Journal of Computers Vol. 34 No. 4, August 2023, pp. 15-24
doi: 10.53106/199115992023083404002

Face Recognition Method Based on Lightweight Network SE-ShuffleNet V2

Hong-Rong Jing[1,2], Guo-Jun Lin[1,2*], Zhong-Ling Liu[1,2], Jing-Li[1,2], Li He[1,2], Xuan-Han Li[1,2],

Hong-Jie Zhang[1,2], Shun-Yong Zhou[1,2]

[1] College of Automation and Information Engineering, Sichuan University of Science & Engineering, Yibin 644000, China
[2] Artificial Intelligence Key Laboratory of Sichuan Province, Sichuan University of Science & Engineering, Yibin 644000, China

{2366464126,386988463,1084400802,1712492521,1191401486,568317750, 2369095868,14409186} @qq.com

Received 10 July 2022; Revised 4 November 2022; Accepted 4 January 2023

Abstract. We develop a more efficient lightweight network based on SE-ShuffleNet V2 to address the issues of large parameter sizes and sluggish feature extraction rates in large networks in the field of face recognition. First, to increase the network's accuracy and inference speed, the ReLU activation function of the original ShuffleNet V2 basic unit is swapped out for a segmented linear activation function. Second, the SE attention mechanism is added to the lightweight network ShuffleNet V2, which may improve the effective feature weights and decrease the invalid feature weights, and the SE attention causes the network to focus on more helpful features. Finally, the addition of the Arcface loss function enhances the face recognition network's capacity for categorization. Experiments indicate that the SE-ShuffleNet V2 network that we created achieves superior performance under the parameters of position and age. Particularly, the LFW accuracy is 99.38%. The algorithm presented in this research significantly increases face recognition accuracy when compared to the original ShuffleNet V2 network, therefore the additional parameters and longer inference times can be disregarded. To match the accuracy of substantial convolutional networks, we developed the lightweight SE-ShuffleNet V2.

Keywords: ShuffleNet V2, activation function, SE attention, Arcface, face recognition

1 Introduction

An important area of research in computer vision is face recognition [1]. Faces are utilized in a rising number of mobile devices [2] due to their advantages of convenience, originality, and irreproducibility [3]. From the advent of convolutional neural networks through 2017, the prevalent trend has been to construct networks that are deeper and more complicated in order to improve accuracy. Due to the large amount of calculation required, conventional face recognition models have high demands for GPU computing power and memory storage capacity of mobile devices. Large models are challenging to run on machines with poor arithmetic performance and storage limitations. Therefore, many academics have consistently attempted to design new lightweight networks, which have had positive effect in image classification tasks, without significantly lowering the model performance. Three things must be considered by lightweight networks: few parameters, quick speed, and high accuracy. Real-world face recognition tasks typically attain maximum accuracy with a constrained computing budget [4]. The challenge of creating a model that is both lightweight and performs well is currently a popular area of study. Since 2017, an increasing number of lightweight-weight networks have been built. MobileNet series from Google [5], EfficientNet series [6], ShuffleNet series from Kuang Shi, and GhostNet from Huawei [7] are the more developed lightweight networks. According to experimental findings [8], lightweight networks cannot be employed for high-precision face recognition tasks since their accuracy is inferior to that of state-of-the-art large convolutional networks. The advantage of MobileNet V1 is that it has the most basic network architecture and fewest parameters. It is also the first network to be successful in the lightweight field. The suggested DW and PW convolution, which separates the parameters regulating the size and depth of the network, makes it simple to optimize the network and reduces the computation of ordinary convolution to one-ninth of the original one. Low recognition accuracy and subpar recognition performance for face recognition in natural environments, particularly for face datasets including pose and age, are drawbacks of MobileNet V1. The advantage of MobileNet V2

* Corresponding Author

is that it adopts ResNet's residual block structure and introduces the inverted residual structure, which is distinct from the traditional residual structure, but first employs PW convolution to carry out the up-dimensional operation and subsequently the down-dimensional operation. The performance of the V2 network is further enhanced and the network's parameters are decreased by applying the inverse residual structure. MobileNet V2's drawback is that despite adding a lot of parameters, the network performance is not improved. The advantage of MobileNet V3 is that it updates the network modules based on the V2 network, which is just an inverted residual module modified. The redesign of the activation function and the addition of SE attention are the two most significant features of the V3 network. Another disadvantage of MobileNet V3 is the large rise in the number of parameters and the lengthy feature extraction process. ShuffleNet makes use of DW convolution and the ResNet module's network design features. ShuffleNet has almost six times less parameters than ResNet while maintaining the same number of setup channels. ShuffleNet has the advantage of having a small number of parameters in the network and a high recognition accuracy. In order to address this disadvantage, we carefully considered the benefits and advantages of popular lightweight convolutional neural networks for face recognition and created the SE-ShuffleNet V2 lightweight neural network. Convolutional neural models in the SE-ShuffleNet V2 collection are quite effective. Additionally, because to variances in age and pose, different face photos of the same individual show a tremendous deal of variety. For face recognition tasks, a challenging issue is how to get the computer to recognize the relatively mild inter-class variance despite the interference of large intra-class variation.

