Living Face Verification via Multi-CNNs

Peiqin Li[*], Jianbin Xie, Wei Yan, Zhen Li, Gangyao Kuang

School of Electronic Science, National University of Defense Technology, Changsha, Hunan, 410073, P. R. China

ARTICLE INFO

ABSTRACT

In face verification applications, precision rate and identifying liveness are two key factors. Traditional methods usually recognize global faces and can not gain good enough results when the faces are captured from different ages, or there are some interference factors, such as facial shade, etc. Besides, the false face attack will pose a great security risk. To solve the above problems, this study examines how to achieve reliable living face verification based on Multi-CNNs (convolutional neural networks) and Bayes probability. Participants are required to make several expressions in random orders and contents, to ensure their liveness. Then, an effective component-based method is proposed for face recognition, and synthesized multiple CNNs can help reflecting intensities of different components. Eventually, the similarity between faces is calculated by integrating the results of each CNN, with the help of Bayes probability. Comparative experiments demonstrate that our algorithm outperforms traditional methods in face recognition accuracy. Moreover, our algorithm has unique preponderance in that it can verify the liveness of users, which can achieve higher security.

1. INTRODUCTION

With the rapid progress of face recognition technology, face authentication turns into an excellent way in security nowadays. In a traditional face authentication system, the camera captures face images of the people who need to be identified, then the features of the face are extracted and compared with enrolled ones. Finally the result is gained by calculating the similarity. The traditional system of simple face recognition has a shortage that simulating prosthetic targets can easily cheat it. In other words, if one person does not come in the front of the camera, instead his paper photograph, or a digital picture shown on a PAD is used by other people. They may potentially pass the identification. To avoid such problems, the authentication system requires that the image to be processed is captured from real faces, not other prostheses. Thus the living target identification has become one of the well-studied problems in computer vision [1].

Some effective face recognition algorithms have been studied. For face recognition, in [2, 3], an original face image is converted into a corresponding sketch image, after which recognition is conducted by sketches. Certain handcrafted descriptors, such as local binary pattern (LBP), scale invariant feature transform (SIFT), and histogram of oriented gradients (HOG) [4–6] has the effect of comparing faces. Various methods are applied to detect facial key points, as such methods used for recognition can accurately locate the face. However, such studies are employed for static face recognition, which may be easily attacked.

In order to verify that faces are captured from living persons, some methods have been generated. For example, 3D models are used to add stereo information of faces [7–9]. Although these methods may have reliable effects, they typically require additional and special imaging equipment.

Deep learning, a class of machine learning techniques mainly developed since 2006, can generate numerous stages of nonlinear information processing in hierarchical architectures, which are employed in feature learning and pattern classification [10]. Especially for face recognition, Convolutional Neural Network (CNN) is an excellent widely applied model [11–14]. In recent years, CNN is used by some famous companies and research teams for face detection or recognition, such as DeepID3 [15], NormFace [16], DeepFace [17], FaceNet [18], and so on. Generally speaking, deep learning methods turn a global face image as input data, and face analysis is accomplished internally. On one hand, this process shows the intelligent advantage of deep learning, on the other hand, feature extraction of a whole face increases the complexity of networks. Recognition accuracy may declines, especially when there are age differences between the enrollment and verification of human faces.

In this paper we propose a robust and useful algorithm, which takes advantage of CNN-Bayes probability jointed scheme for living person recognition. The rest of this paper is organized as follows: the framework of our algorithm introduced in Section 2, recognition of the face with constrained dynamic expression in Section 3, experimentation and comparison of the obtained results with competing methods in Section 4, and conclusions in Section 5.

[*]*Corresponding author. Email:* lipeiqin_nudt@163.com

2. FRAMEWORK OF DYNAMIC FACE RECOGNITION

We design a process for living identity verification. Firstly, participants are asked to make designated expressions. The expressions include closing left or right eyes, opening and closing the mouth, knitting and releasing the nose. Both the types and orders of expressions are random. Then, the faces with different expressions are recognized by our innovative algorithm. Concretely, we investigate the peculiarities of facial components as key targets to identify faces with different expressions. Based on such methods, only living persons can respond to the demands, and optimized methods can ensure the accuracy of identification. Meanwhile the whole algorithm is much easier to be achieved than traditional living identification. The framework is shown in Figure 1.

