The Improvement of the Traditional Cascade Correlation Network and the Combination with the Convolutional Neural Network

Yuanfei Fan[1]

[1] Research School of Computer Science, Australian National University, Canberra Australia
u7155106@anu.edu.au

Abstract. As a feed-forward neural network, cascaded neural network has great advantages in face detection, mainly from its fast learning speed and ability to determine the network depth by itself. This paper first implemented a cascaded- correlation neural network based on the SFEW principal component data set, and the accuracy of TOP1 was 22%. Later, based on the shortcomings of the cascaded-correlation neural network, the network was improved and a new algorithm was constructed with an accuracy of 24%. Finally, the improved network and convolutional neural network are combined to train the original SFEW data set, and the accuracy rate is 29%. From the result point of view, the new algorithm are better than the traditional methods, but there is still a gap compared with the accuracy of SOTA in the same data set of 46.28%.

Keywords: Cascade Networks, Cascade layer, hidden neuron, Convolutional Neural Network, SFEW

1 Introduction

In the past time, due to the continuous progress of computer vision, face recognition technology has been widespread, and in the present environment, facial emotion recognition plays a vital role in face recognition research. In order to ensure the universality of the data set, the data set needs to have a large age span, different facial resolutions, and different degrees of occlusion. Based on the above several elements, I choose the SFEW data set as the data basis of this paper.

Compared with the general back-propagation neural network, the cascade-correlation neural network, as a feed-forward neural network, can greatly reduce the complexity of the network, because it has the characteristic of defining its own network depth, so it can better match the complexity of the problem to be solved. Therefore, it has many advantages in the field of image recognition, which is also the reason why I choose to cascade-correlation neural network as the basic algorithm in this paper.

However, the traditional cascade-correlation neural network still has many shortcomings, such as the need to train neurons continuously, resulting in a longer training period, the initial network has insufficient interpretation of non-linearity due to the small number of layers, etc. Therefore, the purpose of this paper is to improve the traditional cascade-correlation neural network in order to eliminate the shortcomings, and combine the network and deep learning to use the convolutional neural network to further improve the performance of facial expression classification.

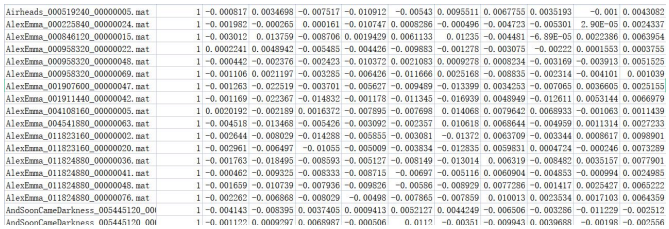
From the results, based on the principal component SFEW data set, the improved cascade correlation network proposed in this paper not only retains the good generalization properties of the construction cascade algorithm, but also effectively solves various deficiencies in the cascade correlation network. Moreover, for the original SFEW data set, a better recognition performance can be obtained through the combination with deep learning.

2 Method

2.1 Data Set

Principal component SFEW data set

Data set analysis. The data set used in this article is SFEW[1], which includes two parts: LPQ feature and PHOG feature. PHOG feature was used for most of the time in this paper. In the original data set, there are a total of 675 pieces of data, each of which contains the first 5 principal components of the corresponding picture,as the Fig. 1 shows. Among them, there are 100 expressions of angry, Fear, Happy, Neutral, Sad and Surprise, and 75 expressions of Disgust.