With few parameters and quick feature extraction, the lightweight SE-ShuffleNet V2 network developed in this paper is compared against MoblieNet V1, MobileNet V2, ShuffleNet V2, ResNet-34, and ResNet-50 in the same experimental setting. The results demonstrate that SE-ShuffleNet V2 has recognition accuracy comparable to large networks, a small number of parameters, and feature extraction that is quick.

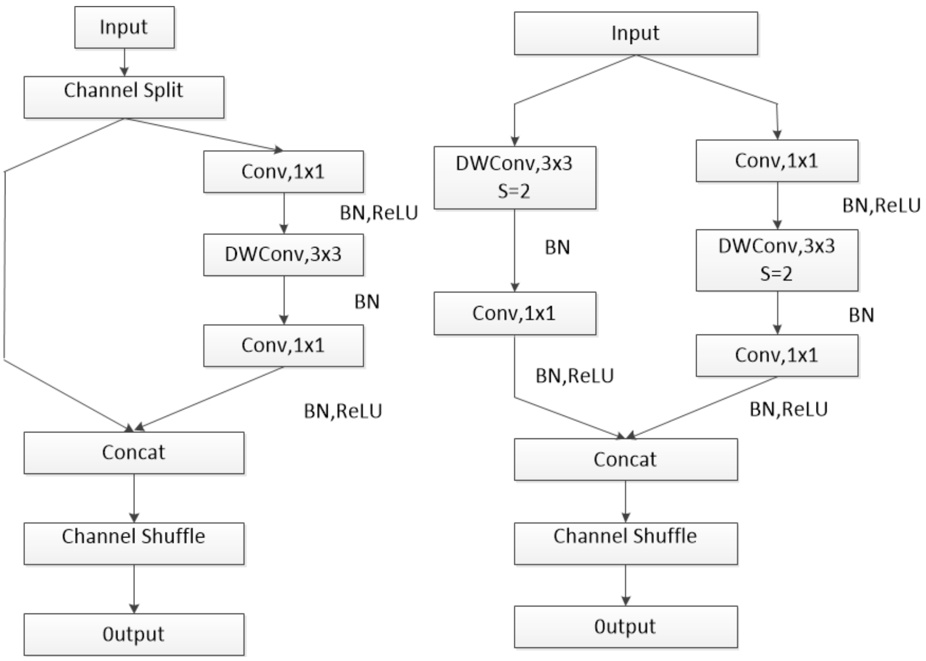
The content of this paper is structured as follows. Section I is the introduction. Section II is the ShuffleNet V2 network model. Section III is the SE-ShuffleNet V2 lightweight network. This section is divided into four sections: the first introduces the enhanced activation function, the second introduces the SE attention mechanism, the third introduces the loss function Arcface, and the fourth introduces the overall network design. Section IV is the experimental results and analysis. Section V is the conclusion. The last section is the references.

2 ShuffleNet V2 Network Model

The metric used to evaluate the computational complexity of lightweight networks is FLOPs (floating point operations). However, FLOPs can only assess the complexity of the model theoretically, and it does not use it only as an evaluation metric to measure the goodness of the network model. The actual inference time, memory access cost and parallelism factor of the network should also be considered [9]. Lightweight network ShuffleNet: It is a lightweight network designed specifically for mobile (e.g. cell phones, drones, small robots). ShuffleNet V2 [10] is an upgraded version of the proposed ShuffleNet V1. Although the accuracy of MobileNet V2 network is the highest, we select ShuffleNet V2 as the classification network in this paper because of the combination of training time, model size, top1 error rate. It outperforms ShuffleNet V1 and MobileNet V2 in terms of accuracy with the same complexity.

