3D Human Face Recognition Using Sift Descriptors of Face's Feature Regions

Nguyen Hong Quy[1], Nguyen Hoang Quoc[1], Nguyen Tran Lan Anh[1], Hyung-Jeong Yang[2], and Pham The Bao[1]

[1] Faculty of Math and Computer Science, University of Science, Ho Chi Minh City, Vietnam
[2] Department of Computer Engineering, Chonnam National University, South Korea
nhquy@hotmail.com, {nhquoc,hyungjeong}@gmail.com,
{ngtlanh,ptbao}@hcmus.edu.vn

Abstract. Many researches in 3D face recognition problem have been studied because of adverse effects of human's age, emotions, and environmental conditions on 2D models. In this paper, we propose a novel method for recognizing 3D faces. First, a 3D human face is normalized and determined regions of interest (ROI). Second, SIFT algorithm is applied to these ROIs for detecting invariant feature points. Finally, this descriptor, extracted from a training image, will be stored and later used to identify the face in a test image. For performing reliable recognition, we also adjust parameters of SIFT algorithm to fit own characteristics of the template database. In our experiments, the proposed method produces promising performance up to 84.6% of accuracy when using 3D Notre Dame biometric data-TEC.

Keywords: 3D face recognition, SIFT descriptors, range images.

1 Introduction

Nowadays, many fields such as finance, banking, stock market, etc. require high level of security. The need of fast and precise human identification in business transactions has become urgent. A lot of biometric technologies (e.g. fingerprint, iris, face) are exploited due to their high reliable characteristics. Although human face contains less invariant features than others, it still gains much potential and suitable low-price security applications. Currently, face recognition systems only focus on 2D images taken by normal digital cameras. Despite of achieving very good results, it cannot satisfy researchers because these 2D images may flatten special features appearing in the face as well as there exists restrictions caused by objective effects of light, noise, facial emotions, etc. So its important depth information may be lost. For this reason, 3D human face recognition algorithms have been studied more and more since not only faces captured by 3D models contain a lot of information but also they are not affected by negative effects.

As clarified in surveys of 3D face recognition methods, their main researches can be divided into two categories [1]: one processes only pure 3D data and the other is a

combination of 2D and 3D data. Approaches belonging to the second type were promoted quite late in 2000. Most of them try to take results obtained from both 2D and 3D models for producing better conclusions. Wang et al. in [2] presented an idea of this combination by describing feature points using Gabor filter responses in a 2D gray-level image and point signature in a 3D domain. Later, an approach proposed by Gang Pan et al. [3] was to automatically extract ROI of facial surface by considering bilateral symmetry plane and localization of nose tip. In [4], [5], [6], [7] Principal Component Analysis (PCA) algorithm and its modified versions were applied to get principal components of range images as feature points. After that, various distance measurements were used for 3D face recognition or classification. J. Cook et al. [8] combined Log-Gabor Templates for variation expression of depth and texture data with Mahalanobis Cosine metric as the distance measure for solving the problem. Suggested in [9] by Pamplona Segundo et al., a scale-invariant face detection approach were proposed by using boosted cascade classifiers in range images as input for real-time 3D face recognition system. Jahanbin S et al. [10] motivated a novel multimodal framework based on local attributes such as 2-D and 3-D Gabor coefficients and the anthropometric distances. Finally three parallel face recognizers were considered to form a recognition system at matching score level. However processing very large volume of 3D information is still the most challenge of 3D recognition models. Important facial characteristics can be skipped for reducing processing time during this stage. For adapting to real-time systems, a balance of accuracy and quickness in 3D face recognition algorithms is needed. Derived from above issues, developing a simple and high reliable 3D recognition model is the key point of our proposed method.

In this paper, a novel face recognition method by applying a 2D processing technique into pure 3D face images is represented. In Section 2, we describe the proposed method in details. Section 3 gives experiments on 3D Notre Dame biometric data-TEC database. Finally, conclusions are drawn from our work in Section 4.

2   Methodology

2.1   General System

To describe points of a face in our 3D model, data is input in the form of three matrices and position of the top of its nose is always given. An example of our input data is shown in Fig. 1 after mapping three data matrices to a 2D face image.

Fig. 1. An input face after mapping from 3D to 2D space including its depth value

2.2   Face Normalization

When sampling data, it is very difficult and inconvenient to capture well human faces as our standard criteria. We might accept samples that have slight deviation in both three dimensions. Although this flexibility does not affect SIFT descriptors [15], the input should be rotated into its front face after this step for an easy ROI extraction.