Airheads_000519240_00000005.mat	1	-0.000817	0.0034698	-0.007517	-0.010912	-0.00543	0.0095511	0.0067755	0.0035193	-0.001	0.0043082
AlexEmma_000225840_00000024.mat	1	-0.001982	-0.000265	0.000161	-0.010747	0.0008286	-0.000496	-0.004723	-0.005301	2.90E-05	0.0024337
AlexEmma_000846120_00000015.mat	1	-0.003012	0.013759	-0.008706	0.0019429	0.0061133	0.01235	-0.004481	-6.89E-05	0.0022386	0.0063954
AlexEmma_000958320_00000022.mat	1	0.0002241	0.0048942	-0.005485	-0.004426	-0.009883	-0.001278	-0.003075	-0.00222	0.0001553	0.0003755
AlexEmma_000958320_00000048.mat	1	-0.000442	-0.002376	-0.002423	-0.010372	0.0021083	0.0009278	0.0008234	-0.003169	-0.003913	0.0051525
AlexEmma_000958320_00000069.mat	1	-0.001106	0.0021197	-0.003285	-0.006426	-0.011666	0.0025168	-0.008835	-0.002314	-0.004101	0.001039
AlexEmma_001907600_00000047.mat	1	-0.001263	-0.022519	-0.003701	-0.005627	-0.009489	-0.013399	0.0034253	-0.007065	0.0036605	0.0025155
AlexEmma_001911440_00000042.mat	1	-0.001169	-0.022367	-0.014832	-0.001178	-0.011345	-0.016939	0.0048949	-0.012611	0.0053144	0.0066979
AlexEmma_004108160_00000005.mat	1	0.0020192	-0.002189	0.0016372	-0.007895	-0.007698	0.014068	0.0079642	0.0068933	-0.001063	0.0011439
AlexEmma_004541880_00000063.mat	1	-0.004518	-0.013468	-0.005426	-0.003092	-0.002357	0.010618	0.0068644	-0.004959	0.0011314	0.0027233
AlexEmma_011823160_00000002.mat	1	-0.002644	-0.008029	-0.014288	-0.005855	-0.003081	-0.01372	0.0063709	-0.003344	0.0008617	0.0098901
AlexEmma_011823160_00000020.mat	1	-0.002961	-0.006497	-0.01055	-0.005009	-0.003834	-0.012835	0.0059831	0.0004724	-0.000246	0.0073289
AlexEmma_011824880_00000036.mat	1	-0.001763	-0.018495	-0.008593	-0.005127	-0.008149	-0.013014	0.006319	-0.008482	0.0035157	0.0077901
AlexEmma_011824880_00000041.mat	1	-0.000462	-0.009325	-0.008333	-0.008715	-0.00697	-0.005116	0.0060904	-0.004853	-0.000994	0.0024985
AlexEmma_011824880_00000048.mat	1	-0.001659	-0.010739	-0.007936	-0.009826	-0.00586	-0.008929	0.0077286	-0.001417	0.0025427	0.0065222
AlexEmma_011824880_00000076.mat	1	-0.002262	-0.006868	-0.008029	-0.00498	-0.007865	-0.007859	0.010013	0.0023534	0.0017103	0.0064359
AndSoonCameDarkness_005445120_00(	1	-0.004143	-0.008395	0.0037405	0.0009413	0.0052127	0.0044249	-0.006506	-0.003286	-0.011229	-0.002512
AndSoonCameDarkness_005445120_00(	1	-0.001122	0.0009297	0.0068987	-0.005506	0.0112	-0.00351	-0.009043	0.0039688	-0.00198	-0.002556

Fig.1. Data in Principal component SFEW data set

Data pre-processing. First, traverse the data set to find whether there are errors or incomplete data,as the Fig. 2 shows, and remove them from the data set. Because the existence of these data will cause errors in the loss during training, which will lead to problems such as the model's failure to fit.

Hangover_001949614_00000089.mat	3	0.99401	-0.006083	-0.016374	0.0009698	0.0064847	NaN	NaN	NaN	NaN	NaN

Fig.2. errors in data set

Then, each column of the data set is normalized to avoid many problems, such as the input data being too small or too large, which causes the gradient to fail to drop, and the difficulty of model convergence.

Finally, the processed data is divided into training set and test set according to the ratio of 9:1.

Original SFEW data set

Data set analysis. The distribution of picture types in this data set is basically the same as that of the principal component data set, but the data is a picture with a size of 720*576, as shown in the Fig. 3.

Fig.3. Image in Original SFEW data set

Data pre-processing. First, filter errors and incomplete data the same as the principal component data set. Then, after traversing all the pictures in the data set, it is found that there are a large number of highly similar pictures in the data set,as shown in the Fig. 4. If part of these data is added to the test set and the other part is added to the training set, it will affect the evaluation of model performance. Therefore, the data set and the test set can no longer be divided randomly and can only be filtered manually. Finally, 70 pictures that did not appear in the training set were selected as the test set, 10 of which were in each expression category. The remaining 605 pictures are used as the training set.

Fig.4. Similar pictures in the data set

Then compress all pictures to a size of 224*224, and add a horizontal flip with a probability of 0.5 to the training data for data augmentation. The purpose of data enhancement is to improve the generalization ability of the model. Finally, the image data is standardized, and the pixel value is adjusted to (-1, 1).

