Pose Normalization for Robust Face Recognition Based on Statistical Affine Transformation

Xiujuan Chai[1, 2], Shiguang Shan[2], Wen Gao[1, 2]

[1]Vilab, Computer College, Harbin Institute of Technology, Harbin, China, 150001
[2]ICT-YCNC FRJDL, Institute of Computing Technology, CAS, Beijing, China, 100080
xjchai@vilab.hit.edu.cn, {sgshan, wgao }@jdl.ac.cn

Abstract

A framework for pose-invariant face recognition using the pose alignment method is described in this paper. The main idea is to normalize the face view in depth to frontal view as the input of face recognition framework. Concretely, an inputted face image is first normalized using the irises information, and then the pose subspace algorithm is employed to perform the pose estimation. To well model the pose-invariant, the face region is divided into three rectangles with different mapping parameters in this pose alignment algorithm. So the affine transformation parameters associated with the different poses can be used to align the input pose image to frontal view. To evaluate this algorithm objectively, the views after the pose alignment are incorporated into the frontal face recognition system. Experimental results show that it has the better performance and it increases the recognition rate statistically by 17.75% under the pose that rotated within 30 degree.

Keywords: Pose-invariant face recognition, pose estimation, pose alignment, affine transformation, face recognition

1. Introduction

Face recognition and the related research activities have significantly increased in these past few years and several successful commercial face recognition systems have emerged in the biometrics markets [1,2]. But at the same time, some unavoidable problems appear in the variety of practical applications, such as, the people are not always frontal to the camera, so the pose problem is a big obstacle for the face recognition system to be prevalence.

Pose-invariant face recognition allows the pose to vary when recognizing, especially allow the face to rotation in depth, including up/down rotation and left/right rotation. In essence, the difference between the same people under the varied poses is larger than the difference between the distinct persons under the same pose [5]. So it is difficult for the computer to do the face identification when the poses of the probe and gallery images are different.

Besides this reason, there also exist large difficulties in collecting the face images under various poses. The training images using to recognize can't cover all the poses. And the more the multi-poses samples the training set have, the slow the performance of the recognize system is. So the difficult of pose-invariant recognition is the face samples of multi-poses can't be achieved easily and cover all the poses. There are many existing approaches proposed to recognize face under varying pose.

Many subspace-based techniques have been used to tackle pose-invariant face recognition. Murase and Nayar used a parametric eigen-space method by representing each known person by compact manifolds in the subspace of the eigenspace [6]. Pentland et al used a view-based subspace method by producing separate subspaces each constructed from faces at the same viewpoint [7]. Recognition is performed by first finding the subspace most representative of the test face and then matching using a simple distance metric in this subspace. Analytical subspace method is used by Valentin and Abdi [8]. Characteristic subspace method is used by McKenna et al [9].

Active Appearance Model is also used in pose-invariant face recognition [10]. It performs the recognition by deforming a generic face model to fit with the input image and gains the control parameters as the feature vector, to be classified. Because of the process of optimal and iteration, it takes longer time.

Another mainstream method is to generate virtual views from one previous known view using kinds of algorithms. For example: V. Blanze and T. Vetter proposed a morphable model method to synthesis 3D faces [15]. First derive a morphable face model by transforming the shape and texture of the examples into a vector space representation. Then an input new face can be modeled by forming linear combinations of the prototypes. But the presupposition is that there are plenty of precise 3D face image data. The similar algorithm such as the parallel transform algorithm using the optical flow computation proposed by Beymer and Poggio exploits the image-based linear 2D face models [11-13]. The performance of parallel transform algorithm is enslaved to the precision of the optical flow computation.

Our investigation is also explored the image synthesis strategy to tackle the pose problem. The method can be described as follows: divide the face region into three rectangles. Through studying the correspondence information between the specific pose and the frontal pose, the affine transformation parameters can be calculated. So the normalized face image in some pose can be geometrically warped into the frontal pose. Having the virtual frontal pose view, then the frontal face recognition algorithm can be used to recognize the given face image.

The remaining part of this paper is organized as follows. In Section 2, we describe our pose-invariant face recognition system framework in general. In Section 3, the pose alignment algorithm is introduced in detail. The performance evaluation of our pose normalization algorithm by the face recognition system is presented in Section 4. A short conclusion and some discussions are given in the last Section.

