper number of layers, and more abundant features can be extracted, so that the model can avoid the problem of gradient disappearance and information loss in the deep network. Meanwhile, Depthwise Separable Convolution and SE Block are used to improve the computational efficiency and multi-scale feature fusion capability of the models.

Depthwise Separable Convolution (DSC) is a key module in many efficient neural network architectures. It includes Depthwise Convolution and Pointwise Convolution as shown in Fig. 5 [41]. It uses Depthwise Convolution instead of common convolution [42, 43]. The computational cost of a common convolution as shown in Equation (1).

$$\text{cost} = h * w * Cin * Cout * K * K \tag{1}$$

Where h and w stand for the height and width of the feature map respectively, Cin stands for the number of channels of the input feature map, $Cout$ stands for the number of channels of the output feature map, K stands for the size of the convolution kernel. The calculated cost of DSC as shown in Equation (2).

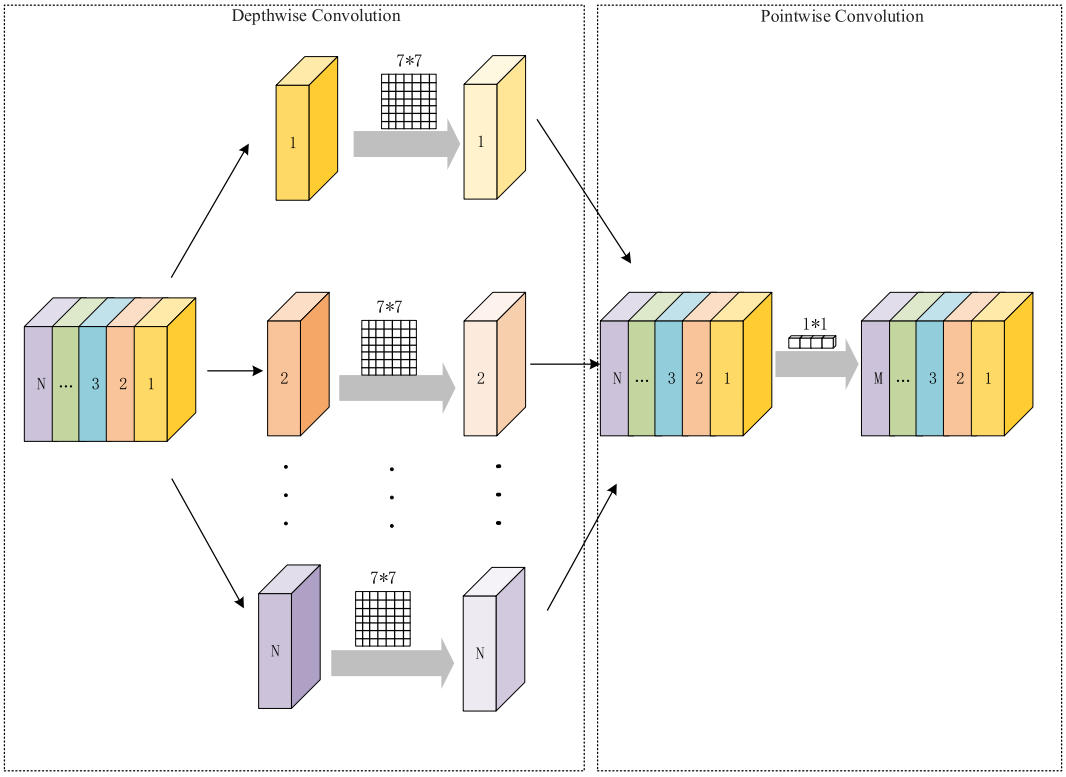

Fig. 5. Depthwise Separable Convolution structure.

