Applied Mathematics and Computation 321 (2018) 721–730

Contents lists available at ScienceDirect

Applied Mathematics and Computation

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

Fast face recognition based on fractal theory

Zhijie Tang[a], Xiaocheng Wu[a,*], Bin Fu[b], Weiwei Chen[a], Hao Feng[a]

[a] School of Mechatronic Engineering and Automation, Shanghai University, Yanchang Road, Zhabei District, Shanghai 200072, China
[b] Shanghai Charpie Intelligent Technology Co., Ltd., China

ARTICLE INFO

Keywords:
Face recognition
Fractal theory
Fractal code

ABSTRACT

Nowadays, people are more and more concerned about accuracy, rapidity and convenience in the process of personal identification. In the field of biology and computer vision, a variety of methods have been proposed, while a proper method for face recognition is still a challenge. Although some reliable systems and advanced methods have been introduced under relatively controlled conditions, their recognition rate or speed is not satisfactory in the general settings. This is especially true when there are variations in pose, illumination, and facial expression. This paper proposed a fast face recognition method based on fractal theory. This method is to compress the facial images to obtain fractal codes and complete face recognition with these codes. Experimental results on Yale, FERET and CMU PIE databases demonstrate the high efficiency of our method in runtime and correct rate.

1. Introduction

Recently, a large number of biological features have been are applied to identity recognition, such as iris recognition, fingerprint recognition, gait recognition and face recognition. These biological features are easy to use, to distinguish and difficult to forge. Compared with other methods, non touching and aggression are the biggest advantages and features of face recognition. As a hot topic, more and more attention has been focused on the face recognition. Face recognition is considered to have broad application prospects in video surveillance, access control system, criminal investigation and other fields [1–7].

General face recognition methods can be broadly divided into two categories of local and global approaches [8]. The task of those local methods is to extract different local features. For another, global approaches process the entire image and make a general template for the face [8]. It should be noted that some deep learning methods such as Convolution Neural Network (CNN) and tensor face also achieve good results.

Global approaches usually adopt a projection technique to manipulate the image as a whole and create a general template for each face pattern. The main work is to find the best template which can describe the test object. Eigenface and Fisherface are the most famous methods in this category. In the eigenface, Principle Component Analysis (PCA) is proposed and can reduce the dimension effectively. It projects images into a low-dimension space and seeks a linear transformation matrix that maximizes the data variance in the projection subspace [9]. Another linear projection is insensitive to variation in lighting direction and facial expression which is implemented by Fisher's Linear Discriminant Analysis (LDA). LDA is a supervised scheme that aims at minimizing the within-class variances as well as maximizing the between-class distances in the projection subspace [9]. However, we often meet the problems of small sample size or high dimensional data in face

* Corresponding author.
E-mail address: wxc001@foxmail.com (X. Wu).

https://doi.org/10.1016/j.amc.2017.11.017
0096-3003/

classification and recognition tasks. Therefore, the traditional LDA is not generally available for our direct use due to the fact that the within-class scatter matrix is always singular [6].

Local methods take another path. These approaches process different parts of the image to obtain salient features which are used to learn patterns of different people. Support Vector Machine (SVM) is proposed and used to classify the features extracted from a set of facial components in a component-based system. To extract local topographic representations for objects, Local Feature Analysis (LFA) was mentioned in [8].

For the deep learning methods, CNN is capable of learning local features from the input images and complete recognition. A typical CNN classifier is consisted of a CNN with altering sequence of convolution, sub-sampling layers for feature extraction and a neural network in the last layer for classification [10]. For the tensor face, it is actually used for a multidimensional array. The vector and matrix can be the representation of the first and second order tensors. Higher order tensors have more information, therefore researchers want to use this ability for face recognition and gain better results [11].

In the respect of fractal coding, Tan and Yan have made great contributions in this field. They first put forward the concept of Fractal Neighbor Distance (FND) which is a way of ranging. The definition of the degree of similarity between images which have taken fractal coding is used as the classification and identification criteria [12–14]. In [13], the speed of target recognition is analyzed using the principle of FND is associated with ultimate compression factor of Iterated Function System (IFS). Weighted Fractal Neighbor Distance (WFND) is proposed in [12]. The study finds that the regions of eyes and nose of everyone contain most features of a face and further improve the original method based on different parts of the face with different weighted coefficients.

It should be noted that when the face image is evenly distributed in the frame, it is generally symmetrical. This feature is especially effective for fractal compression, which can help us accelerate the encoding speed. After we complete the process of encoding and get the corresponding fractal codes, Fractal Neighbor Distance based Classification (FNDC) which has the direct connection to FND is presented in this paper and can meet the requirement of rapid identification. Different people are in our training library and each person has several samples under various conditions. The difference between the different samples from the same person is the within-class difference. The difference between different people is the between-classes difference. By these two differences, FNDC can ensure the recognition rate and accelerate the identification speed at the same time.