2.2 Cascaded Correlation Neural Network

The traditional cascaded correlation neural network is shown in the Fig. 5.The initial network is a small model, its input and output directly connected by the full connection layer.Then, the simple neural network is trained until the loss of the network is no longer diminished.

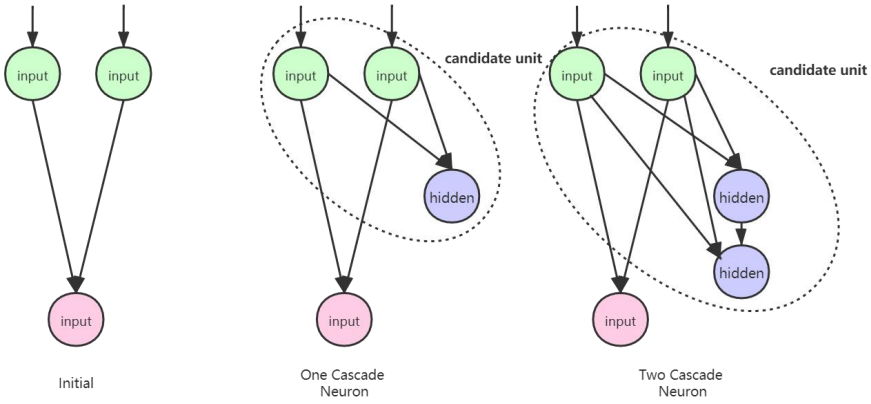


Fig. 5. The structure of the Cascor network

The second step is to construct a candidate unit, as the middle and right diagrams in Fig. 5 shows, which includes all the inputs and the existing hidden neurons.

$$V_p = \varphi(I_p \cdot W) \tag{1}$$

$$S = \sum_o |\sum_p (V_P - \overline{V})(E_{o,p} - \overline{E_o})| \tag{2}$$

The third step is to build a neuronal training network and input all the data and labels of the current candidate units.Here I replaced the original correlation with the covariance S of the residual of the candidate unit and the network according to Fahlman's method[3] , as shown in the formula, where φ is the activation function, E is the residual error, and V is the output of the activation function. The maximum value of S calculated through the network. After the optimal solution is obtained, the output of the model is added to the network as a hidden neuron, and the previous weights are frozen at the same time.

Finally, retrain the network and repeat the above process until the loss reduction of the entire network is less than the set threshold.

2.3 Improved Cascade Neural Network

The traditional cascading correlation network has many drawbacks, firstly, there is only one fully connection layer

between input and output in the initial network, which large-scale input may cause the network parameter dimension to be too large and training difficult.

Second, since the cascaded-correlation network needs two training stages, one is the training of the whole network, the other is the training of neurons, so more time is needed in a training cycle.

Third, the cascaded correlation network adds a hidden neuron every time, so there may be a large number of individual neurons when reaching the optimal model. The combination of these individual neurons and the input may also lead to the large dimensions of network parameters .

Based on the above three questions, I got the inspiration from Suisin Khoo and Tom Gedeon's paper[2] and designed a new constructional cascade network on principal component SFEW dataset. As shown in the Fig. 6 below.

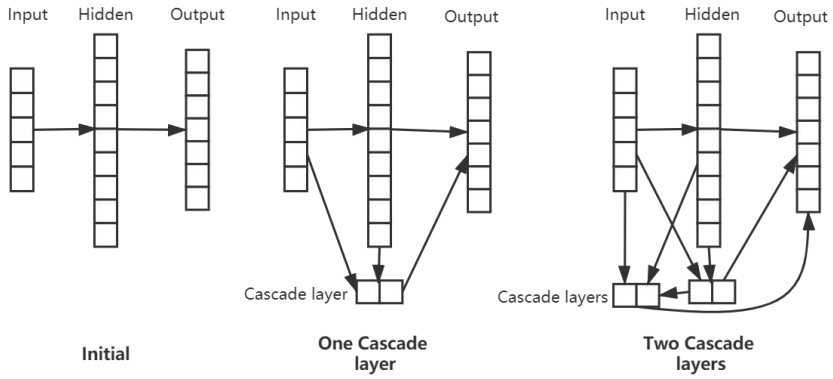


Fig. 6. The structure of the Improved Cascade Neural Network on principal component SFEW dataset

The first improvement : I made some changes to the initial network. Compared with the cascaded-correlation network, my initial network added a hidden layer between the input and output. Since this data set was the principal component SFEW dataset, one-dimensional vectors are input and and there is less data, the structure of the hidden layer was designed as 1 by 25. In this way, compared with smaller hidden layers, although the network becomes larger, it can better improve the generalization performance and accuracy of the model. Each neuron in the hidden layer is connected to the input neuron, and also to each output neuron.

