Eng Int Syst (2020) 2: 125–130

Engineering
Intelligent Systems

An Improved Facial Expression Recognition Algorithm

Yilihamu Yaermaimaiti*

School of Electrical Engineering, Xinjiang University, Urumqi 830047, China

The recognition of facial expression images is susceptible to non-uniform illumination factors, which may reduce the recognition rate. In view of this, an improved facial expression recognition algorithm is proposed. Firstly, the pattern-oriented edge magnitudes (POEM) histogram of the corresponding facial expression image is obtained through calculating the characteristic quantity of the facial expression image by the POEM. The histogram is created as the POEM texture histogram of the central characteristic point and the texture characteristic information of the facial expression feature points are obtained Secondly, the improved incremental non-negative matrix factorization (IINMF) algorithm is used to train the category information of face image samples to extract the face image representation vector. Canonical correlation analysis (CCA) is then used to combine the characteristic information of the POEM texture histogram and the eigenvector of the facial expression image extracted by IINMF to obtain the syncretic eigenvector of the facial expression image. Finally, the nearest neighbor classifier is used to classify and obtain the final identification result. The experimental results show that the proposed algorithm has a high recognition rate for facial expression recognition under non-uniform illumination and has excellent robustness and real-time results

Keywords: Facial expression; non-uniform illumination; POEM; IINMF; CCA.

1. INTRODUCTION

Facial expression recognition is an identification technology based on facial expression image characteristics. As a biometric authentication intelligent technology, it has been widely used in computer vision and pattern recognitions such as identification, security monitoring, and e-commerce, among others [1–4]. Among scholars, interest in facial expression recognition technology has increased greatly within the last few years [5–6]. Most existing facial expression recognition techniques can achieve good results only under ideal conditions, such as uniform illumination, with only small changes in expression and gestures, and no occlusion. However, in practical applications, it is difficult to ensure that the terminal remains in a fixed position, resulting in a difficulty in ensuring ideal imaging conditions. Therefore, facial expression recognition methods under non-ideal conditions have become a problem to be solved in order to reach the goal of mobile identity recognition. The research on facial expression recognition under non-uniform illumination is still one of the major current challenges in the research of facial expression recognition technology.

In recent years, facial expression recognition algorithms have mainly included the conventional local binary pattern (LBP) algorithm, conventional incremental non-negative matrix factorization (INMF) algorithm, SIEF characteristic algorithm, LDP algorithm and other characteristic extraction algorithms. Although each of these algorithms can improve the facial expression recognition effect to some extent, they all still have certain drawbacks. The conventional LBP algorithm has invariance to grayscale and rotation, and is an effective non-global texture description operator, but it relies too heavily on the gray value of the central point pixel and is not sensitive to illumination change [7]. Although the conventional INMF algorithm effectively avoids the recalculation problem after adding new training samples of facial expression images to the base matrix, the lack of category information of the initial training samples and the newly added training samples may lead to recognition failure, which affects the accuracy of the whole facial expression

*Email of Corresponding Author: 65891080@qq.com

recognition classification [8] The SIFT algorithm can detect key points in the image and is also a local characteristic descriptor, but the algorithm characteristics are all invariant for scale, rotation and affine transformation, and the calculation is time-consuming which makes it difficult to achieve real-time requirements [9]. The LDP algorithm is robust to noise [10], but the LDP code value has difficulty when reacting to the local eigenvector of the center pixel [11].

In view of the above factors, this paper proposes a facial expression recognition algorithm that combines POEM and IINMF. The algorithm uses the POEM algorithm and the IINMF algorithm to extract the eigenvector of facial expression images, and uses CCA to mix the facial expression eigenvector proposed by the above two algorithms together to obtain the final facial expression image characteristics.

