2019 2nd International Conference on Safety Produce Informatization (IICSPI)

Face Key Point Location Method based on Parallel Convolutional Neural Network

Zhou Pu, Kai Wang, Kai Yan
Guangdong University of Technology, Guangzhou 510000, China
1920327855@qq.com

***Abstract*—Based on convolutional neural network and face detection algorithm, this paper proposes a training sample expansion strategy, and a parallel convolutional network face detection algorithm for face features, occlusion and illumination detection, combined with Relu activation function and Dropout random regularization strategy. Network training not only speeds up the convergence of the network, but also improves the generalization ability. On this basis, the software based on face detection and feature point location is designed to realize the automatic loading of images and the face recognition function, to achieve accurate positioning of the face points, and to locate experiments on the LWF face database. The results show that the method is greatly improved in accuracy and reliability, and it can achieve robust and accurate estimation of key points.**

Keywords—Parallel convolutional neural network, Face key point location, Deep learning, Face detection

I. INTRODUCTION

Face detection is an indispensable part of face recognition. The effect of face detection directly affects the recognition effect. Face detection refers to the process of determining the position, posture, and size of all faces (if any) in a video or still image. After decades of research, face detection has become an independent research topic, which has been widely recognized by scholars at home and abroad. Today, with the rapid development of technology, the application of face detection is becoming more and more extensive. For example, Ma Yun, the president of Alibaba, put forward the idea of brushing his face, which is expected to become a reality in the near future [1]. The key technologies of face detection will be further integrated into the fields of security, identity authentication, autofocus, human-machine interaction and image search to benefit smart life.

Face feature point detection technology plays an important role in face feature extraction [2]. Although the research of face detection technology has developed rapidly in recent years, even accurate face positioning results are affected by factors such as posture, angle and illumination, thus affecting the subsequent recognition effect. Accurate facial key point positioning results can be used for attitude correction and position calibration to enhance recognition. At the same time, face feature point calibration technology can also be applied to face animation synthesis, expression analysis, attitude analysis, fatigue judgment, 2D/3D modeling, video face tracking and face local information extraction. Face key point positioning is an indispensable pre-processing part in applications such as face recognition. Even if the image is rotated according to the detected binocular coordinates, the eyes are set to the horizontal plane, which can effectively improve the accuracy of face recognition. Complex preprocessing methods are also inseparable from accurate face key location algorithms.

Convolutional Neural Network (CNN) was originally applied in handwritten character recognition [3], and has been greatly developed in recent years and gradually applied to various fields. The convolutional neural network can extract the detailed structure information from the input image, and at the same time make the structural information have spatial invariance such as feature rotation, which is very suitable for the detection and recognition problem in the image. Based on this, a new facial feature point localization method based on parallel convolutional neural network is proposed. In the study of convolutional neural networks, the design of network structure is one of the most important problems. Different network structures have a great influence on the recognition results. In the traditional repetitive and down sampling network structure, the number of network layers is too large and the structure is complex. Therefore, a parallel convolutional neural network topology is designed and implemented.

II. CONVOLUTIONAL NEURAL NETWORK

Convolutional neural network is a new artificial neural network proposed by combining traditional artificial neural network and deep learning. It optimizes the traditional neural network structure by introducing weight sharing, local perception and pooling layer. Convolutional neural networks in convolutional neural networks are only connected to neurons in a small area of the upper layer, rather than to all neurons in the upper layer, and the weights of neurons on the same feature mapping surface It is shared. The application of weight sharing and local perception strategies greatly reduces the parameters that need to be learned, making the training of convolutional neural networks more efficient. The characteristics of the output of the pooling layer have invariances such as translation and rotation space, which makes the convolutional neural network have good robustness to the effects of translation and rotation. Through the alternate use of the convolutional layer and the pooled layer, the convolutional neural network completes the learning process from image local to global. The operation of the convolutional layer can be expressed as Equation (1).

$$X^{(l,k)} = f\left(\sum_{p=1}^{n_l-1}\left(W^{(l,k,p)} \otimes X^{(l-1,p)}\right) + b^{(l,k)}\right) \quad (1)$$

Where: $X^{(l,k)}$ represents the k-th feature map of the first layer output, n_l represents the number of layers of the l layer feature map, and $W^{(l,k,p)}$ represents the p-th feature map of the l-1 layer to the k-th feature map of the l layer The filter used when mapping the map. The generation of each set of feature maps of the l layer requires n_l-1 filters and an offset. Assuming that the size of the filter is $h \times w$, the number of parameters of the layer l convolutional layer is $n_{l-1} \times n_l \times h \times w + n_l$.