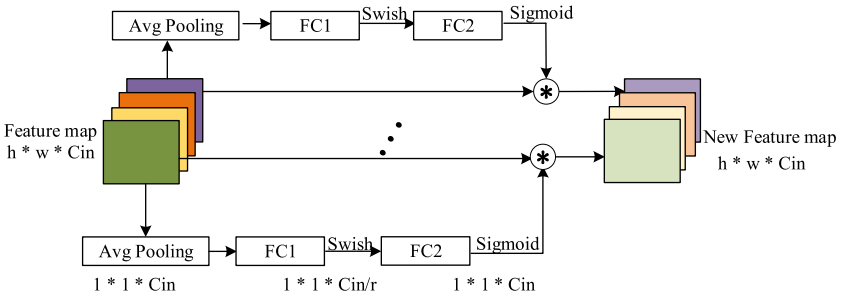

Fig. 6. SE Block structure.

$$\text{cost} = h * w * Cin * \left(K^2 + Cout\right) \tag{2}$$

Compared with traditional convolution, DSC reduces the computation costs by $1/K^2$ by reducing the number of parameters. Thus, the training time of the model is reduced. At the same time, Depthwise Convolution uses 7 * 7 convolution nuclei to increase the size of the receptive field. The model has the ability to extract global features and context information. The kernel size of Pointwise Convolution is 1*1. It combines the features of different channels in a weighted way. Make the model pay more attention to important features. In the residual part of the feature map, the deep feature is extracted by DSC, and then the channel number of the feature matrix is adjusted by convolution with kernel size of 1*1. The shallow features of the direct mapping part can be fully integrated with the deep features.

Each region of the image shows different importance for traffic sign recognition task. But CNN assumes that the features of all regions are equally important. SE Block is an attentional mechanism used to enhance the model's focus on important features as shown in Fig. 6 [44]. It adjusts the importance of feature channels by dynamic weights. Global Averaging Pooling operation is used for reducing the channel dimension of input feature maps to a scalar through. Then, two fully connected layers is used for learning the relationship between the channels and obtain a weight vector representing the importance of each channel of the feature map. The channel weight is multiplied with the original input feature map to weight the importance of the channel.

Specifically, suppose the input feature map is $X \in R^{h*w*Cin}$. Where h, w, and Cin stand for the height, width, and number of channels

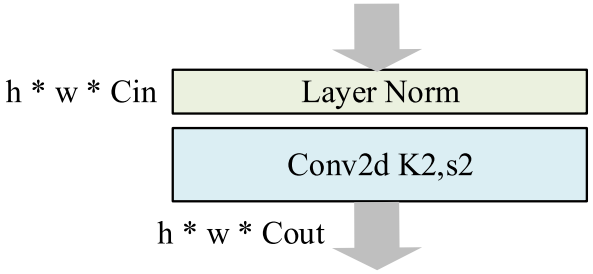

Fig. 7. Sample Block structure.

Table 2
Experimental platform configuration.

CPU	Intel Core i7-7700 K @ 4.20 GHz
GPU	NVIDIA GeForce GTX 3090
Memory	16 GB
Operating System	Ubuntu 18.04
Deep learning framework	PyTorch 2.0

of the feature map. The calculation formula of the SE module as shown in Equation (3).

$$\begin{aligned} \alpha &= AvgPool(X) \\ z &= Swish(W_1\alpha + b_1) \\ f &= \delta(W_2 z + b_2) \\ y &= f * X \end{aligned} \tag{3}$$

Where *AvgPool* stands for global averaging pooling operation, *Swish* stands for modified linear unit, δ stands for sigmoid function, W_1, W_2, b_1 and b_2 stand for weights and bias parameters,α stands for vectors after averaging the features of each channel, and z stands for the intermediate result after passing through the first fully connected layer. f stands for the channel weight vector through the second fully connected layer, y stands for the weighted output features, and * stands for element multiplication.

ConvNeSe Block solves the problems of gradient disappearance and performance degradation during model training effectively, and improves the training efficiency of the model. At the same time, the multi-scale feature extraction capability of the model is improved by using Inverted Residuals structure to fuse the feature maps of different scales and SE Block to learn the weights of different feature maps.

