ISSN: 0976-9102(ONLINE)
DOI: 10.21917/ijivp.2012.0051


# PERSONAL AUTHENTICATION USING PALMPRINT WITH SOBEL CODE, CANNY EDGE AND PHASE CONGRUENCY FEATURE EXTRACTION METHOD

**Jyoti Malik[1], G. Sainarayanan[2] and Ratna Dahiya[3]**
[1,3]*Department of Electrical Engineering, National Institute of Technology, Kurukshetra, India*
E-mail: [1]jyoti_reck@yahoo.com and [3]ratna_dahiya@yahoo.co.in
[2]*HCL Technologies Limited, Chennai, India*
E-mail: sai.jgk@gmail.com

***Abstract***

***Palmprint recognition refers to recognizing a person on the basis of palmprint features. In this paper, we have proposed a palmprint based biometric authentication method with improvement in accuracy, so as to make it a real time palmprint authentication system. Several edge detection methods, Directional operator, Wavelet transform, Fourier transform etc. are available to extract line feature from the palmprint. In this paper, Sobel Code operators, Canny edge and Phase Congruency methods are applied to the palmprint image to extract palmprint features. The extracted Palmprint features are stored in Palmprint feature vector. The corresponding feature vectors are matched using sliding window with Hamming Distance similarity measurement method. In this paper, a Min Max Threshold Range (MMTR) method is proposed that helps in increasing overall system accuracy by reducing the False Acceptance Rate (FAR). The person authenticated by reference threshold is again verified by second level of authentication using MMTR method. Experimental results indicate that the MMTR method improves the False Acceptance Rate drastically. The accuracy improvement leads to proposed real time authentication system.***

***Keywords:***

***Canny Edge, Hamming Distance, Palmprint Identification, Phase Congruency, Sliding Window, Sobel Code***

## 1. INTRODUCTION

Biometric identification of a person by his/her physiological or behavioral characteristics, like face, finger, palmprint, gait, signature, voice etc. has become increasingly popular in modern personal identification and verification systems [3][4]. Here, palmprint biometric is one of the most desirable biometric that can independently authenticate a person by palmprint features. Palmprint acts as a reliable biometric because it cannot be duplicated and the features in a palmprint are permanent. The palmprint biometric has several advantages over other biometric methods like, low cost capturing device, easy to collect, user friendly, unique, permanent features etc.

Palmprint features include geometry features, line features, minutiae points, delta point features. Several methods are available in the literature to extract palmprint features. The extraction of palm lines using stack filter [12], derivative of Gaussian [13], Fourier transform [14], wavelet transform [15] have been used earlier. In this paper, the palmprint line feature that includes principal lines, wrinkles and ridges is extracted using Sobel Code operator, Canny edge and Phase congruency method [15-17]. Palmprint line features are extracted and stored in Palmprint feature vector that are matched by Hamming Distance similarity measurement.

The rest of the paper is organized as follows: Section 2 defines the palmprint authentication system. Section 3 explains about feature extraction by Sobel Code operator, Canny edge and Phase congruency method. Section 4 discusses the feature matching by hamming distance and sliding window method. Section 5 discusses about the Min Max Threshold Range (MMTR) method. Section 6 explains the experimental results. Section 7 includes the conclusion.

## 2. PALMPRINT AUTHENTICATION SYSTEM

In this paper, the palmprint authentication system is divided in following two subsystems:

(a) Pre- Authentication System

(b) Authentication System

In Pre-authentication system, a database of Palmprint features is prepared. In addition, Reference threshold and Min Max threshold values are also identified and stored in database. These values will be used in Authentication system.

In Authentication system, the authenticity of a person is identified with the help of Reference threshold and Min Max threshold values stored in Pre-authentication system database.