For face recognition, six kinds of components are chosen for the operation: left brow, right brow, left eye, right eye, nose, and mouth. They are detected and located with the help of facial landmarks. On faces of known categories, these components and entire faces are classified accordingly to train CNN models. The template face and unknown faces are handled separately, and the results are integrated to calculate the complete similarity of faces by Bayes probability.

The contributions of our study are summarized as follows:

- The dynamic faces with different expressions can efficiently ensure the target is living, thus the safety of identification system can be enhanced.

- Besides whole-face images, an effective component-based method is added for face recognition, and the peculiar features of components are distinct and stronger than those of single global face.
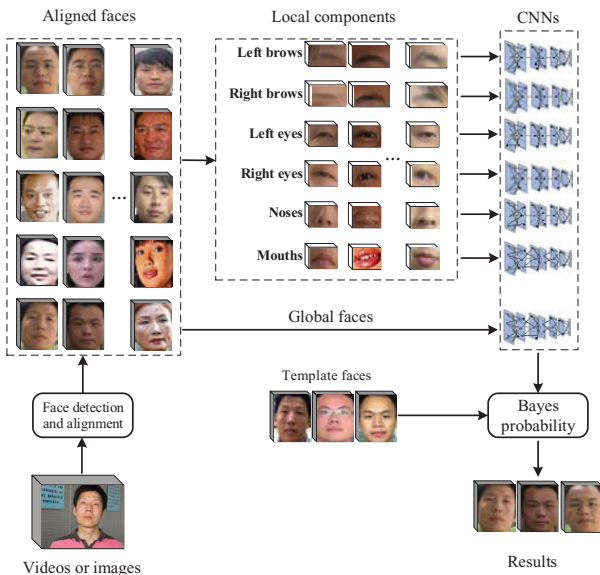
- Several ordered CNN models are used to extract features. This order can avoid limitations of normal CNN.

- Bayes probability is introduced to synthesize multiple CNNs. This method can effectively reflect the peculiar intensities of different components.

3. DYNAMICALLY FACE RECOGNITION

3.1. Facial Component Location

To strengthen the adaptability of global variations, pre-treatments such as adjustment and normalization of global brightness are needed. Then considering face usually has conformable local gray value, while other regions have diverse gray value, such as textures and graphic patterns on clothes, we should take advantage of the gray-similarity to detect possible face areas. Clustering analysis can achieve the pattern recognition based on the similarities to judge automatically [19]. Therefore, we make use of K-Means for clustering analysis, in order to reduce the object regions of face detection. The process is expounded as follows:

Algorithm 1：Clustering analysis based on K-Means

Input : Sample pixels x of image containing faces, the number of whole patterns k.

Output : The final clustering classes $\{S_1, S_2, \cdots, S_k\}$.

1: **Initialization**: Randomly choose centers of clustering as $C_1^{m1}, C_2^{m2}, C_3^{m3}, \cdots, C_k^{mk}$, where $m1, m2, m3, \cdots, mk$ separately means the number of iteration operation.

2: **while** $C_i^{r+1} \neq C_i^r, (i = 1, 2, 3, \cdots, k)$ **do**

3: 　**for** $i = 1 : k$

4: 　　**if** $d(x, C_i^r) = \min(d(x, C_j^r), j = 1, 2, 3, \cdots, k)$

5: 　　　$S_i^r \leftarrow x$

6: 　　　$C_i^{r+1} = \dfrac{1}{n_i^r} \sum_{x \in C_i^r} x, (i = 1, 2, 3, \cdots, k)$

7: 　　　n_i^r is the number of samples belonging

8: 　　　to the S_i^r .

9: 　　**end if**

10: 　**end for**

11: **end while**

12: Cluster samples x from I into k patterns.