The remaining of the paper is organized as follows: Section 2 introduces the fractal theory and the steps of fractal encoding. Section 3 describes how to complete the face recognition based on fractal codes and presents the novel method FDNC. Section 4 describes the experimental results and proves the validity of FNDC. Finally, the conclusion is provided in Section 5.

2. Fractal coding theory and method

2.1. Segmentation of range blocks and domain blocks

In the whole process of image segmentation, the range blocks and domain blocks can take any shape, but generally are rectangular. And the area of the domain block is usually larger than the range block to ensure that the corresponding mapping is a contraction transformation.

The whole coding process is to find the fittest domain block for the range block and seek out the corresponding contraction mapping, contrast scaling and luminance shift. Once it is hard to find the most matched object for the current range block, we have to split it into smaller sub blocks. The above operations are repeated until the requirements are met, or the range block cannot be divided.

2.2. Determination of contractive mapping

The matching process between domain blocks and range blocks occupies most of the time of fractal encoding. There are 8 basic affine transformations for each domain block and each range block has to compare with them one by one. The matching process is shown in Fig. 1.

(1) The domain block is compressed to ensure it has the same size as the range block.
(2) The domain block after compression also has to take 8 basic affine transformations.
(3) Calculate the values of contrast scaling and luminance shift in the matching process (least squares method can be used here).

The calculation procedure of values of contrast scaling and luminance shift is introduced as follows:

Supposed $\{d_{ij}(\ \ i=1,2,\ldots,n \quad j=1,2,\ldots,n\ \ )\}$is the pixel value which is obtained by an affine transformation and $\{r_{ij}(\ \ i=1,2,\ldots,n \quad j=1,2,\ldots,n\ \ )\}$ is the pixel value of the range block, then the contrast scaling s and luminance shift o under the best conditions should make the R value in Eq. (1) minimum.

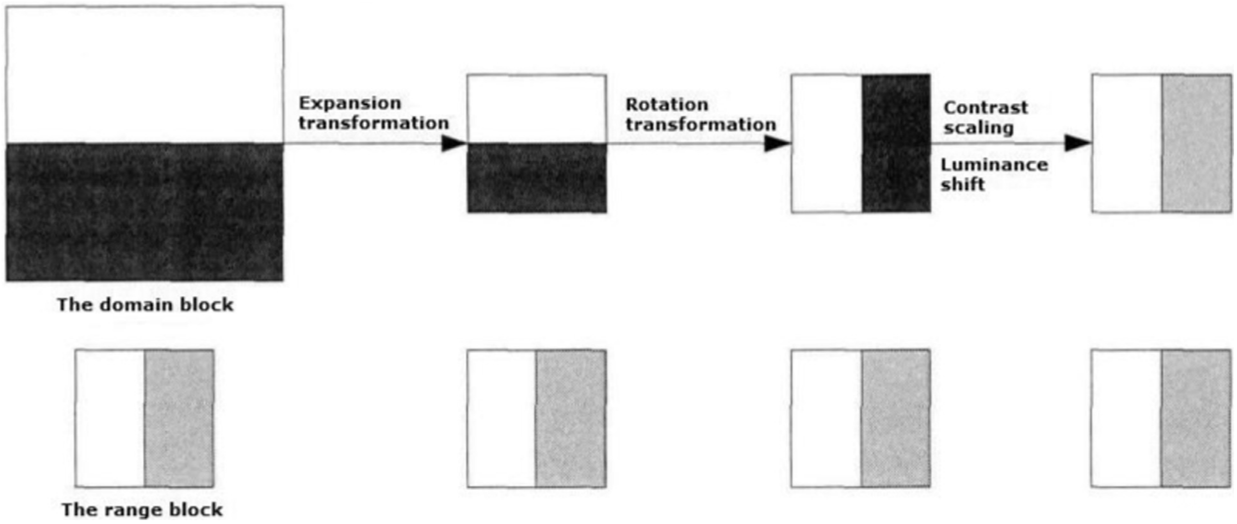

Fig. 1. The matching process between the domain and range blocks.

$$R = \sum_{i=1}^{n} \sum_{j=1}^{m} (s \cdot d_{ij} + o - r_{ij})^2 \tag{1}$$

When R gets the minimum, it can be concluded: $\frac{\partial R}{\partial s} = 0$ and $\frac{\partial R}{\partial o} = 0$.
So,

$$s = \frac{mn \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} r_{ij} - \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} \sum_{i=1}^{n} \sum_{j=1}^{m} r_{ij}}{\sum_{i=1}^{n} \sum_{j=1}^{m} {d_{ij}}^2 - (\sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij})^2} \tag{2}$$

$$o = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} r_{ij} - s \cdot \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij}}{mn} \tag{3}$$

According to Eqs. (2) and (3),

$$\begin{aligned} R = \frac{1}{mn} \Bigg[& \sum_{i=1}^{n} \sum_{j=1}^{m} r_{ij}^2 + s \left(s \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij}^2 - 2 \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} r_{ij} + 2o \sum_{i=1}^{n} \sum_{j=1}^{m} r_{ij} \right) \\ & + o \left(mno - 2 \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} \right) \Bigg] \end{aligned} \tag{4}$$

If $mn \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij}^2 - \left(\sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} \right)^2 = 0$, then $s = 0, o = \frac{1}{mn} \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij}$.