ShuffleNet [11] introduces the structure of group convolution (group convolution). The workflow of the group convolution layer is to first divide the input and output channels to the feature map into N groups, and then perform convolution operations in each group. Each group has one Nth of the feature maps, and they correlate the groups with each other without affecting each other, and each gets the output. The effect of dividing the channels into N groups is to reduce the computational complexity caused by the 1 x 1 convolution operation. Although grouped convolution is effective in reducing the computational effort of the network, it has a drawback. We only related the output of a cluster the input of that cluster, which leads to the isolation of information between different clusters and prevents information fusion across clusters. The lack of information interaction and integration between channels inevitably affects the feature extraction capability of the network, and we use PW convolution in MobileNet to solve this problem. ShuffleNet uses a very convenient way, the Channel Shuffle method [12]. It makes it possible to fuse information from different channels without convolutional operations. This method ensures that the feature mappings of different channel subgroups exchange information without increasing the computational effort. This operation is microscopic and also does not affect the backpropagation of the network.

ShuffleNet is an efficient, lightweight convolutional network, similar to the residual structure, and the differences between the two are: (1) using DWconv instead of the ordinary 3x3Conv, which reduces the complexity of the model; (2) finally using Concat operation instead of Add operation. Fig. 1(a) One of them first performs a Channel segmentation at the input side, dividing the features of the input face into two branches, where the main branch contains three convolution operations, the middle operation is a 3×3 depth divisible convolution, and the two ends are 1×1 point convolution, using the BN and ReLU activation functions, and the side branch is a constant mapping of the residual network. The main branch of the structure in Fig. 1(b) is also a 3 convolution operation with a depth-divisible convolution of step S of 2. The side branch contains a depth-divisible DW convolution of step S equal to 2 and a PW convolution. At the output side, feature fusion is performed by channel splicing (Concat) to merge the output feature maps of the main and side branches, and further channel shuffle is performed on the merged feature maps. The ShuffleNet V2 network structure is shown in the Table 1, where the Stage layer is composed of a stack of the basic units mentioned above, and Repeat is the number of repetitions. The first one in each Stage is the basic unit with step size 2, which is mainly used for down sampling, and the others are all basic modules with step size S of 1. It is possible to design networks of different complexity by scaling the number of channels of the output of each layer in the network structure.



(a) Basic base block (b) Down sampling block

Fig. 1. Basic unit of ShuffleNet

Table 1. ShuffleNet V2 1.5x structure

Layer	KSize	stride	Repeat	Output size	Channels
Conv1	3x3	2	1	112x112	24
MaxPool	3x3	2	1	56x56	24
Stage2		2	1	28x28	176
		1	3	28x28	176
Stage3		2	1	14x14	352
		1	7	14x14	352
Stage4		2	1	7x7	704
		1	3	7x7	704
Conv5	1x1	1	1	7x7	1024
GlobalPool	7x7				
FC				1x1	1000

3 Improved Lightweight Network SE-ShuffleNet V2

3.1 Improved Activation Function

In this paper, a Hard_Swish nonlinear activation function is added to the basic unit of ShuffleNet V2 network, which can improve the accuracy of the classification network in face recognition tasks when Hard_Swish is used instead of ReLU, which improves on the Swish [13] activation function. Swish's formula is:

$$Swish(x) = x \times Sigmoid(x) \quad . \tag{1}$$

The advantage of the activation function of Equation 1 is that it can improve the accuracy of the classification network, but the disadvantage is that the Sigmoid function will increase the computation of the network, which is unfriendly to mobile devices and unfavorable to real-time. Hard Sigmoid is used to replace the Sigmoid function to solve the problem of high computation.

$$Hard\_Sigmoid(x) = \frac{\mathrm{Re}LU6(x+3)}{6} \quad . \tag{2}$$

$$Hard\_Swish(x) = x \times \frac{\mathrm{Re}LU6(x+3)}{6} \quad . \tag{3}$$

In this paper, the replacement Sigmoid's segmented linear activation function Hard_Swish [14] is compared with Swish's nonlinear activation function, and the difference between the two in the model's accuracy is not significant. Equation III uses the ReLU6 function, which increases the inference speed of the network, and the difference in speed between the two is not significant compared to the direct use of the ReLU activation function. From the perspective of mobile deployment, such a substitution improves the efficiency of the model's operation on mobile platforms, while also eliminating the accuracy loss of the Sigmoid function in Swish in calculating the values. They improved the accuracy of the network while keeping the speed comparable to ReLU. After multiple convolutions, the cost of applying nonlinear functions to the network is much lower because we usually halve the feature map size of each layer after each convolution, and using the activation function of Equation 3 is more beneficial to implement lightweight networks. Therefore, in this paper, the basic unit module of ShuffleNet in Fig. 1 of the network makes the activation function of Equation III to improve the inference speed of the model.