Horizontal Rotation. For rotating a face image horizontally, we try to balance the deviation of two tops of nostrils illustrated as red points in Fig. 2. Since position of the top of the nose is given, it is quite easy to determine these two points.
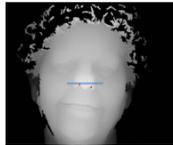

Fig. 2. A necessary case of horizontal rotation based on two unbalanced tops of nostrils

Let *ep* be a threshold defined as a hard constraint to normalize the face. If the deviation of two tops of nostrils is larger than *ep*, rotation of the face image will be done. Depending on the direction of face deviation (i.e. left or right), the image may be rotated 1° or -1° horizontally according to the following Algorithm 1.

Algorithm 1. Horizontal rotation
1: define *ep* threshold
2: **while** (| *leftNostrils.y – righNostrils.y* | > ep)
3:      **if** (*leftNostrils.y* > *righNostrils.y*) rotate *image* 1° horizontally
4:      **else** rotate *image* -1° horizontally
5:      get *leftNostrils*; get *righNostrils*;
6: **end while**

Vertical Rotation. To conclude vertical deviation of a face, we compare the input data with its quantization form. First a straight line is drawn paralleling to the horizontal coordinate axis in the quantization map (see Fig. 3). Our target is then to rotate the face so that the vertical axis of the face is perpendicular to this line. In this case, top of the nose is chosen as the center to separate its left and right side.
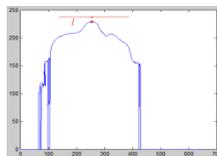

Fig. 3. A quantization map of a face deviated vertically in its right side

Let l be a length of the straight line drawn from the top of the nose to its left and right side. *SoL* and *SoR* are defined as the left and right sides of the top in the quantization map, respectively. Algorithm 2 is used to rotate the face vertically.

Algorithm 2. Vertical rotation

1: define *ep* threshold
2: **while** (| *SoL* – *SoR* | > *ep*)
3:     **if** (*SoL* > *SoR*) rotate *image* 1° vertically
4:     **else** rotate *image* -1° vertically
5:     *Line* ← *image* [*Nose.x* – *l* : *Nose.x* + *l, Nose.y*]
6:     *SoL* ← sum(*Line*[1 : *l*]); *SoR* ← sum(*Line*[*l*+1 : *l*+100]);
7: **end while**

Since the order of vertical and horizontal rotations can affect the result of face normalization, rotating first horizontally and then vertically gains better results than in the reverse order. If we first rotate in vertical direction, the straight line cannot be perpendicular to the vertical axis of the face any more after its horizontal rotation. As a result, its rotated result becomes incorrect. Hence, we decide to perform the horizontal rotation before doing the vertical one.

2.3   ROI Extraction

After normalizing, the face image is very close to a front face. Next we will find biometric features of the face and then apply a local extreme method to determine ROIs in the image. There are four ROIs including two eyes, a nose, and a mouth that we concern. Finally rectangles will enclose each ROI.

As [11], human face can be divided on a golden ratio (see Fig. 4(a)). This ratio is considered as special harmony of the face. Moreover, in 1994 Farkas introduced facial anthropometric measurement points [12], that will transform simultaneously and whose positions are rarely changed during the variation of a face. In this paper, we only choose some of these points to extract ROIs. As shown in Fig. 4(b), these selected points are on a line along the nose.

(a)   (b)

Fig. 4. (a) Golden ratio and (b) some invariant points on a face

By applying a local extreme method into a quantization chart of the line along the nose, we can obtain local extreme points as anthropometric measurement points (see Fig. 5).
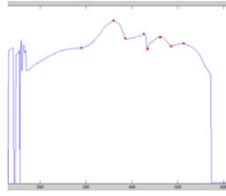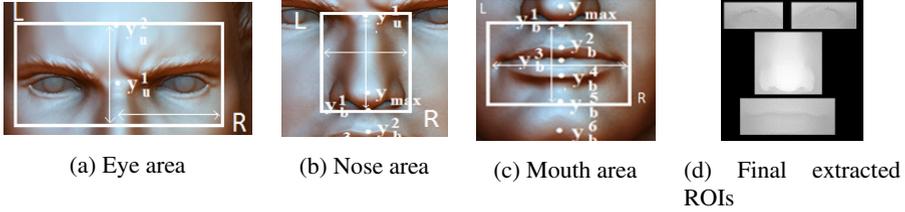
Fig. 5. A quantization chart of the line along the nose