2.3. Fractal encoding

The key to realize real time recognition of fractal image is to improve the speed of fractal coding. It should be noted that the face image with a certain degree of symmetry is different from the general graphs. As is shown in Fig. 2(a,b), the matched domain block D_i of the range block R_i can be found in the symmetric region. It is very beneficial to encode and save so much time that we don't have to match all D_i with R_i. However, in practice, we need to pay attention to the fact that the symmetry center is not necessarily in the center of the image. We need to enlarge the candidate region properly like in Fig. 2(c).

If we cannot find the right domain block in the extended area S_0, we have to enlarge the search scope to S_1, S_2 or S_3 according to the current location of the range block, as shown in Fig. 2(d,e). When R_i is located in the left part, the search scope will be in the S_2. Conversely, if R_i is located in the right part, the search scope will be in the S_1. An alternative condition is that R_i is located in the middle, then the scope will be extended to S_3. Of course, there will be a situation that we cannot find the matching domain block even in S_1, S_2 and S_3. Once this happens, we will expand the search to the full image. By reducing the search range, we can save a lot of coding time.

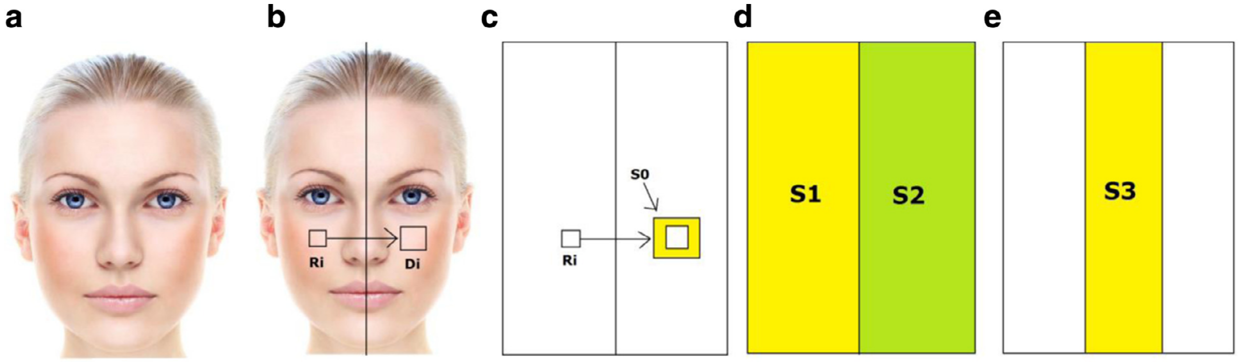

Fig. 2. Face image and the schematic diagram.

2.3. Fractal decoding

Before introducing fractal decoding, we need to give an outline of the compression mapping. For compression mapping w_i: $F \to F$, there is a s($0 < s < 1$) which makes $d_2(w_i(f), w_i(g)) \leq s \bullet d_2(f,g)$, where $d_2(.,.)$ is a measure of the function space L^2. Variable values are changed in the process of integration:

$$d_2^2(w_i(f), w_i(g)) = |s_i|^2 \det(A_i) \int_{D_i} |f(x,y) - g(x,y)|^2 dxdy \\ \leq |s_2|^2 \det(A_i) d_2^2(f,g) \tag{5}$$

In Eq. (5), $\det(A_i)$ is the determinant of matrix A_i, and s is the coefficient of contrast control.

Because w_i is a compression transformation, we can obtain:

$$|s_2|^2 |\det(A_i)| < 1 \tag{6}$$

The range block set $\{R_i\}$ represents the image of I^2 which is divided into non overlapping blocks. Namely, $I^2 = \cup R_i$ and $R_i \cap R_j = \phi$ $(i \neq j)$. The domain block set $\{D_i\}$ is slightly different from $\{D_i\}$, and it can be overlapped and each area is larger than those in $\{R_i\}$. Supposed that the mapping $\tilde{w}_i$ from $\{D_i\}$ to $\{D_i\}$ is on function space F, then

$$w_i(f)(x,y) = s_i f(\overset{-1}{w_i}(x,y)) + o_i \tag{7}$$

The appropriate s_i to ensure that the current w_i is a compressed mapping. W: $F \to F$ is defined, and

$$W(F)(x,y) = w_i(F)(x,y),\ (x,y) \in R_i \tag{8}$$

As long as the transformation set $\{w_i\}$ is selected correctly, the final result will be similar to the original image through repeated iteration of F.