After these treatments, the searching range of Active Shape Model (ASM) [20] can be reduced, and the whole process can achieve higher speed. The location of facial component is annotated on training image set. Supposing that there are *n* face images in the training set and each face has *m* landmark points, the shapes can be represented as vectors

$$X^i = \left(x_0^i, y_0^i, x_1^i, y_1^i, \cdots, x_{m-1}^i, y_{m-1}^i \right)^T, i = 0, 1, ..., n - 1 \quad (1)$$

Here x_j^i and y_j^i are the coordinates of the *j*th landmark in *i*th face image.

Figure 1 | Framework of our algorithm. Living faces are detected and aligned, then several useful local facial components can be located, and respective CNNs are trained to extract features of each local components and global faces, finally Bayes probability is used for calculating the similarity.

Aligning faces in set X, we can get the new set

$$\hat{X}^i = \left(\hat{x}_0^i, \hat{y}_0^i, \hat{x}_1^i, \hat{y}_1^i, \cdots, \hat{x}_{m-1}^i, \hat{y}_{m-1}^i\right)^T, i = 0, 1, ..., n-1 \quad (2)$$

And the average template of face is

$$\overline{X} = \frac{1}{n} \sum_{i=0}^{n-1} \hat{X}^i \quad (3)$$

The deviation between samples and average template can be calculated as

$$dX^i = \hat{X}^i - \overline{X} \quad (4)$$

And the covariance matrix is

$$S = \frac{\sum_{i=0}^{n-1} \left(dX^i\right)\left(dX^i\right)^T}{n} \quad (5)$$

We assume the nonzero eigenvalue vector and eigenvector are λ_i and p_i, then a face shape can be represented as

$$X = \overline{X} + P_t B_t \quad (6)$$

Here $P_t = (p_1, p_2, ..., p_t)$ is a matrix of the first t eigenvectors of the covariance matrix, and $B_t = (b_1, b_2, ..., b_t)$ is a vector that indicates the variation from X to $\overline{X}$ in the direction of $p_1, p_2, \cdots p_i$. Across the training set, the variance of the ith parameter vector p_i is given by the corresponding eigenvalue vector λ_i. To ensure the generated shape is similar to those in the training set, b_i should meet the constraint condition as $|b_i| \leq 3\sqrt{\lambda_i}$, which is a commonly-used configuration in the ASM domain.

Face shapes in videos and images can be searched and extracted, and then facial components can be located with the help of landmark points. Several examples are presented in Figure 2.

3.2. Face Recognition with CNN and Bayes Probability

Convolution network is a multilayer perceptron designed to identify 2D shapes. This network structure is highly invariant to translation, scaling, tilting or common deformation.

The layer of CNN can be divided into convolutional layers and sub sampling layers. The convolution layer is followed by a sub sampling layer to reduce computation time and establish spatial and structural invariance, whereas a convolutional filtering is carried out from S to C. After a number of such layers, the final eigenvector is obtained.

In Figure 3, f_x is a digital filter, b_x is a bias parameter, C_x is the feature map of convolutional layer, W_{x+1} means the value of sub sampling, b_{x+1} is the corresponding weight. This connection shows how to obtain the sub sampling layer S_{x+1}.

Alternating between the convolution layer and the sampling layer, CNN consists of forward propagation and backward propagation. Forward propagation is calculated from input parameters to output results. The output of the previous layer is the input of the current layer. By activating the function, the output of the current layer is
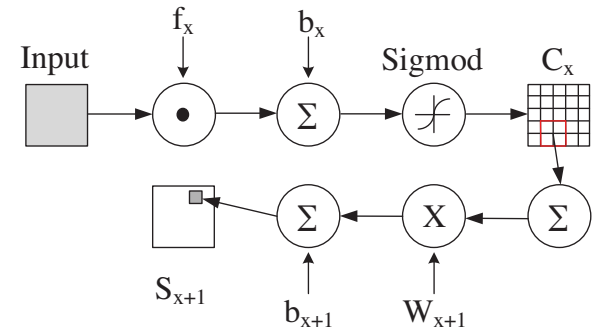

Figure 2 | Examples of facial component location. Red points mean the landmark points of facial components. After facial component location, it is necessary to increase the consistency of images. This task is conducted by aligning, which includes translocation, zooming, and rotation.