3.2 SE Attention Mechanism

Convolution trained the attention mechanism to get the importance of each feature channel. When we finish the model training, the model can then automatically go to enhance the useful features based on the weight information saved by the training data, which will eliminate the perturbation of some unimportant features and help the network learn the important feature information. The SE network in the paper [15] is not a complete structure, but a substructure that can be combined with other classification models. In the paper, the authors inserted SENet into a variety of existing classification networks, all with excellent results. Of course, SENet embedded in some of the original classification networks inevitably increases some parameters and computational effort, but it is still acceptable in the face of the improved results.

SENet Needs to Perform Three Operations. First is the compression operation. The global average pooling operation is used to perform feature compression on the dimensions, and we match the output dimensions with the number of input feature channels to get the global compressed feature volume of the current feature map (Fig. 2).

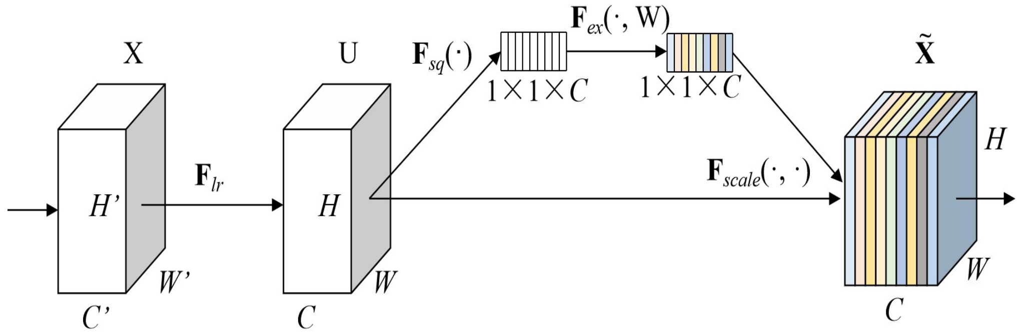


Fig. 2. SENet structure

$$F_{sq}(u_k) = \frac{1}{W\text{x}H}\sum_{i}^{W}\sum_{j}^{H}u_k(i,j)\ . \tag{4}$$

u_k is the input image when the input is W*H*C. K is the number of channels when the dimension is i*j. F_{sq} results from the first operation of the SE attention mechanism.

Next is the Excitation operation, which comprises two FC layers. The purpose of the initial FC layer is to implement dimensionality reduction with a decay factor of 16, and then to perform the dimensionality increase operation to the original dimension of 1*1*C, using the Sigmoid activation function to compress the weights to between 0 and 1. The two FC layers are used to increase the nonlinearity of the network and to reduce the number of parameters in the network. Let the weights be generated for each feature channel.

W_1 and W_2 are two FC layer operations. The dimensionality with a decay factor r of 2. Then the result is W_2 after a dimensionality increase to 1*1*C. After that, the result is F_{ex}.

Finally, there is a Reweight operation. A matrix multiplication step is applied. We superimpose the weight of the Channel of the stimulus output on the first feature. We achieve the reallocation in the Channel dimension.

3.3 Loss Function Arcface

We often use the Softmax function as the final classifier in convolutional neural networks used for multi-classification tasks. Combining it with deep learning algorithms can achieve better results. Loss function in face recognition. The loss function is a very important step in face recognition and belongs to the category of metric learning. In order to improve the performance of the model as much as possible, this paper uses the recently proposed Arcface loss function [16], which is got by modifying the loss function based on the traditional Softmax loss. However, the Softmax loss function can ensure good separability between classes, but the intra-class distance of features is scattered in a large range, and the intra-class features are not compact enough, and even the spacing of some intra-class features is larger than the inter-class distance. Its functional form is as follows.

$$\text{Loss} = -\frac{1}{N}\sum_{i=1}^{N}\log\left(\frac{e^{w_{y_i}^T x_i + b_{y_i}}}{\sum_{j=1}^{n} e^{w_{y_j}^T x_i + b_j}}\right)\ . \tag{5}$$

$w_{y_j}^T x_i + b_j$ are the output of the fully connected layer. Softmax mainly considers whether it can classify correctly and lacks constraints on intra-class and inter-class distances. Inamateur’s terms, it means that it is poor at classifying similar, not like faces. In order to improve the effectiveness of face recognition, the loss function must constrain the feature vectors to be as compact as possible within classes and as separate as possible between classes, besides ensuring separability. Therefore, an additive angle margin loss is used, using the arc-cosine function to calculate the angle between the current features and the weights. Then, an additional angular margin m is added to the target angle. To facilitate the calculation, so the bias is made equal to 0. The inner product of the weights and the input features are expressed as follows.