3. Face recognition based on fractal theory

3.1. Fractal neighbor distance

Euclidean distance can be defined as follows: Suppose M and N are two images in Euclidean space whose height and width are H and W respectively. $M_{(i,j)}$ and $N_{(i,j)}$ are the pixel gray values of corresponding points, under the condition of $0 \leq i \leq H$ and $0 \leq j \leq W$. Then the Euclidean distance between two images can be calculated by:

$$d(M,N) = \|M - N\|_2 = \sqrt{\sum_{k=0}^{H} \sum_{l=0}^{W} \left(M_{(k,l)} - N_{(k,l)}\right)^2} \tag{9}$$

According to Section 2, we already know the encoding and decoding processes. Now, if the input image is P and the target image is Q, f_p is the fractal code of P and f_q is the fractal code of Q. Then the FND between P and Q can be defined as:

$$d_{FN}(P,Q) \equiv d(f_p(P), f_q(P)) \tag{10}$$

In Eq. (10), we decode the same image with two different fractal codes. According to Eq. (11) and Fig. 3, we can find that the final image is only related to the fractal code and independent of the input image.

$$\lim_{n \to \infty} \overset{n}{f_j}(P_i) \approx X_j \tag{11}$$

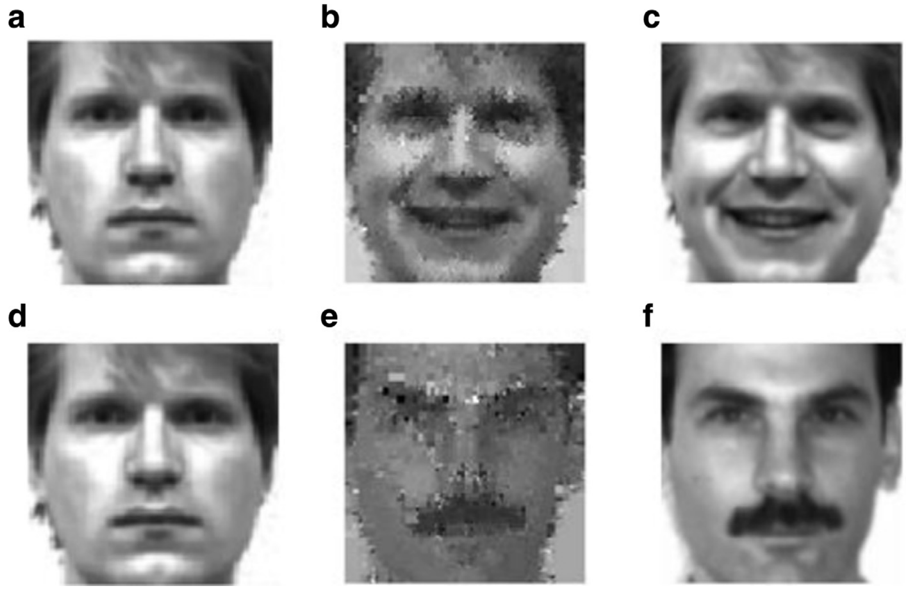

Fig. 3. Iterative process (decoding the same image with two different fractal codes).

This means that in the training library, we just need to store the fractal codes of sample images which can greatly save storage space. Taking 100 samples in Yale database as an example, the storage space is reduced by more than half.

The more similar the two pictures are, the shorter the distance is between the corresponding points. These two methods are both based on this theory. There are two main reasons why we use FND to face recognition: on the one hand, the iterative process mentioned above makes the decoding image have certain adaptability on illumination levels. And in this way, the difference between the two images (grayscale value subtracted) must be less than the value obtained directly. On the other hand, the operation of decoding iterations is actually a blurring process. The iterated image cannot be as clear as the original one. The target image recovering from its fractal code will lose some details, which is called the high frequency information. This makes the image blur to a certain extent and reduces the influence of micro-expressions and gestures on recognition.

3.2. Fractal neighbor distance based classification

When FND is used for face recognition, the input image needs to be compared with each object in the training library. There must be a suitable sample whose FND with the input image is the smallest. Then we can output the best matching image by decoding that sample's fractal code [15–17]. This method is becoming less and less suitable as the training library grows larger.

To accelerate the recognition speed, this paper presents a novel method called Fractal Neighbor Distance based Classification (FNDC). When we encode those images to construct the training library, it is important for us to classify different samples of the same person into one class. The main flow chart is shown in Fig. 4. We set up two thresholds: between-classes threshold K1 and within-classes threshold K2. The following is an example which explains how we operate.

