A Face Recognition Algorithm Based on LBP-EHMM

Tao Li[1], Lingyun Wang[1], Yin Chen[1, *], Yongjun Ren[1], Lei Wang[1] and Jinyue Xia[2]

Abstract: In order to solve the problem that real-time face recognition is susceptible to illumination changes, this paper proposes a face recognition method that combines Local Binary Patterns (LBP) and Embedded Hidden Markov Model (EHMM). Face recognition method. The method firstly performs LBP preprocessing on the input face image, then extracts the feature vector, and finally sends the extracted feature observation vector to the EHMM for training or recognition. Experiments on multiple face databases show that the proposed algorithm is robust to illumination and improves recognition rate.

Keywords: Face recognition, LBP, feature extraction, EHMM.

1 Introduction

Face recognition is a hot topic in the field of pattern recognition and artificial intelligence. Traditional methods mostly use template matching based on feature extraction methods such as principal component analysis and linear discriminant analysis to realize face recognition. Principal component analysis (PCA) uses K-L transform to extract the main components of the face and form the feature face space for processing [Martnez and Kak (2004); Turk and Pentland (1991)]. Linear discriminant analysis (LDA) is to project the original data into a transformation space by maximizing the objective function and divide the original data in this new space [Howland and Park (2004)]. They describe the face image from a global perspective and can extract the global features of the image better, but they are easy to be affected by illumination and position. After years of development, face recognition technology has made considerable progress, and has a high recognition rate under ideal conditions. However, due to the complexity of face recognition and the multiple effects of objective conditions, there are still many problems not well solved in real-time recognition, such as illumination changes, expression changes, occlusion and so on, in which the changes in lighting caused the most serious interference.

Compared with the traditional method, the embedded hidden Markov model (EHMM) can efficiently represent the dynamic time series signal, and the specific structure of the face is consistent with the state sequence of the Markov chain in EHMM [Dong (2009)]. Since the feature observation vector extracted by DCT in EHMM is relatively sensitive to

---

[1] College of Electronic and Information Engineering, Nanjing University of Information Science and Technology, Nanjing, 210044, China.

[2] International Business Machines Corporation (IBM), New York, USA.

[*] Corresponding Author: Yin Chen. Email: cy_nuist@foxmail.com.

illumination changes, LBP is used as image preprocessing to reduce the influence of illumination change [Viola and Jones (2001); Wei (2010); Xiao, Qiang, Song et al. (2012)]. The experimental results show that the proposed method reduces the influence of illumination changes while improving the face recognition rate.

2 Preliminary

2.1 LBP

Ojala proposed in document [Ojala and Pietikainen (1996)] that the local texture feature of the image is represented by the relationship between the gray values of the local pixels. The basic idea of the LBP operator is to take the gray values of the central pixels as the threshold and compare the gray values of each pixel in the circular neighborhood with the thresholds to generate the binary pattern of the region to represent the local texture features [Ojala, Pietikainen and Maenpaa (2002); Xia, Yuan, Lv et al. (2019); Xia, Ma, Shen et al. (2018)]. For any LBP operator, its formula is:

$$LBP_{P,R} = \sum_{i=0}^{P-1} s(g_i - g_c)2^i, s(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}$$
(1)

In the formula, $g_i (i = 0, ..., P - 1)$ represents P pixels with the center pixel g_c as the center and R as the circumference of the radius. For different (P, R) values, there are different LBP operators, and graph 1 is 3 different LBP operators.
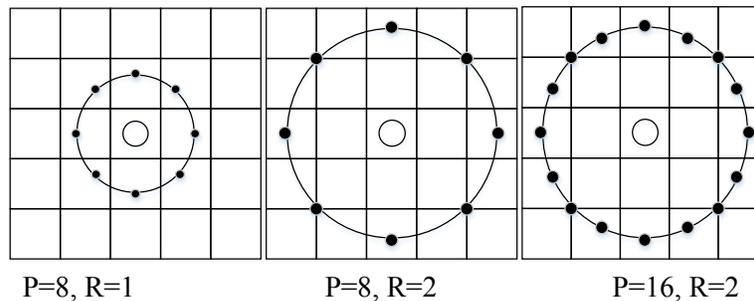