$$W_j^T x_i = \| W_j \| \| x_i \| \cos\theta_j \ . \tag{6}$$

We then regularized the weights regarding the features L2, $\|W_j\| = 1$ and $\|x_i\| = 1$. Thus, the learned to embed features are distributed on a hypersphere of radius s. Since the embedded features are distributed around the center of each feature on the hypersphere, we add an additional angle m between the weights and the features. The goal is all to make the feature vector more compact within classes and more separated between classes on the hyperspace. The loss function of Arcface is:

$$\text{Loss} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s(\cos(\theta_{y_i}+m))}}{e^{s(\cos(\theta_{y_i}+m))}+\sum_{j=1, j\neq y_i}^{n} e^{s\cos\theta_j}} \ . \tag{7}$$

N: Batch size
n: Number of categories
s: Hyperspherical radius
m: Angular Margin

3.4 Overall Network Framework

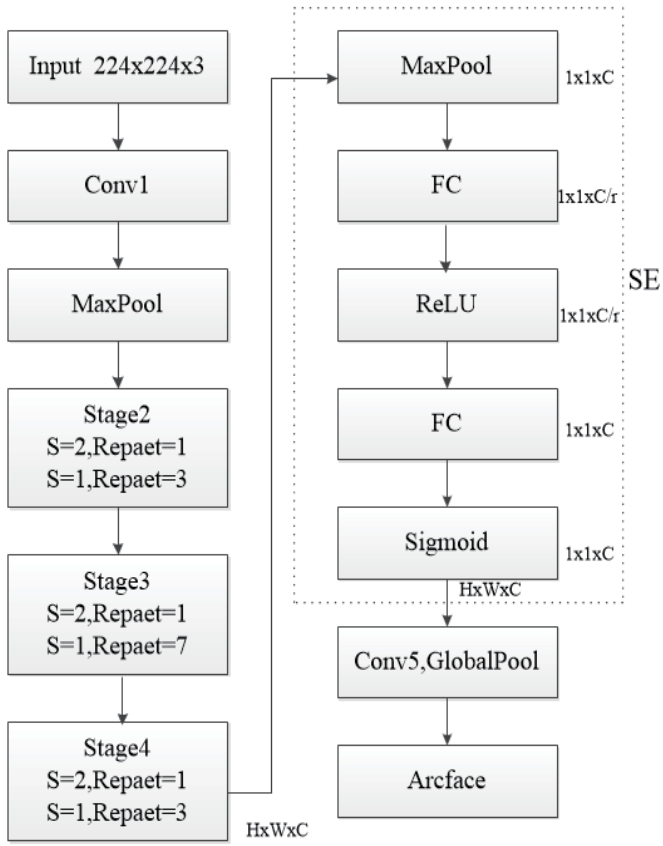


Fig. 3. Designed SE-ShuffleNet V2 network

In this paper, the ShuffleNet V2 network structure is further improved to design a network called SE-ShuffleNet V2 network. The improved network structure is shown in Fig. 3. The ReLU activation functions inside the basic units Stage2, 3 and 4 of Fig. 1 in the ShuffleNet V2 network are replaced with the Hard_Swish activation function of Eq. 3. Existing studies have shown that adding an attention mechanism to the model of convolutional neural network can significantly improve the performance of the model, so this study also adopts this idea by adding an SE module only after the Stage4 layer to improve the recognition accuracy. An Arcface classification loss function is also added to the original network with the purpose of expanding the inter-class margins and reducing the intra-class variance, thus improving the recognition ability of the trained model. The training process is that the input image is a color image with an aspect of 224, and after the first convolutional layer, the input is first down sampled using Fig. 1(b) with a step size of 2. The rest are used to then pick up the feature maps got from multiple base modules, and then the SE attention increases the weight of important features, and the SE operation does not change the dimensionality of the features, and finally the global average pooling is used to get a 1×1 feature map.

