BIOMETRIC RECOGNITION BY FUSING PALMPRINT AND HAND-GEOMETRY BASED ON MORPHOLOGY

*Wei-Chang Wang, Wen-Shiung Chen and Sheng-Wen Shih**

VIP-CCLab., Dept. of Electrical Engineering, National Chi Nan University, Taiwan
*Dept. of Computer Science and Information Engineering, National Chi Nan University, Taiwan

ABSTRACT

This paper presents a novel biometric recognition system fusing the palmprint and hand geometry of a human hand based on morphology. We utilize the image morphology and concept of Voronoi diagram to cut the image of the front of the whole palm apart into several irregular blocks in accordance with the hand geometry. Furthermore, statistic characteristics of the gray level in the blocks are employed as characteristic values. The experimental results show that the proposed system has an encouraging performance. The false acceptance rate (FAR) and false rejection rate (FRR) are reduced down to 0.0035% and 5.7692%, respectively.

Index Terms—Biometric Recognition, Palmprint, Hand Geometry, Morphology.

1. INTRODUCTION

Biometrics refers to the automatic identity authentication of a person on the basis of one's unique physiological or behavioral characteristics. To date, many biometric features, including fingerprint, hand geometry or palmprint, face, handwritten signature, voice, DNA, retina, and iris, have been studied and applied to the authentication of individuals. Compared with the other physical characteristics, palmprint and hand geometry has several advantages: (i) non-intrusive, (ii) stable features, (iii) low cost, and (iv) high user acceptance.

In the past time, most of palmprint and hand geometry recognition methods were proposed individually. Most palmprint recognition systems must cut out a square region of interest (ROI) [1]. The ROI of our method is the front of the whole palm. Obviously, we try to explore more discriminative characteristics of palm that may be omitted by traditional ROI. Because hand shape is irregular, we need powerful tools for image processing, morphological algorithms. Morphological operations used in this paper can be found in [2]. In this paper, we investigate and design a biometric recognition system fusing the features of palmprint and hand geometry. The recognition system is expected to provide a simple, low-cost, non-contact, comfortable and user-friendly acquisition mechanism that is apt to apply to mobile devices or notebooks for personal authentication.

2. OVERVIEW AND IMAGE ACQUISITION

2.1. System Overview

The proposed framework consists of four modules: image acquisition, image pre-processing, feature extraction, and recognition modules, as shown in Fig. 1. First, the mechanism of hand image acquisition module is the same as that of PHIA module designed in [3]. The pre-processing module utilizes the length of middle finger to define the region of palm. The feature extraction module employs the image morphology and concept of Voronoi diagram to divide the palm image into a number of irregular blocks in accordance with the hand geometry. Furthermore, some statistical measurements of the gray level for each block are extracted as feature values. Then, we present a two-stage recognition module based on the features of hand-shape and palmprint of a hand. At the first stage, we utilize the number of stripe regions of a palm to be the feature value. At the second stage, we encode the means or variances of the pixel gray levels of the blocks as feature values.

2.2. Image Acquisition Module

The mechanism of hand image acquisition module is as same as that of PHIA module designed in [3]. However, the hand image donators were asked to stretch and open their fingers as possible as they can, when the pictures of their hands were being taken. Basically, the hand image acquisition module employs fixed light source, black background and fixed gesture.

3. PRE-PROCESSING MODULE

The pre-processing module is to define a region that would be always the same for each different image of the same palm and to enhance the image of the region. The flowchart of pre-processing module is shown in Fig. 2 and described as follows.