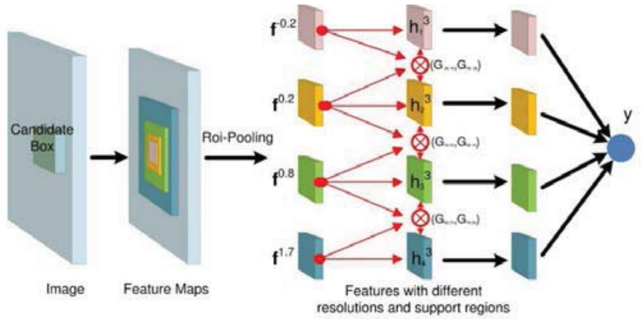

Fig. 1 Convolutional neural network schematic

Common pooling operations have maximum pooling, averaging pooling, etc. The convolutional neural network in this paper uses only the maximum pooling. After the pooling operation, the size of the feature map is reduced to the original 1/step according to the step size. The general form of maximum pooling can be expressed as Equation (2).

$$X^{(l+1,k)}(m,n) = \max_{0<a,b<s}\left\{X^{(l,k)}(m \bullet step + a,\ n \bullet step + b)\right\} \quad (2)$$

Where: $X^{(l+1,k)}(m,n)$ is the value at the coordinate (m,n) of the kth group map of the output of layer $l+1$; s is the size of the pooled window, and step is the step size when the pooled window is moved. In this paper, s and step are both set to 2. The convolutional neural network updates all connection weights and offsets between neurons for the purpose of backpropagation to minimize the error function. Considering the face key location task, the square sum loss function is used, which can be expressed as Equation (3).

$$E^N = \frac{1}{2N}\sum_{i=1}^{N}\left\|y_i - d_i\right\|_2^2 \quad (3)$$

Where: N is the number of nodes in the neural network output layer, y is the predicted value of the neural network, and d is the manually labeled value. In this paper, we use the random gradient descent [4] algorithm to update all the connection weights and offsets between neurons. The final loss function is expressed as Equation (4).

$$E = \frac{1}{m}\sum_{j=1}^{m}\left(\frac{1}{2N}\sum_{i=1}^{N}\left\|y_i^j - d_i^j\right\|_2^2\right) + \frac{1}{2}\eta\left\|W\right\|_2^2 \quad (4)$$

Where: m is the number of training samples used for each backpropagation, and W is the weight matrix for each layer in the network. The weight matrix W in the convolutional neural network is updated during backpropagation. The network is initialized with a random value matrix W_0 before network training begins, and the weight matrix W_{t+1} updated after t+1 iteration can be expressed as Equation (5).

$$W_{t+1} = W_t - \lambda \bullet \frac{\partial E}{\partial W_t} \quad (5)$$

III. Research on 3 Face Feature Point Calibration Algorithm

Facial features include eyes, nose, mouth, eyebrows and borders. These features can be used not only to analyze facial expressions, age, beauty index, but also as an important criterion for identifying the identity of a person's face, so accurate feature points the positioning result becomes a prerequisite for extracting facial features [5]. The mainstream feature point location algorithms include: 5-point, 25-point, 83-point, and so on. The basic flow of the feature point calibration algorithm includes: firstly, performing face detection on the input image to obtain an accurate face position; secondly, performing feature point detection according to the detail texture information in the face frame; finally, specifying a number of feature points Calibration is in the face area. At present, both Uche and Face++ provide a cloud technology experience platform for enthusiasts to learn.

This article uses a 5-point method. Among them, five points include: left eye pupil center point (LE), right eye pupil center point (RE), nose tip point flash, left mouth corner point (LM) and right mouth corner point (RM). Accurate feature point positioning results can be used to correct the angle and posture of the face, thereby improving the accuracy of face recognition. The excellent feature point calibration algorithm not only can obtain the feature point position efficiently and accurately, but also has certain robustness to the face being affected by expression, posture, rotation, occlusion and illumination. This paper chooses to use convolutional neural networks for the study of feature point calibration algorithms.

