International Journal of Applied Science and Engineering

Face recognition system with feature normalization

Wen-Chang Cheng[1], Hung-Chou Hsiao[2*], Da-Wei Lee[1]

[1] *Department of Computer Science and Information Engineering, Chaoyang University of Technology, Taichung, Taiwan, R.O.C.*
[2] *Department of Information Management, Chaoyang University of Technology, Taichung, Taiwan, R.O.C.*

ABSTRACT

This study proposed an identity verification system that uses face recognition. The system features face detection as well as facial feature extraction and comparison methods. Early methods of face detection involved using specific approaches coupled with a classifier to extract features and detect faces. Although these methods can detect faces quickly, they generate a high false positive rate. Recent face detection methods based on a deep learning structure are extremely accurate but time-consuming. This study realized a face detection method based on the histogram of oriented gradient. The proposed method is not as accurate as deep learning; however, it is fast and can complete instant computing. Early methods of face recognition also involved using feature extraction methods coupled with a classifier to complete face recognition; however, these methods were not extremely accurate. The emergence of deep learning has facilitated greatly increasing the accuracy of face recognition. A deep learning-based method requires the entire deep learning structure to be retrained when a system needs to add a new user. This requirement is not feasible in actual applications. A researcher has therefore proposed the FaceNet method, which uses deep learning structure to extract eigenvectors and calculates the distance between eigenvectors as a measure of face similarity. Thus, the entire deep learning structure does not need to be retrained when a new user is added to the system. In this paper, FaceNet was used to extract the eigenvector of a face. However, the experiments of this study showed that facial features extracted using FaceNet are unevenly distributed in different dimensions, and using the calculated distance of the eigenvector as a measure of face similarity will yield inaccurate results. Therefore, this study proposed a facial feature normalization comparison method. The experimental results verified that the proposed method can achieve more than 98% accuracy and can be applied in practice.

Keywords: Deep learning, Feature extraction, Feature comparison, Histogram of oriented gradient, Feature normalization.

OPEN ACCESS

Received: August 5, 2020

Accepted: October 15, 2020

Corresponding Author:
Hung-Chou Hsiao
s10814902@gm.cyut.edu.tw

Publisher:
Chaoyang University of Technology
ISSN: 1727-2394 (Print)
ISSN: 1727-7841 (Online)

1. INTRODUCTION

Face recognition is a method of verifying the identity of a person by using biometrics. Using unique facial features, facial recognition can be applied to verification problems of different areas, particularly activities involving computer security, such as border inspections, building access control, criminal identification, and user identification. Face recognition can be branched into detection and recognition, both of which are issues of concern in image recognition. Previous studies have proposed a number of face detection methods. Viola and Jones (2001) proposed a face detection structure consisting of Haar-like Adaboost and Cascade classifiers. Because this face detection system can very quickly detect faces, it has been widely used in numerous applications. This approach has been realized in the OpenCV (2021) function base; however, false positives were frequently reported after actual tests. The accuracy of face detection has improved considerably in recent years, thanks to the success of deep learning technologies.

Jiang and Learned-Miller (2017) proposed face detection using Faster region-based convolutional network (R-CNN), which has previously demonstrated impressive results in a large-scale visual identity competition of ImageNet (2016). Therefore, Faster R-CNN can also accurately detect faces. However, after it was tested, Faster R-CNN was found to be overly sensitive; it only detected a side or part of the face and was unable to extract all facial features, which negatively affects facial recognition. Moreover, Faster R-CNN requires graphics processing unit (GPU) (Graphics processing unit, 2021) to accelerate the calculation process. In this study, a face detection method based on the histogram of oriented gradients (HOG) (Dalal and Triggs, 2005) was developed. Although the proposed method is not as accurate as Faster R-CNN, it can instantly complete computing and detect the whole face without using a GPU.

Face recognition ensues after facial images are extracted. Traditionally, facial features are extracted from an image; therefore, a number of feature extraction methods have been applied in face recognition. For example, Zhao et al. (2011) proposed a face recognition method based on local binary patterns (LBP). Shu et al. (2011) also developed a HOG-based face recognition approach. These methods were mostly short of accuracy or limited in application, rendering them not applicable in practice. Similarly, the recent emergence of deep learning technologies has facilitated not only overcoming face recognition problems but also greatly increasing the accuracy of face recognition. Subsequently, a multitude of deep convolutional neural networks (DCNN) based facial recognition systems have been proposed, including Faster R-CNN (Ren et al., 2017), Ensemble-CNN (Cheng et al., 2018), and FaceTime (Sladojevic et al., 2017). Because these methods primarily use DCNN at the core of recognition, when the system requires adding a new user, the entire DCNN must be retrained, which is not feasible in actual applications. Hence, Florian et al. (2015) proposed to improve this problem by introducing FaceNet, which uses DCNN for extracting the eigenvector of a facial image and directly uses the calculated distance of the eigenvector as a measure of face similarity. Thus, the entire DCNN does not need to be retrained when a new user is added to the system. This approach was also used in this study to extract the eigenvector of a face. Even so, the experiments of this study showed that facial features extracted using FaceNet are not evenly distributed, and using the calculated distance of the eigenvector as a measure of face similarity will yield inaccurate results, specifically if a facial image database is not trained by FaceNet. In this study, the facial features extracted using FaceNet were first normalized, and the calculated distance of the normalized eigenvector was used as the measure of face similarity. The experimental results verified that normalization can substantially increase the accuracy rate and reduce the false positive rate. In addition, Francesca et al. (2019) compared different technologies in view of the shortcomings and limitations of 2D + 3D models, especially analyzed the limitations and advantages of FER technology for traditional and deep learning (and focused on 3D solutions). The paper is the best introductory guide for future researchers.