The second improvement: As shown in the middle diagram in Fig. 6, the hidden layer is connected to a new network structure. This new network is called the cascade layer, which replaces the previous cascade neuron. Treadgold and Gedeon has a similar algorithm[4]. The size of the cascade layer is designed to be 1 by 2, because the hidden layer is designed to be large, so the total number of neurons needs to be controlled. Similar to the cascaded-correlation neural network, a cascade layer is added to the network when the loss is not reduced in the process of initial model training. Since the network is still a cascade-correlation structure, adding the cascade layer requires freezing the weight of the previous network, and only trains the new cascade layer. The input of each cascade layer comes from three parts. The first part is the mapping from the input layer, the second part is the mapping from the hidden layer, and the last part is the mapping from all the previous existing cascade layers,as the right diagram in Fig. 6 shows. Finally, the sum of these three parts forms a new cascade layer to join the network.

Through the above two changes, the deficiencies of the cascaded-correlation network proposed before can be solved to a certain extent. The problem of large-dimensional network parameters can be solved by adding a hidden layer. The addition of the hidden layer is more conducive to the extraction of non-linear information from the network and is conducive to improving the accuracy of the model. Moreover, the replacement of cascade neurons by the cascade layer can solve the multiple training problems of the cascade-related neural network. The cascade layer no longer needs to train the network separately but directly joins the network and can be trained together with the original network. At the same time, the emergence of the cascade layer can add multiple neurons at a time, which can improve efficiency and no longer need to be combined with input data to generate large-dimensional parameters.

Since it is impossible to use extracted principal component data in actual use, it is still necessary to further reform the network so that it can train the complete image data.

2.4 Cascaded Convolutional Neural Network

As mentioned at the end of the above, in the actual classification, the input data cannot be the extracted principal component information, but the original image, so the original SFEW data set will be used in this part. Correspondingly, the structure of the network should also change. The main structure is still the same as Fig. 6, but the input data of the previous network is a one-dimensional vector, so the network uses a linear layer except the output layer , but after replacing the input data with 224*224*3 pictures, each hidden layer of the network needs to be transformed into a convolutional layer to obtain image features.

However, after changing the form of the hidden layer, we will find that the entire network has only one hidden layer, so the feature extraction of the image will be too thin, and the image data will not be able to obtain accurate and complete information after passing through the network, which will lead to the model not being well recognized effect.

Therefore, this paper proposes to use a convolutional neural network to replace the input module for image feature extraction, and then utilize cascade layers as a new classification module in the network to modify the classification effect of the main classifier, so that as the cascade layer increases, It is possible to avoid the problem of poor performance of the main classifier in the network leading to poor classification effect of the model. The final network structure is shown in the Fig. 7

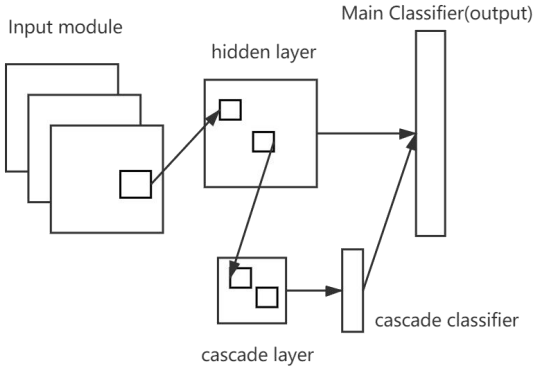


Fig.7. Cascaded Convolutional Neural Network

Next, this paper will introduce the used convolutional neural network.

ResNet (Residual Network): The VGG network puts forward a point of view that the deeper the network, the stronger the expressive power of the network[8]. With this view, the CNN network has developed from the 7th layer of Alexnet to the 19th layer of the VGG network and finally has the 22nd layer of Googlenet. However, as the number of network layers deepens, the results show that blindly increasing the depth does not bring about further improvement in classification performance, but will cause network convergence to become slower, and the classification accuracy also becomes worse. This is because the increase in network depth is accompanied by vanishing gradients and exploding gradients.

