Frontiers in Signal Processing, Vol. 4, No. 1, January 2020
https://dx.doi.org/10.22606/fsp.2020.41006 

# Face Recognition Based on MTCNN and Convolutional Neural Network

Hongchang Ku, Wei Dong*

Southwest Minzu University, Key Laboratory of Electronic and Information Engineering, State Ethnic Affairs Commission, Chengdu, 610041, China
Email: 1626152817@qq.com

**Abstract.** MTCNN is a face detection method based on deep learning, which is more robust to light, angle and facial expression changes in natural environment, and has better face detection effect. At the same time, the memory consumption is small, and real-time face detection can be realized. Therefore, a method based on MTCNN and improved convolution neural network is proposed in this paper. Firstly, MTCNN is used to detect and align faces. Then, the output image is used as the input data of the improved convolution network, and multi-level convolution training is carried out. Finally, the accuracy of the model is tested.

**Keywords:** MTCNN, face detection and alignment, convolutional neural network, face recognition.

## 1 Introduction

With the rapid development of technology, face recognition is more convenient than other human body recognition systems such as fingerprints, irises, and DNA. It does not require compulsory participation and can solve problems without affecting people's normal life. It has the advantages of low cost, high user acceptance and high reliability, and has broad application prospects in identification, security monitoring, human-computer interaction and other fields. Traditional face recognition process includes four stages: face detection, face alignment, feature extraction and face classification. The most important stage is feature extraction, which directly affects the accuracy of recognition. At present, in the restricted environment, the traditional neural network method has better results in face recognition, but in the unrestricted environment, due to the complexity of the face image leading to large intra-class changes, as well as the inter-class changes caused by the external light and background, the traditional neural network face recognition method often fails to achieve the desired results.

Therefore, a method based on MTCNN and deep convolution neural network is proposed in this paper. On the input raw face image data, face detection and face alignment are performed using MTCNN. Then, the deep convolution neural network is used to extract face features, and face recognition is carried out. Finally, the LFW database and CASIA-WebFace data set are used to realize face recognition in the depth learning framework TensorFlow.

## 2 Basic Principles

### 2.1 Face Detection

MTCNN is a method of face detection and alignment based on deep convolution neural network[1], [2], [5], [10] that is to say, this method can accomplish the task of face detection and alignment at the same time. Compared with the traditional method, MTCNN has better performance, can accurately locate the face, and the speed is also faster, in addition, MTCNN can also detect in real time.

MTCNN consists of three neural network cascades, namely P-Net, R-Net, and O-Net. In order to achieve face recognition on a unified scale, the original image should be scaled to different scales to form an image pyramid before using these networks.

The first network P-Net is a full convolutional network used to generate candidate window and border regression vectors. Bounding box regression is used to correct candidate boxes, and then non-maxima are

 

used to suppress these combined overlapping candidate boxes. The structure of P-Net network is shown in Figure 1.

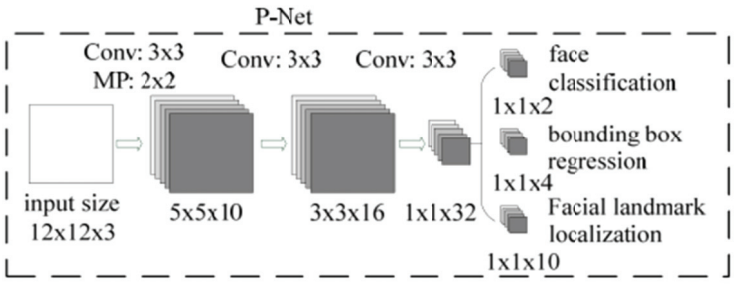


**Figure 1.** P-Net network structure

The results of P-Net are relatively rough, so further tuning is further done using R-Net. The R-Net structure diagram is shown in Figure 2. This structure is very similar to P-Net. It will enter into R-Net through the candidate window of P-Net, reject most of the false windows, and continue to use bounding box regression and NMS merging.

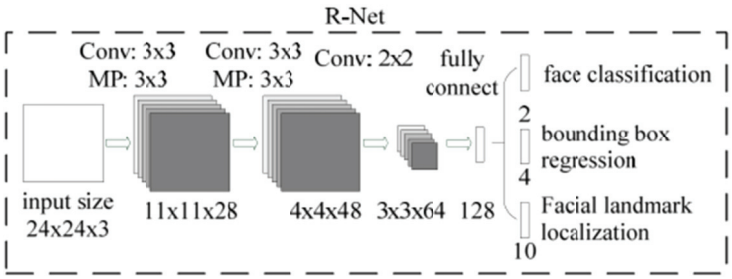


**Figure 2.** R-Net network structure

Finally, O-Net is used to output the final face frame and feature point positions. Similar to the first two steps, the difference is to generate 5 feature point positions. The O-Net network structure is shown in Figure 3.