3.3. Sample block

In this paper, we propose a Sample Block consisting of a Layer Normalization layer and a convolution layer with kernel size of 2*2 as shown in Fig. 7. The stability of training can be improved by normalizing the feature matrix. Some information is lost during downsampling at the pooling layer. Therefore, we use convolutional layers for spatial downsampling to reduce information loss and improve feature representation.

4. Experiment and analysis

In order to demonstrate the superiority of our proposed method in traffic sign recognition task. Experiments and analyses are performed on the public dataset GTSRB. The configuration of the experimental platform as shown in Table 2.

4.1. Training settings

A total of 70 epochs are trained on the GTSRB dataset. The batch size is 16 during training.

We used the Cross Entropy Loss Function (CELF) to evaluate the model [45]. CELF measures the difference between the model's predicted results and the real results effectively, It is used to evaluate the classification ability of the model, optimize the parameters of the model, and improve the accuracy of the model. The formula for calculating the CELF as shown in Equation (4).

$$loss = -\frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{K}(Y\ln(y) + (1-Y)\ln(1-y)) \tag{4}$$

where N stands of the number of samples, K stands of the number of categories, Y stands of the true label of the sample, and y stands of the prediction probability value of the sample.

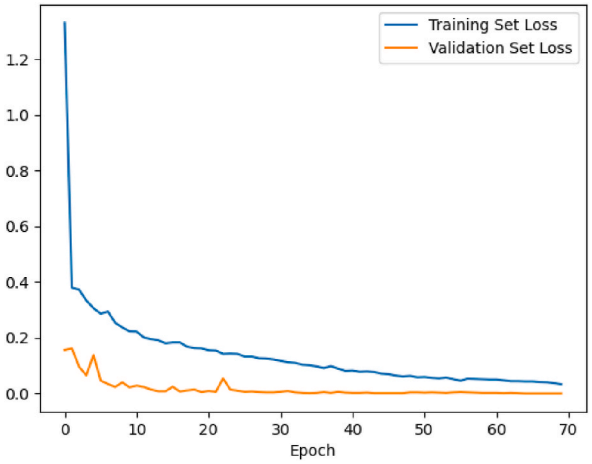

Fig. 8. Training set and validation set loss curves.

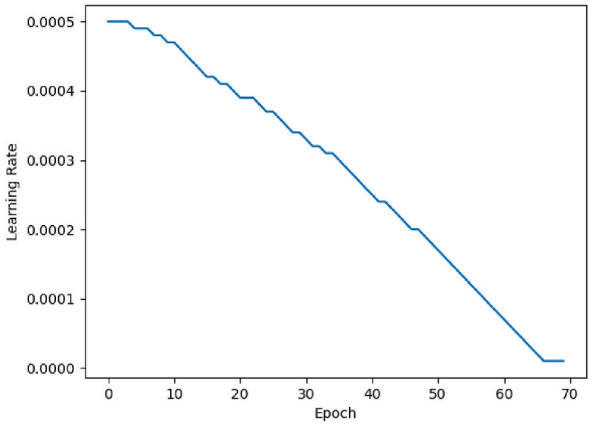

Fig. 9. Learning rate decay curve.

AdamW optimizer is used to optimizing the model parameters with a weight decay rate of 5e-2. The AdamW optimizer adds a mechanism for Weight Decay to the Adam optimizer [46]. It separates weight decay from gradient calculation and applies it to parameter updating, which can improve the stability and convergence performance of deep learning models effectively. The AdamW implementation as shown in Equation (5).

$$
\begin{aligned}
&m_t = \beta_1 m_{t-1} + (1-\beta_1)\mathrm{g}_t \\
&v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t{}^2 \\
&\hat{m}_t = \frac{m_t}{1-\beta_1{}^t} \hat{v}_t = \frac{v_t}{1-\beta_2{}^t} w_t = w_{t-1} - \frac{\alpha}{\sqrt{\hat{v}_t + \varepsilon}} * \left(\hat{m}_t + \lambda_w w_{t-1} \right)
\end{aligned}
\tag{5}
$$

Where m_t stands for the First Moment Estimation, v_t stands for the Second Moment Estimation, g_t stands for the gradient at the current iteration moment, $\widehat{m_t}$ and $\widehat{v_t}$ stand for corrections to first and second moments , β_1 and β_2 stand for the decay factors of the first and second moments respectively, α stands for the learning rate; ϵ is a very small constant to prevent the denominator from being 0, λ_w stand for the weight decay coefficient.