The vanishing gradients means that the basis of the BP algorithm is the chain rule of derivatives and the chain rule will make the calculation result very small after multiple calculations, which will cause the parameters to basically not fluctuate as the network deepens.Then when the gradient propagates to the shallow network, the shallow network cannot receive loss information. Gradient explosion is the opposite of it.

The solution to this problem in ResNet is very simple, it changed each unit to the bottleneck structure as shown in the Fig. 8[7]. The shortcut is equivalent to a path that can skip the weighted layer, and the gradient at the weight is added with a gradient without attenuation.So even if the depth of the network is very deep, the gradient from the deep layer can also reach the shallow network, making this Network parameters can be effectively trained.Through the design of the bottleneck structure, ResNet solves the problem of disappearing and exploding gradients as the network depth increases. This allows the gradient to be spread to the shallow network without obstacles for better training results.

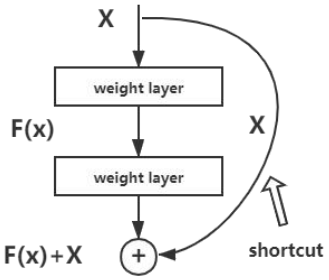


Fig.8. Bottleneck structure

Moreover, the batch normalization layer is added to ResNet. Batch Normalization normalizes the input data of each layer to N(0,1) so that it is concentrated near the mean with a small variance.It makes the activation input value fall in the area where the nonlinear function is more sensitive to the input, so that a small change in the input will lead to a larger change in the loss function, which means that the gradient becomes larger to avoid the problem of vanishing gradients. And the larger the gradient means the fast learning convergence speed, which can greatly speed up the training speed.

2.5 Train Methodology

Data set: For principal component SFEW data set, construct the data set class FrameDataset, extract three kinds of information of the data set, one is the overall content, one is the input data, and the other is the label.

For original SFEW data set, construct the setdata class, extract the data information, label, and names of the picture , and perform pre-processing and data augmentation on the picture. And take out 10% of the training set as the validation set.

Cascor: The algorithm contains two kinds of networks. The first is the overall training network, which is designed to have only one fully connected layer. Since there are a total of 7 categories, the output is 7 units. The second is the network used to train candidate units. Considering the overall training time, it is also designed as a fully connected network with only one layer, and the output dimension is set to 1. Because it is a multi-classification problem, the activation function of the overall network is set as the softmax function, and the loss function is the cross-entropy loss function[6]. Since the candidate unit network generates individual neurons, the activation function is set to the sigmoid function. The algorithm uses the Adam optimization algorithm to update the whole weights, because Adam has higher computational efficiency, less memory requirements, and the ability to handle large-dimensional parameters[5]. In the training process, the initial network is used for training at the beginning. When the loss of the network is less than the set threshold, the candidate unit is trained, and the output is added to the network as a neuron. In the part of adding neurons, since it is necessary to ensure that the neurons are added and the parameters before the neurons are frozen, I directly connect the new neurons with the input data and input them into the overall network as a new input. This not only ensures the addition of new neurons, but also freezes the previous parameters, which perfectly implements the algorithm.

Improved cascade network: As mentioned in the method section, the improved algorithm only requires one network instead of two. The initial network consists of a hidden layer and an output fully connected layer. Since the data size in the data set is 1 by 5, the hidden layer is also set to be linear. And, since there are 7 categories in total, the output layer is 7 units. After adding the cascade layer, each newly added cascade layer is connected to the existing cascade layer, and the output is used as the input to connect to the fully connected layer of the network output. The weight update of the network still uses the Adam optimization algorithm[5], and the loss function selects the cross-entropy loss[6]. Similarly, because it is a multi-classification problem output, the fully connected layer still uses the softmax activation function. But other layers in the network, such as the hidden layer and the cascade layer, all use the sigmoid activation function. In the training process, start training with the initial network, until the loss reduction is less than the set threshold, a new cascade layer is added to the network and freeze the previous network weight. In the calculation of the

cascade layer, the input of the current network, the output of the hidden layer, and the output of all existing cascade layers are extracted respectively, and these values are linearly processed and then added as the input of the new cascade layer. The input of the final output layer is the sum of the output of the hidden layer and the input of all cascaded layers.