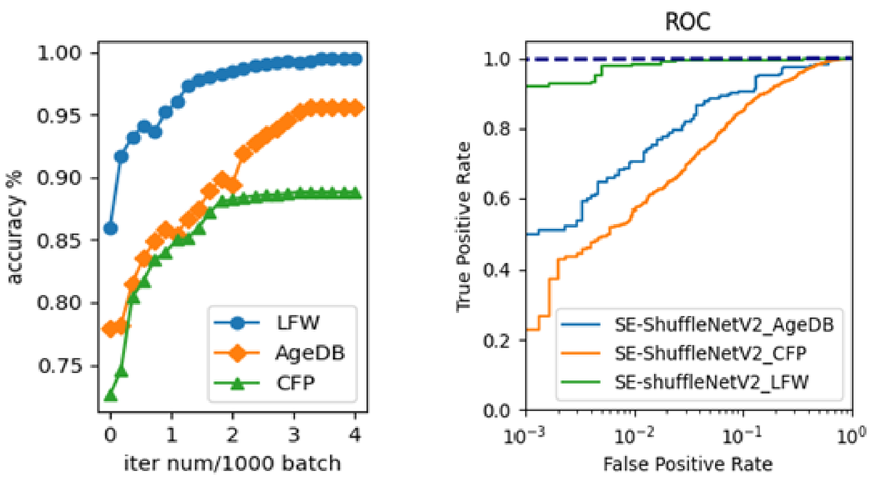
4 Experimental Results

4.1 Experimental Data Introduction

We usually use LFW as a test set for face recognition, in which there over than five thousand individuals with 13,000 face images. The AgeDB dataset has about 16,000 face images. The average age inside each class is 30 years. The CFP dataset contains 500 identities, each with 10 front faces and 4 side faces, with about five thousand images. This experiment uses the unrestricted face dataset with a training to test the division ratio of 9:1.

4.2 Experimental Results and Analysis

The experiments are based on the pytorch framework and the Python language is used to implement the network model and the corresponding evaluation metrics in this paper. We set the experimental parameters: learning rate 1/e, and the optimizer uses Adam to optimize the parameters for training. The training set accuracy of SE-ShuffleNet V2 network is shown in Fig. 4(a), when the network training reaches 3000 steps, the network has basically converged and the training accuracy does not increase and change smoothly. The training accuracy of LFW is 99. 46%, the training accuracy of AgeDB is 95. 66%, and the training accuracy of CFP is 88. 83%. Fig. 4(b) shows the ROC curves of the network designed in this paper got into three unrestricted datasets, and the closer the ROC curve is to the upper left corner represents the superior performance of the algorithm.



(a) Accuracy of the training set

(b) ROC curve

Fig. 4. Experimental results in this paper

The horizontal coordinate of the ROC curve is the probability of false positive (FPR) and the vertical coordinate is the probability of true positive (TPR). The ROC curve is used to measure the binary classification effect of different models, and the criterion for judging the algorithm is that the curve closer to the upper left corner represents the better algorithm.

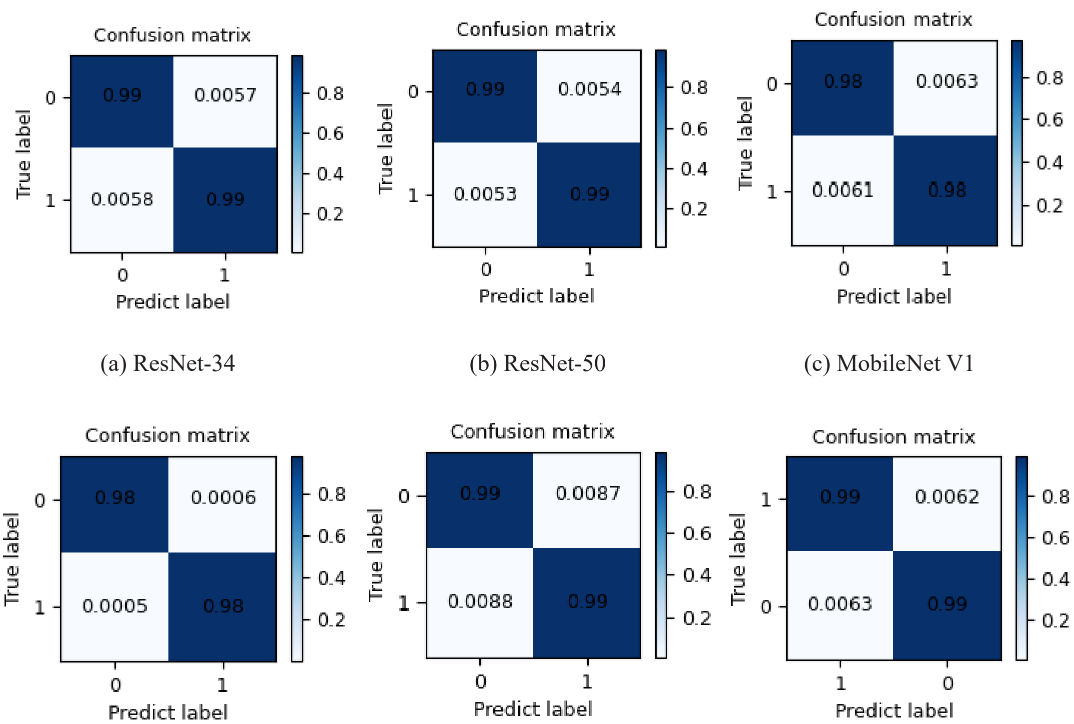
$$TPR = \frac{TP}{TP + FN} \quad . \tag{8}$$

$$FPR = \frac{FP}{FP + TN} \quad . \tag{9}$$

$$ACC = \frac{TP + TN}{TP + TN + FN + FP} \quad . \tag{10}$$

Equations (8), (9), (10) symbols represent the following meanings.
TP: The number of positive class faces divided into positive class faces
FN: The number of positive class faces divided into negative class faces
FP: The number of negative faces divided into positive faces
TN: Number of negative faces classified as negative faces