We use the learning rate to control the convergence speed of the model. The initial learning rate is 5e-4. The decay strategy of learning rate consists Warmup and Cosine annealing [47,48]. In the Warmup part, the learning rate is increased to the initial learning rate gradually. The goal is to avoid problems with the model such as overfitting or gradient explosion at the beginning of training of the model. In the Cosine annealing part, the learning rate will decrease in the form of the cosine function gradually to avoid problems such as oscillation or overfitting during the training of the model. Combining the above two parts, a smooth decay strategy of learning rate can be realized. The use of high learning rate can accelerate the convergence of the model and make the model approach the optimal solution quickly. As the model approaches the optimal solution, the high learning rate may cause the parameters to oscillate around the optimal solution, so the learning rate needs to be reduced gradually.

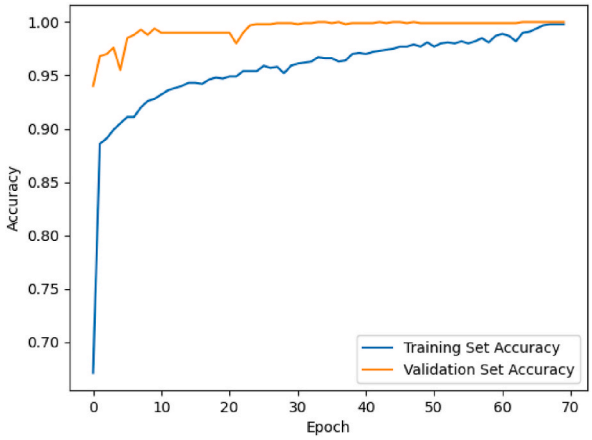

Fig. 10. Accuracy curves of training set and validation set.

Table 3
Precision, Recall and F1-score of each category in the GTSRB dataset.

Traffic sign label	Precision	Recall	F1-score
Speed limit 20 km/h	1.0	1.0	1.0
Speed limit 30 km/h	0.989	1.0	0.994
Speed limit 50 km/h	0.995	0.999	0.997
Speed limit 60 km/h	0.995	0.982	0.988
Speed limit 70 km/h	0.997	0.995	0.996
Speed limit 80 km/h	0.994	0.989	0.994
Speed limit 100 km/h	1.0	1.0	1.0
Speed limit 120 km/h	1.0	0.98	0.990
End of speed limit 80 km/h	1.0	0.990	0.995
No passing	0.998	1.0	0.999
Right-of-way at the next intersection	1.0	1.0	1.0
No passing for vehicles over 3.5 metric tons	1.0	1.0	1.0
Priority road	0.999	0.999	0.999
Yield	0.985	0.997	0.991
Stop	1.0	1.0	1.0
No vehicles	1.0	0.986	0.993
Vehicles over 3.5 metric tons prohibited	0.994	1.0	0.997
No entry	1.0	1.0	1.0
General caution	0.995	0.987	0.991
Dangerous curve to the left	0.996	1.0	0.998
Dangerous curve to the right	0.989	0.998	0.994
Double curve	0.989	1.0	0.994
Bumpy road	1.0	0.993	0.996
Slippery road	0.988	1.0	0.994
Road narrows on the right	1.0	1.0	1.0
Road work	0.99	0.99	0.99
Traffic signals	0.994	1.0	0.997
Pedestrians	1.0	1.0	1.0
Children crossing	0.994	1.0	0.997
Bicycles crossing	1.0	1.0	1.0
Beware of ice/snow	0.980	1.0	0.99
Wild animals crossing	0.993	1.0	0.996
End of all speed and passing limits	1.0	1.0	1.0
Turn right ahead	0.984	1.0	0.992
Turn left ahead	0.994	0.986	0.990
Ahead only	1.0	0.997	0.998
Go straight or right	0.999	0.999	0.999
Go straight or left	0.987	0.997	0.992
Keep right	0.997	0.980	0.989
Keep left	0.990	0.990	0.990
Roundabout mandatory	0.989	0.989	0.989
End of no passing	1.0	1.0	1.0
End of no passing by vehicles over 3.5 metric tons	1.0	1.0	1.0