This paper is organized as follows. Section 2 introduces the process of verifying identity by using face recognition. Section 3 describes the face recognition methods, including HOG-based face detection, FaceNet facial eigenvector extraction method, and feature comparison method. Section 4 presents the experimental results, and section 5 concludes this study.

2. IDENTITY VERIFICATION PROCESS

This section introduces the processing using face recognition to verify a person's identity. Fig. 1. illustrates the process. Once the identity verification system is activated, the system extracts images from the camera and then detects faces on the extracted images. If a face is detected, facial features are extracted; otherwise, the system returns to the image extraction step. After facial features are extracted, the features are compared. The process of facial feature comparison differs by methods. When the comparison confirms that the identified person is the user of the identity verification system, the user is logged into successfully, concluding the entire identity verification process; otherwise, the system returns to the image extraction step. The subsequent sections introduce the methods of face detection and facial feature extraction and comparison in practice.

3. METHODS

This section describes the use of HOG-based face detection, FaceNet facial eigenvector extraction method, and feature comparison method.

3.1 Face Detection Based on Histogram of Oriented Gradients

A HOG-based face detection method involves first extracting HOG-based features from input image blocks through sliding window, and then using a linear classifier to determine whether the object is a face. HOG is commonly used in computer vision and image processing to describe local features. It forms features through the histogram of local cells and blocks of static images. Fig. 2 demonstrates an example. After an image is loaded (Fig. 2(a)), the gradient of the image is first calculated (Fig. 2(b)) and then the image is divided into cells, which are small regions of equal sized $n \times n$ pixels, as shown by each square block in Fig. 2(c). Next, the histogram of these cells is calculated, and a certain number of cells are grouped into slightly larger regions, called blocks. For example, a 2×2 cell forms one block. The histogram obtained for each block is L2-normalized, and the eigenvectors containing the histograms from all of the blocks are grouped together to form HOG eigenvectors in the histogram of the image (Fig. 2(d)). Because HOG can effectively describe image structures, using HOG can quickly and effectively detect faces. However, its effect in face recognition is not as good as in face detection. In this study, face detection was completed by using the Dlib (2020) function base.

