

# Palmprint Verification using Gradient Maps and Support Vector Machines

Chun-Wei Lu<sup>†</sup>, Ivy Fan<sup>†</sup>, Chin-Chuan Han<sup>\*</sup>, Jyh-Chian Chang<sup>#</sup>, Kuo-Chin Fan<sup>††</sup> and H.Y. Mark Liao<sup>†</sup>

<sup>†</sup>Inst. of Info. Sci., Academia Sinica, Taiwan

<sup>††</sup>Department of Comp. Sci. and Info. Eng., National Central University, Taiwan

<sup>\*</sup>Department of Comp. Sci. and Info. Eng., National United University, Taiwan

<sup>#</sup>Department of Comp. Sci. and Info. Eng., Chinese Culture University, Taiwan

**Abstract** — With the urgent demand in information security, biometric feature-based verification systems have been extensively explored in many application domains. However, the efficacy of existing biometric-based systems is unsatisfactory and there are still a lot of difficult problems to be solved. Among many existing biometric features, palmprint has been regarded as a unique and useful biometric feature due to its stable principal lines. In this paper, we proposed a new method to perform palmprint recognition. We extract the gradient map of a palmprint and then verify it by a trained support vector machine (SVM). The procedure can be divided into three steps, including *image preprocessing, feature extraction, and verification*. We used the multi-spectral palmprint database prepared by Hong Kong PolyU [14] which included 6000 palm images collected from 250 individuals to test our method. The experimental results demonstrate our proposed method is reliable and efficient to verify whether the person is genuine or not.

## I. INTRODUCTION

Recently, personal authentication has become a vital and highly demanded technique as a foundation of many applications, like security access system, time attendance system, and forensics science. In the past decade, many biometric features have been used for identification purpose, including fingerprint, iris, retina, face, hand geometry, and palmprint. A palmprint, as one of the extremely important biometric features, has attracted great attention in these years due to its reliable line structure, ridge structure, and texture information. Combining these features together makes a powerful feature representation scheme. However, there is still great room to improve the accuracy, the efficiency, and the power of anti-spoofing.

In order to deal with the aforementioned problems, a large number of researches have been proposed in the past decade [1-10] [14-17]. Algorithms related to palmprint feature extraction can be categorized into three types: 1) subspace feature learning; 2) line-like feature extraction; and 3) texture-based coding. The algorithms fall into the subspace learning category include principal component analysis (PCA) [1], fisher's linear discriminant analysis (LDA) [2], and locality preserving projection (LPP) [3]. The algorithms belonging to this category consider a palmprint image as a whole and the representation of the image is accomplished by performing dimensionality reduction. Although a subspace-based

representation scheme is suitable for representing palmprint texture information, the requirement of establishing a large database is difficult for most of the researchers working in this field.

As to the line-like feature representation scheme, one needs to detect and extract salient line segments from a palm image. The line features of a palm image are, in fact, the most significant features that can be used to characterize a palm image. Han et al. [4] use Sobel and morphology operations to extract the palm features. Lin et al. [5] extract the principal palmprint features by applying the hierarchical decomposition mechanism to the region-of-interest (ROI) regions, which include directional and multi-resolution decompositions. Zhang et al. [6] proposed a modified finite random transform to explicitly extract the principal lines even though a palmprint image contains many long and strong wrinkles.

Another direction of palmprint recognition research is to perform texture-based coding. Usually, texture-based features can better characterize a palmprint and they are robust to illumination change. In [7], Zhang et al. proposed a new palmprint coding approach which extracts the orientation information by Gabor filter and encodes the orientation response into a 3-bit competitive code.

In this paper, we propose a four-step preprocessing process to identify the location of a normalized ROI. The details of feature extraction are described in order to construct the gradient map. SVM is employed and it is equipped with the one-against-all strategy to verify each individual palmprint. Finally, some experimental results are shown to demonstrate the effectiveness of the proposed approach.

## II. PREPROCESSING

The whole procedure is described in details as follows:

Step 1: *Binary thresholding*. To extract the foreground objects from the background, binarization is an essential step to convert a grayscale image into a binary form. By applying the *mode method* [11], the histogram of a grayscale image is analyzed first to automatically determine a local minimum between two local maxima as a threshold. When a grayscale value is lower than this threshold, we set the pixel value as '0'; otherwise, the pixel value is set '255' to segment the palm shape as shown in Fig. 1 (a).

