Future Generation Computer Systems 111 (2020) 375–380

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Multi-view face recognition using deep neural networks

Feng Zhao [a], Jing Li [a], Lu Zhang [a,b], Zhe Li [a], Sang-Gyun Na [a,*]

[a] *Department of Business Administration, Wonkwang University, 460 Iksandae-ro, Iksan, Jeonbuk 54538, Republic of Korea*
[b] *Department of Office Management, Hebei GEO University, Shijiazhuang 050031, China*

ARTICLE INFO

Article history:
Received 9 January 2020
Received in revised form 25 April 2020
Accepted 1 May 2020
Available online 15 May 2020

Keywords:
Deep learning
Face region
Face image recognition
Deep neural network
PCA feature dimension reduction

ABSTRACT

F Face recognition has been widely used in modern intelligent systems, such as smart video surveillance, online payment, and intelligent access control system. Existing face recognition algorithms are prone to be attacked by various face presentation attacks (face-PAs), such as printed paper, video replay, and silicone masks. To optimally handle the aforementioned problems, we formulate a novel deep architecture to increase the accuracy of multi-view human face recognition. In particular, in the first place, a novel deep neural network is built for deeply encoding the face regions, where a novel face alignment algorithm is employed to localize the key points inside faces. Subsequently, we utilize the well-known PCA for reducing the dimensionality of the deep features and simultaneously, removing the redundant and contaminated visual features. Thereafter, we propose a joint Bayesian framework in order to evaluate the similarity of feature vectors and highly competitive face classification accuracy can be achieved. Comprehensive experiments were conducted on our compiled CAS-PEAL dataset and achieved a 98.52% face recognition performance. Moreover, our proposed face recognition system can robustly handle various face recognition attack under various contexts.

1. Introduction

In recent years, face recognition has attracted lots of attention in plenty of domains. The relevant techniques can be employed in different intelligent systems, for example, smart phone unlocking, online payment and access control system [1–3]. The objective of face recognition is to localize/detect and track various human faces by leveraging the captured images. This technique plays a highly important role in biological verification. Each face recognition system captures a face image from one or multiple persons by utilizing a camera, and thereafter it compares the human face with the face samples that are already fed into the face database to fulfill the recognition. Face recognition exhibits the feature of non-contact, wherein it may not be descriptive for the identified person to deliberately join the feature collection. In this way, the output of rejecting the identified person will be identified. In this way, human face recognition technology is highly useful to biometric identification domain.

Nowadays, the pervasively used face recognition suffers from multiple challenges in the real applications. More detailed, in the real-world application scenarios, face angles, expressions, lighting effects, age changing. And therefore, face image clarity issues, face regions will become smaller, the pixels will become higher. This will enforce the deep learning platform requires a larger scale input image. This will make the entire deep learning system infeasible for low-pixel images to be identified by the proposed deeply-learned visual models. The recognition task toward human faces for each monitoring device requires a real-time-level response operation. It is worth emphasizing that, however, lots of human face recognition algorithms nowadays require one/multiple models to be seamlessly combined. This setup will meet the requirements of real-time human face recognition. The parallel computational resources highly requires a competitive system equipment. There are also tough problems regarding the data training process. Nowadays, many large companies are leveraging the big data technique for data training. Moreover, the deep neural networks with varied architectures are becoming larger and larger. This requires for the equipment with a high performance, wherein the cost indicates the problem to be analyzed by application. We plan to fulfill human face recognition on mobile platforms with smaller GPU.

The earliest reference on face recognition was a paper published by scientist Gaiton in "Nature" in 1988 [1]. But in the mid to late 1960s, the study of automatic face recognition was pervasive. The earliest automatic face recognition research was the work published by chan and Bledsoe in 1965 [2]. The key techniques of human face recognition (1965–1990) generally investigated face recognition algorithms by leveraging geometric features, such as Harmon and Lesk [3] by tuning the geometric

* Corresponding author.
E-mail addresses: do00npx@163.com (F. Zhao), 99moonsea@163.com (J. Li), zhanglu@hgu.edu.cn (L. Zhang), qingdaolizhe@outlook.com (Z. Li), nsghy@wku.ac.kr (S.-G. Na).