A. Convolutional neural network topology

The structure of the convolutional layer and the down sampling layer alternately arranging not only the local and specific features of the front layer can be merged into a more global and abstract structural feature expression, which is beneficial to classification, and also simplifies the network structure in the process of feature dimensionality reduction. To eliminate interference from redundant information. The end of the convolutional network is usually the fully connected and output layers. The fully connected layer rearranges the two-dimensional feature map into a one-dimensional feature vector in a certain way, and the output layer implements classification and function approximation. In the feature point calibration problem [6], since we want to obtain the coordinates of the five feature points, the number of nodes in the network output layer is 10, representing the horizontal and vertical coordinate values of the five output points. The specific network structure is shown in Figure 2.

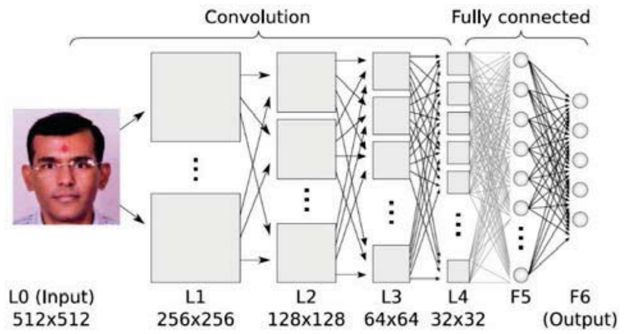

Fig. 2 Characteristic point calibration structure

In Fig. 2, it is assumed that the coordinates of the upper left corner and the lower right corner of the face frame detected by the face detection algorithm are (x_1, y_1) and (x_2, y_2), respectively, and the actual coordinates of the center point of the left eye pupil are (x, y), and the output of the output layer node is normalized. The value (a, b) is calculated as Equation (6).

$$\begin{cases} a = \dfrac{x\text{-}x_1}{x_2\text{-}x_1} \\ b = \dfrac{y - y_1}{y_2 - y_1} \end{cases} \tag{6}$$

B. Local weight sharing strategy

The local weight sharing strategy is to divide the input feature map into A and other regions. The shared weight kernel weight is shared within each region, which is equivalent to extracting a local texture feature. The schematic diagram of the local weight sharing strategy is shown in Figure 3.

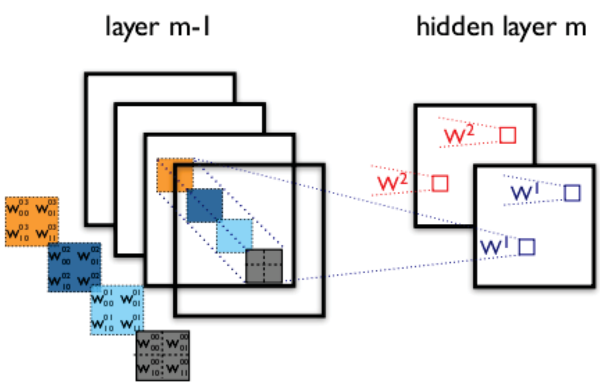

Fig. 3 Local weight sharing strategy

The local weight sharing strategy divides the original image into equal-sized regions, shares weights in each region, and extracts a texture feature of the region. Since the structural features inside each sub-area are relatively simple, the extracted features are more representative. Of course, the number of sub-areas can be selected according to the needs, and the number of sub-areas is too large, which increases the amount of calculation and leads to over-fitting [7]. Therefore, it is especially important to select the number of sub-areas suitable for detecting objects.