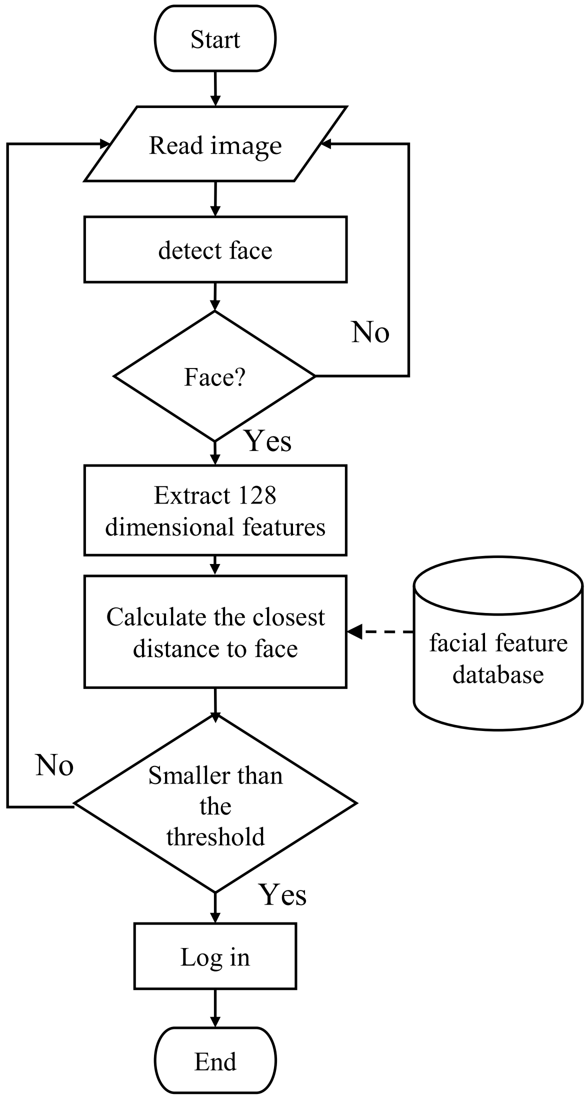


Fig. 1. Identity verification process

3.2 FaceNet Facial Feature Extraction

Developed by Google (Florian Schroff, Dmitry Kalenichenko and James Philbin, 2015), FaceNet is a face recognition technique that uses DCNN to complete feature conversion from images of a face. Fig. 3 illustrates the structure of the FaceNet model, in which an input face image I is mapped onto eigenspace $X \in R^{128}$. The most important technique in this method is the use of triplet loss in the learning of the entire system. In other words, the purpose of DCNN training is to minimize the distance between all the eigenvectors of the same person and maximize the distance between the eigenvectors of a different person. Hence, a triplet loss is composed of two different face images I_A and $I_{A'}$ both of which have the same identity and a face image I_B of a different identity. The selection criteria of $I_{A'}$ is the image of the same person with the farthest dimensional distance from I_A, while the selection criteria of I_B is the image of the different person with the closest dimensional distance from I_A. Finally, face image **I** can be mapped onto a 128-dimensional eigenvector through the DCNN of FaceNet, and then the similarity of the eigenvectors is calculated.

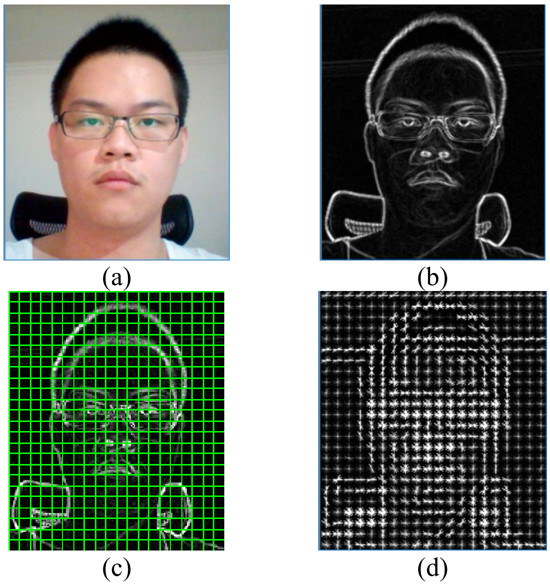


Fig. 2. Schematic of HOG feature descriptor (a) input image (b) the gradient of the image (c) The image is divided into cells, which are small regions of equal sized n×n pixels (d) HOG eigenvectors in the histogram of the image

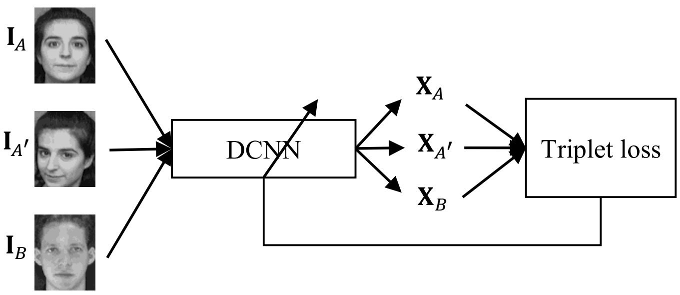


Fig. 3. The structure of the FaceNet model

Fig. 4 illustrates the concept of triplet loss. In the training process, the triplet related to the left half of Fig. 4 is searched from the database to be used as the input—that is, I_A and $\mathrm{I}_{A'}$ which have the same identity but maximum distance between the eigenvectors, and I_A and I_B which have different identity but minimum distance between the eigenvectors. Next, the model is trained so that the distance X_A and $\mathrm{X}_{A'}$ between eigenvectors extracted from I_A and $\mathrm{I}_{A'}$ can be greater than the distance X_A and X_B between eigenvectors extracted from I_A and I_B plus α. The purpose of α is to prevent the loss from becoming negative. Hence, the loss function can be defined as Equation (1):

$$Loss = [\|\mathrm{X}_A - \mathrm{X}_{A'}\| - \|\mathrm{X}_A - \mathrm{X}_B\| + \alpha] \quad (1)$$

This study hopes to minimize loss function through training. In Equation (1), α is a constant aimed at increasing the difference between $\|\mathrm{X}_A - \mathrm{X}_{A'}\|$ and $\|\mathrm{X}_A - \mathrm{X}_B\|$. After the model is trained, the eigenvector extracted from the face of the same identity will exhibit a certain level of invariability. In other words, the trained model can extract highly similar features from face images that have the same identity but are taken at different times and under different environmental and lighting conditions. Therefore, the DCNN in Fig. 3 is considered a method that can extract representative features from a face image.

Because FaceNet directly learns a face image and maps the image onto an eigenspace, it directly uses distance as a measure of face similarity. Therefore, FaceNet is a method of extracting the eigenvectors of a face and can be used in face recognition, face verification, and face clustering. The advantage of this method is its high efficiency, with each face only requiring 128-dimensional eigenvectors. FaceNet exhibits 99.63% accuracy in Labeled Faces in the Wild database, and 95.12% accuracy in YouTube Faces Database (Schroff et al., 2015).