2. BRIEF DESCRIPTION OF THE POEM ALGORITHM

The basic theory of the POEM algorithm is that the effective information data spread by the gradient magnitude and the edge's two-dimensional orientation can properly indicate the surface and state characteristics of facial expression images, the state after combining the facial expression image with the gradient operator has a significant improvement based on the horizontal and vertical directions. Assuming a two-dimensional face image, the effective information of the face gradient is shown in equation (1):

$$\begin{cases} G_x(x, y) = I(x, y) - I(x-1, y) \\ G_y(x, y) = I(x, y) - I(x, y-1) \end{cases} \tag{1}$$

Among them, $I(x, y)$ represents a random gray value in the facial expression image. The gradient magnitude and gradient direction of the facial expression image can be expressed as equation (2):

$$\begin{cases} G(x, y) = \sqrt{G_x(x, y)^2 + G_y(x, y)^2} \\ O(x, y) = \arctan \frac{G_y(x,y)}{G_x(x,y)} \end{cases} \tag{2}$$

The steps for POEM to extract facial expression features are as follows:

(1) Calculating the gradient two-dimensional image. Given a two-dimensional facial expression image, the gradient map is calculated; the gradient map and the orientation map of each pixel are then obtained. Thus for each pixel point of the two-dimensional facial expression image, the face information is a vector of a two-dimensional face image covering the magnitude and direction of the gradient. The gradient direction of the pixel points of the two-dimensional facial expression image is $q_i\ (i = 1, 2, \cdots, m)$ and the range is $q_i(i = 1, 2, \cdots, m)$ and the range is $0\tilde{\pi}$.

(2) Recording the image block of $w \times w$ centered on the pixel point q as $Cell$, a local time histogram is constructed with $Cell$ block as the basic unit. This histogram represents the intrinsic texture characteristic of the central two-dimensional pixel point of the time period.

(3) The final POEM histogram is constructed for the central 2D pixel of each stage, in a similar form to the LBP encoding process. The difference being that the LBP was estimated by its central two-dimensional pixel points and nearby two-dimensional pixel points, while the POEM was estimated by the $Cell$ block with the central two-dimensional pixel as the core and the $Cell$ block of the other two-dimensional pixel.

(4) For each gradient direction, the encoding process was centered on the central two-dimensional pixel block, and the remaining two-dimensional pixel blocks in the circle were calculated as follows:

First, in each gradient direction, for the central two-dimensional pixel q, in a circular range with the diameter R, n pixels and $Cell$ block were used to perform encoding:

$$POEM^{\theta_i}_{R,w,n}(q) = \sum_{j=1}^{n} f\left(S\left(I_q^{\theta_i}, I_{c_j}^{\theta_i}\right)\right) \cdot 2^j \tag{3}$$

In the equation, I_q and I_c are the central two-dimensional pixels and their neighboring pixels respectively, and $S(\ldots)$ is a similarity function. $f(x)$ is a binary function whose threshold is p, and its value is given by:

$$f(x) = \begin{cases} 1, & x \geqslant p \\ 0, & x < p \end{cases} \tag{4}$$

The POEM values in different directions of the intrinsic characteristic point q were then joined, and a concatenated histogram of the intrinsic texture characteristics of the characteristic points were obtained:

$$POEM_{R,w,n}(q) = \left\{POEM^{\theta_1}_{R,w,n}, \ldots, POEM^{\theta_m}_{R,w,n}\right\} \tag{5}$$

This allows POEM eigenvalues for each 2D pixel to be obtained.

3. IINMF

Although the conventional IINMF algorithm can be decomposed and formed into a low rank matrix mode, that is, data is broken into component elements and stored using this element data, reducing the amount of space required to store the data, unfortunately this creates a further issue: when adding new sample data for each training run, it is necessary to iteratively calculate the basic matrix W and coefficient matrix H generated by the previous factorization. Thus, under the structure of a large range of initial training data, the time taken to resolve re-iterations is considerable. In particular, the conventional IINMF uses the gradient descent method to update the incremental part h_{k+1} of the coefficient matrix, and its initial value is set to the end column vector of the current coefficient matrix H, namely:

$$(h_{k+1})_{init} = h \tag{6}$$

Although this h_{k+1} initialization method is very convenient, it does not make good use of the category information of the newly added samples, this will inevitably interfere with the effectiveness of data convergence. At the same time, because the iteration initial data setting of the h_{k+1} is not appropriate, it is easy to deviate from the local minimum when using the gradient descent method for optimization, which further affects the classification accuracy. For this phenomenon, the conventional IINMF can be improved with the below method based on the category information between the training samples and the new samples. Given a set of k training samples and their corresponding category information, the set can be expressed as equation (7):

$$V=\left\{v_1^1, v_2^1, \cdots, v_{K_1}^1, v_1^2, v_2^2, \cdots, v_{K_2}^2, \cdots, v_1^C, v_2^C, \cdots, v_{K_C}^C\right\} \tag{7}$$

In this equation, $v_i^c \in R^{n\times 1}$ represents the No. i image sample data in the c-type data, C is the total amount of the sample categories, and K_C is the sample data of the c-type data, and satisfies $K = K_1 + K_2 + \cdots + K_C$.