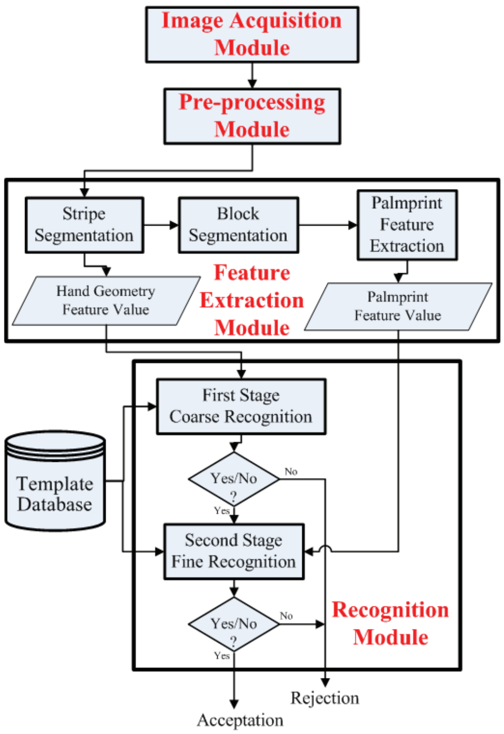


Fig. 1: System diagram.

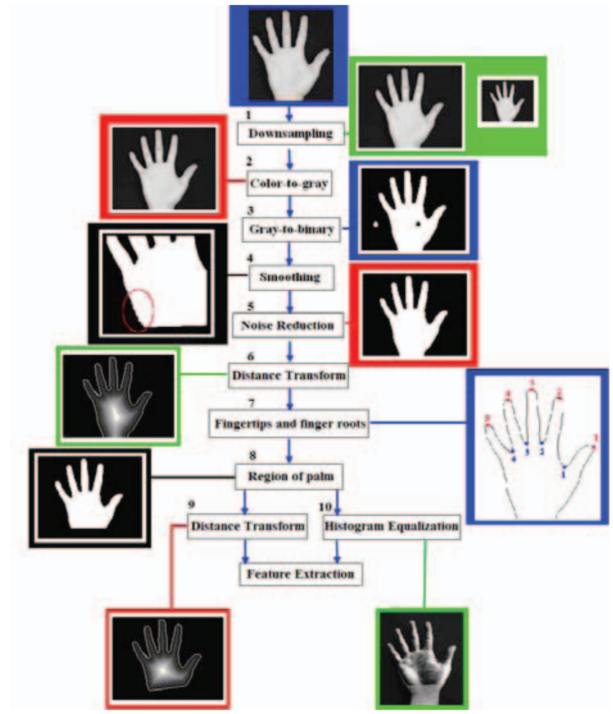
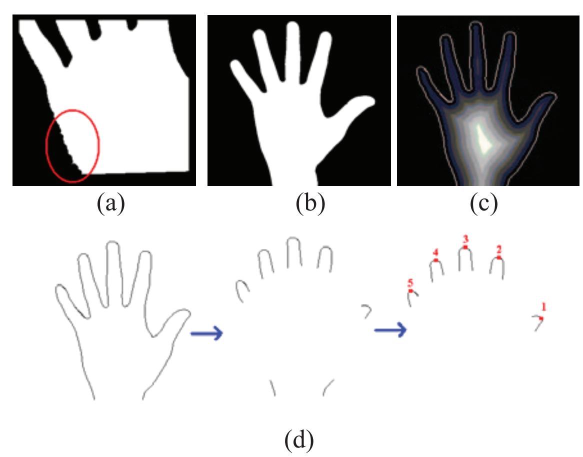


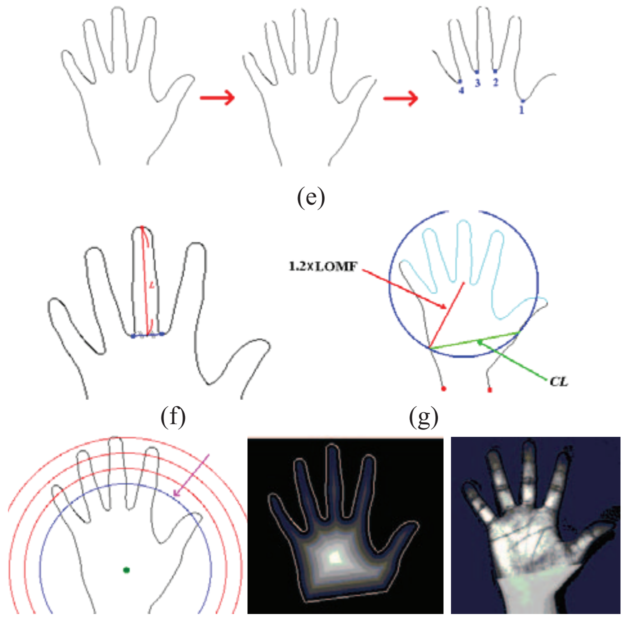
Fig. 2: Flowchart of pre-processing module.