(a) Eye area   (b) Nose area   (c) Mouth area   (d) Final extracted ROIs

Fig. 6. Facial ROI extraction

First, an eye area is illustrated in Fig. 6(a) based on three points y_u^3, y_u^1, and y_{max} and described as

$$\left\{(x, y) \mid x_{(L_3)} \leq x \leq x_{(R_3)}, \left(y_u^1 + y_{max}\right)/2 \leq y \leq y_u^2\right\} \tag{1}$$

Next, nose area is illustrated in Fig. 6(b) based on two points y_u^1 and y_b^1 and described as

$$\left\{(x, y) \mid x_{(L_1)} \leq x \leq x_{(R_1)}, y_b^1 \leq y \leq y_u^1\right\} \tag{2}$$

Finally, mouth area is illustrated in Fig. 6(c) based on two points y_b^5 and y_b^1 and described as

$$\left\{(x, y) \mid x_{(L_2)} \leq x \leq x_{(R_2)}, y_b^5 \leq y \leq y_b^1\right\} \tag{3}$$

Based on the facial golden ratio, the width of rectangles enclosing these eyes, nose, and mouth ROIs are estimated respectively (see Fig. 6(d)) as below

$$\mid x_{(L_3)} - x_{(R_3)} \mid = \frac{\mid y_u^1 - y_b^3 \mid}{\varphi} \tag{4}$$

$$\mid x_{(L_1)} - x_{(R_1)} \mid = \frac{\mid y_{max} - y_b^4 \mid}{\varphi} \tag{5}$$

$$\mid x_{(L_2)} - x_{(R_2)} \mid = \mid y_{max} - y_b^4 \mid \tag{6}$$

2.4   Feature Extraction Using SIFT Descriptor

SIFT descriptors was proposed by Lowe [13] in 2004 and there have been many its improvements [15], [16]. Compared to others, this method has its own advantages of detecting image features at different scales relied on scale space theory [17] as well as being invariant to Affine transformations and illuminative changes. SIFT method processes in three main stages: locating features named as key-points on an image, generating detectors, and generating descriptors. These key-points usually contain distinct characteristics that help improving the efficiency of the matching stage. In this paper, we use an improvement of SIFT descriptors done by Andrea Vedaldi (University of California) [16].

According to Lowe, SIFT descriptor depends on two parameters. They are the number of octave and the number of picture in each octave (numlevels) [13]. If we can choose a good number of octave and numlevels, good features will be found and the identification will be performed better (see Fig. 7).
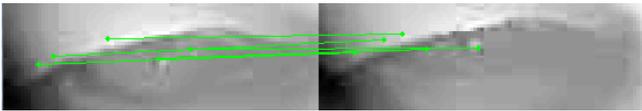

Fig. 7. An illustration of SIFT descriptors at a good level of numlevels for a left eye of the same person captured from two different pose directions

2.5   Matching

In this stage, key-points of two unknown face images are matched to find a set of similar points in these faces. Basically, pairs of similar key points have to locate in similar ROIs in the face (eye to eye, nose to nose, and mouth to mouth). Thus we evaluate the similar location of each pair of two similar key points. Each key point of the pair is calculated its relative distance to the top of the noise. Then, if the difference between these two relative distances is less than a threshold λ, this pair is considered to have similarities in both feature and position. Meanwhile, two similar key points located in the different positions will be removed. By counting the number of pairs of similar key points, we can evaluate the similarity in two faces.

In our proposed method, the given result is only the closest face to the face we want to identify. It cannot answer whether the result is correct or not. As we know, different persons usually have at least one eye that is absolutely dissimilar from the corresponding eye of others (explained later). In few cases, if both eyes have similar key points, the subtraction of the number of similarities key points between two eyes of the same person is lower than of two different persons. Next, we perform two statistics in our database corresponding to two above comments respectively. Following Table 1 shows results of the first statistics. We took arbitrary 10 persons out of 80-person dataset in succession to build more than 800 observations.

For the second statistics, we first call *SL* as the number of similar key points of the left eye and *SR* as the similar points of the right eye between the input face and the current matching face. In the case that a person have similar key points in both his/her eyes, an experiment to measure |*SL* − *SR*| is represented in Fig. 8.