Cascaded Convolutional Neural Network: First, remove the last fully connected layer and the 2d adaptive mean convergence layer of the ResNet network, then use the remaining part as the backbone of the network. This is because this algorithm only needs to obtain the feature of each image, and the subsequent classification results are responsible for the cascaded network structure. Because the number of output channels of the last layer of ResNet feature extraction is 512, the number of input and output channels of the hidden layer are both 512. Because the feature information extracted by ResNet should be kept as much as possible, the size of the convolution kernel ,the stride and the padding of the hidden layer are consistent with the last layer of ResNet, which are 3, 1, and 1, respectively. The connection and calculation method of the cascade layer is the same as the previous part of the improved cascade network. Since it is a cascading-correlation network, it is still necessary to freeze the weights when joining the cascade layer. The input and output channels of cascade layer are both set as 512. At the same time, because the output structure of the cascade layer is only used to assist in correcting the main classifier, it needs to be restricted, the size of the cascade layer is smaller than the hidden layer. When ResNet is used as the feature extraction module, the output size of the cascade layer is set to 4*4. The weight update still uses the Adam optimization algorithm, and the loss function selects the cross-entropy loss function. However, the activation function is replaced with the Relu function, which can increase the sparseness of the data and improve the fitting speed. At the same time, the batch normalization layer is added after the hidden layer and the cascade layer to further improve the convergence speed. Finally, a 2d adaptive mean convergence layer is added before each classifier to convert 512 channel feature maps into 512 pixels for classification.

3 Results and Discussion

As mentioned above, I used two different data sets to evaluate the three methods I designed. First, in the principal component SFEW data set, there are two types of data, one is Local Phase Quantization (LPQ) features, and the other is Pyramid of Histogram of Gradients (PHOG) features. In the experiment, I tested PHOG features on Cascaded Correlation Neural Network and Improved Cascade Neural Network. The problem to be solved is the facial emotions in 7 categories. The information of each facial emotion is input to the network through a size of 1 by 5, and the 7 output neurons respectively represent the probability of each category.

For principal component SFEW data set

Cascaded Correlation Neural Network: Table 1 shows the top1 and top2 accuracy of the test set after adding different numbers of neurons to the cascade correlation neural network in PHOG.

Table 1. Top1 and top2 accuracy of the test set vs number of neurons in PHOG(lr = 0.01, batchsize =64)

number of neurons	Top1 accuracy %	Top2 Accuracy%
0	17	21
5	20	29
10	21	32
15	22	37

From the above table, we can see that as the number of cascade neurons increases, the accuracy of the test set continues to rise, indicating that as the number of cascaded neurons containing existing network information increases, it has a benign gain to the learning of the network and improves the recognition effect of the model.

Improved Cascade Neural Network: From Table 2, when the PHOG feature is used for the experiment,the top1 and top2 accuracy of the test set in the improved cascade network.

Table 2. Top1 and top2 accuracy of the test set vs number of cascade layers in PHOG(lr = 0.01, batchsize =64)

number of cascade layers	Top1 accuracy %	Top2 Accuracy%
0	16	21
3	18	41
5	23	40
7	24	44
10	17	22

Through the comparison, it can be found that when the cascade layer just starts to increase, the accuracy of the test set continues to improve, but when it reaches more than 10 layers, the accuracy begins to drop rapidly. This is because the network model is too large after 10 layers, and the input data volume is small, which leads to the phenomenon of over-fitting of the model.

Table 3 is the comparison between the accuracy of the cascaded-correlation network and the improved network, as well as the training time. The training time is the time between the start of training and the decrease in the overall loss of the network less than the set threshold.

Table 3. Comparison between Cascor and Improved Cascade in PHOG,optimal results within 10 cascade structures(lr = 0.01, batchsize =64)

Network structure	Top1 Accuracy%	Top2 Accuracy%	Training time
Cascor	21	32	~120 min
Improved Cascade	24	44	~35 min

We can see that both the training time and accuracy of the improved network are higher than that of the cascaded correlation network, so it is proved that the improved algorithm does make up for many defects of the traditional Cascor network.

Second, I used the original SFEW data set to evaluate the cascaded convolutional network.The input data of the network is a picture of 224*224*3, and the output is the probability corresponding to the 7 expressions.