First, the input image (1,920×2,560 pixels) is down-sampled to a lower resolution of 480×640 pixels. Then, the color image is converted to a gray-level image. Furthermore, the gray-level image is binarized. Because the binarization possibly generates serrate edges, as shown in Fig. 3(a), we borrow "opening" operation from morphology to smooth the serrate edges of a palm. Then, we use two morphological methods to reduce noise. Accordingly, we get a clear pattern of the palm, as shown in Fig. 3(b). Then, we calculate distance map [4] of the image and define the point with the greatest value (i.e., brightest point, as shown in Fig. 3(c)) as the Center Of a Palm (i.e., COP) [4].

Next, we locate the fingertips and finger roots. First, we employ Sobel filters to extract the boundary of a palm from the binary image. Then we delete the boundary points in the shrinking circle centered at COP in a bid to form 5~7 connected components, as shown in Fig. 3(h). If the number of connected components formed is greater than five, then we delete those components connected to the lowest row of the image. Then, we identify the farthest points from the COP in every connected component, and define them to be fingertips, as shown in Fig. 3(d). In order to locate finger roots, all the fingertips and some of their neighboring points in a complete hand contour are deleted. The components connected to the lowest row of image are also deleted. Furthermore, each of the nearest points away from the COP in four connected components is defined as a finger root which is numbered in order from right to left, as shown in Fig. 3(e). For left hand, the numbering order is reversed.

The length of line L is defined as the length of the middle finger (*LOMF*), which is the distance between the point of the third fingertip and the midpoint of the second and third finger roots, shown in Fig. 3(f). The *LOMF* is used to define the ROI of a palm. A circle with radius equal to 1.2 times the *LOMF* and centered at the midpoint of the second and third finger roots would cross the hand contour on many points. We connect the lowest two crossing points with a straight line segment, as the line segment *CL*, shown in Fig. 3(g). The region enclosed by the contour of the palm and above the line segment is defined as the ROI of a palm. We calculate the distance map of the ROI to redefine the COP, as shown in Fig. 3(i). We perform histogram equalization only on the ROI in gray-level image to enhance the palm image. An example of the result is shown in Fig. 3(j).





(h) (i) (j)

Fig. 3: (a) Example of serrate edges of a palm. (b) Hand pattern. (c) Example of Distance Transform. (d) The process of defining the fingertips. (e) The process of locating the finger roots. (f) Illustration of LOMF. (g) Illustration of defining ROI of a palm. (h) Illustration of a shrinking circle. (i) Distance transform of ROI of a palm. (j) A sample of locally enhanced image.

4. FEATURE EXTRACTION MODULE

The feature extraction module consists of three parts: stripe segmentation, block segmentation and palm feature extraction. The flowchart of feature extraction module is shown in Fig. 4(a).

4.1. Stripe Segmentation

The feature information used in this work is bimodal that fuses the features of hand-shape and palmprint, which is extracted by using mathematical morphology. First of all, we define an ***n*-th level stripe region**. The n-th level stripe region is the region that the binary palm image is eroded n-1 times subtracts the binary palm image eroded n times, where n is a positive integer. The radius of eroding disk, r_e, is defined as

$$r_e = LOMF/z, \quad (1)$$

where z is a constant, called the fixed ratio factor.

We use the binary palm image eroded n-1 times to subtract the binary palm image eroded n times to obtain the stripe regions, n=1, 2, 3, …, as shown in 4(b). The process of eroding the binary image of the palm with a disk proceeds till the eroded palm image is too small to erode. Note that, actually, the number of segmented stripe regions would be always the same for each different image of the same palm.

