Facial Landmark Detection based on Cascade Neural Network

Xuzhan Chen¹,²,*, Luxi Huang³,⁴, Yingming Chen³,⁴
¹Xidian University, Xi’an 710126;
²The 20th Research Institute of China Electronics Technology Group Corporation, Xi’an 710068
³National Time Service Center, Xi’an 710600
⁴University of Chinese Academy of Sciences, Beijing 100049
*Corresponding Author: Xuzhan Chen

Abstract: A three-stage cascade convolutional neural network is proposed in this paper to learn the facial features. In order to improve the robustness and accuracy of facial landmark detection, several parallel sub-networks are involved in each stage of network. The convolutional neural network performs global advanced feature extraction for the entire facial region in the preceding stage, so as to obtain more accurate positioning. Since the first-stage input covers the whole image, the global facial information can be effectively used to weaken or even avoid the detection errors caused by occlusion, large posture changes, extreme illumination and etc. During facial detection, the first-stage network will make initial prediction of facial landmarks, and then get the facial area narrowed down and accurate according to the prediction results; the second-stage network will take the narrowed-down facial area above as the input, the three parallel convolution networks are used to classify the facial landmarks for quadratic estimates, so as to improve the accuracy of prediction results based on the results of first-stage network. The input of the third-stage network is limited to the small neighborhood of the second-stage prediction results, which can make the prediction results more accurate. The experimental data shows that our method has certain advantages in detection accuracy and reliability.

1. Algorithm review

There are two basic approaches for facial detection: 1. obtain a rectangular candidate box of multiple facial regions through a certain search window, and then make further screening and fine-tuning of the target position[1, 2, 3]. In short, this method can be considered as a testing process “from large to small”. 2. Directly search for the facial feature points, such as the positions of eyes and other facial features, and then add a box according to the positions of facial features, so as to obtain the estimated target area, which is a processing process “from small to large”[4]. Specifically, the first approach is to sets a classifier for each facial landmark and make decisions based on local areas. The multiple candidate areas will seem like facial feature points or the suitable candidate areas cannot be found because the local features may be blurry or damaged. In contrast, the second method, which directly predicts the location (or shape parameter) of facial landmarks, is more effective because it is not needed to scan the sample images. In consideration of this, the method of direct prediction of facial features is employed in our work.
Obviously, good initialization is critical to iteratively update the positions of facial landmarks. In general, the positions of facial features are initialized by the “average face” sampled from training set. Nevertheless, due to the differences among the samples, the facial features predicted by the average face may be far away from the real positions, and the results of training iterative update may not be the global minimum. In addition, many methods are unable to accurately predict the facial features due to that the features extracted are not clear, so the prior information plays a vital role. The processing complexity will increase exponentially with the increase of image size, which has put forward higher requirements for regression or classification model.

In our work, a three-stage convolutional neural network was used to detect the facial features. In the cascade convolutional neural network used in this paper, the first-stage network took a whole image as the input to initially predict the facial landmarks, and narrow down the range of face simultaneously, thus laying a foundation for further accurate estimation of the later stage. The advantage of using a whole image as the input lied in that it could effectively avoid the local minimum problem of some methods, so as to make full use of the texture information, extract the global advanced features and excavate the deeper structure of higher level. Even in the case of local blur or damage in the test sample, the facial landmarks could be effectively predicted.

The downscoping image output in the first stage was used as the input of the second-stage network to carry on the second estimation of facial landmarks. Three parallel convolutional networks were used to estimate the five key points of the left and right eyes, the nose, and the left and right corners of mouth respectively, and the positions of the estimated key points and the corresponding positions of the first-level network were averaged to further improve the accuracy of prediction results.

The third-stage network took the output position of the second-stage network as a prior condition, and made further estimation and prediction. Since the network task load of the later stage was relatively small compared with that of the former stage, it was only responsible for fine tuning in a small range around the output of the second stage, so the number of network layers was also less than that of the preceding stage. In conclusion, this paper introduced the cascade neural network and adopted the parallel and serial network architecture, aiming to improve the accuracy and reliability of the estimation.

2. Structure of cascade convolutional neural network
In the following paragraphs, the methods employed in our work are discussed from two aspects: macro cascade structure, and single network structure.