https://doi.org/10.1016/j.future.2020.05.002
0167-739X/

feature parameters exhibited in facial front images. Based on this, they designed a human identification framework by adopting the deep feature representation framework. Since the above research is in its infancy, it has not been practically used. From 1991 to 1997, lots of progresses is made in human face recognition. During this time, the well-known Eigenface algorithm [4] was proposed. Such technique was also proposed by the Massachusetts Institute of Technology and there are a rich number of researches on face recognition by updating this method. In 1992, Brunelli and Poggio [5] implemented face recognition algorithm by leveraging structural features based on the so-called template matching. And important conclusions are made: human face recognition algorithm by leveraging template matching performs better than face recognition using structural features.

Face modeling technique has been developed rapidly over the past three decades. Particularly, in recent years, face recognition has becoming the key topic in pattern recognition and machine learning. As far as we know, there are a rich number of very famous enterprises and universities in China and abroad with a rich set of project support. Since 1960, systematic research and exploration of face recognition has been emerged. And in 1980s, based on the fast development of computing infrastructure and optical imaging techniques, face recognition technique has been remarkably enhanced. During the late 1990s, face recognition is frequently used in many real-world applications. In many real-world applications, advanced machine learning algorithms, recognition efficiency/accuracy is becoming the key to the success of face recognition system. Currently, face recognition is pervasively used in access control systems requiring inspecting areas, monitoring systems for suspicious human/vehicles, financial systems for verifying customer identifies, smartphones identifying, and paying using human faces.

Besides artificial intelligence, human face recognition technique involves a rich set of disciplines, for example, such as physiology, biology and cognition. The advancement of human face recognition requires the seamless combination of a set of disciplines. Other disciplines could also develop effectively by optimally leveraging the face recognition, which plays an indispensable role to the generation and advancement of new research topics. Generally, it is observable that face recognition algorithms have a great scientific impact and social significance, and has substantially enhanced academic and industrial activities, and exhibits some far-reaching effects. Machine learning [6] functions as the key research direction in the AI community. AI can be considered as a multidisciplinary product that optimally combines multiple disciplines such as probability statistics, information theory, computer complexity theory, neurobiology, cybernetics, psychology, and philosophy. More importantly, machine learning's objective is to simulate human cognitive processing through algorithms, and subsequently learn the generalization rules from massive-scale visual data. Based on this, new samples are identified using laws to predict the new samples. After the development for many years, plenty of machine learning tools have been developed and great success is achieved in theory, algorithms, and applications. Notable achievements include the Decision Tree, Support Vector Machine (SVM) and Artificial Neural Network (ANN) [7].

2. Proposed method

Artificial God Network (ANN) is a large-scale reasoning system of non-homogeneous non-religious self-religion, which is a research result of the ancient theological sciences, and a large-scale religion. From the philosophy of the physical angle, the abstract of the gods, the model of the erection, the unequal connection method, the unequal connection method. A large number of points (or the origin of the deity) of the gods of the world.

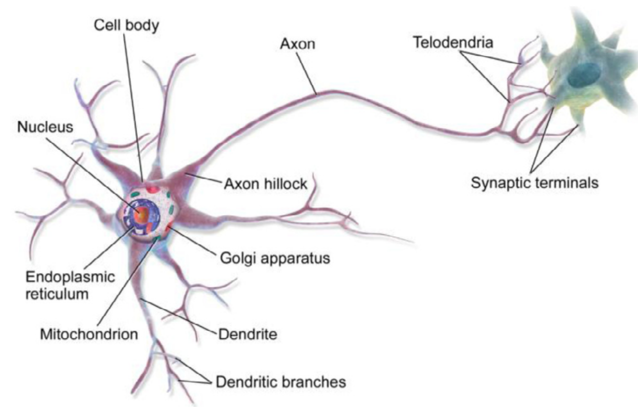

Fig. 1. Biological neuron diagram.

2.1. Neural networks

The structure of human brain is complex, but the basic unit of nervous system is neuron, which is the basic unit of human brain processing information. Fig. 1 shows the basic structure of neurons, including cell bodies, dendrites and axons. Cell body is the microprocessor of neuron, which receives all signals for simple processing; the function of dendrite is to receive signals from external or other neurons; axon is the signal processed by cell body. Output to other neurons [8].