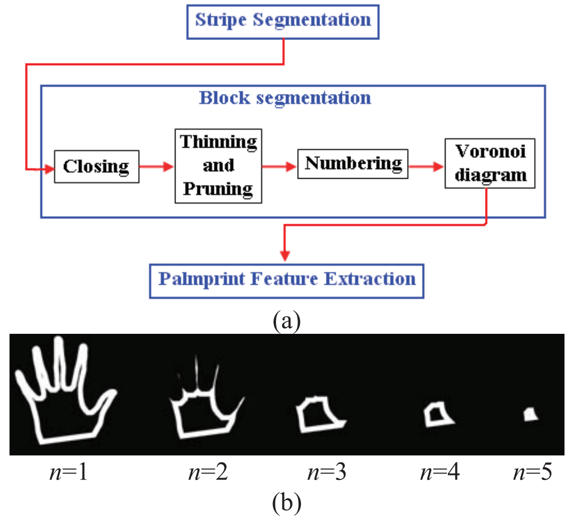


Fig. 4: (a) Flowchart of feature extraction module. (b) Examples of n-th stripe region, n=1, 2, 3, ….

4.2. Block Segmentation

After segmenting the ROI of a palm into several stripes, a block segmentation process is introduced to segment each stripe into a number of irregular shape blocks with different sizes. Sometimes a special kind of block, called *odd zone*, will appear. An odd zone is defined as any connected component other than the biggest connected component, like the connected component in the circle shown in Fig. 5(b). The block segmentation process, as shown in Fig. 4, is described as follows.

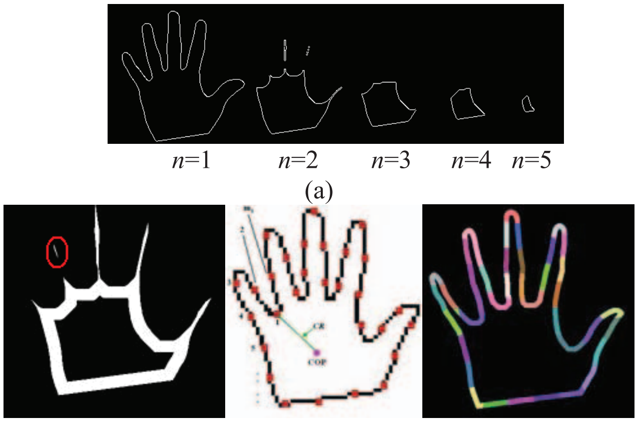
First, we use Sobel filters to extract the boundary of binary palm image eroded n-1 times, as shown in Fig. 5(a). After deleting odd zones, we borrow "closing" operation from morphology to turn the contour into one connected component. Then, we employ "thinning" and "pruning" operations of morphology to extract a simple and thin boundary.

Next, we define location points for implementing block segmentation. First, we equally divide the boundary points into m_n sets, where m_n means the number of blocks into which n-th level stripe region to be segmented. Then we mark the first points of every set with numbers in order, and call them location points, as shown in Fig. 5(c). And for all levels, the starting point should be on the line that connects the COP and the fourth finger root, as the line *CR* shown in Fig. 5(c). For the left hand, the numbering order is reversed. In order to segment the stripe region, if a point P_a in the n-th level stripe region is the closest point to the location point i, then the point P_a is assigned to block i. This is the concept of Voronoi diagram. An example of block segmentation of the first level stripe region is shown in Fig. 5(d).

4.3. Palmprint Feature Extraction

Finally, the entire ROI of a specific palm is segmented into

(m_1+m_2+m_3+…+m_n) blocks totally. For each block, a statistical quantity, such as mean or variance, of the gray levels is computed as the palmprint feature value.



n=1 n=2 n=3 n=4 n=5

(a)

(b) (c) (d)

Fig. 5: (a) Examples of the outer contours of n-th level stripe regions. (b) An example of odd zone. (c) Numbering outer contours of 1-st level stripe region. (d) The block segmentation of 1-st level stripe region.

5. TWO-STAGE RECOGNITION

Recognition module is composed of two stages. At the first stage, it utilizes the number of stripe regions of a palm to be the feature value. Second, it encodes the means or variances of the pixel gray levels of the blocks as feature values. The procedure of two-stage recognition, shown in Fig. 1, is described as follows.