After the training sample V is decomposed by the non-negative matrix factorization algorithm, a coefficient matrix H having class information can be obtained through equation (8):

$$H=\left\{h_1^1, h_2^1, \cdots, h_{K_1}^1, h_1^2, h_2^2, \cdots, h_{K_2}^2, \cdots, h_1^C, h_2^C, \cdots, h_{K_C}^C\right\} \tag{8}$$

It is generally believed that when both v_i^c and v_j^c belong to the c-type, the corresponding coefficient vectors h_i^c and h_j^c after factorization should be similar. Based on this idea, the iteration initial value of the increment h_{k+1}^c of the coefficient matrix belonging to the c-type is set as the average vector of the c-type training data, as follows:

$$\left(h_{k+1}^c\right)_{init} = \overline{h^c} = \frac{1}{K_c}\sum_{j=1}^{K_c} h_j^c \tag{9}$$

Iteration is then conducted by the gradient descent method to obtain the increment h_{k+1} of the final coefficient matrix and the latest base matrix W_{k+1} after adding a new sample. The specific algorithm steps are as follows:

(1) Initialize non-negative matrices W and H;

(2) For the data image set V covering k initial training samples, iteration should be conducted according to the following requirements until the convergence condition is reached:

$$H_{aj} \leftarrow H_{aj}\frac{\left(W^T V\right)_{aj}}{\left(W^T W H\right)_{aj}} \tag{10}$$

$$W_{ia} \leftarrow W_{ia}\frac{\left(V H^T\right)_{ia}}{\left(W H H^T\right)_{ia}} \tag{11}$$

(3) Each time a new training sample v_{k+1}^c is added, initialization h_{k+1}^c will be conducted according to its category information, as follows:

$$\left(h_{k+1}^c\right)_{init} = \frac{1}{K_c}\sum_{j=1}^{K_c} h_j^c \tag{12}$$

(4) Update W_{k+1} and h_{k+1}^c according to following requirement until the convergence conditions are met:

$$\left(W_{k+1}\right)_{ia} \leftarrow \left(W_{k+1}\right)_{ia} \times \frac{\left(V_k H_k^T + v_{k+1}^c \left(h_{k+1}^c\right)^T\right)_{ia}}{\left(W_{k+1} H_k H_k^T + W_{k+1} h_{k+1}^c \left(h_{k+1}^c\right)^T\right)_{ia}} \tag{13}$$

$$\left(h_{k+1}^c\right)_a \leftarrow \left(h_{k+1}^c\right)_a \frac{\left(W_{k+1}^T v_{k+1}^c\right)_a}{\left(W_{k+1}^T W_{k+1} h_{k+1}^c\right)_a} \tag{14}$$

(5) Add the updated h_{k+1}^c to the last column of the current coefficient matrix H_k, i.e. $H_{k+1} = \left[H_k, h_{k+1}^c\right]$. In the above formulas, $i = 1, 2, \cdots, n$, $c = 1, 2, \cdots, C$, $j = 1, 2, \cdots, m$, $a = 1, 2, \cdots, r$

4. CCA

CCA is a statistical analysis method that expresses the relationship between multiple variables, which illustrates the correlation between two sets of feature vectors. That is, for a pair of zero-mean eigenvectors $x \in R^p$, $y \in R^q$, Rq, it is used to find a pair of base vectors (w_x, w_y) to make their eigenvector projections $x_i^* = w_x^i x$ and $y_i^* = w_y^i y$ correspond to the base vector, $i = 1, 2, \cdots, d$ $(d \leqslant \min(p, q))$, so that there is a maximum correlation between them, and the same eigenvectors are not correlated between the projection results on the base. Thus, the analysis of fewer canonical variables is required to create the correlation analysis between the eigenvectors x and y.