2. System Overview

In our system, face images are classified into 7 poses (excluding the upward and downward situations). The normalized class samples are shown in Figure 1. Our goal is to recognize the non-frontal face image by aligning non-frontal poses to frontal views. The pose-invariant recognition system diagram is given in Figure 2. At first, we estimate the pose of the given face through the specific subspace method. In this phase, all classes of images are normalized to the same image size and the same distance between two eyes. The precise locations of irises can be got by the region growing search method proposed by B.Cao[14]. Then according to the estimated poses, the pose alignment algorithm is used to generate the frontal face image which can be regarded as the input of the frontal face recognize algorithm and the recognize result is gotten.

Figure 1. The normalized train samples of 7 different poses. In succession, the poses are: Right 40 degree, Right 25 degree, Right 15 degree, Frontal, Left 15 degree, Left 25 degree, Left 40 degree.

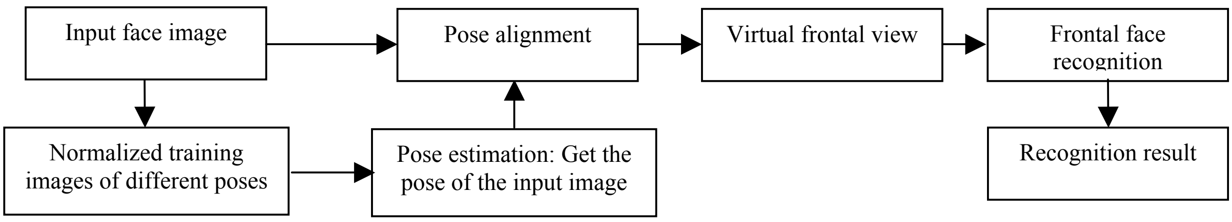

Figure 2. The diagram of the pose-invariant face recognition system.

3. Pose Alignment Based on Statistical Transformation

In this section, an affine transformation algorithm based on statistic is described to solve the problem of the virtual frontal view generation. The algorithm partitions the face into 3 rectangle regions and estimates the 7 groups of pose parameters with each pose having 200 face images. Those images have been manually labeled with some face profile points. The one-to-one rectangle mapping relations between the specific non-frontal pose and the frontal pose will be found. For the images under different poses, the corresponding normalizations are used. So, first we must estimate the distances between two eyes by uniforming the vertical distance between eyes and jaw.

In Figure 3, the top of the rectangle is the top vertex of the brow, the left boundary of the first rectangle is the left visor, the right boundary of the third rectangle are the right visor, and the bottom of the rectangle is the rock bottom of the jaw. We have enough information to get the affine transform relation between the corresponding two rectangles. The affine transform strategy is used to model

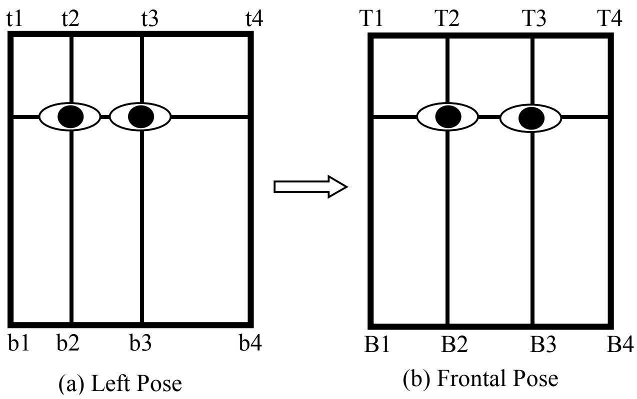

Figure 3. The sketch map of the affine transformation to align the pose (take the left pose to frontal pose as example). The corresponding rectangles are respectively: rectangle [t1,t2,b2,b1] and rectangle [T1,T2,B2,B1], rectangle [t2,t3,b3,b2] and rectangle [T2,T3,B3,B2], rectangle [t3,t4,b4,b3] and rectangle [T3,T4,B4,B3].

the images mapping relation under different poses.

The formula description of this algorithm is as follows:

(1) Given the 8 vertices of 2 rectangles (R_p and R_f) under left pose and frontal pose.

(2) For the 4 vertices (x_i, y_i) $i = 1,2,3,4$ in R_f, after the transformation, can get the corresponding 4 vertices (x_i', y_i') $i = 1,2,3,4$ in R_p. So the affine transform parameters can be calculated through this equation:

$$\begin{bmatrix} 1 & x_1 & y_1 & x_1y_1 \\ 1 & x_2 & y_2 & x_2y_2 \\ 1 & x_3 & y_3 & x_3y_3 \\ 1 & x_4 & y_4 & x_4y_{41} \end{bmatrix} \cdot \begin{bmatrix} a_0 & b_0 \\ a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \end{bmatrix} = \begin{bmatrix} x_1' & y_1' \\ x_2' & y_2' \\ x_3' & y_3' \\ x_4' & y_4' \end{bmatrix} \qquad (1)$$

Let use X_f, A, X_p denote the three matrices, and the Equation (1) can be simplified as:

$$X_f \cdot A = X_p \qquad (2)$$

So the parameters can be calculated by:

$$A = X_f^{-1} \cdot X_p \qquad (3)$$

As the images are normalized, the corresponding points under the different poses have the same y coordinates. So finally the parameters are equal to zero except a_0 and a_1.