Our nervous system is connected by a large number of neurons through network structure. It is a very intelligent and complex processing system. It receives information from the external environment in a very unique way, then analyzes and processes it, and finally controls the body to respond.

Inspired by biological neurons, we can get the simplest neural network model, a single-layer perceptron from 0 to 1, whose network contains only one neuron, as shown in Fig. 2. Inspired by biological neurons, one can derive the simplest neural network model, a single-layer perceptron from 0 to 1, whose network contains only one neuron.

In the diagram, x1, X2 sum X3 input signal, y output signal, W1, W2 sum W3 input signal weight. Calculation of deviations, deviation amount B, special value, immediate B = W0, Kachi Yugami, y-weight wi-like function, immediate Eq. (1)

$$Y = f(W^T X) = f(\sum_{i=0}^{3} W_i X_i) \tag{1}$$

In the single-layer perceptron, the activation function f() selects the sign(x) function, and its formula (2) is as follows:

$$sign(x) = \begin{cases} 1 & x > 0 \\ 0 & x = 0 \\ -1 & x < 0 \end{cases} \tag{2}$$

The above is the simplest neural network model-a single neuron. So how does it adjust the weights through self-learning and ultimately achieve the expected output? This is due to the learning rule of the perceptron. The rule is that the learning signal is equal to the difference between the expected output and the actual output of the neuron, that is, Eq. (3) is as follows:

$$r = d_j - o_j \tag{3}$$

In formula (3), d_j—expected output; o_j—actual output, $o_j = f(W_j^T X)$ substitutes $o_j = f(W_j^T X)$ into formula (2), and formula (4) is obtained:

$$f(W_j^T X) = sign(W_j^T X) = \begin{cases} 1 & W_j^T \geq 0 \\ -1 & W_j^T < 0 \end{cases} \tag{4}$$

Fig. 2. An example of a single neuron.

Then the weight adjustment formula (5) is as follows:

$$\Delta W_j = \eta[d_j - sign(W_j^T X)X] \tag{5}$$

In the above formula, if there is an error in the network structure, the weight needs to be adjusted, and since d_j and $sign(W_j^T X)$ take only two values of -1 and 1, the formula (5) can be written as the formula (6) as follows:

$$\Delta W_j = \pm 2\eta X \tag{6}$$

In formula (6), η is the learning rate of the network structure. Its usual range is. In the actual training process, it is necessary to pay attention to the appropriate selection of the learning rate. When the learning rate is too large, it is easy to make the weight adjustment fluctuation. Too big; when the learning rate is too small, the weight adjustment is too slow.

2.2. Convolutional neural network

Convolutional neural network (CNN) is a multi-layer characterization algorithm for processing high-dimensional grid data (i.e., tensor). The network directly performs feature learning on the input image through convolution operation, avoiding complex artificial feature processing, and is therefore favored by researchers. CNN conducted in-depth analysis, mainly including convolution (Conv), pooling (pooling), full connection (FC), etc. The principle and rules of each step are as follows.

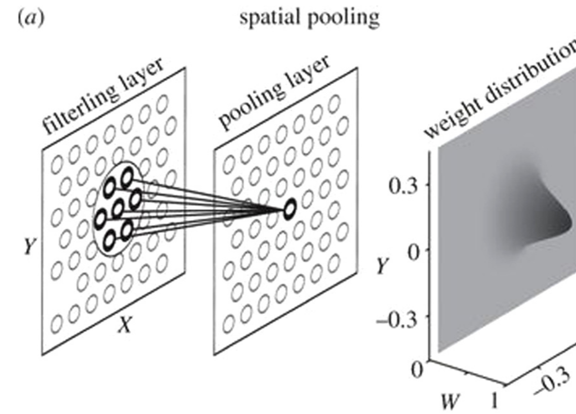

Fig. 3. An example of spatial pooling.