Taking the Yale database as an example, operations are introduced below.

There are 15 people in the Yale database and everyone has 11 different gray-scale images. Five samples of each person are selected randomly for training and the rest for testing. A (a1, a2, a3, a4, a5), B (b1, b2, b3, b4, b5),..., O (o1, o2, o3, o4, o5) are in the training library. There are different classes and each class has different samples. For convenience, we take the first sample of each class as a representative of this class. Supposed that the input image is *P*,

$$\begin{aligned} T1 &= \min\{d_{FN}(P, a1), d_{FN}(P, b1), \ldots, d_{FN}(P, o1)\} \\ &= \min\{d(f_x(P), f_{a1}(P)), d(f_x(P), f_{b1}(P)), \ldots, d(f_x(P), f_{o1}(P))\} \end{aligned} \tag{12}$$

Certainly the between-classes *K1* which we set will be slightly larger than this ideal value *T1*, which is to make sure the input image can enter the matching class. If P is an image of class A, the next search will take place in A class under normal conditions. Then,

$$\begin{aligned} T2 &= \min\{d_{FN}(P, a1), d_{FN}(P, a2), \ldots, d_{FN}(P, a5)\} \\ &= \min\{d(f_x(P), f_{a1}(P)), d(f_x(P), f_{a2}(P)), \ldots, d(f_x(P), f_{a5}(P))\} \end{aligned} \tag{13}$$

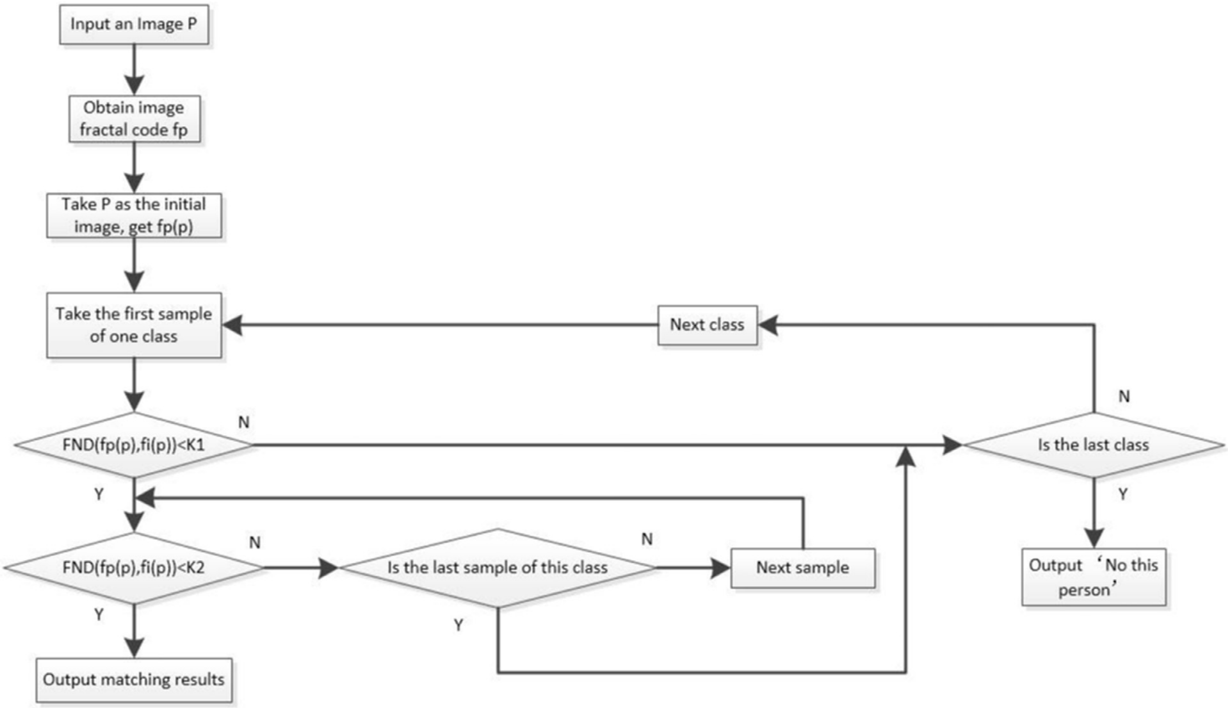

Fig. 4. The flow chart of face recognition.

Similarly, within-classes threshold *K2* which we set will be slightly larger than the ideal value too. *K2* is used to prevent us from missing the correct class. Repeating the above operation on FERET and CMU PIE databases, and then according to Fig. 5, we set $K1 = T1 + 1.3$ and $K2 = T2 + 0.4$.