P=8, R=1   P=8, R=2   P=16, R=2

Figure 1: LBP operator corresponding to different P and R values

The $LBP_{P,R}$ operator has gray-scale invariance. As long as the illumination does not change the order of the local pixel values, the value obtained by the $LBP_{P,R}$ operator is also the same. Fig. 2 shows the solution process of $LBP_{8,1}$.
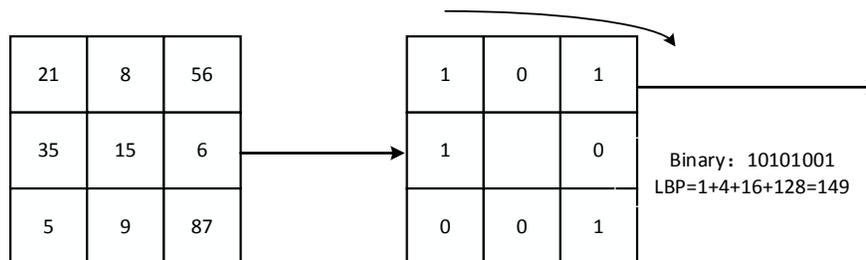

Binary: 10101001
LBP=1+4+16+128=149

Figure 2: $LBR_{8,1}$ operator

An M×N image with M×N eigenvalues, each of which is one of 2^P integer values. The $LBP_{P,R}$ feature reflects the local texture features of the image, and the global texture features of the entire image can also be obtained by $LBP_{P,R}$ [Jin, Wang, Ji et al. (2018); Xia, Neal, Athanasios et al. (2017)]. Fig. 3 is a comparison of the image before and after LBP processing.

2.2 Hidden Markov Model (HMM)

HMM is a double stochastic process describing time-varying signals by means of probability and statistics. One of them is the Markov chain, that is, the Markov process with discrete state and time. This is a basic stochastic process. Another random process is to describe the statistical correspondence between the state and the observed value. The first order discrete hidden Markov process generally includes 4 aspects: ①N, the number of implied states; ②M, the total number of different observation symbols; ③A, state transition probability distribution or transition matrix; ④B, observation probability matrix or called emission matrix. Using a shorthand notation, the HMM can be represented as the following three-parameter form λ=(A,B,π). Samaria first applied HMM to face recognition in the literature [Samaria (1995)] and proposed the HMM model. It is based on the fact that a frontal face image contains six distinct feature areas: hair, forehead, eyes, nose, mouth and chin. Even if the head is deflected or tilted, the order of them remains constant from top to bottom [Qiang (2008)]. Therefore, the six feature regions can be abstracted into six states, and the observed sequence is generated by the six states. The six states are abstract and have no specific meaning and can only be estimated by observing the sequence. Its state structure diagram is shown in Fig. 3.
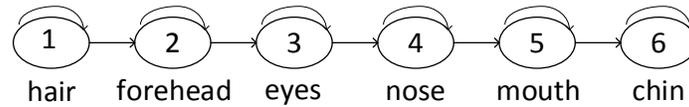

Figure 3: One-dimensional HMM state structure diagram

2.3 EHMM

One-dimensional HMM is good at one-dimensional signal processing, so it has made great breakthroughs in the field of speech. But for image processing and recognition, the image is a two-dimensional signal. Although it can be expanded into one-dimensional data like a feature face, the expanded one-dimensional data has lost spatial information. In the later calculation, there is a large amount of computation and is not suitable for real-time applications. Therefore, in the real-time face image recognition system, the standard one-dimensional HMM method is difficult to meet the requirements, and another two-dimensional model needs to be searched to model the image to achieve high-efficiency and high-precision recognition.