(1) Convolution operation

The essence of convolution is to operate on two functions f and g to obtain a new representation function. Since the training is based on the face image, only the discrete convolution operation is explained. The formula (7) is as follows:

$$S(i,j) = (F * K)(i,j) = \sum\sum F(m,n)K(i-m, j-n) \tag{7}$$

In the above convolution formula, the first parameter (i,j) is input data, the second parameter $F(m,n)$ is called "core function", m, n is the size of the filter, and $S(i,j)$s the output feature map result value.

(2) Pooling

The two aggregation methods in the pool are shown in Fig. 3. One is to take the average value, called the average pool; the other is to take the maximum value, called the maximum pool. In fact, if only dimensionality reduction is considered, the same goal can be achieved without sampling. For the pool, it is more important to use the maximum or average method to make the feature extraction "shift invariant". That is to say, even in the case where there are multiple pixel displacements in the image, a stable feature set can be obtained. Among them, the average pool mainly extracts the average value of the feature points in the region, and the effect of maintaining the relative background is better; the maximum pool mainly extracts the maximum value of the feature points of each region, and the texture extraction effect is better.

The combination of convolution and pooling adds a strong prior experience to convolutional neural networks, emphasizing local continuity and correlation of images while maintaining translational invariance.

(3) Full connection operation

Through the above analysis, convolution is a method of feature extraction, pooling is a method of dimensionality reduction, and then the fully connected layer is used as the final data output layer. Therefore, the fully connected layer acts as a classifier in the entire CNN network, realizes the projection space mapping function, and projects the extracted results from one feature space to different feature spaces. Eq. (8) is as follows:

$$r = w_{mn}h + b \tag{8}$$

2.3. Activation function

In traditional neural networks, the use of activation functions primarily increases the nonlinear modeling capabilities of the network, so in general the activation function is a nonlinear function. The sigmoid function [9] or the tanh function [10] is used as an activation function to simulate activation. In CNN, the convolution operation is actually a linear operation, which only includes addition and multiplication in computation, but one of the most important theoretical foundations in deep learning is that the vector of a feature space must be transformed by nonlinear transformation. Linearity can be achieved by mapping to another space. The Activation Function is a means of introducing nonlinearity.

The Sigmoid function, formula (9) is as follows:

$$f(z) = \frac{1}{1+\exp(-z)} \tag{9}$$

The Tanh function, formula (10) is as follows:

$$f(z) = \tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \tag{10}$$

For the fitting of neural networks, the process is to constantly superimpose similar data and features, and the function of the activation function is to segment the data categories by nonlinear functions, thus obtaining a better simulation. Combined results. When the model is built, the activation function should be further selected according to the specific task.

In recent years, new nonlinear activation functions have emerged on the basis of ReLU, such as LcakyReLU [11], PReLU [12], RReLU [13], ELU [14] and so on.

The ReLU formula (11) is as follows:

$$f(x) = \max(0, x) = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \tag{11}$$

The Leaky ReLU formula (12) is as follows:

$$f(x) = \begin{cases} x & \text{if } x \geq 0 \\ \alpha x & \text{if } x < 0 \end{cases} \tag{12}$$

where $\alpha \in (0, 1)$ is a fixed value, usually set to 0.01.

The PReLU formula is as shown in (12), where $\alpha \in (0, 1)$ is a parameter learned through a neural network.

The RReLU formula is as in (12), where $\alpha \sim U(l, u)$, $l < u$ and $l, u \in (0, 1)$.

The ELU formula (13) is as follows:

$$f(x) = \begin{cases} x & \text{if } x \geq 0 \\ \alpha(e^x - 1) & \text{if } x < 0 \end{cases} \tag{13}$$

Where $\alpha > 0$.

2.4. Batch normalization

In the deep neural network training process, the initialization of parameters is very demanding, because the parameters of the input layer are constantly changing, and the distribution of the current layer is constantly changing, which leads to the setting of a training process. Small learning rate. This phenomenon is called Internal Covariate Shift. In order to solve this problem, Batch Normalization [15] came into being. In CNN, usually the Batch Normalization operation is applied to the nonlinear activation function before subtracting its own mean value from each dimension and dividing by its own standard deviation.