3.3 Facial Feature Comparison

As mentioned above, FaceNet is an effective facial feature extraction method. Hence, when facial features are compared, distance is used as a measure of face similarity. Assume that a database contains the information of *N* people, each of whom has one face sample. The face images are converted into a 128-dimensional eigenvector, indicated by $\mathrm{Y}^n \in R^{128}, n = 1,2,\dots,N$. When the system uploads a face image of an unknown person and converts it into a 128-dimensional eigenvector $\mathrm{X} \in R^{128}$, the distance d^n between this eigenvector and the eigenvector of each face image in the database is calculated as follows:

$$d^n = \sum_{i=1}^{128}(x_i - y_i^n)^2, n = 1,2,\dots,N. \quad (2)$$

The face n^* with the smallest distance and smallest threshold value is extracted for final comparison and is defined as follows:

$$n^* = arg \min_{n=1,2,\dots,N} d^n \quad (3)$$

The threshold value *Th* (in Fig. 5) in the aforementioned method is 0.6, which is the optimal result obtained experimentally by Dlib (Davis King, 2017). This feature comparison method is easy and quick to use, and each person in the database requires only one sample for comparison. However, experimental verification shows that comparing the similarity of a single image is not as accurate as expected and produces considerable false positives, making it difficult to apply in practice. Therefore, the *k*-nearest neighbors (*k*-NN) algorithm was used to increase accuracy.

k-NN (*k*-nearest Neighbors Algorithm, 2021) determines the class of an unknown input by using data to determine the classes of the closest *k* points. *k*-NN is commonly used in machine learning. Assume that a comparison database contains the information of *N* people, each of whom has *M* face samples. The distance in Equation (1) is calculated for the eigenvalue $\mathrm{X} \in R^{128}$ of an unknown input and the $N \times M$ eigenvector in the database, and *k* eigenvectors with the smallest distance and smallest threshold value are extracted. Finally, the face that appeared most frequently in the *k* eigenvectors is calculated for final comparison. Fig. 5 shows the schematic of *k*-NN. Assume that *k* = 5; in this example, three out of the five closest features with distance smaller than the threshold value are triangles. Hence, the system can determine that the features of the unknown face are the triangle class.

The aforementioned method uses FaceNet to map the face image onto a 128-dimensional eigenvector and assigns a fixed threshold value. This method considers the 128-dimensional features to be equally important; however, in the experimental process, obvious dimensional changes were observed in a few of the 128-dimensional features of every person. The features extracted from the same person should have a certain level of invariability. Under this circumstance, using a fixed threshold value cannot accurately separate the range of facial features of different people. To improve this problem, the standard deviations of each dimension were normalized. Feature normalization is a method of scaling data features to a specific range so that when features of different units or magnitude are compared for similarity, the comparison result is not affected by a specific feature. In other words, feature normalization enables each feature to exert the same effect on the result. Only after data processing is completed can a fixed threshold value be used. Fig. 6 illustrates an example of 2-dimensional feature distribution (select arbitrarily from 128-dimensional, F1 and F2). The data in Fig. 6(a) are more widely distributed along the F1 axis than along the F2 axis, at which point if the threshold value is used to distinguish whether the unknown data are of the same class, the result will cover numerous areas that are not of the same class.

Cheng et al., International Journal of Applied Science and Engineering, 18(1), 2020179

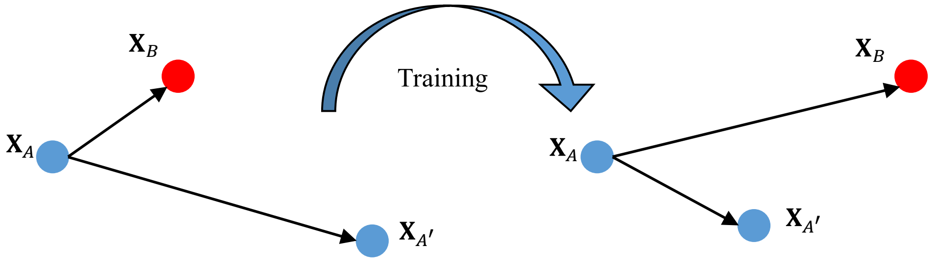


Fig. 4. The concept of triplet loss

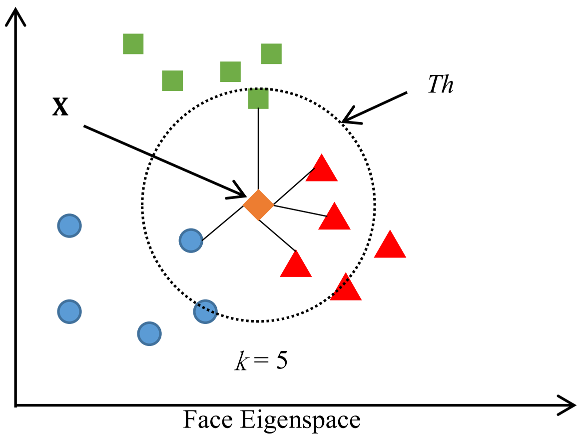


Fig. 5. Schematic comparison of facial features by using *k*-NN

Following normalization, if Fig. 6(b) uses a fixed threshold value, the covered area is obviously considerably more reasonable.