The confusion matrix can represent a classifier result, and in image accuracy evaluation, it is mainly used to compare the classification result with the actual measured value, and we can display the accuracy of the classification result inside a confusion matrix (Fig. 5).



(a) ResNet-34 (b) ResNet-50 (c) MobileNet V1

(d) MobileNet V2 (e) ShuffleNet V2 (f) SE-ShuffleNet V2

Fig. 5. Confusion matrix of six comparison experiments in LFW

We compared SE-ShuffleNet V2 with ResNet-34, ResNet-50, MobileNet V1, MobileNet V2, the ShuffleNet V2 algorithms in the same environment and the same parameters. Six confusion matrices were derived in the LFW dataset. The values inside the confusion matrices were normalized to facilitate the calculation of accuracy. The confusion matrix can show the result of a classifier. The accuracy of the classification results can be represented in the confusion matrix. The smaller value of the subdiagonal of the confusion matrix represents the higher accuracy rate. The SE-ShuffleNet V2 network designed in this paper outperforms other lightweight networks and can be comparable to large networks. To analyze and compare the performance of the networks designed in this paper, two large CNN models ResNet-34 and ResNet-50 and three lightweight CNN models MobileNet V1, MobileNet V2, ShuffleNet V2 1.5× are introduced in this experiment and compared with the networks designed in this paper. The different networks have the same hyper parameter settings during training. We compared the above networks in five evaluation metrics: accuracy, FLOPs, number of parameters, model size, and inference time. We show the experimental structure of the three data sets in Table 2.

Table 2. Experimental Results with Other Comparison Networks

Models	LFW	AgeDB	CFP	Parameters/ 10^6	Time /ms	FLOPs /10^8	Model size/MB
ResNet-34	99.43%	96.01%	94.46%	21.8	530.43	36.7	99.69
ResNet-50	99.47%	96.25%	94.67%	25.6	647.65	41.2	182.1
MobileNet V1	98.39%	90.01%	87.54%	4.2	111.34	1.07	7.14
MobileNet V2	98.95%	92.13%	88.23%	3.4	98.29	3.20	8.75
ShuffleNet V2	99.13%	94.38%	88.48%	2.3	92.78	3.04	9.66
SE-ShuffleNet V2	99.38%	95.64%	88.81%	2.305	92.79	3.047	9.67

As can be seen from Table 2, the SE-ShuffleNet V2 network constructed in this experiment achieved 99.38% accuracy in LFW, which is only lower than two large CNN networks and the highest among the lightweight networks. The SE-ShuffleNet V2 network is fast compared with other networks, so the SE-ShuffleNet V2 network model has a significant advantage in running inference speed. It is effective to exchange better recognition accuracy by introducing SE attention, Hard_Swish and Arcface functions. The simultaneous SE-ShuffleNet V2 network with less number of parameters can achieve the effect of real-time. The performance is not perfect in the five evaluation metrics in Table 2, but it is more helpful compared to the large network models. The inference time is the average time to recognize 100 images. The three datasets in the large CNN model ResNet-50 model has the highest accuracy, but the largest number of parameters, the speed is the slowest compared to other models, and the inference time in mobile devices is 647.65 ms, which does not meet the criteria of real-time for mobile devices. From the table, the performance comparison between the network created in this paper and other lightweight networks shows that the SE-ShuffleNet V2 in this paper achieves better accuracy compared to the other three lightweight networks, showing that the SE-ShuffleNet V2 is a network with excellent performance. Although it does not achieve the accuracy of the 50-layer residuals of the enormous network, the model in this paper is much better than the enormous network in the two evaluation indexes of the number of parameters, inference time, and model size. SE-ShuffleNet V2 is an efficient lightweight network model, and the recognition effect on the test set is ideal. It solves the problems of difficulty of deploying large CNN models on mobile devices and the low recognition accuracy of lightweight CNN models, and is more suitable for deployment on mobile devices, which can basically meet the demand for recognition accuracy on mobile devices.