(3) Using the estimated parameters, we can generate the virtual frontal view through the polynomial warping. For the given image, its pose class is first determined by the subspace method. Then the affine transformation vector used for warping is decided. For each point (x, y) in the frontal view, first determine which rectangle it is in, then decide which two parameters should be used to map from the given image.

Define (x', y') as the transformed value (x, y) and it can be expressed as:

$$\begin{cases} x' = a0 + a1 \cdot x \\ y' = y \end{cases} \qquad (4)$$

Let $\begin{cases} x1 = \text{int}(x') \\ y1 = y' \end{cases}$, and $dx = x' - \text{int}(x')$, so the luminance value $f(x, y)$ can be computed by:

$$f(x, y) = (1 - dx) \cdot f'(x1, y1) + dx \cdot f'(x1 + 1, y1) \qquad (5)$$

where $f'(x, y)$ is the luminance of point in the given non-frontal image.

4. Experiment Results

Experiments are carried out on the subset of the Facial Recognition Technology (FERET) database with 200 image samples each pose (in total 1400 images). All face images are normalized according to the different criterions associated with different pose classes. To get the affine transformation parameters, some key points which represent face structure information are marked in each image manually, and we estimate the average values of the 8 vertices (t1,t2,t3,t4,b1,b2,b3,b4) which consist of the 3 rectangles for each pose. The vertex values are shown in Table 1. Take the frontal face image as the baseline, map the 3 rectangles to the corresponding rectangles under other poses, and calculate 6 groups of affine transformation parameters. We define a0 and a1, b0 and b1, c0 and c1 as the mapping parameters of the left rectangle, the middle rectangle, and the right rectangle respectively. All the parameters are given in Table 2.

To test whether our pose alignment algorithm has a good performance for recognizing non-frontal face images, we take the frontal views generated by this pose normalization strategy as the input of the frontal face recognition system, compare the right recognition rate using original pose images and the pose-aligned images respectively. The recognition system adopts the simple nearest neighbor method. Experimental results show that this pose normalization algorithm based on statistical transformation has better performance on pose alignment. The results are presented in Table 3. Some masked face images after pose alignment and the original masked images that used to do face recognition are displayed in Figure 4. It is easy to know from the Table 3 that the recognition rate has increased averagely by 17.75% for the pose rotation in depth within 30 degree, better than the performance when pose rotation is greater.

Table 1. The 8 vertices coordinate values statistic from train images of the 7 different poses.

	Left 40	Left 25	Left 15	Frontal	Right 15	Right 25	Right 40
t1	（21，34）	（18，34）	（14，34）	（10，34）	（7，34）	（3，34）	（2，34）
t2	（30，34）	（30，34）	（30，34）	（30，34）	（31，34）	（31，34）	（32，34）
t3	（57，34）	（58，34）	（59，34）	（60，34）	（60，34）	（60，34）	（60，34）
t4	（85，34）	（83，34）	（81，34）	（79，34）	（74，34）	（71，34）	（68，34）
b1	（21，104）	（18，104）	（14，104）	（10，104）	（7，104）	（3，104）	（2，104）
b2	（30，104）	（30，104）	（30，104）	（30，104）	（31，104）	（31，104）	（32，104）
b3	（57，104）	（58，104）	（59，104）	（60，104）	（60，104）	（60，104）	（60，104）
b4	（85，104）	（83，104）	（81，104）	（79，104）	（74，104）	（71，104）	（68，104）

Table 2. Affine transformation parameters for each pose.

	a0	a1	b0	b1	c0	c1
Left 40	16.500000	0.450000	3.000000	0.900000	-31.421053	1.473684
Left 25	12.000000	0.600000	2.000000	0.933333	-20.947368	1.315789
Left 15	6.000000	0.800000	1.000000	0.966667	-10.473684	1.157895
Right 15	-5.000000	1.200000	2.000000	0.966667	15.789474	0.736842
Right 25	-11.000000	1.400000	2.000000	0.966667	25.263158	0.578947
Right 40	-13.000000	1.500000	4.000000	0.933333	34.736842	0.421053

Table 3. Evaluate the performance of the pose alignment algorithm.