For original SFEW data set

Table 4. Top1 accuracy of the test set vs number of cascade layers (lr = 0.01, batchsize =10)

number of cascade layers	Top1 accuracy %
0	21%
1	26%
2	29%
3	27%
4	19%

From Table 4, we can find that with the addition of the cascade layer, the classification effect can be significantly improved at the beginning, but after the increase to 3 cascade layers, the accuracy rate begins to decrease. This is because too many cascade layers are added, and the proportion of the cascade classification module's influence on the main classifier increases and thus has a negative impact on the main classifier, resulting in poor classification results.

Table 5. Top1 accuracy of the test set vs different network structures (lr = 0.01, batchsize =10,epoch=100)

Network structure	Top1 Accuracy%
CasCNN(Cascaded Convolutional Neural Network)	29%
Improved Cascade	15%
ResNet	24%
CasCNN transfer learning	21%

Table 5 counts the classification effects of 4 different networks under the influence of the same set of hyperparameters. Among them, CasCNN transfer learning directly use ResNet pre-training weights for transfer learning. It can be seen from the results that if an improved cascade network without CNN as input is used, the classification effect is particularly poor. This is because the network is too shallow to express the features, the classification effect is very poor. The method of transfer learning is not very good either, because the pre-training weights used in transfer learning are based on the ImageNet data set, so this weight cannot extract the features needed by the network on the SFEW data set. CasCNN is better than ResNet, indicating that the addition of the cascade layer can indeed improve the classification performance of the main classifier.

However, though the results of these three algorithms are higher than the baseline 19% accuracy of the classification using a non-linear SVM in data set paper, there is still a big gap with SOTA's 46.28% accuracy rate[1]. I think that the reason for such a big gap is due to the feature extraction of the picture. The facial expressions of the people in the data set only account for a small proportion. Therefore, in the process of extracting features, the network may not finally obtain the facial features we need, which resulted in incorrect classification.

4 Conclusion and Future Work

In this paper, I introduced three network structure algorithms to implement facial emotion recognition based on the two different SFEW data sets. The first is to build a traditional cascading correlation network, which is built through two-part training. After testing on principal component SFEW data set, it can be obtained that the top2 accuracy of this network is around 40%, with a fluctuation of around 3% and the top1 accuracy is around 22%. The second network structure introduces the concept of cascading layer. Many shortcomings of the traditional network are solved by the addition of the cascade layer. The test has also verified my ideas. Compared with the traditional network, the training time is greatly shortened and the top2 recognition rate of the test network reached 45% and the effect was better than before. The third network uses a combination of improved networks and convolutional neural networks for actual image classification. This network optimizes the classification results of the main classifier through cascading layers, thereby improving the classification ability of the model. The final experiment also proved that my design, the top1 accuracy of CasCNN is 29% which is better than the one without cascade layers. Both of these networks surpass the baseline in the data set paper, but there is still a gap between the best results[1].

In the future, I will try to add target detection to the network to extract the position of the face in each image, so that the data trained by the network will no longer have a lot of interference results. Moreover, I will try not only to cascade classification modules, but to cascade different convolutional neural networks, and use multiple models to extract features at different positions of the picture, so that the features obtained are more conducive to improving the performance of the entire model.

References

1. Dhall, A., Goecke, R., Lucey, S., & Gedeon, T. (2011, November). Static facial expressions in tough conditions: Data, evaluation protocol and benchmark. In 1st IEEE International Workshop on Benchmarking Facial Image Analysis Technologies BeFIT, ICCV2011.
2. Khoo S, Gedeon T. Generalisation Performance vs. Architecture Variations in Constructive Cascade Networks[C]//International Conference on Neural Information Processing. Springer, Berlin, Heidelberg, 2008: 236-243.

3. Fahlman S E, Lebiere C. The cascade-correlation learning architecture[R]. CARNEGIE-MELLON UNIV PITTSBURGH PA SCHOOL OF COMPUTER SCIENCE, 1990.
4. Treadgold, N.K., Gedeon, T.D.: Exploring Architecture Variations in Constructive Cascade Networks. In: IEEE Int. Jt. Conf. on Neural Networks, pp. 343–348 (1998)
5. Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980, 2014.
6. Rubinstein R Y, Kroese D P. The cross-entropy method: a unified approach to combinatorial optimization, Monte-Carlo simulation and machine learning[M]. Springer Science & Business Media, 2013.
7. He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
8. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.