Table 4
Successful identification of complex background.

Traffic sign image	Traffic sign label	Confidence
	Dangerous curve to the left	99.9%
	No passing	100%
	Road work	100%
	Keep right	99.9%
	Speed limit 80km/h	99.89%
	No passing for vehicles over 3.5 metric tons	100%

Table 5
Comparison between our CNN and other states of the art CNN classifier on GTSRB.

Model	param (M)	FLOPs (G)	Top-1 Acc. (%)
VGGNet [50]	138.36	15.47	91.74%
ResNet [50]	44.55	7.87	95.57%
ENet [21]	**0.9**	**0.21**	98.60%
MCDCNN [51]	38.5	7.14	99.50%
PFANet [30]	27.7	4.99	97.21%
ConvNeSe(ours)	26.87	4.46	**99.85%**

4.2. Main result

The change trend of training set loss decreases rapidly and becomes stable, and the training set loss converges to 0.33. The verification set loss shows a decreasing trend and fluctuates in the 5th and 20th rounds of training, and converges to 0.001 as shown in Fig. 8. The learning rate shows a downward trend and converges to 0.00001 as shown in Fig. 9. As the learning rate decreases to a stable state gradually, the accuracy of training and verification of the model converges gradually.

In the accuracy curve, the accuracy of the training set increases gradually and converges to 99.85%. The accuracy of verification set also showed a trend of gradual improvement and converges to 99.99% as shown in Fig. 10. The results show that the model is reliable in traffic sign recognition. It can be applied to many fields such as intelligent transportation system, which can improve the efficiency and convenience of intelligent traffic management [49]. The recognition rates of Precision, Recall and F1-score of each category in the GTSRB test set are above 98% as shown in Table 3. It shows that the recognition performance of the model is balanced for all categories. In addition, the model can recognize traffic signs with complex background well as shown in Table 4.

The ConvNeSe model is compared and analyzed with other advanced convolutional neural network models as shown in Table 5. ConvNeSe has fewer parameters and higher accuracy than VGGNet and ResNet. ENet has fewer parameters and FLOPs than the ConvNeSe model, which has advantages in computing resources and storage space, but its accuracy is lower than that of the ConvNeSe model. MCDCNN and PFANet are higher than ConvNeSe in model parameters and inference time, but lower than ConvNeSe in accuracy. In summary, ConvNeSe model performs well in accuracy and speed of inference. So it can be applied to scenarios that require high accuracy.

4.3. Ablation experiment

SE Block improves model accuracy by capturing contextual information and long-term dependencies. In order to verify the validity

Table 6
Ablation results on GTSRB dataset.

Model	Use SE Block	Param (M)	FLOPs (G)	Top-1 Acc. (%)
ConvNeSe-T	No	26.87	4.42	99.45%
ConvNeSe-T	Yes	26.90	4.44	99.85%
ConvNeSe-S	No	50.21	8.70	99.51%
ConvNeSe-S	Yes	50.23	8.86	99.89%

of SE Block, we conduct a series of ablation experiments on GTSRB as shown in Table 6. ConvNeSe-T and ConvNeSe-S are two different model structures, which T stands for the tiny model and S stands for the small model. Although the accuracy of the ConvNeSe-T is lower than the ConvNeSe-S slightly, the number of model parameters and FLOPs is much smaller than the ConvNeSe-S. In addition, the number of parameters and FLOPs of models without SE Block are slightly lower than those with SE Block, but the accuracy dropped a little. In particular, accuracy of ConvNeSe-T reduces from 99.85% to 99.45%. In summary, adding SE Block to ConvNeSe T and ConvNeSe S models can improve model performance. In particular, the improvement is especially evident in ConvNeSe-T.