2.1 Macro cascade structure
The network detection subjects in this paper included the center of left and right eyes, nose tip, and mouth corners. As shown in fig. 1, in the first stage, a deep convolutional network was used to make regional prediction of feature points, the input scope for the second-stage network was narrowed down according to the prediction results of the first-stage network, and three parallel convolutional neural networks were used to make re-prediction. L21 predicted the center of the left eye, the center of the right eye, with the prediction region covering the upper 80% area of the output image of first-stage network. L22 only predicted the position of nose tip, covering the middle 80% area in the output image of the first-stage network. L23 predicted the positions of the left and right mouth corners, which covered the lower 80% area of the output image of the first-stage network. The output of the second-stage network was the mean value of the predicted results of the three sub-networks and the first-stage network. The neighborhood centered on the predicted location of the second-stage served as the input of the third stage to limit the increment of the third stage compared with the output of the second stage. This was because the purpose of introducing the later stage was to improve the positioning accuracy on the basis of the former stage, but the range of the range of post adjustment needed to be limited due to the unreliable local information.
Specifically, with regard to the regression problem among various neural networks, the facial landmarks relative to the positions of input image might change in a large range due to the diversity of facial posture changes and the uncertainty of image acquisition locations. Therefore, the input image region corresponding to the first stage needed to be as large as possible to cover all possible positions of feature points. However, every coin has two sides. A large range of unrelated region would obviously cause some interference to the prediction results, thereby causing the prediction location to be not accurate enough. Hence, on the basis of the initial prediction of facial landmarks of face image in the first-stage network, the face images were cropped based on the prediction results to remove irrelevant information in the input samples, and the cropped face images were used as the input of the post-stage network, so as to improve the prediction accuracy of the post-level network. In other words, the boundary box obtained from the output of the first-stage network was used as a priori condition for the second-stage network, so as to further estimate the position of facial features. Then, we limited the adjustment of the third level network to key points within a small region based on the prior knowledge of the second-stage output. It was just because of this limitation that, the estimated results would not be disturbed by other regions. Nevertheless, the lack of global information would result in the lower reliability of predictions. In order to reduce the propagation and expansion of errors, the third-stage network was set to be a shallow structure.

2.2 Single network structure

The single-network structure in the cascade neural network is introduced below with sub-network L11 as an example. As shown in fig. 2, there are four convolution layers, three pooling layers, and two fully-connected layers[5]. In addition, tanh is used as the activation function in this paper.

Fig. 2 the network structure of L1

In terms of weight sharing, it is well known that the advantages of weight sharing lie in that certain features may appear anywhere in the image, and it can prevent gradient dispersion to some extent. However, it may not be the best solution to directly apply weight sharing to facial landmark detection. As an intuitive example, the eyes and mouth show quite different deeper features in spite of the similar marginal structures. Therefore, the method of local weight sharing was employed in this paper.
3. Implementation details of cascade neural network

The corresponding input images in the network of this paper are gray scale images. The convolutional layers are set to be \( \text{Con}(s, n, \alpha, \beta) \), in which, \( s \) is the size of convolution kernel, \( n \) is the number of mappings in the convolutional layer, and \( \alpha \) and \( \beta \) indicate the weighted shared parameters. The feature images are divided into \( \alpha/\beta \) image blocks, with each image block sharing the weight, namely, the local weight sharing. By default, the stepping quantity of the convolution kernel is 1 pixel in both directions. If there are a total of \( m \) feature images, and the size of each feature image is \( h \times w \), then the operation of convolutional layer can be expressed as

\[
y_{i,j}^{(t)} = \tanh \left( \sum_{r=0}^{m-1} \sum_{k=0}^{n-1} x_{i+r,k,j+l}^{(r)} w_{i,j}^{(r)} + b^{(r)} \right)
\]

Where, \( i = \Delta h \cdot u, ..., \Delta h \cdot u + \Delta h - 1 \), \( j = \Delta w \cdot v, ..., \Delta w \cdot v + \Delta w - 1 \), \( t = 0, ..., n-1 \), \( \Delta h = \frac{h-s+1}{\alpha} \), \( \Delta w = \frac{w-s+1}{\beta} \), \( u = 0, ..., \alpha - 1 \), \( v = 0, ..., \beta - 1 \).

X and y represent the input and output of the current layer. To be more specific, W is the weight matrix, and b is the bias. m feature images in the previous layer are connected with \( m \times s \times s \) convolution kernels. After the obtained characteristic graph is superimposed with the same bias value, \( n \) feature images are formed in the convolution layer through the nonlinear processing by activation function \( \tanh \). The weight matrix and bias are not the same even within the same feature image.

\( P(s) \) is used to represent the pooling operation, of which, \( s \) is the side length of square pooling area. Maximum sampling is used as the pooling mode in this paper, with the pooled regions not overlapping. The pooling results include a gain coefficient \( g \) and an offset \( b \), which are non-linearized later by \( \tanh \). The weight sharing mode of gain and offset parameters is in a similar way to the previous layer.

\( P(s) \) can be expressed as

\[
y_{i,j}^{(t)} = \tanh \left( g^{(u,v)} \cdot \max_{0 \leq x_i, j \leq s} \{ x_{i,j}^{(t)} \} + b^{(u,v)} \right)
\]

The fully-connected layer \( F(n) \) could be expressed to be

\[
y_j = \tanh \left( \sum_{i=0}^{m-1} x_i \cdot w_{i,j} + b_j \right), j = 0, ..., n-1
\]

where \( n \) and \( m \) are the number of neurons in the current layer and the previous layer.