Step 2: *Border tracing*. After the binarization step, the *inner border tracing* algorithm [11] is adopted to trace the contour of a palm shape. In the beginning, the starting point  $W_m$  is set at the middle point of the intersection line segment formed by the wrist and the bottom margin of the palm shape as shown in Fig. 1 (b). Then, all contour pixels of the palm shape are traced in the anti-clockwise direction. The eight neighboring directions are applied to describe the relative position of those traced points more precisely. The coordinates of contour pixels would be recorded sequentially as  $P_1, P_2, \dots, P_n$ .

Step 3: *Finger-webs locating*. Since  $W_m$  and  $P(i)$  have been determined in the previous steps, we can calculate the Euclidean distance between the middle wrist point and each contour point to draw a distribution diagram. The pattern of the diagram is similar to the geometric shape of a palm. To meet the speed and reliability requirements, a contour and curvature based method [12] is adopted to find the location of the four finger-webs ( $FW_1, FW_2, FW_3$ , and  $FW_4$  as indicated in Fig. 1 (b)) and the four finger tips.

Step 4: *ROI extraction*. In this step, the locations of the second and the fourth finger-webs are selected as the datum points to determine the ROI from the palm region. One thing to be noted is that a user would put his/her hand at any position and direction with pegs-free device. So the final step is ROI normalization, and this step will need to handle rotation and scaling. The line  $\overline{FS}$  is determined by connecting  $FW_2$  and  $FW_4$  as shown in Fig.1 (b). The palm image is rotated with angle  $\theta$  between  $\overline{FS}$  and the vertical line. For the scaling part,  $FW_2$  and  $FW_4$  are employed to normalize each ROI to the same size.

Fig. 1 (c) shows a normalized ROI. The proposed normalization process has two advantageous points. First, any displacement of a palm image can be adjusted. Second, the redundant information can be removed and the important information will be retained.

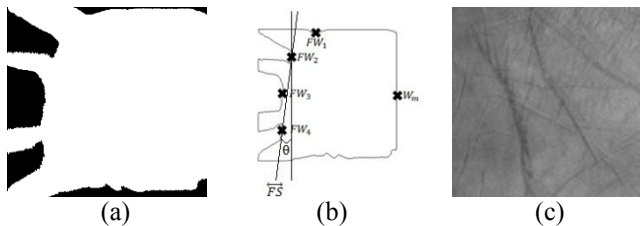


Fig. 1. (a) Binarized palmprint image. (b) Location of starting point  $W_m$  and four datum points  $FW_1, FW_2, FW_3$ , and  $FW_4$ . (c) Normalized ROI.

### III. FEATURE EXTRACTION

After pre-processing, the next step is to perform feature extraction. Usually, a palmprint is composed of principal line segments, wrinkles, and ridges. Principal line segments of a palmprint have been proven to be one of the most effective features in palmprint verification. By observing a segmented ROI carefully, line features possess lower grayscale values against the remaining regions. After the preprocessing stage, we describe how the principal line segments are extracted.

#### A. Gradient Map Construction

The line-like features can be considered as edges. Therefore, we can apply edge detectors to extract them. Conventional edge detection methods can be categorized into two types – gradient-based and laplacian-based. Since a Laplacian-based edge detector is more sensitive to noises, we adopt a gradient-based edge detector – the well-known Sobel edge detector to detect edges and then construct a gradient map.

The direction of the gradient at point  $f(x,y)$  can be calculated by four directional operators –  $S_1, S_2, S_3$ , and  $S_4$  with the angle of  $0^\circ, 45^\circ, 90^\circ$ , and  $135^\circ$ , respectively. The function of these operators is expressed as follows:

$$f_g = \max_{i=1 \text{ to } 4} (f * S_i), \quad (1)$$

where  $f_g$  is the gradient map of decomposed images, and symbol  $*$  denotes the convolution operator. Every maximum received by applying the above four directional operators is selected as the representative gradient of its corresponding coordinate.