Stage 1: *Coarse recognition*. If the number of stripe regions on the tested image is different from that of the training image, then we claim that these two images are taken from two different palms. Otherwise, the tested image enters into Stage 2 recognition.

Stage 2: *Fine recognition*. We use linear support vector machine to classify feature vectors with the same length into two classes, one class is claimed user and the other class is not claimed users.

6. EXPERIMENTAL RESULTS

We implement and test the proposed schemes on our own hand image database. It comprises 1,560 hand images captured from 260 different hands. Three user's hand images and 777 (259×3) non-user's images are used to train an SVM hyperplane to separate these two classes. Means of the gray levels in all the blocks are employed as the feature values. *NB* stands for the number of blocks in n-th level stripe region. *NB* is defined by the following rule. If $NB=k$ for n=1, then we set $NB=\frac{k}{2}-(\frac{k}{4}-1)(n-2)$ for n>1. From In addition, we implement the "eigenpalm" method [6] on our hand image database and compare them.

Table 1, when z=4.65, this system achieves a desirable performance. In addition, we implement the "eigenpalm" method [6] on our hand image database and compare them.

Table 1: Result of experiment of fixed ratio factor

z	$NB(n=1)$	$NB(n>1)$	FAR (%)	FRR (%)
1.00	40	20-9(n-2)	0.0035	34.3590
2.40	40	20-9(n-2)	0.0035	9.7436
2.80	40	20-9(n-2)	0.0084	7.9487
3.30	40	20-9(n-2)	0.0054	6.7949
4.65	40	20-9(n-2)	0.0035	5.7692
5.00	40	20-9(n-2)	0.0040	5.8974

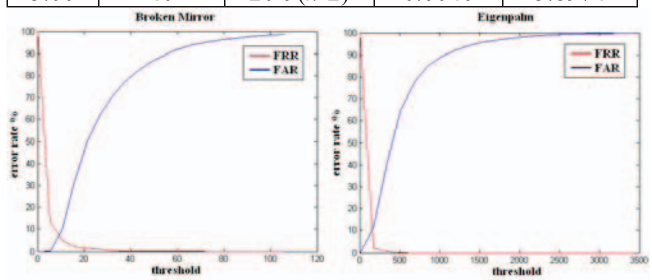


Fig. 6: Comparison of our method to "eigenpalm."

7. CONCLUSION

This paper presents a biometric recognition system fusing palmprint and hand geometry based on morphology. We used the geometry of hand to segment the images of palms, and employed statistical characteristics of the palmprint as feature values. So our method utilizes the image of the front of the whole palm. Our method can obtain a better recognition rate, and is more memory-efficient as compared with the "eigenpalm" method. In addition, our system does not need the process of size normalization. Altogether, this method achieves a low FAR and FRR.

8. REFERENCES

[1] C. C. Han, H. L. Chang, C. L. Lin and K. C. Fan, "Personal authentication using palmprint features," *Pattern Recognition*, vol. 36, no. 2, pp. 371-381, 2003.

[2] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Prentice Hall, New Jersey, 2002.

[3] W.-S. Chen, Y.-S. Chiang, C.-M. Hsieh, S.-W. Shih and Y.-H. Chiu, "Biometric verification with hybrid features of palmprint and hand geometry based on wavelet transform and statistical moments," *18th IPPR Conference on Computer Vision, Graphics and Image Processing*, Taipei, Taiwan, 2005.

[4] G. Borgefors, "Distance transformations in digital images," *Computer Vision, Graphics, and Image Processing*, vol. 34, no. 3, pp. 344-371, Jun. 1986.

[5] A. Kumar and D. Zhang, "Personal recognition using hand shape and texture," *IEEE Trans. on Image Processing*, vol. 15, no. 8, pp. 2454-2461, 2006.

[6] G. Lu, D. Zhang and K. Wang, "Palmprint recognition using eigenpalms features," *Pattern Recognition Letters*, vol. 24, no. 9-10, pp. 1463-1467, Jun. 2003.