We use *K1* to perform a fuzzy operation which can help us find the corresponding class and reduce the narrow range effectively. If there is more than one class, then we can find the suitable object by *K2*. The whole process uses the idea of classification to reduce recognition time.

4. Experiment results

In this section, FNDC proposed in this paper is estimated on three databases. For comparison, some popular feature extraction methods such as PCA, LDA, CNN and Tensor and some methods related to fractal coding such as FND and WFND were used in our experiments. All methods in this paper were validated by computer simulation using Matlab software on an Intel AMD A6-3420 M CPU 1.5 GHz machine with 4 G RAM.

4.1. On the Yale database

The Yale face database is created by computational vision and control center of Yale University, includes 165 images of the 15 volunteers, and contains influence factors like light, facial expression and postures. For convenience, we trimmed each image to 100×80 pixels. All samples of one person are shown in Fig. 6 [18].

There are different sample sizes (three, four, five and six samples of each person) to train and the rest to test. We ran the whole system 10 times. The average recognition rate and the average running time of each method are shown in Table 1. According to the table, it is easy to find that the recognition rate is higher and higher with the increasing number of samples. At the same time, the recognition time is getting longer and longer. The recognition rates of each method are similar while FNDC proposed in this paper is hardly affected by the number of training samples.

4.2. On the FERET database

In order to promote the research and practical application of face recognition algorithm, the US Department of Defense's Counterdrug Technology Transfer Program (CTTP) launched a Face Recognition Technology (FERET) engineering, including a general face library and universal test standards. This database involves variations in facial expression, illumination, gesture and age. We performed our method on a subset of the FERET database. The subset is consisted of 200 people, and each one has seven images. All samples of one person are shown in Fig. 7 [18].