EHMM was first used by Nefian et al. [Nefian and Hayes (1999)] for face recognition and achieved good results. EHMM is based on a one-dimensional HMM. It is obtained by nesting a new HMM in each state of a one-dimensional HMM. Generally, the state of the outer HMM is called a super state, and the inner embedded HMM is called an embedded state. It can be

represented by an extended triplet $\lambda = \{\Pi_0, A_0, A\}$. The relevant parameters are briefly introduced as follows:

1) $\Pi_0 = \{\pi_{0,i}\}$, $1 \leq i \leq N_0$, initial distribution of super state, where $\pi_{0,i}$ is the superstate i probability and N_0 represents the number of superstates.

2) $A_0 = \{a_{0,ij}\}$. $1 \leq i, j \leq N_0$, the state transition probability matrix of the super state, where $a_{0,ij}$ is the transition probability from super state i to j.

3) $\Lambda = \{\Lambda^1, \ldots \Lambda^k, \ldots \Lambda^{N_0}\}$, the HMM description is embedded, where Λ^k describes the corresponding kth HMM.

4) $\Lambda^k = \{\Pi_l^k, A_l^k. B^k\}$ includes an initial state distribution Π_l^k of the kth embedded HMM, a state transition probability matrix A_l^k and a state probability matrix $B^k = \{b_t^k(O_{t_0,t_1})\}$, and O_{t_0,t_1} represents an observation sequence located in the t_1th row of the t_0th row.

The state structure description of EHMM in the face recognition system is shown in Fig. 4. The six states of the hair, the eyes, and the like in the vertical direction are super-states, and the plurality of groups of HMMs embedded in the horizontal direction are referred to as an embedded state. EHMM can reflect the two-dimensional structural features of the face in the model, and the state transition in the horizontal direction is limited to the inside of the super state. It is simpler than the general two-dimensional HMM. It is a better model for describing and recognizing the face.
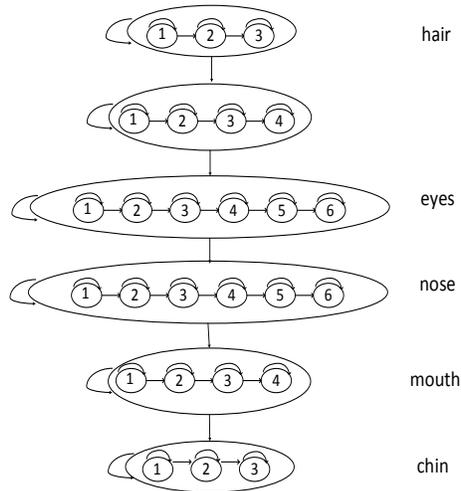

Figure 4: EHMM state structure diagram

EHMM training

The training of the embedded hidden Markov model is to determine a set of optimized EHMM parameters for each individual. Each model can be trained in single or multiple images. The specific steps are as follows:

1) The DCT transform parameters are initialized, the face image is sampled, and each sample window DCT transform is calculated to obtain a DCT coefficient matrix.

2)   A general EHMM model $\lambda = \{\Pi_0, A_0, \Lambda\}$ is established, and the number of model states and the number of possible observations corresponding to each state are determined, and the DCT transform coefficient vector is input as an observation sequence into the EHMM.

3)   The training data is evenly divided, corresponding to N states, and the initial parameters of the model are calculated.

4)   Using the one-dimensional Viterbi algorithm to adjust the division of the kth super state embedded HMM on this input sequence, so that the $P(O|\lambda^{(k)})$ output is the largest, and the output of $P(O|\lambda^{(k)})$ is taken as the observation probability of $\lambda^{(0)}$. After all the observation sequences are calculated, the Viterbi algorithm is implemented on $\lambda^{(0)}$, so that the output of $P(O|\lambda^{(k)})$ is the largest.