#### B. Iterative Threshold

In the constructed gradient map, many small wrinkles were detected together with regular edges. Usually, these wrinkles will affect the genuine matching and degrade the verification accuracy. Under these circumstances, we adopt the iterative threshold method proposed in [13] to eliminate the above mentioned redundant wrinkles. The detailed procedure is described as follows:

Step 1: Initialize the threshold,  $Th(i)$ , by a random number ranging from 0 to 255 to segment the gradient map into two parts – *feature points* and *non-feature points*.

Step 2: At the  $i_{th}$  iteration, calculate  $\mu_n(i)$  and  $\mu_f(i)$  as the mean of the non-feature points and the feature points, respectively. The threshold is determined by step 3 of the previous iteration.  $\mu_n$  and  $\mu_f$  can be calculated as follows:

$$\mu_n(i) = \sum_{(x,y) \in \text{non-feature points}} f(x,y) / N_n, \quad (2)$$

$$\mu_f(i) = \sum_{(x,y) \in \text{feature points}} f(x,y) / N_f, \quad (3)$$

where  $N_n$  and  $N_f$  represent the pixel number of non-feature points and feature points, respectively.

Step 3: Set  $Th(i+1)$  as a new threshold for the next iteration.

$$Th(i+1) = (\mu_n(i) + \mu_f(i)) / 2. \quad (4)$$

Step 4: Terminate the iteration process whenever  $Th(i+1) = Th(i)$ , otherwise return to step 2.

The gradient map of an ROI can then be transformed into binary format and the principal feature points are preserved.

#### C. Morphological Operation

Although an iterative thresholding process can effectively eliminate some unwanted points, some holes and isolated blobs still remain. A morphological operator with square structure elements is adopted to solve the problem. By sliding the structure elements across an image and taking the convolution, one is able to process the image based on the provided morphological operators.

The two operations that we introduce to operate on the target image are dilation and erosion. By applying these two morphological operators, one is able to filter out noises and non-feature points.

#### D. Count Filter

After performing the above three steps, some salient points are retained in the processed ROIs, and a *count filter* [5] is employed to construct feature vectors. A count filter is suitable for our system, because it can characterize the pattern of principal palmprint features. The operation of this filter is expressed as follows:

$$C_f(x, y) = \text{count\_filter}(FP(x_i, y_i)), i \in \{1, 2, \dots, \frac{m}{n} \times \frac{m}{n}\}, (5)$$

where  $C_f(x, y)$  is the number of  $FP(x_i, y_i)$  sitting inside the range of  $\frac{m}{n} \times \frac{m}{n}$  block.

After the operation of a count filter, we decompose the processed ROIs into several non-overlapping blocks, and calculate the number of feature points in each block to construct feature vectors. For example, if we decompose an image into 16 divisions along  $x$  and  $y$  axis, then the total number of blocks will be  $16 * 16 = 256$  (as shown in Fig. 2) and the dimension of a feature vector will be 256. Obviously, it can solve the offset problem of feature points in images.

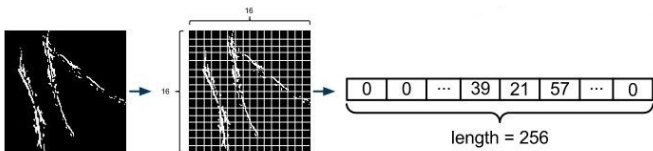


Fig. 2. The construction of gradient map with  $16 * 16$  blocks, and the dimension of a feature vector will be 256.

#### IV. MODELING AND VERIFICATION

The support vector machine (SVM) is an effective tool for data classification. The goal of SVM modeling is to find a set of appropriate hyper-planes in a high dimensional space and then use these hyper-planes to classify input samples. The middle of an SVM margin is an optimal hyper-plane to separate two classes. Therefore, this margin provides the largest distance from sample points to it. All points sitting on the boundaries are called support vectors which are calculated by numerical optimization during the training phase. While the samples are non-linearly separable, these samples would be mapped to a higher dimension feature space by a non-linear function. Polynomial and radial basis function (RBF) are two non-linear kernel functions that are commonly used. Since the samples taken in our problem domain are high dimensional features, we project these samples into a two dimensional feature space by principal component analysis. By analyzing the data distribution empirically, we choose RBF as our kernel functions.