Table 1. A statistics of the first comment about the similarity of key points in human eyes

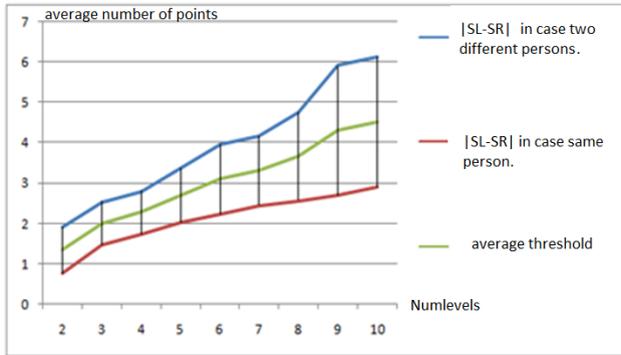| Numlevels	At least one eye has no similar key point
2	606(75.8%)
3	599(74.9%)
4	594(74.3%)
5	593(74.1%)
6	542(67.8%)
7	511(63.9%)

Fig. 8. A statistics of the second comment about the |*SL* − *SR*| value between the same person and different persons

Based on the above statistics, we suggest an adaptive way to identify whether the input face belongs to the given dataset or not. The matching result is not correct (that means it is not posed from the same person) if it satisfies one of the following conditions:

$$SL=0 \text{ or } SR=0 \tag{7}$$

$$SL > 0 \text{ and } SR > 0 \text{ and } |SL - SR| \geq \alpha \tag{8}$$

where α is an average value corresponding to each Numlevel chosen in the green line shown in Fig. 8. Algorithm 3 is used to describe how to match the given input with the existing system database.

3   Results

To demonstrate performance of our proposed 3D human face recognition system, we have carried out different evaluations on the Notre Dame biometric 3D-TEC dataset (including 440 poses of over 90 persons). Our program runs on a PC equipped with Intel Core 2 Duo, CPU 2x2.0GHz, 3GB RAM, Windows 7 Professional in the environment of Matlab 2011.

   To evaluate the performance of face recognition, we perform our model in two data collections. They are selected randomly by choosing a quarter and a third of the dataset. Due to the ROI extraction and our remarks on the matching stage, we can both save a lot of cost for the identification and achieve high accuracy of recognition. As shown in Fig. 9, the minimum average cost is approximate 13.2 seconds in Matlab. Furthermore, as described in Table 2, the maximum precision is up to 84.6%.

Algorithm 3. Matching

Input: *inputImage*
1: define α, λ
2: max $\leftarrow$ 0
3: **for all** *imageSet*
4:      $\theta \leftarrow getSIFT(inputImage, imageSet[i]$);
5:      $SL \leftarrow |\theta_{leftEye}|$; $SR \leftarrow |\theta_{rightEye}|$;
6:      $S \leftarrow SL + SR + |\theta_{Nose}| + |\theta_{Mouth}|$;
7:      **if** (SL=0 | SR=0 | (SL>0 & SR>0 & $|SL - SR| \geq \alpha$))
8:           **continue**
9:      **if** (*max < S*) *max $\leftarrow$ S*; ω = i; **end if**
10: **end for**
11: **if** (*max* = 0) we cannot find any similar face.
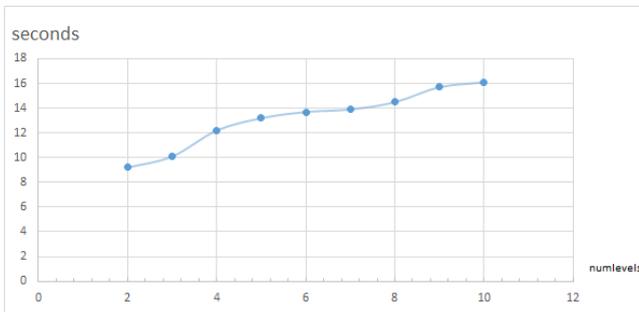**Output:** *imageSet[ω]*

Fig. 9. Average implementation cost for our face recognition model

Table 2. The precision though the numlevel with a quarter of database and one-third of database

Numlevels	Precision with a quarter of dataset (%)	Precision with a third of dataset (%)
2	70.5	63
3	72	65.3
4	75.4	67.2
5	81	68.3
6	84.6	71.3
7	84.3	70.2