	Left 40	Left 25	Left 15	Right 15	Right 25	Right 40
Recognition rate no pose alignment	14%	27%	63%	47%	26%	11.5%
Recognition rate after pose alignment	21.5%	51%	74%	70%	39%	15%
Increase rate	7.5%	24%	11%	23%	13%	3.5%

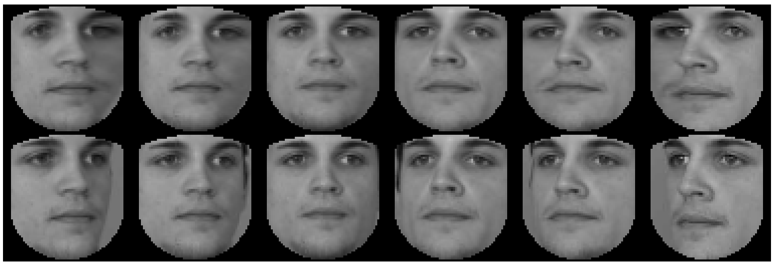

Figure 4. The masked face images for the frontal face recognition. The first row is the revised views after pose alignment with our algorithm. The last row is the masked original face images under non-frontal poses that haven't done pose corresponding to the above pose.

5. Conclusion

The pose normalization algorithm based on the statistical transformation is presented in this paper. The proposed algorithm can solve the pose-invariant face recognition problem to some extent, especially when face rotation angle is less than 30 degree. Experiments show that our method has a good performance for normalize the non-frontal face images, and can dramatically increase the recognition rate with a very low time complexity.

Whereas, the face rotation in depth is a non-linear transformation, so simple affine transformation can only model the pose variant tendency partly. For the great pose variability, this strategy can't precisely learn the vary modes of the different poses. So the more complicated and elaborated model may be studied to solve the pose normalization in the future.

6. Acknowledgment

This research is sponsored partly by Natural Science Foundation of China (No.69789301), National Hi-Tech Program of China (No.2001AA114190 and No. 2002AA118010), YinChen Net. Co. (YCNC)

References

[1] R. Brunelli and T. Poggio. Face Recognition: Features versus Template. TPAMI, 15(10), pp.1042-1052, 1993

[2] R. Chellappa, C.L. Wilson et al. Human and Machine Recognition of Faces: A survey. Proc. of the IEEE, 83(5), pp. 705-740, 1995

[3] P.J. Phillips, H. Moon. The FERET Evaluation Methodology for Face-recognition Algorithms. IEEE TPAMI, 22(10) pp. 1090-1104, 2000

[4] D.M. Blackbum, M. Bone, P.J. Phillips. Facial Recognition Vendor Test 2000: Evaluation Report. 2001, http://www.frvt.org/frvt2000/

[5] D. Beymer and T. Poggio. Face Recognition From One Example View. ICCV, Boston, MA, pp. 500-507, 1995

[6] H. Murase and S.K. Nayar. Visual Learning and Recognition of 3-D Objects form Appearance. International Journal of Computer Vision, 14:5-24,1995

[7] A. Pentland, B. Moghaddam and T. Starner. View-based and Modular Eigenspaces for Face Recognition. IEEE CVPR, pp. 84-91, 1994

[8] D. Valentin and H. Abdi. Can a Linear Autoassociator Recognize Faces From New Orientations. Journal of the Optical Society of America A-Optics, Image Science and vision, 13(4), pp. 717-724, 1996.

[9] SmMcKenna, S. Gong and J.J. Collins. Face Tracking and Pose Representation. British Machine Vision Conference, Edinburgh, Scotland, 1996.
[10] T. Cootes, G. Edwards, and C. Taylor. Active Apperance Models. ECCV
[11] D. Beymer. Face Recognition Under Varying Pose. AI Memo 1461
[12] D. Beymer and T. Poggio. Face Recognition From One Model View. Proc. *Fifth Int'l Conf.* Computer Vision, 1995
[13] T. Vetter and T. Poggio. Linear Object Classes and Image Synthesis From a Single Example Image. IEEE Trans. PAMI
[14] B. Cao, S. Shan,W. Gao. Localizing the iris center by region growing search. Proceeding of the ICME2002
[15] V. Blanz, T. Vetter. A Morphable Model for the Synthesis of 3D Faces. SIGGRAPH'99 Conf. Proc., Los Angeles, USA, 1999.