A typical SVM is a binary classifier. However, the problem we encounter is a multi-classification problem. Therefore, we need to train a multi-class classifier. The way we adopted is to build a classifier of this sort through combining more than

two binary classifiers and make them work together. Since the problem is an  $n$ -class problem, we adopted the one-against-all approach to construct a model. Since the one-against-all approach is exclusive, it can prevent the intruders from logging into the system. Therefore, we use it to evaluate the performance of the proposed method.

#### V. EXPERIMENTAL RESULTS

##### A. PolyU Multi-spectral Palmprint Database

The PolyU multi-spectral palmprint images were collected from 250 volunteers. The database contains 195 males and 55 females and their age distribution was from 20 to 60 years old. The palmprint images were collected at two different time periods within a window of 9 days. In each time period, all individuals were asked to provide 6 images for each of the left and right palms, respectively. Therefore, the total number of images on each band was 6,000 from 500 different palms. During the same time period, four images were collected from four different bands at one shot, including red, green, blue, and near-infrared illumination. The resolution of the original images in the database was  $352 * 288$  with 256 gray-levels.

##### B. Palmprint Verification by Single Band

Among the four-band images of the PolyU palmprint database, we chose the blue band to verify our proposed method because it has strong line structures.

We use three values to evaluate the efficiency of our proposed method. First, the false acceptance rate (FAR) which measures the frequency of falsely accepted imposters. Second, the false rejection rate (FRR) which measures the frequency of falsely rejected genuine clients. Third, the equal error rate (EER) is defined as a value where FAR equals to FRR.

In the first experiment, we applied three experiment sets with different amount of training samples. The first set which contained 31 individuals was selected at random as training data, and the rest of 469 individuals were used as testing data. The second and third set contained 51 and 101 individuals, respectively. The results of receiver operating characteristic (ROC) curves were evaluated by one-against-all strategy shown in Fig. 3 (a), it indicates the more training data would result in lower FAR and FRR.

##### C. Anti-spoofing Verification

A good biometric verification system should have capability of anti-spoofing to prevent the intruders from logging into the system. The second experiment is described as follows: we split the testing data into two parts, the inside group and the outside group. In the inside group, the users have registered in the database, and the outside group represents some intruders. In this experiment, 100 individuals were picked in random as genuine to train the models. The inside test verified the accuracy of genuine attempt, and the outside test was employed to evaluate the efficiency of intruder prevention. As shown in Fig. 3 (b), the EER value of inside test was as low as 0.5437% and the value of outside test was 1.193%.

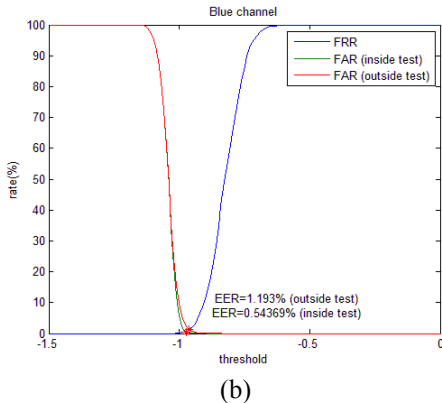
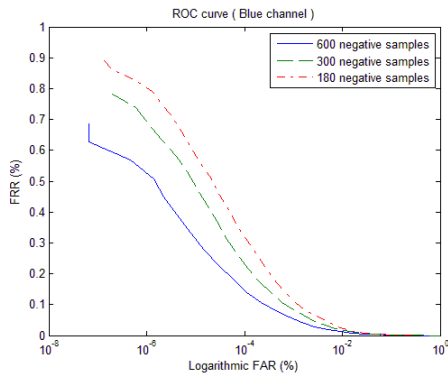


Fig. 3. Verification results are evaluated by blue band. (a) ROC curve with different amount of negative samples. (b) Inside test and outside test.

Unlike the matching method of Zhang et al. [14] by using the Hamming distance, our proposed method need to train the models. Finally, we compare the performance of our method with others which use the same database [15] - [17]. Among the three chosen methods, the best EER was 1.20%. However, our EER was as low as 0.8344% as shown in Fig. 4.