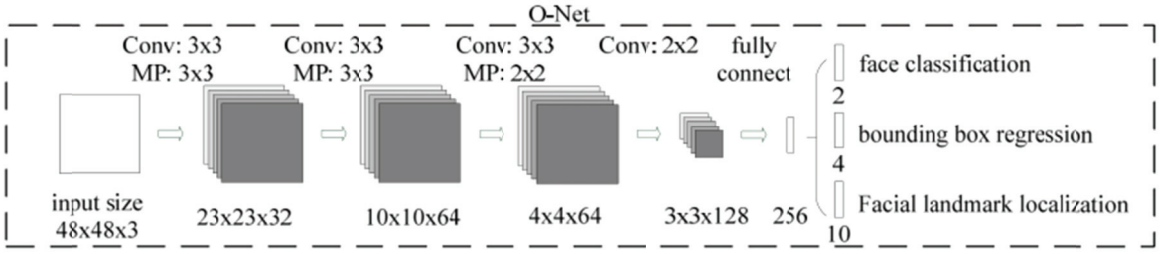


**Figure 3.** O-Net network structure

Each network in MTCNN has three parts of output, so the loss is also composed of three parts. For the face detection part, directly use the cross entropy loss function:

$$L_i^{det} = -\left(y_i^{det}\log(p_i) + \left(1 - y_i^{det}\right)(1 - \log(p_i))\right) \quad (1)$$

In the formula, $p_i$ represents the probability of inputting a face, and $y_i^{det}$ det represents a true label.

For the box regression and the five feature point decisions are all regression problems, so use the common Euclidean distance to find the loss. Boundary box regression loss function:

$$L_i^{det} = ||\hat{y}_i^{box} - y_i^{box}||_2^2 \quad (2)$$

 

where $\hat{y}_i^{box}$ is predicted by the network, and $y_i^{box}$ is the actual real background coordinate.

Key point decision loss function:

$$L_i^{landmark} = ||\hat{y}_i^{landmark} - y_i^{landmark}||_2^2 \quad (3)$$

where $\hat{y}_i^{landmark}$ is predicted by the network, and $y_i^{landmark}$ is the actual real key point coordinate. Finally, the three losses are multiplied by their own weights and then added together to form the final total loss.

### 2.2 Convolutional Neural Networks

Convolutional neural networks are a kind of feedforward neural networks with convolutional computation and deep structure. They are one of the representative algorithms of deep learning. Convolutional neural network (CNN) extracts high-level semantic information from raw data input layer by layer through stacking a series of operations such as convolution, convergence and non-linear activation function11.

In convolutional neural networks, the role of the convolutional layer is to train fewer parameters to extract feature information for the input data. The biggest advantage of the convolutional layer compared with the full connection is that the network is locally connected, and the amount of parameters that need to be trained is small, which is conducive to constructing a deeper and larger network structure to solve more complicated problems. The role of the pooling layer is to reduce the size of the feature map. In order to speed up the network training and reduce the amount of computational data, the convolutional neural network uses a pooling layer behind the convolutional layer to reduce the amount of data, the pooling operation can not only make the feature dimension extracted by the convolution layer smaller, reduce the amount of computing data, but also reduce the degree of over-fitting of the network to some extent and improve network performance. The function of the fully connected layer is to map the feature map of a two-dimensional image onto a one-dimensional feature vector. Through the full connection, the feature map of any dimension can be mapped into the vector of the specified dimension[12], [13], [14].

## 3 The Algorithm in This Paper

Because the traditional method can detect the face with good frontal/vertical/light, but it cannot detect the face with bad side/skew/light, so this method is not suitable for field application. The MTCNN algorithm is more robust to light, angle and facial expression changes in the natural environment, and the face detection effect is better; At the same time, the memory consumption is not large, real-time face detection can be realized. So this paper designs a method based on MTCNN and improved VGGNet for face recognition.

This article mainly uses the improved VGGNet network structure15. VGGNet is in principle no different from ordinary CNN. Its main feature is the local receptive field of small, all of them use 3×3 filters, and the network structure is deep. Its input is 224×224 RGB image, there are 5 maximum pooling layers, three fully connected layers, one softmax layer. All hidden layer activation functions use the nonlinear activation function ReLU. After the first and second fully connected layers, dropout technology is also used to prevent network overfitting. And the structure is very simple, the entire network uses the same size convolution kernel size (3×3) and maximum pool size (2×2), making the model's feature extraction ability stronger, and the number of parameters is less. However, it still consumes a lot of computing resources, and there are still many parameters used, resulting in more memory usage. Most of the parameters are from the first fully connected layer.