In general, the base vector (w_x, w_y) can be obtained using a standard function in the following equation (15).

$$\left(w_x, w_y\right) = \arg\max_{w_x, w_y} \frac{w_x^T S_{xy} w_y}{\sqrt{w_x^T S_{xx} w_x \cdot w_y^T S_{yy} w_y}} \tag{15}$$

In the above equation, S_{xy} represents the cross-covariance matrix between x and y, S_{xx} and S_{yy} represent the covariance matrices of x and y respectively. In this paper, the result z_f of the linear change is used as an eigenvector, and the combined eigenvector after the projections of x and y are used for classification, as shown in equation (16):

$$z_f = \begin{pmatrix} w_x^T x \\ w_y^T y \end{pmatrix} = \begin{pmatrix} w_x & 0 \\ 0 & w_y \end{pmatrix}^T \begin{pmatrix} x \\ y \end{pmatrix} \tag{16}$$

5. OVERVIEW OF THE SYNCRETIC FACE RECOGNITION ALGORITHM

The proposed algorithm uses the POEM algorithm and the IINMF algorithm to respectively extract the eigenvector of facial expression image, and adopted the CCA method to combine the facial expression eigenvector proposed by the above two algorithms to obtain the final facial expression image characteristics The algorithm process is as follows:

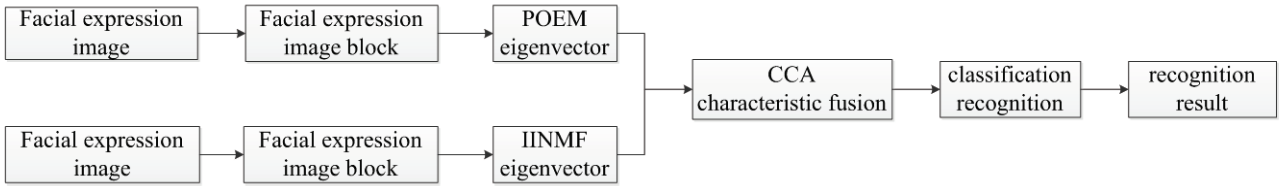

Figure 1 Flow Chart of Facial Expression Recognition Algorithm

(1) n facial expression images are randomly selected as a training sample, the rest of the samples are used as test samples; each original facial expression image is divided into equal size blocks to facilitate characteristics extraction

(2) The eigenvectors of POEM extracted from each training sample sub-image after segmentation is denoted by H^k_{POEM} $(k = 0, 1, 2, \cdots, 15)$, and the histograms of all the block sub-images of each facial expression image are then connected together and denoted by $H_{POEM} = \left[H^0_{POEM}, H^1_{POEM}, \cdots, H^{15}_{POEM}\right]$.

(3) Extracting improved incremental non-negative matrix (IINMF) factorization is represented by H^k_{IINMF} $(k = 0, 1, \cdots, 15)$, and the eigenvectors of all the block sub-images of each picture are then connected together, denoted by $H_{IINMF} = \left[H^0_{IINMF}, H^1_{IINMF}, \cdots, H^{15}_{IINMF},\right]$.

(4) Using the characteristics fusion strategy of equation (16) to combine the histogram characteristics of POEM with the IINMF factorization histogram features to obtain the final syncretic characteristics.

(5) Using the nearest neighbor classifier for classification and identification.

The algorithm identification flow chart based on the fusion of POEM and IINMF is shown in Figure 1:

6. EXPERIMENTAL RESULTS AND ANALYSIS

In order to verify the effectiveness of the proposed method, this paper selected the internationally recognised Cohn-Kanade facial expression dataset for simulation experiments. The dataset is constructed by the American Robotics Institute and the Faculty of Psychology in CMU. This paper selected a data set of 200 samples, with 6 expressions of 21 objects (sorrow, happiness, surprise, fear, anger and disgust). Among them, 15 images were selected from each expression for testing and the rest were used for training.

In this paper, the recognition rate is employed as one of the quantitative evaluation indicators of expression recognition performance, which is defined as:

$$recognition\ rate = \frac{Correctly\ identified\ sample\ size}{Total\ number\ of\ test\ samples} \times 100\% \quad (17)$$

In addition, this paper also used the average recognition time to quantitatively evaluate the computational efficiency of different methods. The recognition time refers to the time from the algorithm receiving the input image to the output classification result, excluding the time of characteristic training.