5)   The Baum-Welch algorithm is used to re-estimate the parameters to iteratively adjust the model parameters until $\lambda^{(0)}$ converges, save the EHMM and end the training. When the training is over, an EHMM model is built into the face database. The specific training process is shown in Fig. 5.
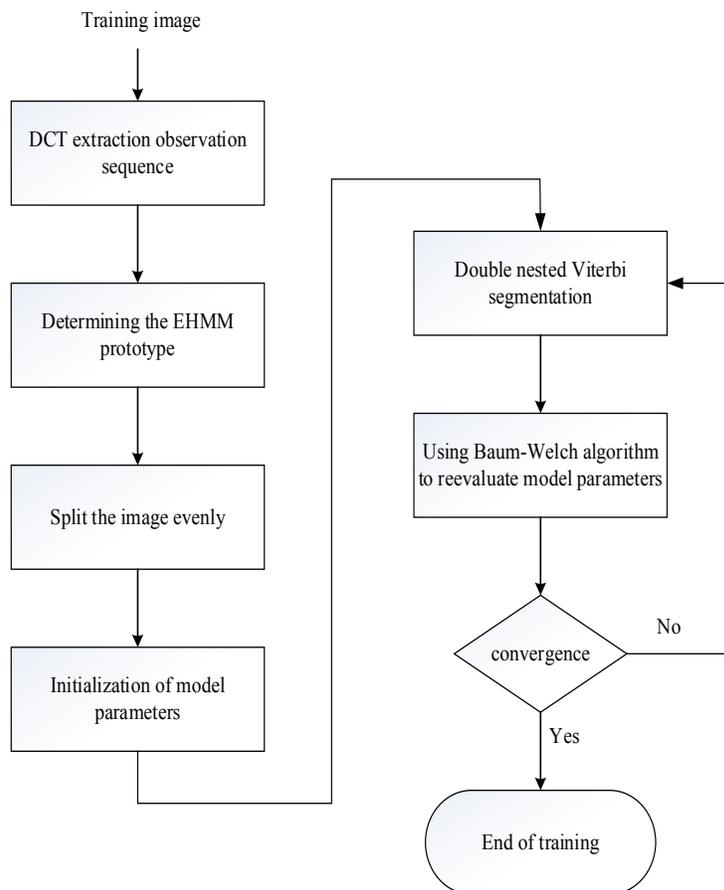

Figure 5: EHMM training process

3 LBP-EHMM-based face recognition algorithm

The real-time face recognition based on LBP-EHMM can be basically divided into four stages: face detection, LBP preprocessing, DCT feature extraction, EHMM training or recognition.

3.1 The Adaboost algorithm

Firstly, the input image is subjected to real-time accurate face detection to segment the face region.

The Adaboost algorithm updates the weight based on whether the samples are correctly classified in each training and the correct rate of the last overall classification. The updated new data set is sent to the next level classifier for training, and finally the classification obtained by each training is cascaded into a strong decision classifier. The device cascades into a strong decision classifier. The algorithm itself is realized by changing the data distribution. The final strong classifier is constructed by cascading different weak classifiers to realize the face detection. Using the classifier can eliminate some unnecessary training data features and focus on the key training data, greatly improving the speed and accuracy of face detection.

3.2 LBP pretreatment

Then LBP preprocessing is performed to reduce the influence of illumination changes.

When the LBP operator is used to extract facial texture features, the statistical histogram of the pattern is usually used to describe the image information. However, too many pattern types may cause the histogram to be too sparse, so it is necessary to effectively reduce the dimensions of the original mode so that the minimum amount of data can represent the information of the image best.

3.3 DCT feature extraction

Then, the appropriate window and moving step are set to perform DCT feature extraction on the face region.