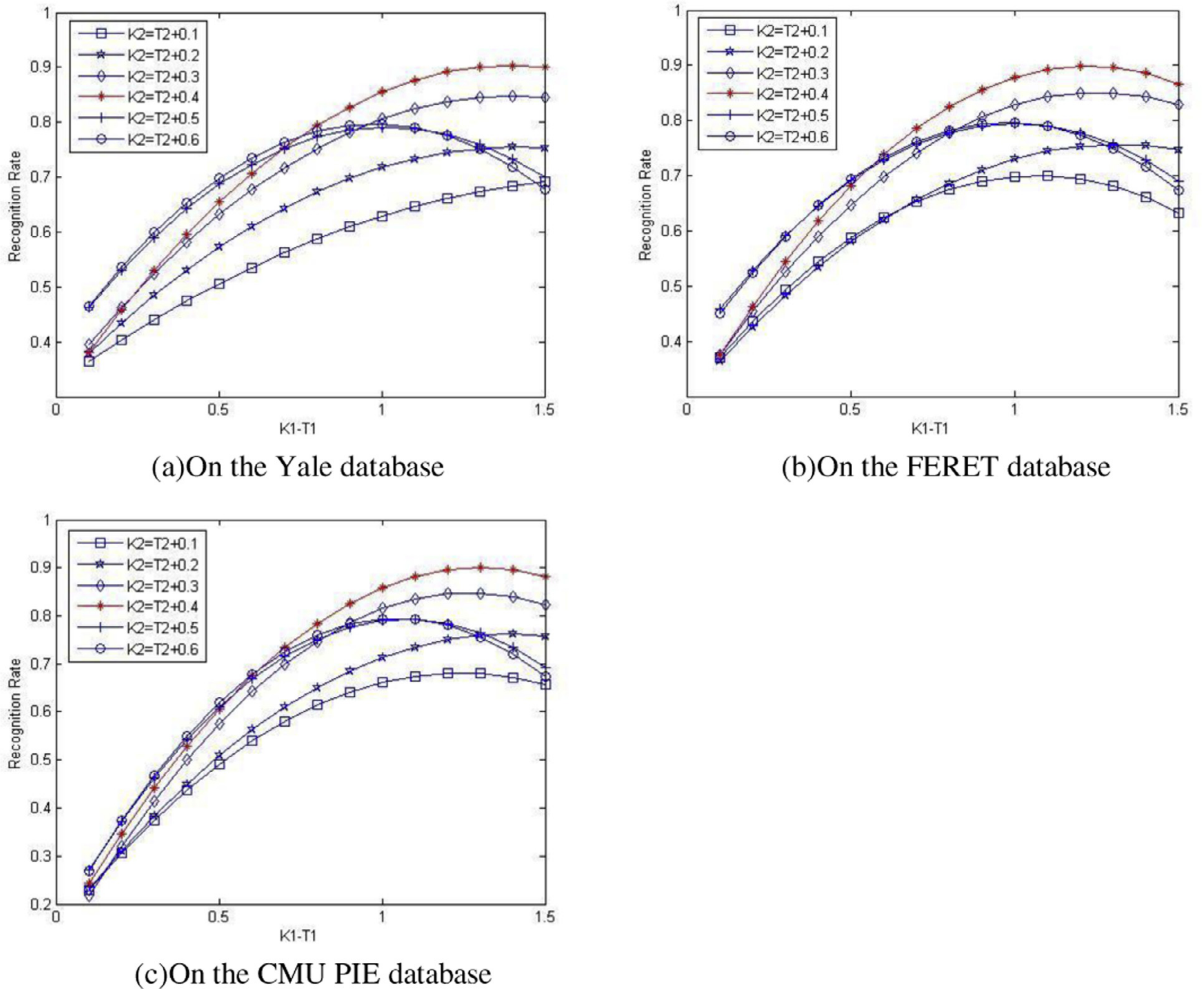

Fig. 5. The influence of $K1$ and $K2$ on the recognition rate.

Fig. 6. Samples on the Yale database.

We randomly selected three, four and five images of everyone to train and the rest to test. The whole system was run 10 times. Table 2 lists the average recognition rate and the average running time of the seven methods.

FERET database is larger than the Yale database. PCA and LDA cannot perform as well as in previous experiment. The recognition rate of the remaining five methods is similar. Although recognition time used in CNN and Tensor is superior to FND and WFND, FNDC is almost unaffected by sample size like the experiments we did on Yale database. The overall performance of FNDC is very prominent.

Table 1
Rates (%) and time (s) of each method on the Yale database.

	Three training		Four training		Five training		Six training	
	Rate	Time	Rate	Time	Rate	Time	Rate	Time
PCA	86.1	**0.088**	88.2	0.106	88.3	**0.134**	89.7	0.185
LDA	86.8	0.092	88.7	**0.092**	89.1	0.138	91.2	**0.151**
CNN	87.5	0.132	88.9	0.159	90.3	0.179	92.3	0.187
Tensor	86.9	0.112	88.8	0.135	90.1	0.167	92.4	0.192
FND	86.7	0.312	89.1	0.411	90.6	0.502	93.9	0.621
WFND	86.4	0.302	89.0	0.385	90.5	0.434	92.8	0.492
FNDC	86.4	0.185	88.7	0.197	90.5	0.201	93.6	0.224

Fig. 7. Sample images of one person on the FERET database.

Table 2
Rates (%) and time (s) of each method on the FERET database.

	Three training		Four training		Five training	
	Rate	Time	Rate	Time	Rate	Time
PCA	41.3	3.61	56.4	9.22	62.1	13.11
LDA	43.2	2.77	62.2	4.32	66.8	6.79
CNN	42.5	2.11	63.1	2.57	71.8	3.53
Tensor	42.9	2.35	63.8	2.96	71.7	3.75
FND	42.6	3.61	64.5	4.76	72.7	6.21
WFND	42.8	3.54	63.9	4.22	71.1	6.03
FNDC	42.5	**1.57**	64.2	**1.68**	72.3	**1.83**

Fig. 8. Sample images of one person on the CMU PIE database.

4.3. On the CMU PIE database

The CMU PIE face database includes 68 subjects with 41,368 face images as a whole. Each subject contains 13 different poses, 43 different illumination conditions, and 4 different expressions. In our experiment, 60 face images of each individual were used and all of them were transformed to gray scale. Partial samples of one person are shown in Fig. 8.

Ten, fifteen and twenty images of everyone were randomly selected to train and the rest to test. The whole system was run 10 times. Table 3 lists the average recognition rate and the average running time of the seven methods.

The database is further expanded and the training sample library is further enlarged at the same time. Those general methods like PCA and LDA and deep learning methods like CNN and Tensor, which are composed of four stages: simple preprocessing, face detection, feature extraction and use the classifiers to recognize. The experiment result is similar to what we did on the FERET database. FNDC using class information took the least amount of time and its recognition rate

Table 3
Rates (%) and time (s) of each method on the CMU PIE database.

	Ten training		Fifteen training		Twenty training	
	Rate	Time	Rate	Time	Rate	Time
PCA	50.5	3.77	66.2	10.22	78.1	15.25
LDA	54.2	2.87	68.3	5.62	81.6	7.87
CNN	55.1	2.12	71.5	3.23	85.3	4.12
Tensor	55.2	2.45	70.4	3.54	84.8	4.02
FND	56.1	3.69	68.9	4.82	85.6	6.94
WFND	55.8	3.25	67.6	4.33	84.5	6.11
FNDC	54.0	**1.62**	67.2	**1.72**	84.9	**1.95**

was very close to the highest one. In the comprehensive performance, especially on the respect of identification speed, FNDC is far superior to other methods.