The computer used for the experimental test was configured with an Intel® Core™ i7 CPU and 4GB of RAM; the simulation environment was matlab2014b. According to the above quantitative evaluation index, the performance of the algorithm proposed in the paper was compared with the Gabor algorithm, the CS-LDP algorithm and the improved non-negative matrix algorithm. The effectiveness of the proposed method was verified.

6.1 Performance Evaluation

The six facial expressions were first identified by this paper's algorithm. Figure 2 shows that the algorithm recognition rate was higher than 90% for all six of the expressions. The average rate reached 93.45%, with the recognition rate for happiness reaching 100%. This indicates that the algorithm had a high recognition rate for the recognition of all six kinds of expressions.

The six kinds of expressions were then identified by different algorithms, the Gabor algorithm, the CS-LDP algorithm and the improved non-negative matrix algorithm, and compared to the algorithm proposed in this paper. The results are shown in Figure 3.

Figure 3 shows the recognition rate of the algorithm proposed in the paper is the highest, mainly because the algorithm based on the POEM and IINMF algorithms enhanced the saliency and robustness of the facial expression characteristics, which in turn highlighted the characteristics of the six facial expressions, while the other three algorithms rely on a single algorithm to achieve facial expression characteristics, without the effectiveness of the algorithm proposed in the paper.

6.2 Time Test Analysis

Finally, in order to verify the excellence of the algorithm in this paper, a time test was conducted. Table 1 clearly shows that the proposed algorithm had the fastest facial expression recognition rate, improving on the already impressive IINMF algorithm average time.. The algorithm proposed in this paper is therefore superior to the other algorithms it was compared with and has acceptable real-time performance.