5. Conclusion

In this paper, we propose a lightweight model for traffic sign recognition. It is used in traffic sign recognition successfully. We use Depthwise Separable Convolution, Inverted Residuals structure and SE Block to build a powerful feature extraction module called ConvNeSe Block. Depthwise Separable Convolution and Inverted Residuals structure are used to extract and fuse features at different levels. SE Block is used to pay close attention to important features. Our model not only advances in its simple implementation, but also shows superior ability to classification performance when compared to many recent competitors. The next step of our work focuses on improving the generalization ability of our model. The model also has good classification performance for traffic signs with low resolution. In addition, the study applies ConvNeSe as a backbone for traffic sign detection in complex backgrounds.

CRediT authorship contribution statement

Wei Wei: Writing – review & editing. **Lili Zhang:** Writing – review & editing, Software. **Kang Yang:** Formal analysis. **Jing Li:** Validation, Project administration. **Ning Cui:** Resources. **Yucheng Han:** Data curation. **Ning Zhang:** Data curation. **Xudong Yang:** Investigation. **Hongxin Tan:** Conceptualization. **Kai Wang:** Funding acquisition.

Declaration of competing interest

The authors declare no conflict of interest.

Acknowledgements

This work is supported by The RD Program of Beijing Municipal Education Commission (No. KM202210017006) , the Beijing Science and Technology Association 2021–2023 Young Talent Promotion Project (BYESS2021164), Beijing Digital Education Research Project (BDEC2022619048), Ningxia Natural Science Foundation General Project (2022AAC03757, 2023AAC03889), Beijing Higher Education Association Project (MS2022144), Ministry of Education Industry-School Cooperative Education Project (220607039172210, 22107153134955). The referees' valuable suggestions are greatly appreciated.

References

[1] W. Marcin, A. Zielonka, A. Sikora, Driving support by type-2 fuzzy logic control model, Expert Syst. Appl. 207 (2022) 117798.
[2] B. Janakiraman, S. Shanmugam, R. Pérez de Prado, M. Wozniak, et al., 3D road lane classification with improved texture patterns and optimized deep classifier, Sensors 23 (11) (2023) 5358.
[3] J.H. Chung, D.W. Kim, T.K. Kang, M.T. Lim, ADM-Net: attentional-deconvolution module-based net for noise-coupled traffic sign recognition, Multimed. Tool. Appl. 81 (16) (2022) 23373–23397.
[4] C. Dewi, R.C. Chen, H. Yu, Weight analysis for various prohibitory sign detection and recognition using deep learning, Multimed. Tool. Appl. 79 (2020) 43–44, 32897-32915.
[5] M.A. Abdou, Literature review: efficient deep neural networks techniques for medical image analysis, Neural Comput. Appl. 34 (8) (2022) 5791–5812.
[6] K. O'Shea, R. Nash, An Introduction to Convolutional Neural Networks, Computer Science (2015), https://doi.org/10.48550/arXiv.1511.08458.
[7] D.A. Alghmgham, G. Latif, J. Alghazo, et al., Autonomous traffic sign (ATSR) detection and recognition using deep CNN, Proc. Comput. Sci. 163 (2019) 266–274.
[8] G. Yao, T. Lei, J. Zhong, A review of convolutional-neural-network-based action recognition, Pattern Recogn. Lett. 118 (2019) 14–22.
[9] L. Anis, T. Ghada, S. Anis, M. Abdellatif, et al., Optimal feature selection based on hybridization of MSFLA and Gabor filters for enhanced MR brain image recognition using SVM, International Journal of Tomography & Simulation 27 (3) (2014) 3–5.
[10] A. Ladgham, A. Sakly, A. Mtibaa, MRI brain tumor recognition using modified shuffled frog leaping algorithm, in: 15th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA), IEEE, 2014, pp. 504–507.
[11] E. Cengil, A. Çinar, Z. Güler, A GPU-based convolutional neural network approach for image classification, in: International Artificial Intelligence and Data Processing Symposium (IDAP), IEEE, 2017, pp. 1–6.
[12] A. Krizhevsky, I. Sutskever, G.E. "Imagenet classification with deep convolutional neural networks.", Adv. Neural Inf. Process. Syst. 120 (2012) 1097–1105.
[13] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv (2014) 51–56, 1409.1556.