5 Conclusion

The algorithm SE-ShuffleNet V2 designed in this paper achieves high accuracy in face recognition under unrestricted conditions such as pose and age. In addition, the network SE-ShuffleNet V2 proposed in this paper has the advantages of higher accuracy, smaller number of parameters and faster inference compared to large CNN networks and other lightweight networks, which can be well deployed on mobile devices. Of course, there are some problems, and the algorithm in this paper is not ideal for age and extreme side face recognition, the algorithm needs further improvement. In the actual application scenarios often contain more interference factors, the next step is to use GAN networks to solve the face recognition of age and extreme side faces, enhance the robustness of the face recognition model in different application scenarios.

Acknowledgement

Sichuan Science and Technology Program of China (2020YFSY0027); The Scientific Research Foundation of Sichuan University of Science and Engineering under Grant 2019RC11 and Grant 2019RC12; The artificial intelligence key laboratory of Sichuan province Foundation (2020RZY02); The Opening Project of Key Laboratory of Higher Education of Sichuan Province for Enterprise Informationalization and Internet of Things (2021WYY01).

References

[1] H.-Q. He, J.-Q. Yan, K.-H. Hui, Lightweight Face Recognition Method Based on Depth Residual Network, Journal of Computer Applications 42(7)(2022) 2030-2036.
[2] X.-L. Liu, T.-H. Li, M. Zhang, Face recognition based on lightweight neural network combining gradient features, Laser & Optoelectronics Progress 57(16)(2020) 161005.
[3] X. Han, H.-Y. Zhang, Y.-Y. Zhang, Face expression recognition based on efficient channel attention network, Transducer and Microsystem Technologies 2021(1).
[4] L.-F. Zhou, J.-L. Liu, W.-S. Li, J.-X. Mi, B.-J. Lei, Facial Expression Recognition Based on Deep Binary Convolutional Network, Journal of Computer-Aided Design & Computer Graphics 34(3)(2022) 425-436.
[5] Z. Qin, Z. Zhang, X. Chen, C. Wang, Yu. Peng, Fd-mobilenet: Improved mobilenet with a fast downsampling strategy, in: Proc. 2018 25th IEEE International Conference on Image Processing (ICIP), 2018.
[6] B. Koonce, EfficientNet, Convolutional neural networks with swift for tensorflow, Apress, Berkeley, CA, 2021 (pp. 109-123).
[7] M.-E. Paoletti, J.-M. Haut, N.-S. Pereira, J. Plaza, A. Plaza, Ghostnet for hyperspectral image classification, IEEE Transactions on Geoscience and Remote Sensing 59(12)(2021) 10378-10393.
[8] Y. Martinez-Diaz, M. Nicolas-Diaz, H. Mendez-Vazquez, L.S. Luevano, L. Chang, M. Gonzalez-Mendoza, L.E. Sucar, Benchmarking lightweight face architectures on specific face recognition scenarios, Artificial Intelligence Review 54(8) (2021) 6201-6244.
[9] R. Gong, S. Ding, C.-H. Zhang, H. Su, Lightweight and multi-pose face recognition method based on deep learning, Journal of Computer Applications 40(3)(2020) 704-709.
[10] N. Ma, X. Zhang, H.-T. Zheng, J. Sun, Shufflenet V2: Practical guidelines for efficient cnn architecture design, in: Proc. of the European conference on computer vision (ECCV), 2018.
[11] X. Zhang, X. Zhou, M. Lin, J. Sun, Shufflenet: An extremely efficient convolutional neural network for mobile devices, in: Proc. of the IEEE conference on computer vision and pattern recognition, 2018.
[12] F. Boutros, N. Damer, M. Fang, F. Kirchbuchner, A. Kuijper, Mixfacenets: Extremely efficient face recognition networks, in: Proc. 2021 IEEE International Joint Conference on Biometrics (IJCB), 2021.
[13] K.-P. Cai, L.-J. Ma, Research on face emotion recognition principle and deep learning mode, Journal of Anyang Institute of Technology 20(2)(2021) 49-51, 77.
[14] R. Avenash, P. Viswanath, Semantic Segmentation of Satellite Images using a Modified CNN with Hard-Swish Activation Function, in: Proc. VISAPP 2019 - 14th International Conference on Computer Vision Theory and Applications, 2019.
[15] Z. Deng, R. Yang, R. Lan, Z. Liu, X. Luo, SE-IYOLOV3: An accurate small scale face detector for outdoor security, Mathematics 8(1)(2020) 93.
[16] X.-Y. Mao, Y.-B. Pen, Landmark recognition based on ArcFace loss and multiple feature fusion, Journal of Image and Graphics 25(8)(2020) 1567-1577.