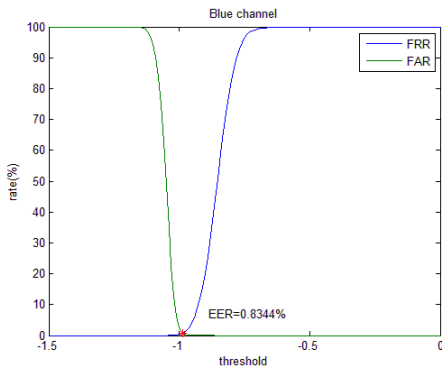


Fig. 4. The EER value at blue band.

## VI. CONCLUSIONS

This paper proposes an effective authentication method by constructing gradient map and only the important feature points are retained. SVM is then adopted to train a set of hyper-planes to separate different groups of samples with the one-against-all strategy. Since the different bands have

different texture information, the performance is evaluated by using blue band. The proposed method received a low EER value as 0.8344%. The encouraging experimental results reveal that our proposed method is reliable for verification system.

## REFERENCES

- [1] T. Connie, A. Teoh, M. Goh, D. Ngo, "Palmprint recognition with PCA and ICA," *Proceedings of Image and Vision Computing New Zealand*, pp. 227-232, 2004.
- [2] X. Wu, D. Zhang, K. Wang, "Fisherpalms based palmprint recognition," *Pattern Recognition Letters*, Vol. 24, pp. 2829-2838, 2003.
- [3] D. Hu, G. Feng, and Z. Zhou, "Two-dimensional locality preserving projection (2dlpp) with its application to palmprint recognition," *Pattern Recognition*, 40, 1, pp. 339-342, 2007.
- [4] C. C. Han, H. L. Cheng, C. L. Lin, K. C. Fan, "Personal authentication using palmprint features," *Pattern Recognition*, vol. 36, pp. 371-381, 2003.
- [5] C. L. Lin, Thomas C. Chuang, and K. C. Fan, "Palmprint Verification Using Hierarchical Decomposition", *Pattern Recognition*, vol. 38, issue 12, pp. 2639-2652, 2005.
- [6] D. Hu, W. Jia, and D. Zhang, "Palmprint verification based on principal lines," *Pattern Recognition*, 41, 4, pp.1316-1328, 2008.
- [7] A.W.K. Kong, D. Zhang, "Competitive coding scheme for palmprint verification," in *Proceedings of International Conference on Pattern Recognition*, vol. 1, pp. 520-523, 2004.
- [8] A. Kumar, H. C. Shen, "Palmprint identification using PalmCodes," in *Proceedings of 3rd International Conference on Image and Graphics* pp. 258-261, 2004.
- [9] X. Wu, D. Zhang, K. Wang, "Fisherpalms based palmprint recognition," *Pattern Recognition Letters*, 24 (15) pp.2829-2838, 2003.
- [10] C. L. Lin, and K. C. Fan, "Biometric Verification Using Thermal Images of Palm-dorsa Vein-patterns," *IEEE Transactions on Circuits and Systems for Video Technology*, vol.14, no. 2, pp. 199-213, 2004.
- [11] M. Sonka, V. Hlavac and R. Boyle, "Image Processing, Analysis, and Machine Vision", Second edition, PWS publishing, New York, 1999.
- [12] Lee T, Hollerer T and Handy AR, "Markerless Inspection of Augmented Reality Objects Using Fingertip Tracking," *IEEE International Symposium on Wearable Computers*, pp. 83-90, 2007.
- [13] T. W. Ridler and S. Calvard, "Picture Thresholding Using an Iterative Selection Method," *IEEE transaction on System, Man and Cybernetics*, vol. 8, pp. 630-632, 1978.
- [14] David Zhang, Zhenhua Guo, Guangming Lu, Lei Zhang, and Wangmeng Zuo, "An Online System of Multi-spectral Palmprint Verification," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 2, pp. 480-490, 2010.
- [15] V. Struc, N. Pavessic, "Phase congruency features for palmprint Verification," *IET signal Process*, vol. 3, issue 4, 2009.
- [16] Ali M., Ghafoor, M., Taj, I.A., and Hayat K., "Palm Print Recognition Using Oriented Hausdorff Distance Transform," *Frontiers of Information Technology (FIT)*, 2011.
- [17] Kumar, A., Hanmandlu, M., Madasu, V.K., and Vasikarla S., "A palm print authentication system using quantized phase feature representation," *Applied Imagery Pattern Recognition Workshop (AIPR)*, 2011.