In the next experiment, we first took 10 persons out of the database. Face images of these persons are not used to train the system. They are considered as "strangers" in the database. We later use them to test our proposed system. As shown in Fig. 10, we finally get 20% of the minimum False Acceptance Rate (FAR) in the case of choosing good SIFT parameters.
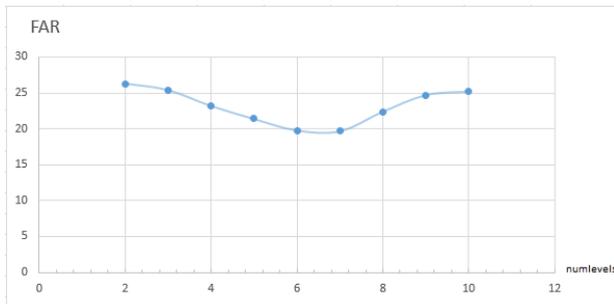
Fig. 10. FAR when testing 10 "strangers" in the training dataset

4   Conclusions

We represented a novel 3D face recognition algorithm to model good human identification in a simple and fast way. Since SIFT descriptors is a suitable method in 2D database, we tried to reduce its high cost of many processing steps as well as limit the dependence on its own parameters for applying into 3D data. In the experiments, our proposed model showed promising results of efficiency and effectiveness. However, this algorithm still needs further improvements. Although its precision and average cost is acceptable, they have not reached the required criteria of a security system yet. And appropriate parameters for SIFT descriptors should be considered more adaptively to restrict coincidental positions of key points.

References

1. Bowyer, K.W., Chang, K., Flynn, P.: A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition. Journal of Computer Vision and Image Understanding 101(1), 1–15 (2006)
2. Wang, Y., Chua, C., Ho, Y.: Facial feature detection and face recognition from 2D and 3D images. Pattern Recognition Letters 23, 1191–1202 (2002)
3. Pan, G., Han, S., Wu, Z., Wang, Y.: 3D Face Recognition using Mapped Depth Images. In: IEEE Computer Society Conference on CVPR (CVPR 2005) Workshops, vol. 03, p. 175 (2005)
4. Yuan, X., Lu, J., Yahagi, T.: A Method of 3D Face Based on Principal Component Analysis Algorithm. IEEE International Recognition Symposium on Circuits and Systems 4, 3211–3214 (2005)
5. Russ, T., Boehnen, C., Peters, T.: 3D Face Recognition Using 3D Alignment for PCA. In: IEEE Computer Society Conference on CVPR (CVPR 2006), vol. 2, pp. 1391–1398 (2006)
6. Taghizadegan, Y., Ghassemian, H., Naser-Moghaddasi, M.: 3D Face Recognition Method Using 2DPCA-Euclidean Distance Classification. ACEEE International Journal on Control System and Instrumentation 3 (2012)
7. Gervei, O., Ayatollahi, A., Gervei, N.: 3D Face Recognition Using Modified PCA Methods. World Academy of Science, Engineering & Technology 4, 254–257 (2010)
8. Cook, J., McCool, C., Chandran, V., Sridharan, S.: Combined 2D / 3D Face Recognition using Log-Gabor Templates. In: IEEE International Conference on Video and Signal Based Surveillance (AVSS 2006), p. 83 (2006)
9. Pamplona Segundo, M., Silva, L., Bellon, O.R.P.: Real-time scale-invariant face detection on range images. In: IEEE International Conference on Systems, Man and Cybernetics (SMC), pp. 914–919 (2011)
10. Jahanbin, S., Cho, H., Bovik, A.C.: Passive Multimodal 2-D+3-D Face Recognition Using Gabor Features and Landmark Distances. IEEE Transactions on Information Forensics and Security 6, 1287–1304 (2011)
11. Rossetti, A., De Menezes, M., Rosati, R., Ferrario, V.F., Sforza, C.: The role of the golden proportion in the evaluation of facial esthetics. Angle Orthodontist 83 (2013)
12. Farkas, L.G.: Anthropometry of the head and face. Raven Press, New York (1994)
13. Lowe, D.G.: Object recognition from local scale-invariant features. In: IEEE International Conference on Computer Vision, vol. 2, pp. 1150–1157 (1999)
14. Lindeberg, T.: Scale-space theory: A basic tool for analysing structures at different scales. Journal of Applied Statistics 21(2), 224–270 (1994)
15. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
16. http://www.vlfeat.org/~vedaldi/code/sift.html
17. Lindeberg, T.: Scale-space theory: A basic tool for analysing structures at different scales. Journal of Applied Statistics 21, 224–270 (1994)