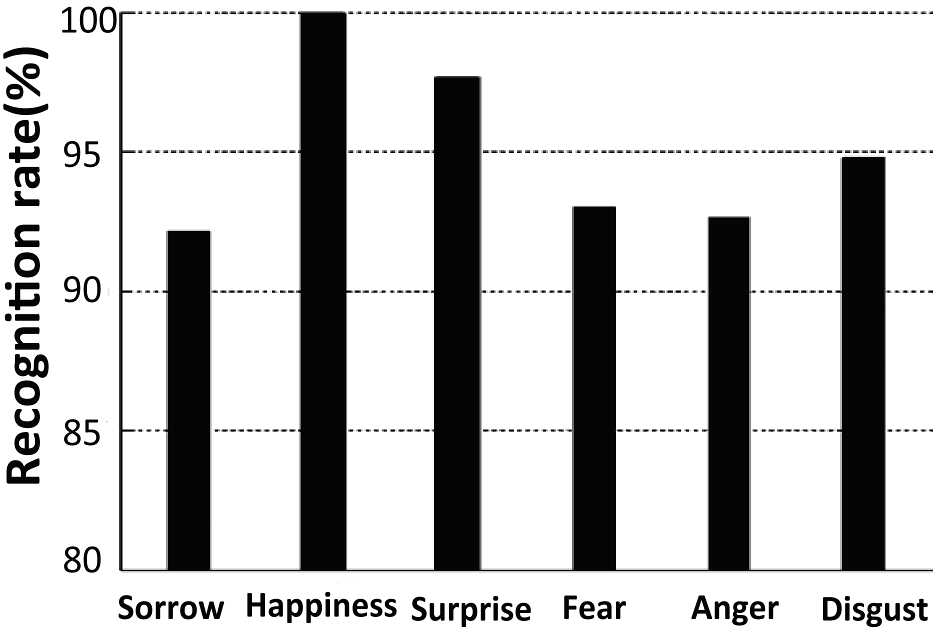

Figure 2 Expression Recognition Rate in This Paper

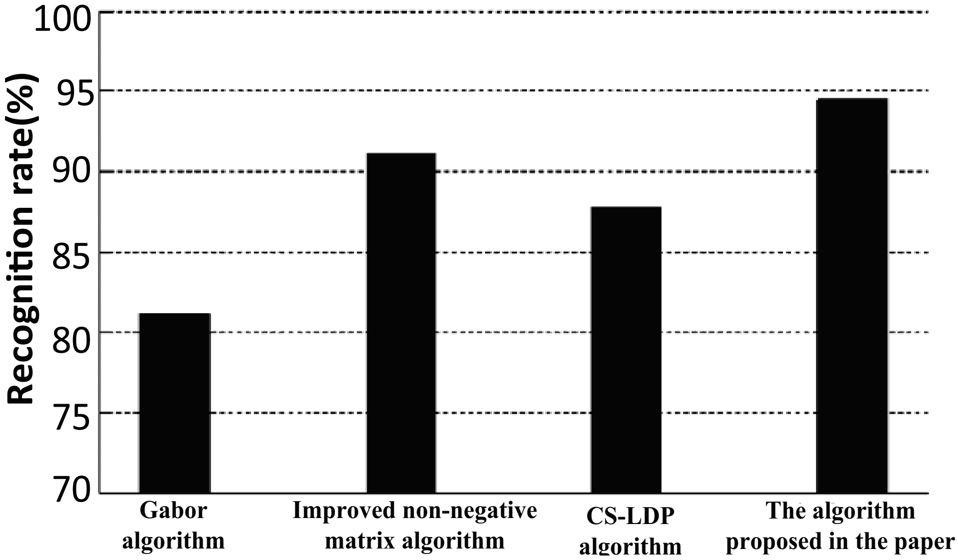

Figure 3 Recognition Rate of Different Algorithms

Table 1 Time test of different algorithms.

Algorithm	The average time of recognition/ms
Gabor	187
CS-LDP	179
Improved non-negative matrix algorithm	156
The algorithm proposed in the paper	149

7. CONCLUSION

This paper proposed a facial expression recognition algorithm based on the POEM and IINMF algorithms. The algorithm used the POEM algorithm and the IINMF algorithm to extract the eigenvectors of facial expression images, and adopted CCA to fuse the eigenvectors of the facial expression images of these above two algorithms; this then obtained the final facial expression image characteristics. Experiments showed that the proposed algorithm can extract facial expression characteristics well in non-uniform light, and had a high recognition rate, stable real-time results and robustness.

ACKNOWLEDGEMENTS

The authors acknowledge the financial support from the National Natural Science Foundation of China (Project No.: 61866037, 61462082).

REFERENCES

1. Faraji M.R., Qi X. Face Recognition Under Varying Illumination Based on Adaptive Homomorphic Eight Local Directional Patterns. Computer Vision IET, 2014, 9(3): 390–399.
2. Lu Y., Wang S.G., Zhao W.T., Wu W. Technology of Virtual Eyeglasses Try-On System Based on Face Pose Estimation. Chinese Optics, 2015, 8(4): 582–588.
3. Chen L., Zhao F. Application of Local Binary Pattern Weighting Algorithm Based on Support Vector Machine in Face Recognition. Bulletin of Science and Technology, 2015, 31(07): 111–114.
4. Wu J., Yan H., Wang J. Facial Recognition Using Local Binary Pattern and Deep Belief Network. Telecommunication Engineering, 2016, 56(10): 1119–1123.
5. Zhang F.Y., Liu Y.Y., Li X.M., Qin J. Multi-View Deep Neural Network Enhanced Forests for Facial Expression Recognition. Journal of Computer-Aided Design & Computer Graphics, 2018, 30(12): 2318–2326.
6. Pei S.W., Yang B.G., Gu C.H. Research on Recognition of Facial Expression Image Based on NIN_Res Net Model. Journal of Chinese Computer Systems, 2018, 39(12): 2681–2686.
7. Zhang J.Y., Zhao H.P., Chen S. Face Recognition Based on Weighted Local Binary Pattern with Adaptive Threshold. Journal of Electronics & Information Technology, 2014, 36(6): 1327–1333.
8. Cai J., Wang W.L., Zheng J.W., Li J.M. Incremental Discriminant Non-Negative Matrix Factorization and its Application to Face Recognition. Journal of Graphics, 2017, 38(05): 715–721.
9. Chen M.T., Yan D.M., Wang G. Algorithm of High-Resolution Remote Sensing Image Matching Based on Harris Corner and SIFT Descriptor. 2012, 17(11): 1453–1459.
10. Uijlings J.R.R., Smeulders A.W.M., Scha R.J.H. Real-time Visual Concept Classification. IEEE Transactions on Multimedia, 2010, 12(7): 665–681.
11. Jain A.K. Data Clustering: 50 Years Beyond K-means. Pattern Recognition Letters, 2010, 31(8): 651–666.