For eigenvectors of *M* face images of the same person in the database, the central value μ^n and standard deviation $\sigma^n, n = 1, \ldots, N$ are calculated, representing the average eigenvector of the person and standard deviation of each eigenvector dimension. The unknown face image is converted into 128-dimension X, which is substituted in Equation (4) to calculate the similarity d^n with the normalized features of each person in the database, which is defined below:

$$d^n = \sum_{i=1}^{128} \frac{(x_i - \mu_i^n)^2}{(\sigma_i^n)^2}, n = 1,2, \ldots, N. \quad (4)$$

Finally, the face n^* with the smallest d^n is identified. If this value is greater than the threshold value *Th*, the output image is the comparison result; otherwise, the user is not in the database.

4. EXPERIMENTAL RESULTS

For performance evaluation, accuracy (ACC) and false positive rate (FPR) are defined as follows:

$$ACC = \frac{TP+TN}{TP+TN+FP+FN}, \quad (5)$$

and

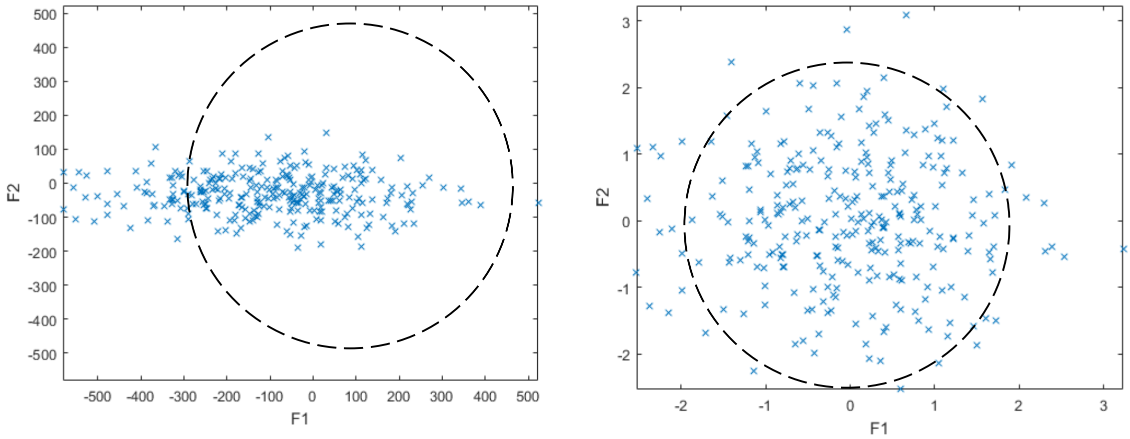
$$FPR = \frac{FP}{TN+FP}, \quad (6)$$

where *TP* (true positive) means that the tested face sample is a member of the database and the system identifies that the tested face sample is the correct member of the database; and *TN* (true negative) means that the tested face sample is not a part of the database and the system also identifies that the tested face sample is not a member of the database. These two scenarios represent that the system provided the correct judgment. *FP* (false positive) means that the tested face sample is not a member of the database but the system determines that the tested sample is a member of the database; and *FN* (false negative) means that the tested face sample is a member of the database but the system determines that the tested sample is not a member of the database.

In this study, system applicability was verified by using the ORL Database of Faces (The Database of Faces, 2001) and the face database provided by the author's research laboratory.

4.1 The ORL Database of Faces

The ORL database contains information of 40 people (as shown in Fig. 7), each of whom has 10 face images. The data were collected between 1992 and 1994 from members of the Speech, Vision and Robotics Group of the Cambridge University. These face images were taken at different times under different conditions, such as lighting, facial expression (e.g., smiling/not smiling, with eyes closed or open), and other details (e.g., with and without reading glasses). All images are front-facing portraits with a simple background. Each image is a grayscale image with 92×112 pixels.



(a) (b)

Fig. 6. Schematic of a 2-dimensional feature distribution (F1-F2) of (a) raw data and (b) normalized data

Fig. 7. The 40 people in the ORL database

Because faces could not be detected from five out of the 400 images in the ORL database (the faces were of a different person), 395 face images were used. These images were converted into 128-dimensional eigenvectors, and 20 people were randomly selected for comparison. Following the comparison method, one image or multiple images of a person were randomly selected as the control database. The remaining photos and photos of another 20 people were used as the test database. Half of the members were selected as the control database; therefore, during the test, some members were not in the control database. Thus, the test can determine whether the system recognizes the faces in the control database and confirm whether the system can exclude the faces that are not in the control database. Because the samples were randomly selected, the same experiment was performed 10 times.