Figure 3 | The connection of CNN.

calculated and passed down layer by layer, thus the output of the current layer can be expressed as:

$$x^l = f\left(W^l x^{l-1} + b^l\right) \quad (7)$$

l means the layers, f is the activation function, such as tanh, sigmoid and relu, in this paper, relu is used.

Spread back propagation is used to calculate the error between results and given sample labels. Considering square variance loss function, for classification problem which contains c categories and N training samples, the error function is shown as:

$$E^N = \frac{1}{2} \sum_{n=1}^{N} \sum_{k=1}^{c} \left(t_k^n - y_k^n\right)^2 \quad (8)$$

t_k^n means k dimensionality of the sample n, y_k^n means the output k of the sample n.

Back propagation will update convolution layer, and each output map may be associated with the convolution of several feature mappings on the previous layer. The general form of the convolution layer is:

$$x_j^l = f\left(\sum_{i \in M_j} x_i^{l-1} \times k_{ij}^l + b_j^l\right) \qquad (9)$$

l means the layers, k is the convolutional kernel, M_j is one of the input features, b is the bias.

The sub sampling operation does not change the number of feature mappings, but only the size. If the size of the sampling operator is $n \times n$, then the size of the feature map becomes $1/n$ of the original feature after sampling. The general form of the sub sampling is:

$$x_j^l = f\left(\beta_j^l \text{down}\left(x_j^{l-1}\right) + b_j^l\right) \qquad (10)$$

down () means a sub sampling function.

In sum total, CNN is a special kind of deep neural network model, its particularity embodies in two aspects, on the one hand, it's connections between neurons are not all connected, on the other hand, some weights of connections between neurons in the same layer are shared. These specialties make it more similar to the biological neural networks, which can reduce the number of the weights and the complexity of the network model. It is very important because the deep structure is difficult to learn.

Convolutional Architecture for Fast Feature Embedding (Caffe) is a deep learning framework made with expression, speed, and modularity in mind. It is developed by Yangqing Jia in Berkeley AI Research (BAIR), and it is specially designed for realizing CNN model easily, and it has the characteristics of modularization, rapidity and openness. Therefore it is widely used to extract face features.

With the help of CNN and Caffe, significant results have been achieved in face recognition. However, there are some inherent shortages. For example, traditional CNN model extracts global features of a whole face, so it can't map local differences accurately. Besides, for a CNN, the existence of facial component parts becomes a very powerful indicator of a face in the image. But the orienteering and relative spatial relationships between these components are not important for CNN [21] as shown in (Figure 4).
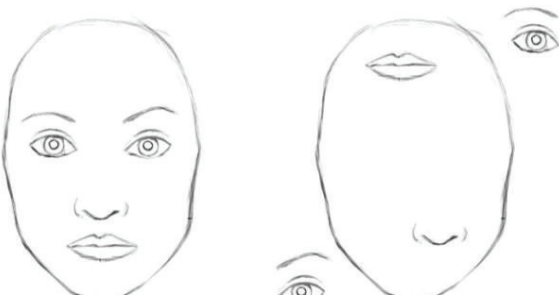

Figure 4 | For CNN, the two graphs are similar because they contain similar elements. Obviously, this is not true.

In contrast, Bayesian Program Learning (BPL) uses written parts and their order to identify characters, and this method has achieved excellent results [22]. Inspired from this, we use multiple CNNs to extract features of ordered facial components, and then calculate the ultimate similarity based on Bayesian probability. The process is presented in Figure 5.

Different CNN networks are used for face and various facial components in this study. Face labels are corresponding to each person, while the labels of components correspond to the types of components. For example, eyes are classified into triangular ones, narrow ones, big ones, and so on. Most kinds are difficult to describe, in fact digital label are directly used. More specifically, the network for global face has 64 layers, each network for other facial components has 20 layers, and the size of global face image is 128×128 pixels, the size of eye or brow is 32×32 pixels, the size of nose or mouth is 64×32 pixels.