During training, a large number of images with different postures and facial positions were input to enhance the robustness of network to postures. In the third stage of cascade neural network, the center of training image block was selected as a neighborhood of facial landmark predicted in the second stage. The maximum offsets in the horizontal and vertical directions of the third stage were 0.05 times of the horizontal and vertical dimensions of the image. The goal of third-stage network was to make more subtle adjustments to the second-stage network to further improve the accuracy of predicted position. The learnable network parameters include weight \( w \), gain \( g \) and offset \( b \), which were initialized by small random numbers and learned by stochastic gradient descent. \( L \) method was used to estimate the learning rate of neurons, and train the neural network until convergence.

4. Experimental results of facial detection based on cascade neural network

In order to verify the face detection algorithm in this paper, the training set and verification set collected by us were used to study the different design of network and cascade structure. Whereafter, it was compared with Haar & AdaBoost face detection algorithm in the public test set without changing the training set. The training set and verification set used in the experiment of this paper had no overlap with the two common test sets.
4.1 Verification experiments on network and cascade structure

We created a dataset including 500 face images. In the experiment, the tagged sample images of keras database were selected for training, and the self-built data set was used for verification. The average detection error and estimated failure rate of each facial landmark was used to measure the algorithm performance. The two parameters could reflect the accuracy and reliability of the algorithm respectively. The detection error was measured as:

\[ err = \sqrt{(x - x')^2 + (y - y')^2} / l, \]

where \((x, y)\) is the real position of landmark, \((x', y')\) is the detection position of model output, and \(l\) is the border size of the image output of the face detector. If the estimation error of a landmark is greater than 5% [6], it is considered a detection failure. Some sample images and detection results from the validation set are shown in fig. 3.

![Fig. 3 some detection results from our validation set](image)

(a) Average error
The figure above shows the detection performance of the three-stage network. On the basis of the first stage, the second stage estimated the key points of three parallel deep networks, thus making detection failure rate greatly reduced. At the third stage, fine-adjustment was carried out on the basis of the second-stage prediction through the shallow network to further improve the accuracy of feature points, which greatly improved the average error rate and detection failure rate. It could be seen from the data above that, the use of multi-stage cascade convolutional neural network could effectively reduce the detection error.

4.2 Comparison with other methods
The experimental comparison was carried out on BioID face database[7]. BioID face database was a set of facial images collected under laboratory conditions. In order to compare with Haar & AdaBoost method[8], we used the model trained on the data set specified above. The results are shown below.

The BioID face database included 1,521 images from 23 volunteers. All images in the face database were full-face, with modest variations in illumination and facial expression only. It could be seen from Fig. 5 that the proposed algorithm was superior to Haar & AdaBoost method in both average error and detection failure rate.

The experimental results above showed that, a convolutional network cascade method was proposed in this paper for effective facial detection. The first-stage network initially estimated the image and made the prediction area of latter stage reduced. The second-stage deep convolutional
network provided the second-stage estimation with high robustness, while the third-stage shallow convolutional network could well adjust the second-stage prediction to improve the estimation accuracy again.

5. Conclusions

A type of three-stage convolutional network was employed in this paper to detect the facial landmarks. In order to improve the prediction accuracy of the later stage, the regions irrelevant to human face were removed while each feature point was initially predicted by the first-stage network. The cropped images obtained from the first-stage network served as the input of the second-stage network to further make the estimated positions of facial landmarks refined. The third-stage network would make fine-adjustment on the basis of the second-stage output, so as to obtain more accurate positioning results. Subsequently, the measured data showed that the average error and detection failure rate of self-built data set of the facial detection model in this paper was less than 1.4% and 1.7% respectively, and the average error and detection failure rate of facial detection model in the BioID face database was less than 1.2% and 1.5% respectively. Finally, through the comparison with Haar & AdaBoost method, it could be seen that the algorithm in this paper was obviously superior in both average error and detection failure rate.

References

[1] P. Sermanet, D.Eigen, X.zhang.et al. OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks[J],arXiv preprint arXiv:1312.6229.2013
[2] Girshick, R. , Donahue, J. , Darrell, T. , & Malik, J. . (2014). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE Computer Society.
[3] Uijlings, J. R. R. , & K. E. A. van de Sande…. (2013). Selective search for object recognition. International Journal of Computer Vision, 104(2), 154-171.
[4] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. In Proc. CVPR, 2012.
[5] Lopes, A. T . , Aguiar , E. D. , Souza, A. F. D . , & Oliveira-Santos, T . . (2016). Facial expression recognition with convolutional neural networks: coping with few data and the training sample order. Pattern Recognition, 61, 610-628.
[6] Sun, Y . , Wang, X. . , & Tang, X. . . (2013). Deep Convolutional Network Cascade for Facial Point Detection. Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. IEEE.
[7] Jesorsky, O., Kirchberg, K. J., & Frischholz, R. (2001). Robust Face Detection Using the Hausdorff Distance. International Conference on Audio-& Video-based Biometric Person Authentication.
[8] Wang, Y . , Ai, H. , Bo, W. , & Chang, H. (2004). Real Time Facial Expression Recognition with Adaboost. International Conference on Pattern Recognition.