Table 1 shows the result of the ORL database experiment. The second column of Table 1 represents the k-NN results when $k = 1$, that is, the result of *1*-NN with threshold value set at 0.6, which is the optimal value obtained following a series of tests by using Dlib (Davis King, 2017). As long as the euclidean distance of two features is within 0.6, the two features can be considered to be features of the same person. The experimental results showed average accuracy of 84.06%. The third column of Table 1 represents the k-NN results when $k = 5$, that is, the result of *5*-NN. In this test, five photos from each person were used as the control database. The experimental results revealed an average accuracy of 86.72%, which is slightly higher than that of the *1*-NN test. The fourth column of Table 1 represents the results of feature normalization. The experimental results revealed an average accuracy of 98.03%, indicating that the feature normalization comparison method was the most accurate. Table 2 compares the false positive rates of the ORL database experiment. False positive rate refers to the percentage of negative samples in a test set that were misjudged as positive samples. In the experiment, most of the errors of the *1*-NN and *5*-NN methods were false positives, and rarely false negatives, indicating that the system can accurately recognize the faces in the database but it cannot effectively exclude faces that are not in the database. However, the false positive rate can be effectively reduced by using the feature normalization comparison method.

Table 1. Comparison of the accuracy (ACC) of the ORL database experiment

Number of experiments	1-NN (%)	5-NN (%)	Feature normalization (%)
1	82.25	85.64	100.0
2	84.79	87.69	97.97
3	88.73	82.56	96.27
4	81.69	84.1	97.63
5	82.82	87.18	95.25
6	85.92	82.56	97.23
7	81.41	86.15	97.97
8	83.66	87.18	99.66
9	83.94	92.82	98.98
10	85.35	91.28	99.32
Avg. *ACC*	**84.06**	**86.72**	**98.03**

Table 2. Comparison of the false positive rates (FPR) of the ORL database experiment

Number of experiments	1-NN (%)	5-NN (%)	Feature normalization (%)
1	33.89	28.57	0
2	28.65	24.74	0
3	22.73	34.69	0
4	36.11	31.0	0
5	34.46	26.04	0
6	28.41	34.34	0
7	36.87	28.13	0
8	32.77	25.77	0
9	31.84	14.74	0
10	29.38	17.35	0
Avg. *FPR*	**31.51**	**26.53**	**0**

4.2 Laboratory Face Database

To verify the applicability of the proposed system, this study used the face database belonging to the Artificial Intelligence Laboratory in the Department of Computer Science and Information Engineering at Chaoyang University of Technology. The control database consisted of 10 members, as shown in Fig. 8. Ten frontal-view face images of each of the 10 members were taken at different angles (± 15º). Another 218 frontal-view face images were collected to test whether the system can exclude members that are nonexistent in the control database. In total, 318 face images were collected.

Because experimental results are directly related to the threshold setting, the settings are described below. In the ORL face database, the distance between any two eigenvectors of face images which have the same identity was calculated, and average distance was roughly 0.31. The distance between the eigenvectors of any two face images which have a different identity was calculated, and the average distance was roughly 0.65. Therefore, using the Dlib-recommended value 0.6 in the ORL database experiment as the threshold value can effectively identify whether the two images are of the same identity. In the laboratory face database, the average distance between any two eigenvectors of face images which have the same identity was 0.28, and the average distance between any two eigenvectors of face images which have a different identity was 0.49. Hence, using 0.6 as the threshold value causes serious false positive primarily because the facial contours of Easterners are not as distinctive as those of Westerners. Subsequently, the average distance between the eigenvectors of different people was less than 0.6. Based on the aforementioned discussions and after many trial and error times, a threshold value of 0.4 was used in the laboratory face database experiment. The experimental results are summarized in Tables 3 and 4, which show that the accuracy of *1*-NN and *5*-NN methods was 94.22% and 97.72%, respectively, and the false positive rate was 5.68% and 2.47%, respectively. By contrast, the accuracy and false positive rate of the feature normalization comparison method were 98.17% and 1.97%, respectively. Compared with the other two methods, the feature normalization comparison produced the best result. Table 5 shows the computing time. Because feature normalization requires more processing steps, it takes longer but is still tolerable.

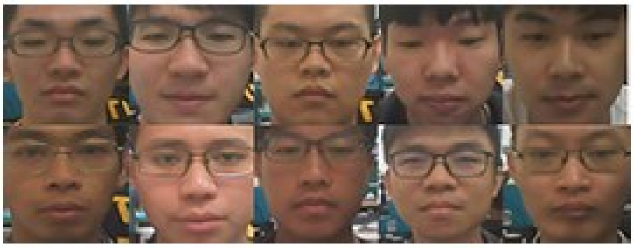


Fig. 8. The 10 users in the laboratory face database

Table 3. Comparison of the accuracy of the laboratory face database experiment (*Th* = 0.4)