After the establishment of two kinds of multiple CNN networks, the features of the template face and the face to be compared can be extracted, named as Feature$_{\text{temp}}$ and Feature$_{\text{target}}$. The probability of the similarity between the two as the same person is marked as P_0. Furthermore, the facial component i is sent to its classification network, and output its classification type S_i and corresponding probability P_i, $i = 1, 2, 3, ..., 7$. If the template face and the component parts to be compared with the face are used as input, the output is expressed as $\left(S_{ij}, P_{ij}\right)$, $i = 1, 2, 3, ..., 7$, $j = 0, 1$. Here 0 represents a template face, and 1 represents a face to be compared.

Naive Bayesian Model (NBM) is originated from classical mathematics theory. It has a solid mathematical foundation and a stable classification efficiency. It is one of the most widely used classification models. NBM needs few parameters to be estimated and is not very sensitive to missing data, and the algorithm is simple [23]. When a sample is given, $P\left(C_i|X\right)$ means the probability that it belongs to each category. C_i represents the category, X represents the test sample, and the maximum probability can be calculated as

$$p\left(c|x\right) = \frac{p\left(x|c\right)p\left(c\right)}{p\left(x\right)} \qquad (11)$$

Similarly, if we get template face X through CNN network, a series of probability values of target face Y, including the similarity of two faces, and the classification results and probability of each facial feature component. The problem can be summed up as: how to determine whether the two faces are the same person? In the form of reference [11], it can be expressed as:

$$p\left(c|(X, Y)\right) = \frac{p\left((X, Y)|c\right)p\left(c\right)}{p\left(X, Y\right)} \qquad (12)$$

Here c indicates that two faces are classified into the same class. To simplify the calculation, we define $p\left(c\right) = S_0$, then $p\left(S_0|(X, Y)\right)$ indicates that two faces are sorted into the same class, that is, facial components are classified to same pattern. At this time, the classification probability of all component parts is calculated as

$$p\left(S_0|(X, Y)\right) = \sum_i \left(P_{i0} \times P_{i1} \times S\right), S = \begin{cases} 0, & \text{if}(S_{i0} = S_{i1}) \\ 1, & \text{if}(S_{i0} \neq S_{i1}) \end{cases} \qquad (13)$$

Here $p\left(S_0|(X, Y)\right)$ is calculated as final probability of whether two faces are treated as the same person. Thus, the face recognition
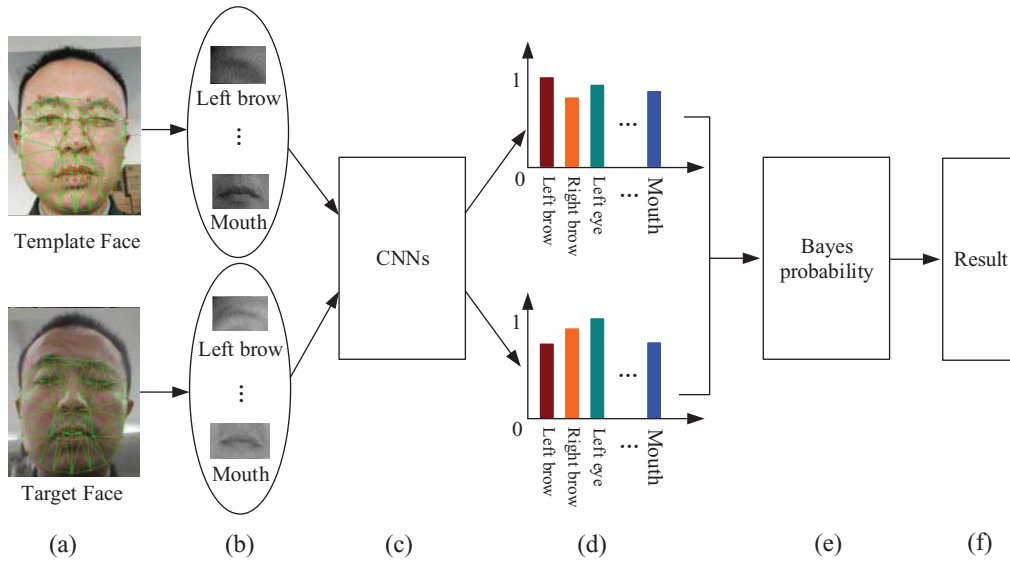
Figure 5 | The whole procedure of face recognition with CNN and Bayes probability. (a) Original faces, (b) Facial components, (c, d) Feature extraction and classification for each kind of component by corresponding CNN, the color of column means different kind of components, and the height of column means the probability value of current classification, (e) Vectors of classification results synthetically calculated by Bayes probability, (f) The final similarity output of two faces.