For input $B = \{x_{1...m}\}$ in a mini-batch, the Batch Normalization calculation process is formula (14):

$$\begin{aligned} \mu_\beta &= \frac{1}{m}\sum_{i=1}^{m} x_i \\ \sigma_\beta^2 &= \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_\beta)^2 \\ \hat{x}_i &= \frac{x_i - \mu_\beta}{\sqrt{\sigma_\beta^2 + \epsilon}} \\ y_i &= \gamma \hat{x}_i + \beta \equiv BN_{\gamma,\beta}(x_i) \end{aligned} \tag{14}$$

The benefits of Batch Normalization are:

(1) You can use a higher learning rate to train;
(2) can avoid over-fitting and speed up the training;
(3) Reduce the L_2 weight attenuation coefficient.

2.5. Similarity metric

2.5.1. European distance

In the literature, one of the most frequently used metrics is the Euclidean distance [16]. It performs the L2 norm on the vector before calculation, as detailed in Eq. (15):

$$d(x_1, x_2) = \left\| \frac{x_1}{\|x_1\|_2} - \frac{x_2}{\|x_2\|_2} \right\|_2 \tag{15}$$

In our implementation, a carefully tuned threshold is adopted, if $d(x_1, x_2)$ it is less than the threshold, it is the same person, otherwise it is different.

2.5.2. Union Bayes

By leveraging the technique of joint Bayesian [17], the high-dimensional local binary mode is combined with the joint Bayesian theory. If pairwise human faces are denoted as x1 and x2, respectively, then X = {x1, x2} is leveraged that optimally combines the two feature vectors, H_I denotes that pairwise human faces belong to the same person, and H_E indicates that two faces belong to different persons. The state-of-the-art face verification algorithms can be deployed to classify vector X to find whether X belongs to H_I or H_E. This operation is similar to compare the sizes of $P(H_I|X)$ and $P(H_E|X)$. Based on Bayesian formulas (16) and (17):

$$P(H_I|X) = \frac{P(X|H_I) \times P(H_I)}{P(X)} \tag{16}$$

$$P(H_E|X) = \frac{P(X|H_E) \times P(H_E)}{P(X)} \tag{17}$$

where $P(H_1) = P(H_E)$.

3. Experimental results and analysis

In practice, face recognition is conduct by collecting a rich set of face images. To our best knowledge, there are plenty of face images distributed on the Internet, which are usually managed in a disordered way. Well-known IT companies like Google, Facebook, and Baidu all released their massive-scale face databases with rich information. However, these data sets have not been publicly released due to the privacy and copyright issues. Owing to the impossibility to obtain very large-scale databases with high face quality as well as rich associated tags. Comparatively, the academic community is in a relatively passive attitude in upgrading the human face recognition technique. More recently, multiple research institutes have handle such problem and subsequently released a series of large-scale and diverse face sets. A few representative face data sets are briefly introduced in the following:

To our knowledge, the MegaFace is the first benchmark for million-scale face recognition algorithms. Totally, it contains one million face images from 69,000 different categories.

The VGG face data set is a face recognition database constructed from scratch. There are 2.6 million pictures and more than 2600 people. The construction process is mainly realized by procedures. A small amount of manual participation is mainly based on the use of Internet search engines and the use of existing personnel. The face recognition method filters the data, generates a large data set, and marks its identity.

The CelebA dataset is carefully-tuned with multiple face attributes, such as hats, glasses, beards, etc. It is built free from the identity attribute from the calibrated face image. This data set is mainly adopted for face attribute understanding.

The MSCeleb was first released by the well-known Microsoft, wherein the data set totally contains 100 million celebrities selected by 100,000 by leveraging their popularity. Afterward, by utilizing the search engine, we annotate each of the 100,000 people, each searched for about 100 pictures. In this way, a total of 100k * 100 = 10M face images are obtained.

The UMDfaces is generally constructed from face images by employing the GoogleScraper web crawler. To carefully refine the purity of the data, the builders have cleaned up the face images. The resulting face database contains a total of >367,000 faces, which belong to 8501 different categories. The face set provides a wealth of face-related information, such as face location, face gestures, 21 facial key points and gender information.

CAS-PEAL [18] is a novel dataset containing 99,450 faces from 1040 volunteers completed by Chinese Academy of Sciences in 2003. The entire face data set involves changes in multiple human faces such as head gestures, expressions, decorations, lighting, background, and temporal. A proportion of the face images from this database is shown in Fig. 4.