DCT is an orthogonal transform method that has been considered as the best optimized transform method for signal processing of speech and images. A two-dimensional transformation of an image separates the high frequency information of the image from the low frequency information. After the image is transformed, most of the coefficients are concentrated in the upper left corner of the image. The coefficients in the upper left corner are the low-frequency information of the face image, which represents the overall feature of the face image. Selecting the coefficient of this part as a feature can achieve the effect of dimension reduction.

The characteristics of DCT extraction focus on the overall attributes of the face, and often lose some of the detailed feature attributes. This part of the detail features often contain important facial features, while the LBP method based on local texture features can can extract the local details of the face better, which makes up for the deficiency of DCT extraction.

3.4 EHMM training

Finally, a set of feature observation vectors is sent to EHMM modeling and the new face image is identified by EHMM. Each group of face images can get an EHMM model. A set of feature observation vectors is obtained in the same way for the face images to be recognized, then the probability of each EHMM model is calculated and the maximum probability is taken as the recognition result [Shen, Yuan, Shen et al. (2018)]. Fig. 6 is a system frame diagram.
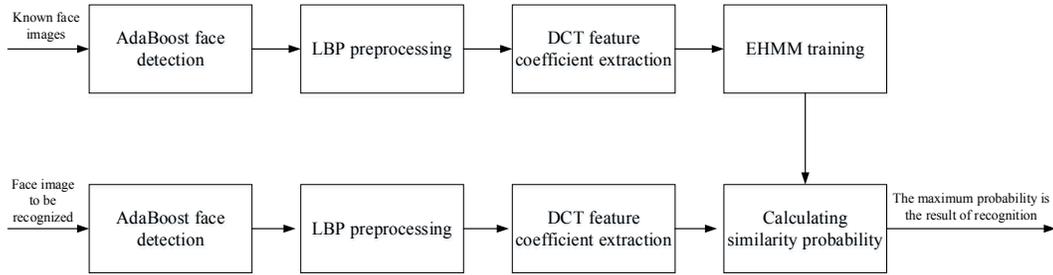

Figure 6: LBP-EHMM real-time face recognition system framework

4 Experiment and result analysis

4.1 Face database experiment

In order to prove the effectiveness of the proposed algorithm, experiments were performed on the ORL, YALE and self-made face databases respectively. The ORL face database consists of 40 people, each of whom is composed of 10 images corresponding to different illumination, posture, expression and decoration. Take the first 5 images of each person, a total of 200 for training, and another 200 for identification. The YALE face database contains 165 images of 15 volunteers, including changes in lighting, expressions, and gestures. Take the first 5 images of each person, a total of 75 for training, and another 90 for identification. The self-made face database contains 20 people, 10 images per person, for a total of 200. Take the first 5 images of each person, a total of 100 images under normal lighting for training, and 100 images with light treatment to identify them. Fig. 7 is part of the self-made face image. The size of the sampling window is: 16×16, The moving step size is: 4×4. The experimental results are shown in Tab. 1.

Table 1: Face database experimental data

Sample library	EHMM accuracy rate	LBP-EHMM Accuracy rate
ORL	96.5%	**98%**
YALE	95.6%	**96.7%**
Self-made	86%	**94%**

Figure 7: Partial images of self-made face database

4.2 Comparison and analysis of experimental results

By comparing the ORL face database and the YALE face database experiment, it can be seen that: The recognition rates of the two algorithms on the ORL face database and the YALE face database are basically the same. Because the illumination of the face image in ORL database changes little, and the illumination changes relatively large in YALE database, the recognition rate of YALE database has decreased. In order to reduce the recognition rate caused by illumination changes, a face database is created and some of the face images are illuminated. From the experimental results, we can see that in the self-made face database, the recognition rate of LBP-EHMM is obviously higher than that of EHMM. The reason is that some of the light in the face image used by the self-made face database is very dark, so that the EHMM misidentification is basically those with dark light. Since the feature observation vector extracted by DCT in EHMM is relatively sensitive to illumination changes, LBP pre-processing is performed before feature extraction to reduce the influence of illumination changes. The results of comprehensive comparison experiments show that the proposed method is robust to illumination changes while maintaining a high recognition rate.