is achieved. Traditional methods usually use the global face as the input of convolutional neural networks. As people grow older, the face shape usually changes and affects the recognition results. Notice that even though the face become fat or thin, shapes of facial components are relatively stable, so this paper combines the global face and local facial components to extract features, thus it has better robustness in adapting to the changes of face age.

4. EXPERIMENTAL RESULTS AND ANALYSIS

We have tested the new algorithm on a personal computer that has an Intel Core CPU with 3.33 GHz and 8 GB DDR.

At large, the existing face recognition methods based on deep learning have achieved excellent results on ordinary image sets, such as the face database of AT&T Lab in Cambridge University [24], Extended Yale B database [25], the CMU pose, illumination and expression database [26], Pubfig [27], and so on. However, these algorithms do not work well on actual scenes, especially when one face is ID photo and the other is monitor image. So we build a face image set, in each group, it contains ID card photo and recent live images. There are 12 300 such groups in the image set, and they mainly contain these differences:

1. With glasses and without glasses.

2. Deformations caused by the change of the ages.

Some examples are shown in (Figure 6):

Comparatively, our experiments are operated with four deep learning methods: simple CNN for global face, optimized CNN including NormFace and DeepID3, and our multi-CNN with Bayes

Figure 6 | Examples of our image set.

probability. In order to record their statistical results, we should introduce some definitions:

Similarity threshold: if two faces whose similarity is greater than the threshold, they will be judged as the same person.

TP_num: the number of true positive.

TN_num: the number of true negative.

FP_num: the number of false positive.

FN_num: the number of false negative.

True positive rate (TPR):

TPR = TP_Num / (TP_Num + FN_Num)

False positive rate (FPR):

FPR = FP_Num / (TN_Num + FP_Num)

False negative rate (FNR):

FNR = FN_Num / (TP_Num + FN_Num)

Error rate curve (ERC):

The similarity threshold is the *x*-axis, then curves are drawn with TPR, FPR and FNR as the ordinate respectively. The intersection of FPR and FNR curves is equal error rate (EER) point, the corresponding abscissa of which is the optimal threshold.

Receiver operating characteristic (ROC):

Take FPR as the abscissa and TPR as the ordinate to draw the curve. The larger area under the curve indicates better effect of classifier.

In this paper, corresponding performances are shown in Figures 7 and 8.

(a)   Simple CNN, the best threshold of EER point is 0.49, the FPR value is 0.42, and the TPR value is 0.58 at this point.

(b)   NormFace, the best threshold of EER point is 0.63, the FPR value is 0.14, and the TPR value is 0.86 at this point.

(c)   DeepID3, the best threshold of EER point is 0.48, the FPR value is 0.12, and the TPR value is 0.88 at this point.