This paper is based on the CPU image of Intel Xeon CPU E5-2620 v2@2.lO GHz, GPU for Nvidia Tesla K20m, 5G memory and 32G memory. The face image database of CAS-PEAL Institute of Computing Technology is Based on the Matlab2014b application, the effectiveness of the face recognition experiment simulation verification algorithm is verified.

For the data set required in this article, we performed the following preprocessing. First, face area detection, then key area detection of face area, and finally face alignment. In face positioning, we use the three points of the left eye, right eye and nose of the face detection area as reference points to adjust the positioning to the same angle. A 200 × 200 image can be obtained through face alignment as a data image for deep neural network training or testing.

4. Discussion

4.1. Feature dimension reduction

Through the CNN model, a 320-dimensional feature vector is obtained to represent the human face. The feature vector is reduced to a low-dimensional space before entering the classifier to better extract information. Experiments show that the CAS-PEAL data set after dimensionality reduction has good accuracy.

We first compare the various models, and then use the PCA dimension reduction method to compare the accuracy of different dimensionality reduction and non-dimensionality reduction feature vectors on CAS-PEAL. The results are shown in Table 1. If the dimension reduction is reduced to a lower dimension, the effect of dimension reduction will be worse due to the loss of information. When we reduce the dimension to a more moderate dimension, we can obtain better accuracy.

4.2. Feature similarity evaluation

This paper compares the accuracy of the Euclidean distance method and the combined Bayesian method on CAS-PEAL. We use three different convolutional neural network models to obtain feature vectors and use two methods for classification. The results are shown in Table 2. Experimental results show that the combined Bayesian method is more effective than the direct use of Euclidean distance.

Table 1
PCA dimension reduction effect comparison.

Dimensionality reduction	CAS-PEAL accuracy rate
No PCA	0.9620
PCA 50	0.9530
PCA 100	0.9664
PCA 150	0.9657
PCA 180	0.9768
PCA 200	0.9564
PCA 250	0.9324
PCA 280	0.9768
PCA 300	0.9213

Table 2
Euclidean distance and joint Bayesian comparison.

Classifier	Model 1 CAS-PEAL accuracy	Model 2 CAS-PEAL accuracy	Model 3 CAS-PEAL accuracy
European distance	0.9812	0.9454	0.9765
Union Bayes	0.9543	0.9760	0.9844

Table 3
Weight attenuation coefficient comparison.

Weight attenuation coefficient	CNN classification accuracy	CAS-PEAL accuracy rate
0.0005	0.7053	0.9487
0.0001	0.7032	0.9498
0	0.7089	0.9507
The convolution layer is taken as 0, and the full connection layer is taken as 0.0005.	0.7267	0.9521

Table 4
Comparison of weight attenuation coefficient of full connection layer.

Full connection layer weight attenuation coefficient	CNN classification accuracy	CAS-PEAL accuracy rate
0.0005	0.7804	0.9686
0.0025	0.7907	0.9735
0.0050	0.7903	0.9737
0.0100	0.7924	0.9749
0.0150	0.7939	0.9695
0.0250	0.7829	0.9708

4.3. Weight attenuation coefficient

During the training process of our adopted CNN, we incorporate the L_2regularization term to the loss function so as to prevent over-fitting and improve the generalization ability, as shown in the below equation:

$$J(W) = J_0(W) + \frac{\lambda}{2}\|W\|^2 \tag{18}$$

As can be seen, the parameter λ denotes the weight attention coefficient. When comparing these weight coefficients, we compare the face recognition accuracy of our CNN with the face verification accuracy on CAS-PEAL. The results are presented in Tables 3 and 4.

When the weight attenuation coefficient of the convolutional layer is set to 0, and the weight attenuation coefficient of the fully connected layer is set to 0.01, the obtained effect is the best.

4.4. Activation function

There are many nonlinear activation functions in the neural network to choose from. We choose three functions: ReLU, PReLU

Fig. 4. CAS-PEAL database part of the face picture.

Table 5
Comparison of activation functions.