So this paper proposes an improved deep convolutional neural network structure. As shown in Figure 4, the last largest pooling layer is modified to be the mean pooling layer. The kernel size of the mean sampling layer is 7×7. This can effectively reduce network parameters while keeping network feature extraction capabilities as much as possible. VGGNet uses SoftMax classifier to classify images, But the SoftMax classifier essentially does not require a distance between each type of vector representation. Moreover, the face images are similar, and it is easy to cause recognition errors, which ultimately leads to poor performance of the trained face recognition model. Therefore, the classification function can be enhanced by improving the loss function16. In order to get better classification results, we should maximize the distance between classes and reduce the distance within the class. So this article uses the loss function combined with SoftMax Loss and Center Loss. Separate the different categories with SoftMax Loss, then use Center Loss to increase the distance between classes and reduce the distance

 

within the class. Center Loss is based on the SoftMax Loss classification and maintains a class center for each category. During training, gradually reduce the distance of class members from the class center and increase the distance between other class members and the center.

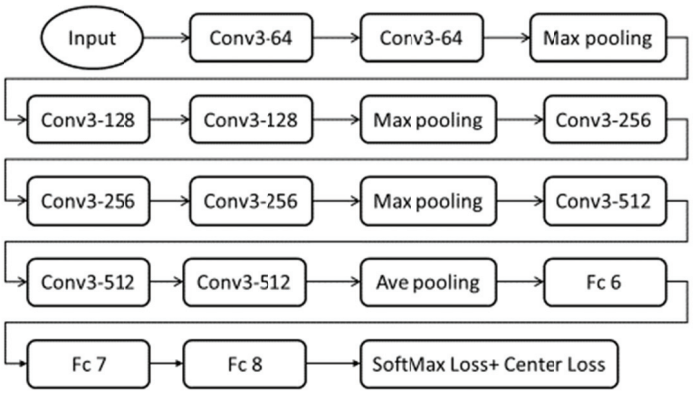


**Figure 4.** Improved VGG network structure

The Center Loss loss function is:

$$L_i = \frac{1}{2}||f(x_i) - c_{y_i}||_2^2 \tag{4}$$

where $x_i$ represents the input face image, $y_i$ represents the category of the face, $f(x_i)$ represents the feature corresponding to the face image, and $c_{y_i}$ represents a category center defined by each category.

The center of multiple images is to add their values together:

$$L_{center} = \sum_{i}^{m} L_i \tag{5}$$

The mixing loss function is:

$$L_i = L_{softmax} + \lambda L_{center} \tag{6}$$

where: $\lambda$ represents the weight of Center Loss.

## 4 Experiment and Analysis

### 4.1 Experimental Data Set and Processing

This experiment is to train the improved VGGNet network model on tensorflow. The CASIA-WebFace database is divided into training set and verification set in a ratio of 4:1. CASIA-WebFace database, which contains 10000 people, a total of 500000 face pictures. Use the training set to train the model, and use the verification set to adjust the super parameters. After obtaining the optimal model on the verification set, the super parameters of the model are used to retrain the final model. Finally, LFW is used to evaluate the final performance. LFW data set contains 5749 people, with 13233 face images, each of which is 250x250 in size.

Pre-processing is required before training and testing. All face images are detected using the MTCNN algorithm, and the five key points are used to align the face images, and finally cut and uniformly scaled to 160×160 pixel RGB images.

### 4.2 Experimental Results and Analysis

In this experiment, a large number of face data in CASIA-WebFace database are used to train the improved VGGNet deep convolution neural network model, and then the verification set is used to adjust the model super parameters repeatedly, and the accuracy of the optimal model in LFW data set test is 98.35%.

 

Finally, the face recognition rate of this method is compared with that of other deep learning algorithms. As shown in Table 1, under LFW data set test, the accuracy of DeepFace model is 97.35%, and that of FaceNet model is 99.63%, the accuracy of DeepID2+ model fused with 25 small networks is 99.47%. In terms of test accuracy, this model is much lower than FaceNet model and DeepID2+ model, and only about 1.18% higher than DeepFace model.

The DeepFace model adopts 3D alignment. When the model is used, the features and classifiers are bound, and the class size of this classifier is bound to the size of the input data class. So for different input data, DeepFace needs different training to ensure accuracy. This model is based on the MTCNN network structure to align the image, and it will not affect the accuracy in different input data. Both this model and FaceNet model use MTCNN for face alignment. The difference is that this paper uses the loss function of SoftMax Loss and Center Loss, which can enhance the distance between classes, reduce the distance within classes, and improve the accuracy of the model. The use of FaceNet model is a triple loss function, which is easy to be dominated by bad data, resulting in poor model.