(d)   Our new algorithm, the best threshold of EER point is 0.51, the FPR value is 0.08, and the TPR value is 0.92 at this point.
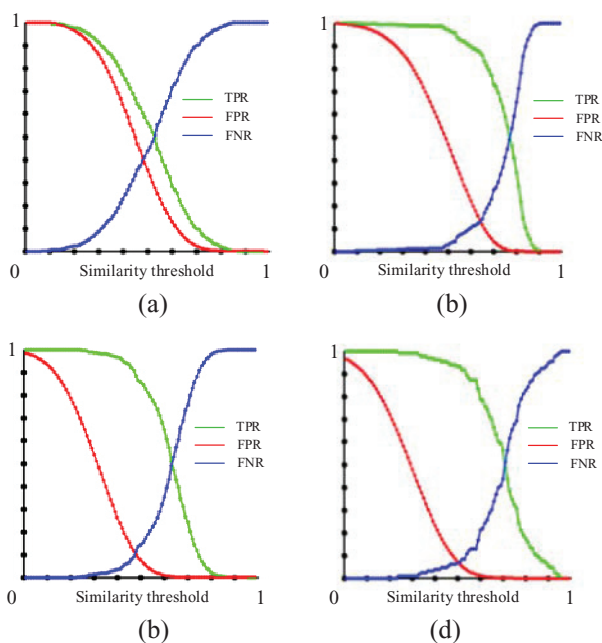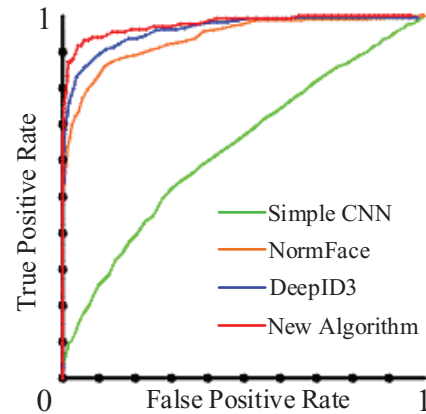

Figure 7 | ERC of contrastive algorithms.

Figure 8 | True positive rate and false positive rate test on our actual face image set.

Table 1 | Average test time.

Algorithm	Time cost per face
Simple CNN	0.08 second
NormFace	0.09 second
DeepID3	0.08 second
Our new algorithm	0.09 second

The smaller FPR value of EER point in the ERC map, and the bigger area of TPR in the ROC map represent better performance. In this experiment, the new algorithm achieves the highest TPR whereas its FPR is lower than that of other algorithms. The experimental results demonstrate the previous theoretical analysis. The new algorithm in this paper is indeed more adaptable to illumination changes, age changes, wearing glasses and other aspects. The face recognition effect is better than the contrast method.

From a computational cost perspective, the overall calculation is related to the number of layers because of the convolution operations. Based on the hardware and test mentioned above, the average test time (but not training time) by CPU is shown in Table 1.

The new algorithm is at the same level as other methods. Since these CNNs are parallel processing, that means the new method does not increase the time complexity.

After aggregating and analyzing the experiment results, the new algorithm has a significant advantage: it can spend the same time to achieve better treatment effect.

5. CONCLUSION AND FUTURE WORK

Traditional CNN face recognition methods extract global features of faces, but they are easily affected in practical applications, because of the following factors:

1.   Although the complete face is pre-processed by correction and normalization, it is unavoidable that the face is skewed and affects the global feature.

2.   There is partial occlusions in the face area, such as wearing glasses.

3.   As age goes on, face may increasingly change.

In this study, face images are captured with randomly changed expressions, and facial components are obtained via optimized pre-treatment. Then various CNN models are designed for the different facial components such as: global face, left brow, right brow, left eye, right eye, nose, and mouth. The CNN models are trained to extract different features. Finally Bayes probability is used to combine global features and local features. The new method can preferably adapt to the above three kinds of effects, and improve the accuracy of personal identification. It can verify the liveness of targets in particular. Therefore, this study has achieved the goal of enhancing accuracy and safety through the new method.

REFERENCES

[1] M. Cho, Y. Jeong, Face recognition performance comparison between fake faces and live faces, Soft Comput. 21(12) (2017), 3429–3437.

[2] B. Xiao, X. Gao, D. Tao, Y. Yuan, J. Li, Photo-sketch synthesis and recognition based on subspace learning, Neuro Comput. 73 (2010), 840–852.

[3] X. Wang, X. Tang, Face photo-sketch synthesis and recognition, IEEE Trans. Pattern Anal. Mach. Intell. 31(11) (2009), 1955–1967.

[4] L. Wolf, T. Hassner, Y. Taigman, Descriptor based methods in the wild, in Faces in Real-Life Images Workshop in ECCV, 2008. https://www.cs.tau.ac.il/~wolf/papers/patchlbp.pdf

[5] D. Lowe, Distinctive image features from scale-invariant key-points, Int. J. Comput. Vis. 60(2) (2004), 91–110.