4.4. Discussion

From the above experiments, we have the following findings:

(1) Sample images on the three databases involve variations in facial expression, pose and illumination, while the proposed method is still able to perform well. It owes to fractal iteration is a blurring process.
(2) When the number of training samples is large like on FERET and CMU PIE databases, FNDC using class information takes the least amount of time and obtain a high recognition rate.
(3) When compression technology is applied to face recognition as a compression technology, we just need to obtain and store fractal codes of training samples, which can help us greatly save storage space. And it has been the key point in face recognition when the size of the database is becoming increasingly large.

5. Conclusions

In this paper, a fast fractal coding method is firstly introduced. The operation of decoding iterations is actually a blurring process which can omit the high frequency information and overcome the effects of illumination, occlusion and micro expression on the experimental results. Just as important, when the fractal coding method is applied to face recognition as a compression technology, we just need to obtain and store fractal codes of training samples, which can help us greatly save storage space. To meet people's requirements of fast face recognition, we propose FNDC which improves the traditional fractal recognition methods using class information to set up between-classes and within-classes thresholds to accelerate the recognition speed. We conducted experiments on Yale, FERET and CMU PIE databases and demonstrate the merits of FNDC.

Acknowledgments

The authors would like to acknowledge the National Natural Science Foundation of China (No. 51005142), Innovation Program of Shanghai Municipal Education Commission (No. 14YZ010) and Shanghai Natural Science Foundation (No. 14ZR1414900) for providing financial support for this work.

References

[1] Q. Gao, L. Zhang, D. Zhang, Face recognition using FLDA with single training image per person, Appl. Math. Comput. 205 (2) (2008) 726–734.
[2] M. Ko, A. Barkana, A new solution to one sample problem in face recognition using FLDA, Appl. Math. Comput. 217 (24) (2011) 10368–10376.
[3] J.Q. Gao, L.a. Fan, L.Z. Xu, Median null(Sw)-based method for face feature recognition, Appl. Math. Comput. 219 (12) (2013) 6410–6419.
[4] X. Li, S. Fei, T. Zhang, Median MSD-based method for face recognition, Neurocomputing 72 (16–18) (2009) 3930–3934.
[5] J. Yang, D. Zhang, J.Y. Yang, Median LDA: a robust feature extraction method for face recognition, in: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, 5, IEEE, 2006, pp. 4208–4213.
[6] L. Li, J. Gao, H. Ge, A new face recognition method via semi-discrete decomposition for one sample problem, Optik 127 (19) (2016) 7408–7417.
[7] L. Li, H. Ge, J. Gao, Maximum-minimum-median average MSD-based approach for face recognition, AEU - Int. J. Electron. Commun. 70 (7) (2016) 920–927.
[8] A. Rikhtegar, M. Pooyan, M.T. Manzuri-Shalmani, Genetic algorithm-optimised structure of convolutional neural network for face recognition applications, IET Comput. Vis. 10 (6) (2016) 559–566.
[9] Y. Liu, D.A. Pados, C.H. Yeh, Two-stage tensor locality-preserving projection face recognition, in: Proceedings of the IEEE 2nd International Conference on Multimedia Big Data, BigMM 2016, 2016, pp. 214–218.
[10] N.P. Ramaiah, E.P. Ijjina, C.K. Mohan, Illumination invariant face recognition using convolutional neural networks, in: Proceedings of the IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems, SPICES 2015, 2015.
[11] Y. Norouzi, M. Kaffashpour-Yazdi, S. Araghi, et al., Fusion of genetic algorithm with tensor based algorithms for face recognition, in: Proceedings of the Iranian Conference on Machine Vision and Image Processing, MVIP, 2016, pp. 96–99.
[12] T. Tan, H. Yan, Face recognition using the weighted fractal neighbor distance, IEEE Trans. Syst. Man, Cybern. 35 (4) (2005) 576–582.
[13] T. Tan, H. Yan, Object recognition based on fractal neighbor distance, Signal Process. 81 (2001) 2105–2129.

[14] T. Tan, H. Yan, The fractal neighbor distance measure, Pattern Recognit. 35 (2002) 1371–1387.
[15] G. Ji, H. Zheng, Z. Liu, X. Ren, Cell image matching based on fractal neighbor distance, Int. Conf. Natural Comput. 4 (2008) 609–612.
[16] Jianji Wang, Nanning Zheng, A novel fractal image compression scheme with block classification and sorting based on Pearson's correlation coefficient, IEEE Trans. Image Process. 22 (9) (2013) 3690–3702.
[17] A.V. Nikonorov, E.Y. Minaev, Fractal recognition of compact artifacts on color images, Br. J. Haematol. 164 (1) (2014) 39–52.
[18] J. Yin, L. Wei, M. Song, W. Zeng, Optimized projection for collaborative representation based classification and its applications to face recognition, Pattern Recognit. Lett. 73 (2016) 83–90.