DeepID2+ model also uses CASIA-WebFace database to divide training set and verification set in 4:1 ratio, and finally uses LFW as the standard face authentication test library to evaluate the final performance. Different DeepID2+ models use SDM method to detect 21 detection points before training. Then, according to these detection points and the factors such as location, scale, channel and horizontal flip, each face is transformed into 400 patches, and 400 patches are trained to 400 160 dimensional vectors using 200 convolution networks. Because the vector dimension generated above is too high, DeepID2+ model uses forward backward greedy algorithm to select the optimal 25 patches, and then generates 25x512 dimension vector, which is still too large, so PCA is used for dimension reduction. At the same time, seven of the best 25 patches are selected in the training process, and the trained seven Bayesian classifiers are fused by SVM to generate a classifier and get the final DeepID2+ model, so DeepID2+ model will use a lot of time and memory in the test. However, the network structure of this model is relatively simple, and to a large extent, it has sparse features, so in terms of time consumption, this model is much lower than DeepID2+, the use time of DeepID2+ is generally more than 35ms, and the use time of this model is usually about 6ms.

**Table 1.** Comparison with other deep learning algorithms

| method | dimension | Number of networks | Accuracy/% | Speed/ms |
|---|---|---|---|---|
| DeepFace | 4096 | 3 | 97.35 | 18 |
| FaceNet | 512 | 1 | 99.63 | 30 |
| DeepID2+ | 512 | 25 | 99.47 | 35 |
| Method of this paper | 4096 | 1 | 98.53 | 6 |

## 5 Conclusion

This work uses a combination of the MTCNN algorithm and the improved VGGNet deep convolutional neural network. The pre-training of the network is carried out by using a large amount of face image data, and the initial weight of the network is adjusted, and then the image of the database is used for feature extraction. This experiment automatically learns the effective features from a large number of face image data. After multi-layer learning, the features that can better represent the face are extracted. The experimental results show that the method has a good recognition result.

**Acknowledgements.** The work of this paper is supported by the Southwest Minzu University Graduate Innovative Research Project (Master Program CX2018SZ94). A special acknowledgement should give to Southwest Minzu University for its experimental conditions and technical support.

## References

1. Zhang K, Zhang Z, Li Z, et al. Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks[J]. IEEE Signal Processing Letters, 2016, 23(10):1499-1503.

 

2. Xiang J, Zhu G. [IEEE 2017 4th International Conference on Information Science and Control Engineering (ICISCE) - Changsha (2017.7.21-2017.7.23)] 2017 4th International Conference on Information Science and Control Engineering (ICISCE) - Joint Face Detection and Facial Expression Recognition with MTCNN[C]// International Conference on Information Science & Control Engineering. IEEE Computer Society, 2017:424-427.
3. Barkan O, Weill J, Wolf L, et al. Fast High Dimensional Vector Multiplication Face Recognition[C]// IEEE International Conference on Computer Vision. 2014.
4. Bengio Y. Learning Deep Architectures for AI[J]. Foundations & Trends® in Machine Learning, 2009, 2(1):1-127.
5. Qian Z, Ge S S, Mao Y, et al. Learning Saliency Features for Face Detection and Recognition Using Multi-task Network[J]. International Journal of Social Robotics, 2016, 8(5):1-12.
6. Cao X, Wipf D, Fang W, et al. A Practical Transfer Learning Algorithm for Face Verification[C]// IEEE International Conference on Computer Vision. 2014.
7. Chen D, Cao X, Wang L, et al. Bayesian Face Revisited: A Joint Formulation[C]// European Conference on Computer Vision. Springer, Berlin, Heidelberg, 2012.
8. Chen D, Cao X, Wen F, et al. Blessing of Dimensionality: High-Dimensional Feature and Its Efficient Compression for Face Verification[C]// Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. IEEE, 2013.
9. Chopra S, Hadsell R, Lecun Y. Learning a Similarity Metric Discriminatively, with Application to Face Verification[C]// IEEE Computer Society Conference on Computer Vision & Pattern Recognition. 2005.
10. Cui Z, Li W, Xu D, et al. Fusing Robust Face Region Descriptors via Multiple Metric Learning for Face Recognition in the Wild[C]// Computer Vision & Pattern Recognition. IEEE, 2013.
11. Taigman Y, Yang M, Ranzato M, et al. DeepFace: Closing the Gap to Human-Level Performance in Face Verification[C]// Conference on Computer Vision and Pattern Recognition (CVPR). IEEE Computer Society, 2014.
12. Lawrence S, Giles C L, Tsoi A C, et al. Face recognition: a convolutional neural-network approach[J]. IEEE Transactions on Neural Networks, 1997, 8(1):98-113.
13. Gu J, Wang Z, Kuen J, et al. Recent Advances in Convolutional Neural Networks[J]. Computer Science, 2015.
14. Schmidhuber J. Deep Learning in neural networks: An overview.[J]. Neural Netw, 2015, 61:85-117.
15. Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. Computer Science, 2014.
16. Schroff F, Kalenichenko D, Philbin J. FaceNet: A unified embedding for face recognition and clustering[J]. 2015.

 