[6] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in Proceeding CVPR, 2005. https://ieeexplore.ieee.org/abstract/document/1467360

[7] U. Prabhu, J. Heo, M. Savvides, Unconstrained pose-invariant face recognition using 3D generic elastic models, IEEE Trans. Pattern Anal. Mach. Intell. 33(10) (2011), 1952–1961.

[8] P. Yan, K.W. Bowyer, Biometric recognition using three dimensional ear shape, IEEE Trans. Pattern Anal. Mach. Intell. 29 (2007), 1297–1308.

[9] J.D. Bustard, M.S. Nixon, 3D morphable model construction for robust ear and face recognition, CVPR. 1 (2010), 2582–2589.

[10] L. Deng, Three classes of deep learning architectures and their applications: a tutorial survey, Microsoft Research, 2013.

[11] Y. Chen, H. Jiang, C. Li, P. Ghamisi, Deep feature extraction and classification of hyperspectral images based on convolutional neural networks, IEEE Trans. Geosci. Remote Sens. 54(10) (2016), 6232–6251.

[12] Y. Wen, K. Zhang, Z. Li, Y. Qiao, A discriminative feature learning approach for deep face recognition, ECCV. 1 (2016), 499–515.

[13] K. Zhang, Z. Zhang, Z. Li, Y. Qiao, Joint face detection and alignment using multitask cascaded convolutional networks, IEEE Signal Proc. Let. 23(10), 1499–1503.

[14] Y. Zhang, D. Zhao, J. Sun, G. Zou, W. Li, Adaptive convolutional neural network and its application in face recognition, Neural Proc. Let. 43(2) (2016), 389–399.

[15] Y. Sun, D. Liang, X. Wang, X. Tang, DeepID3: face recognition with very deep neural networks, Comput. Sci. (2015), arXiv preprint arXiv:1502.00873.

[16] F. Wang, X. Xiang, J. Cheng, A.L. Yuille, NormFace: L2 hypersphere embedding for face verification, Proceedings of the 2017 ACM on Multimedia Conference, ACM. 1 (2017), 1041–1049.

[17] Y. Taigman, M. Yang, M.A. Ranzato, L. Wolf, DeepFace: closing the gap to human-level performance in face verification, CVPR. 1 (2014), 1701–1708.

[18] F. Schroff, D. Kalenichenko, J. Philbin, FaceNet: a unified embedding for face recognition and clustering, CVPR. 1 (2015), 815–823.

[19] A. Bandera, J.M. Pérez-Lorenzo, J.P. Bandera, F. Sandoval, Mean shift based clustering of Hough domain for fast line segment detection, Pattern Recognit. Lett. 27 (2006), 578–586.

[20] T.F. Cootes, C.J. Taylor, D.H. Cooper, J. Graham, Active shape models their training and application, CVIU. 61 (1995), 38–59.

[21] S. Sabour, N. Frosst, G.E. Hinton, Dynamic routing between capsules, NIPS. 1 (2017), 3856–3866.

[22] B.M. Lake, R. Salakhutdinov, J.B. Tenenbaum, Human-level concept learning through probabilistic program induction, Science. 350(6266) (2015), 1332–1338.

[23] T. Kim, B.D. Chung, J.S. Lee, Incorporating receiver operating characteristics into naive Bayes for unbalanced data classification, Computing. 99(3) (2016), 1–16.

[24] ORL Face Set. http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html

[25] A. Georghiades, P. Belhumeur, D. Kriegman, From few to many: illumination cone models for face recognition under variable lighting and pose, IEEE Trans. Pattern Anal. Mach. Intell. 23(6) (2001), 643–660.

[26] T. Sim, S. Baker, M. Bsat, The CMU pose, illumination and expression database, IEEE Trans. Pattern Anal. Mach. Intell. 25(12) (2003), 1615–1618.

[27] N. Kumar, A. Berg, P.N. Belhumeur, S.K. Nayar, Describable visual attributes for face verification and image search, IEEE Trans. Pattern Anal. Mach. Intell. 33(10) (2011), 1962–1977.