5 Conclusion

This paper proposes a real-time face recognition method based on LBP-EHMM. The method firstly performs face detection on the acquired image, and then performs LBP preprocessing on the detected face region to weaken the influence of illumination, and then performs DCT feature coefficient extraction, and finally sent to EHMM for training and recognition. Based on EHMM, the dynamic sequence can be well modeled. LBP is simple in calculation and has gray invariance, which can describe local texture more effectively. The method of this paper combines the advantages of both. The experimental results show that the recognition performance of this method is better than that of single EHMM, especially when the image is dark. In practical applications, with the rapid development of modern computer technology, the processing of more complex algorithms is also very fast, so the choice of LBP-EHMM method has more advantages.

References

Dong, Z. (2009): *Research and Implementation of Face Recognition Based on HMM*. Beijing: Beijing University of Chemical Technology.

Howland, P.; Park, H. (2004): Generalizing discriminant analysis using the generalized singular value decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 8, pp. 995-1006.

Jin, M.; Wang, T.; Ji, Z.; Shen, X. (2018): Perceptual gradient similarity deviation for full reference image quality assessment. *Computers, Materials & Continua*, vol. 56, no. 3, pp. 501-515.

Martnez, A. M.; Kak, A. C. (2004): PCA *vs*. LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 228-233.

Nefian, A. V.; Hayes, M. H. (1999): Face recognition using an embedded HMM. *IEEE Conference on Audio and Video-Based Biometric Person Authentication*, pp. 19-24.

Ojala, T.; Pietikainen, M.; Harwood, D. (1996): A comparative study of texture measures with classification based on feature distributions. *Pattern Recognition*, vol. 29, no. 1, pp. 51-59.

Ojala, T.; Pietikainen, M.; Maenpaa, T. (2002): Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971-987.

Qiang, J. C. (2008): *Research on Facial Expression Recognition Based on Statistical Methods*. Nanjing: Southeast University.

Samaria, F. (1995): *Face Recognition Using Hidden Markov Models* (*Ph. D. Thesis)*, vol. 1, no. 1, pp. 28-34. University of Cambridge.

Shen, X.; Yuan, Y. H.; Shen, F.; Xu, Y.; Sun, Q. S. (2018): A novel multi-view dimensionality reduction and recognition framework with applications to face recognition. *Journal of Visual Communication and Image Representation*, vol. 53, pp. 161-170.

Turk, M.; Pentland, A. (1991): Eigenfaces for recognition. *Journal of Cognitive Neurosicence*, vol. 3, no. 1, pp. 71-86.

Viola, P.; Jones, M. (2001): Robust real-time object detection. *IEEE ICCV Workshop on Statistical and Computational Theories of Vision*.

Wei, G. (2010): *Research on Key Problems of Automatic Face Recognition*. Changchun: Changchun Institute of Optical Precision Machinery and Physics, Chinese Academy of Sciences.

Xiao, W.; Qiang, X. G.; Song, W.; Kuan, L. (2012): Face recognition under complex illumination based on LBP histogram. *Computer Engineering and Applications*, vol. 48, no. 15, pp. 181-186.

Xia, Z. H.; Yuan, C. S.; Lv, R.; Sun, X. M; Neal, N. X. et al. (2019): A novel weber local binary descriptor for fingerprint liveness detection. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*.

Xia, Z. H.; Ma, X. H.; Shen, Z. X.; Sun, X. M; Neal, N. X. et al. (2018): Secure image LBP feature extraction in cloud-based smart campus. *IEEE Access*, vol. 6, no. 1, pp. 30392-30401.

Xia, Z. H.; Neal, N. X.; Athanasios, V. V.; Sun, X. M. (2017): An efficient and privacy-preserving content-based image retrieval scheme in cloud computing. *Information Sciences*, vol. 387, pp. 195-204.