Number of experiments	1-NN (%)	5-NN (%)	Feature normalization (%)
1	90.26	97.39	97.76
2	94.48	95.90	98.51
3	96.43	98.13	98.88
4	96.75	97.76	98.51
5	94.16	97.76	97.76
6	92.86	98.88	98.51
7	94.81	97.76	96.64
8	92.21	97.39	97.39
9	96.43	97.39	98.51
10	93.83	98.88	99.25
Avg. *ACC*	**94.22**	**97.72**	**98.17**

Table 4. Comparison of the false positive rate of the laboratory face database experiment (*Th* = 0.4)

Number of experiments	1-NN (%)	5-NN (%)	Feature normalization (%)
1	9.63	2.75	2.29
2	5.50	4.13	0.92
3	2.29	2.29	1.38
4	1.83	2.75	1.83
5	5.96	2.29	2.75
6	8.72	1.38	1.83
7	6.42	2.29	3.67
8	9.17	3.21	2.75
9	4.60	2.75	1.38
10	2.75	0.92	0.92
Avg. *FPR*	**5.68**	**2.47**	**1.97**

Table 5. Comparison of the computing time of the laboratory face database experiment (*Th* = 0.4)

Number of experiments	1-NN (s)	5-NN (s)	Feature normalization (s)
1	0.340	0.444	1.791
2	0.358	0.504	1.748
3	0.418	0.589	2.171
4	0.467	0.647	2.133
5	0.387	0.512	1.832
6	0.370	0.482	1.875
7	0.359	0.508	1.808
8	0.333	0.447	1.756
9	0.330	0.447	1.760
10	0.339	0.516	1.802
Avg. TIME	**0.370**	**0.509**	**1.867**

We also use a part of the Labeled Faces in the Wild (LFW) (LFW Dataset, 2018) dataset for face recognition experiments. We randomly extract face images of 10 people from the dataset, for a total of 1,434 images. Calculate the 128-dimensional distance of any pair of faces, evaluate the face verification performance on a certain range of distance thresholds (Martin Krasser, 2018), and find the optimal critical value τ. At a given threshold, all 128-dimensional vector pairs of all faces are classified as the same identity or different identities. Since there are more negative samples than positive samples, we use F1 Score (F-Score, 2021) as the optimal threshold for evaluation. Fig. 9 shows the experimental results. The horizontal axis is the distance threshold and the vertical axis is performance. From Fig. 9, it can be found that the critical value τ = 0.62 has the highest accuracy of 93.4%, which is 0.6 recommended by Dlib (Davis King, 2017). Fig. 10 shows the experimental result with normalization. The critical value τ = 156.3 still has the highest accuracy of 92.78%. From the results, it can be found that for the LFW data set, there are good results.

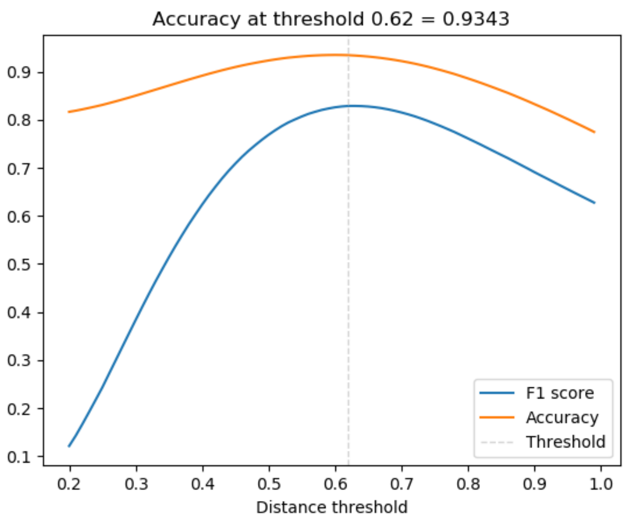


Fig. 9. The results without normalization in the LFW dataset

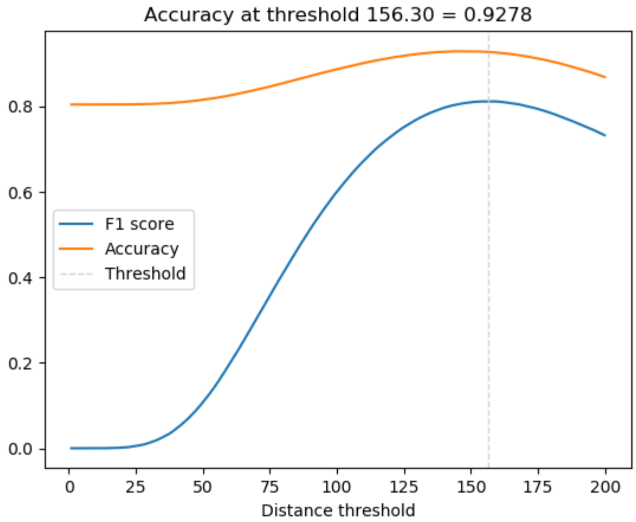


Fig. 10. The results with normalization in the LFW dataset

5. DISCUSSION

In this paper, the self-set threshold of 0.4 is the result of multiple error tests. In the future, we want to find a mathematical model to define the threshold. In addition, we only tested 2D images and did not 3D tests. In fact, 2D face recognition may be affected by environmental factors. We would like to expand the research tentacles to 3D in the future, making it more comprehensive and practical.