[14] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, A. Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.
[15] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, Proceedings of the IEEE conference on computer vision and pattern recognition (2016) 770–778.
[16] A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, H. Adam, MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications, 2017, https://doi.org/10.48550/arXiv.1704.04861.
[17] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.C. Chen, Mobilenetv2: Inverted residuals and linear bottlenecks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4510–4520.
[18] A. Howard, M. Sandler, G. Chu, L.C. Chen, B. Chen, M. Tan, H. Adam, Searching for mobilenetv3, Proceedings of the IEEE/CVF international conference on computer vision (2019) 1314–1324.
[19] M. Tan, Q. Le, Efficientnet: Rethinking model scaling for convolutional neural networks, in: International Conference on Machine Learning, PMLR, 2019, pp. 6105–6114.
[20] M. Tan, Q. Le, Efficientnetv2: smaller models and faster training, in: International Conference on Machine Learning, PMLR, 2021, pp. 10096–10106.
[21] X. Bangquan, W.X. Xiong, Real-time embedded traffic sign recognition using efficient convolutional neural network, IEEE Access 7 (2019) 53330–53346.
[22] C. Zhao, W. Zheng, Fast traffic sign recognition algorithm based on multi-scale convolutional neural network, in: Eighth International Conference on Advanced Cloud and Big Data (CBD), IEEE, 2020, pp. 125–130.
[23] X. Song, H. You, S. Zhou, W. Xie, Traffic sign recognition with binarized multi-scale neural networks, in: 2020 35th Youth Academic Annual Conference of Chinese Association of Automation (YAC), IEEE, 2020, pp. 116–121.
[24] Z. Lan, L. Wang, Z. Su, Traffic sign recognition algorithm based on multi-scale convolution and weighted-Hybird loss function, in: International Conference on Big Data Engineering and Education (BDEE), IEEE, 2021, pp. 84–89.
[25] G. Chen, Y. Deng, Multi-scale CapsNet: a Novel traffic sign recognition method, Frontiers in Signal Processing 3 (2019) 93.
[26] L.I. Zheng-you, G. Jing-bang, Y. Sun, Traffic sign recognition algorithm based on improved residual network, Comput. Mod. 4 (2022) 52.
[27] Z. Wei, H. Gu, R. Zhang, J. Peng, S. Qui, Convolutional neural networks for traffic sign recognition, CICTP (2021) 399–409.
[28] J. Wang, S. Jiang, W. Zhou, Traffic sign recognition based on improved VGG16 algorithm, in: Third International Conference on Computer Vision and Pattern Analysis (ICCPA 2023), SPIE, 2023, pp. 941–946, 12754.
[29] L. Daihui, Z. Shangyou, L. Wenhui, Y. Lei, A new cyclic spatial attention module for convolutional neural networks, in: 2019 IEEE 11th International Conference on Communication Software and Networks (ICCSN), IEEE, 2019, pp. 607–611.
[30] K. Zhou, Y. Zhan, D. Fu, Learning region based attention network for traffic sign recognition, Sensors 21 (3) (2021) 686.
[31] Y. Garg, K.S. Candan, M.L. Sapino, San: scale-space attention networks, in: 2020 IEEE 36th International Conference on Data Engineering (ICDE), IEEE, 2020, pp. 853–864.
[32] J.H. Chung, D.W. Kim, T.K. Kang, et al., Traffic sign recognition in harsh environment using attention based convolutional pooling neural network, Neural Process. Lett. 51 (2020) 2551–2573.