Numbering	Activation function	CNN classification accuracy	CAS-PEAL accuracy rate
1	ReLU	0.7658	0.9616
	PReLU	0.7884	0.9652
2	PReLU	0.7924	0.9751
	ELU ($\alpha = 1$)	0.7823	0.9723
3	PReLU	0.8401	0.9852
	ELU ($\alpha = 0.25$)	0.8376	0.9837

and ELU for comparison. It can be seen from Table 5 that PReLU is better than ReLU and ELU because it can learn more parameters and there is no overfitting.

5. Conclusions

A face image recognition algorithm based on deep neural network is proposed. It includes the use of convolutional neural network to extract facial features, the use of PCA algorithm for feature dimensionality reduction, and the use of joint Bayesian method for vector similarity judgment. Finally, the purpose of improving the accuracy of face image recognition is achieved. After a series of experimental comparisons, the results show that on the CAS-PEAL data set, the accuracy of face image recognition can reach 98.52%.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This paper performance was supported by Wonkwang University in 2020.

References

[1] F. Galton, On personal identification and description, 1888, pp. 201–203.
[2] H. Chan, W.W. Bledsoe, A Man-Machine Facial Recognition System: Some Preliminary Results, Technical report, Panoramic Research Inc, Palo Alto. Cal, 1965.
[3] A.J. Goldstein, L.D. Harmon, A.B. Lesk, Identification of human faces, Proc. IEEE 59 (5) (1971) 748–760.
[4] M.A. Turk, A.P. Pentl, Eigenfaces forrecognition, J. Cogn. Neurosci. 3 (1) (1991) 71–86.
[5] R. Brunelli, T. Poggio, Face recognition through geometrical features, Lecture Notes in Comput. Sci. 588 (12) (1992) 792–800.
[6] D. Michie, D.J. Spiegelhalter, C.C. Taylor, et al., Machine learning, neural and statistical classification, Technometrics 37 (4) (1994) 459.
[7] S. Dreiseitl, L. Ohno-Machado, Logistic regression and artificial neural network classification models: a methodology review, J. Biomed. Inform. 35 (5) (2002) 352–359.
[8] M.W. Ahmad, M. Mourshed, Y. Rezgui, Trees vs neurons: Comparison between Random Forest and ANN for high-resolution prediction of building energy consumption, Energy Build. 147 (2017) S0378778816313937.
[9] X. Yin, J. Goudriaan, E.A. Lantinga, et al., A flexible sigmoid function of determinate growth, Ann. Botany 91 (6) (2003) 753.
[10] E.G. Fan, Extended tanh-function method and its applications to nonlinear equations, Phys. Lett. A 277 (4) (2000) 212–218.
[11] Andrew L. Maas, Awni Y. Hannun, Andrew Y. Ng, Rectifier nonlinearites improve neural network acoustic models, in: Proc. ICML, Vol. 30, 2013.
[12] T. Yoshioka, K. Ohnishi, F. Fang, et al., Noise robust speech recognition using recent developments in neural networks for computer vision, in: IEEE International Conference on Acoustics, 2016.
[13] B. Xu, N. Wang, T. Chen, et al., Empirical evaluation of rectified activations in convolutional network, Comput. Sci. (2015).
[14] L. Yang, C. Fan, L. Yong, et al., Improving deep neural network with multiple parametric exponential linear units, Neurocomputing (2018) S0925231218301255.
[15] S. Ioffe, C. Szegedy, Batch normalization: accelerating deep network training by reducing internal covariate shift, in: International Conference on International Conference on Machine Learning, JMLR.org, 2015.
[16] D. Drusvyatskiy, H.L. Lee, G. Ottaviani, et al., The euclidean distance degree of orthogonally invariant matrix varieties, Israel J. Math. 221 (1) (2017) 1–26.
[17] I. Barra, L. Hoogerheide, S.J. Koopman, et al., Joint Bayesian analysis of parameters and states in nonlinear non-Gaussian state space models, J. Appl. Econometrics 32 (5) (2017).
[18] G. Wen, C. Bo, S. Shan, et al., The CAS-PEAL large-scale chinese face database and baseline evaluations, IEEE Trans. Syst. Man Cybern. 38 (1) (2008) 149–161.