6. CONCLUSIONS

In this paper, an identity verification system by using face recognition was proposed. Three methods were tried: Haar-like based Cascade-Adaboost method, HOG feature combining support vector machine classifier, and deep learning Faster R-CNN face detection method, were attempted, with due consideration to speed and accuracy. Finally, the HOG feature combining SVM classifier was used. In face recognition, FaceNet was used to extract eigenvectors, and the distance between eigenvectors was used as a measure of face similarity. Finally, the *k*-NN method as well as a feature normalization comparison method was also proposed to increase the accuracy of face recognition. Experiments were conducted to test two face databases. The accuracy and false positive rate generated when the proposed method was applied to the ORL database were 98.03% and 0%, respectively. By contrast, the accuracy and false positive rate generated when the proposed method was applied to a laboratory face database were 98.17% and 1.97%, respectively. Compared with the *k*-NN method, the proposed method in this study exhibited the highest accuracy and lowest false positive rate when applied to the two face databases. The experimental results verified that the proposed identity verification system can achieve more than 98% accuracy in existing database and a laboratory database. Thus, the system developed in this study can be applied in practice.

ACKNOWLEDGMENTS

This study was supported by the campus program of Chaoyang university of science and technology (program number: 108F0021101).

REFERENCES

Cheng, W.-C., Wu, T.-Y., Li, D.-W. 2018. Ensemble convolutional neural networks for face recognition. in Proceedings of 2018 International Conference on Algorithms, Computing and Artificial Intelligence (ACAI'18). Dec 21-Dec 23. Sanya. China.

Dalal, N., Triggs, B. 2005. Histograms of oriented gradients for human detection. in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). Jun 20-Jun 25. San Diego. USA.

Davis King. Dlib C++ Library: High quality face recognition with deep metric learning, 2017. Available online: http://blog.dlib.net/2017/02/high-quality-face-recognition-with-deep.html (accessed on 01/15/2021).

Dlib C++ Library, 2020. Available online: http://dlib.net (accessed on 01/15/2021).

Francesca, N., Dagnes, N., Marcolin, F., Vezzetti, E. 2019. 3D Approaches and challenges in facial expression recognition algorithms—A literature review. Applied Sciences. 9, 3904. https://doi.org/10.3390/app9183904.

F-Score, 2021. Available online: https://en.wikipedia.org/wiki/F-score (accessed on 01/15/2021).

Graphics Processing Unit, 2021. Available online: https://en.wikipedia.org/wiki/Graphics_processing_unit (accessed on 01/15/2021).

ImageNet, 2016. Available online: http://www.image-net.org (accessed on 01/15/2021).

Jiang, H., Learned-Miller, E. 2017. Face detection with the faster R-CNN. in Proceedings of the 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition. May 31-Jun 2. Washington. USA.

K-nearest neighbors algorithm, 2021. Available online: https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm (accessed on 01/15/2021).

Labeled Faces in the Wild (LFW Dataset), 2018. Available online: http://vis-www.cs.umass.edu/lfw/ (accessed on 01/15/ 2021).

Martin Krasser. Deep face recognition with Keras, Dlib and OpenCV, 2018. Available online: https://krasserm.github.io/2018/02/07/deep-face-recognition/ (accessed on 01/15/ 2021).

OpenCV, 2021. Available online: https://opencv.org (accessed on 01/15/2021).

Ren, S., He, K., Girshick, R., Sun, J. 2017. Faster R-CNN: Towards Real-Time Object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence. 39, 1137–1149.

Schroff, F., Kalenichenko, D., Philbin, J. 2015. FaceNet: A unified embedding for face recognition and clustering. in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'15). Jun 8-Jun 10. Boston. USA.

Shu, C., Ding, X., Fang, C. 2011. Histogram of the oriented gradient for face recognition. Tsinghua Science and Technology. 16, 216–224.

Sladojevic, S., Arsenovic, M., Anderla, A., Stefanovic, D. 2017. FaceTime—Deep learning based face recognition attendance system. in Proceedings of the 2017 IEEE 15th International Symposium on Intelligent Systems and Informatics. Sep 14-Sep 16. Subotica. Serbia.

The Database of Faces - AT&T Laboratories Cambridge, 2001. Available online: https://cam-orl.co.uk/facedatabase.html (accessed on 01/15/2021).

Viola, P., Jones, M. 2001. Rapid object detection using a boosted cascade of simple features. in Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01). Dec. Kauai. USA.

Zhao, R.-Z., Fang, B., Wen, J. 2011. Face recognition using variance weightiness LBP based on single training image per person. in Proceedings of 2011 International Conference on Wavelet Analysis and Pattern Recognition. Jul 10-Jul 13. Guilin. China.