[33] J.H. Chung, D.W. Kim, T.K. Kang, M.T. Lim, ADM-Net: attentional-deconvolution module-based net for noise-coupled traffic sign recognition, Multimed. Tool. Appl. 81 (16) (2022) 23373–23397.
[34] J.L. Ba, J.R. Kiros, G.E. Hinton, Layer Normalization, 2016, https://doi.org/10.48550/arXiv.1607.06450.
[35] Larsson G., Maire M., Shakhnarovich G., FractalNet: Ultra-Deep Neural Networks without Residuals. (2016).DOI:10.48550/arXiv.1605.07648.
[36] J. Stallkamp, M. Schlipsing, J. Salmen, C. Igel, The German traffic sign recognition benchmark: a multi-class classification competition, in: The 2011 International Joint Conference on Neural Networks, IEEE, 2011, pp. 1453–1460.
[37] P. Sermanet, Y. LeCun, Traffic sign recognition with multi-scale convolutional networks, in: The 2011 International Joint Conference on Neural Networks, IEEE, 2011, pp. 2809–2813.
[38] J. Zhang, M. Huang, X. Jin, et al., A real-time Chinese traffic sign detection algorithm based on modified YOLOv2, Algorithms 10 (4) (2017) 127.
[39] J. Zhang, X. Zou, L.D. Kuang, J. Wang, R.S. Sherratt, X. Yu, Cctsdb 2021: a more comprehensive traffic sign detection benchmark," human-centric comput, Inf. Sci. 12 (2022).
[40] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, C. Igel, Detection of traffic signs in real-world images: the German traffic sign detection benchmark, in: The 2013 International Joint Conference on Neural Networks (IJCNN), IEEE, 2013, pp. 1–8.
[41] F. Chollet, Xception: deep learning with depthwise separable convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251–1258.
[42] H. Srivastava, K. Sarawadekar, A depthwise separable convolution architecture for CNN accelerator, in: 2020 IEEE Applied Signal Processing Conference (ASPCON), IEEE, 2020, pp. 1–5.
[43] M.Z. Asghar, F.R. Albogamy, M.S. Al-Rakhami, J. Asghar, M.K. Rahmat, M.M. Alam, H.M. Nasir, Facial mask detection using depthwise separable convolutional neural network model during COVID-19 pandemic, Front. Public Health 10 (2022) 855254.
[44] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7132–7141.
[45] A. Mao, M. Mohri, Y. Zhong, Cross-entropy loss functions: theoretical analysis and applications, 2023, https://doi.org/10.48550/arXiv.2304.07288.
[46] I. Loshchilov, F. Hutter, Fixing Weight Decay Regularization in Adam, 2018.
[47] Loshchilov I., Hutter F., SGDR: Stochastic Gradient Descent with Restarts. (2016).DOI:10.48550/arXiv.1608.03983.
[48] T. Cazenave, J. Sentuc, M. Videau, Cosine annealing, mixnet and swish activation for computer Go, in: Advances in Computer Games, Springer International Publishing, Cham, 2021, pp. 53–60.
[49] W. Wei, Q. Ke, A. Zielonka, et al., Vehicle parking navigation based on edge computing with diffusion model and information potential field, IEEE Transactions on Services Computing (2023) 1–11.
[50] X. Guo, C. Zhao, Y. Wang, Traffic sign recognition based on joint convolutional neural network model, in: Proceedings of the 2nd International Conference on Big Data Technologies, 2019, pp. 200–203.
[51] D. Cires¸ U. Meier, J. Masci, J. Schmidhuber, Multi-column deep neural network for traffic sign classification, Neural Network. 32 (1) (2012) 333–338.