Suppose $I(h,w,m)$ represents the previous layer feature map parameter of the convolutional layer, the size is $h \times w$, and the number of feature maps is m. $C(s,n,p,q)$ stands for convolutional layer parameters, where s is the convolution kernel size, n is the number of convolutional layer feature maps, and the convolutional layer uses a local weight sharing strategy, which is divided into $p \times q$ equal-sized regions, using sigmoid Activate the function. Then the convolutional layer node output value is calculated as Equation (7).

$$y_{i,j}^{(t)} = sigmoild\left(\sum_{r=0}^{m-1} \sum_{k=0}^{s-1} \sum_{i=0}^{s-1} x_{i+k,j+1}^{(r)} \bullet w_{k,j}^{(r,u,v,t)} + b^{(u,v,t)} \right) \tag{7}$$

IV. Case Analysis

A total of 10,000 training sample sets were used in this section of the experiment, of which 4,150 were from the LFW dataset, 1000 were from the network dataset, and the remaining 4,850 were from the network interception image, and the face frame coordinates and feature point coordinates were obtained. The test set is divided into two subsets, a total of 1872 samples. Subset 1 has a total of 1,440 samples, all from the LFW dataset; a subset of 2, a total of 432 samples, all from the network dataset. The LFW dataset is an unconstrained face dataset in a complex background. Faces differ in terms of skin color, expression, angle, pose, and occlusion. The data set contains 20 face images of the tester in frontal shooting, different lighting conditions, and different expressions, as shown in Figure 4. The face positions in each image of the data set are respectively labeled to obtain the coordinates of the upper left corner and the lower right corner, and the accurate coordinate values of the five feature points are obtained, which are used to calculate the calibration error.

Fig.4 Face key point positioning effect diagram based on parallel convolutional neural network

The training process of convolutional neural networks requires a high degree of inherent parallelism, and there are a large number of floating-point data operations and large-scale matrix operations. A highly parallel-structured graphics processor can solve these problems well. In the same situation, the convolutional neural network training speed in GPU mode far exceeds the training speed in CPU mode. In the training process of CNN (F1), the iteration time of GPU mode is only 34.7 seconds, while in CPU mode, it takes 611.3 seconds. In GPU mode, the speed of Caffe training convolutional neural network is 17.6 times that of CPU. Caffe, an open source deep learning framework that supports GPU acceleration, can dramatically reduce the training time spent on convolutional neural networks.

TABLE I COMPARISON OF ERRORS IN DIFFERENT MODELS

method	Left eye error	Right eye error	Nose tip error	Left mouth angle error	Right mouth angle error	Total mean error
Single CNN	1.564	1.795	2.279	2.078	2.199	1.983
Parallel CNN	0.869	0.864	1.105	1.289	1.289	1.083

V. CONCLUSION

With the rapid development of digital image technology and deep learning technology, the use of deep learning algorithms for information acquisition in the field of digital image has become a research hotspot. As an effective biological feature, human face has been receiving much attention. This paper focuses on the use of convolutional neural network models in the field of deep learning to study face detection and feature point calibration methods, and proposes more efficient face detection and feature point calibration by extending training samples and building cascade structures algorithm. The proposed algorithm has good anti-interference ability for illumination, occlusion, attitude, expression and other interference. Applying the algorithm to face recognition program can improve the accuracy of face recognition.

REFERENCES

[1] Yan Yifan. Face key detection algorithm based on concatenated convolutional neural network. Vol. 2 (2015) No.12, p. 136-150.

[2] Yang Haiyan, Jiang Xinhua, Nie Zuoxian. Research on face key location method based on parallel convolutional neural network. Journal of Computer Applications, Vol.8(2015) No.32, p. 2517-2519.

[3] GU Wei, QIU Weidong. Feature Detection of Beard and Hat Based on Convolutional Neural Network. Information Technology, Vol. 9 (2017) No.13, p. 121-124.

[4] Guo Mingjin, Ni Jiajia, Chen Wei. Face Verification of Mixed Convolutional Neural Networks. Journal of Computer Systems, Vol. 2 (2018) No.27, p. 23-28.

[5] Tang Jie, Chen Renwen, Yu Xiaoqing. Implementation of Vehicle Fatigue Driving Detection System Based on Convolutional Neural Network. Foreign Electronic Measurement Technology, Vol. 1 (2018) No.5, p. 12-15.

[6] Mao Yi. Research on face recognition algorithm based on deep neural network. Vol.1 (2017) No.25, p. 33-36.

[7] GU Jia-ling, PENG Hong-jing. Growth Convolutional Neural Network and Its Application in Face Detection. Journal of System Simulation, Vol. 8 (2009) No.21, p. 2441-2445.