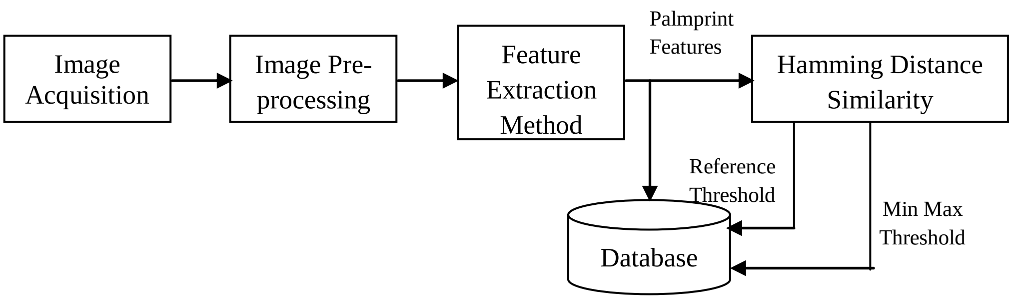



Fig.1. Palmprint Pre-Authentication system

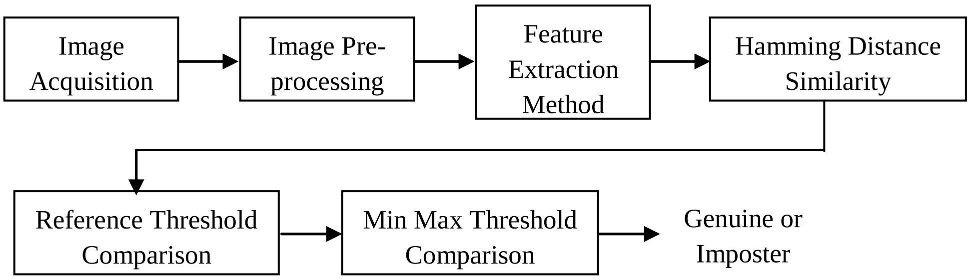



Fig.2. Palmprint Authentication System

## 3. LINE FEATURE EXTRACTION

Palmprint biometric is rich in features like geometry features, line features, datum points, delta features and minutiae features. In this paper, the palmprint line feature that includes principal lines, wrinkles and ridges is extracted using three different edge detection methods.

### 3.1 SOBEL CODE OPERATORS

Sobel Code operators are used to detect edges in specific direction. Sobel Code operators operate in four different directions like (0°, 45°, 90° and 135°). The Sobel Code operators are convolved with the palmprint image and Sobel-Palmprint features are extracted. The sample of 3×3 Sobel Code Operator convolution with the palmprint image is shown in Fig.3.

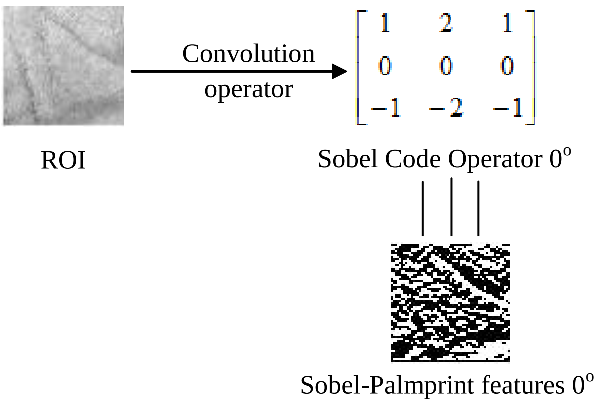



Fig.3. Feature extraction by Sobel Code operators

The matrices for 3×3, 5×5 and 7×7 three different size Sobel Code operators are mentioned in the Fig. 4. The 3×3 Sobel Code Operator is applied to the palmprint image and Sobel-Palmprint features are extracted. The Sobel-Palmprint features extracted are shown in Fig.4.

3×3 Sobel Code Operator

$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \qquad \begin{bmatrix} 2 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -2 \end{bmatrix}$$

Sobel 0° Sobel 45°

$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \qquad \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix}$$

Sobel 90° Sobel 135°

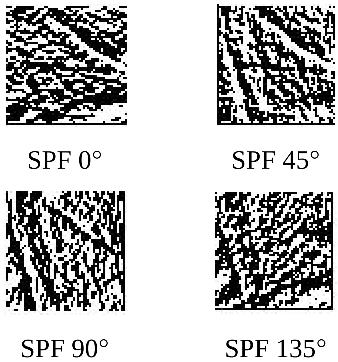



Fig.4. 3×3 Sobel Code Operator and Sobel-Palmprint features for 60×60 palmprint size

Similarly, Sobel-Palmprint features for 5×5 and 7×7 Sobel Code Operators are shown in Fig.5 and Fig.6.

5×5 Sobel Code Operator

$$\begin{bmatrix} 1 & 2 & 3 & 2 & 1 \\ 1 & 2 & 4 & 3 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ -1 & -3 & -4 & -3 & -1 \\ -1 & -2 & -3 & -2 & -1 \end{bmatrix}$$

Sobel 0°

$$\begin{bmatrix} 3 & 2 & 2 & 1 & 0 \\ 2 & 4 & 3 & 0 & -1 \\ 2 & 3 & 0 & -3 & -2 \\ 1 & 0 & -3 & -4 & -2 \\ 0 & -1 & -2 & -2 & -3 \end{bmatrix}$$

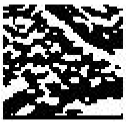
Sobel 45°

$$\begin{bmatrix} 1 & 1 & 0 & -1 & -1 \\ 2 & 3 & 0 & -3 & -2 \\ 3 & 4 & 0 & -4 & -3 \\ 2 & 3 & 0 & -3 & -2 \\ 1 & 1 & 0 & -1 & -1 \end{bmatrix}$$

Sobel 90°

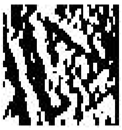
$$\begin{bmatrix} 0 & -1 & -2 & -2 & -3 \\ 1 & 0 & -3 & -4 & -2 \\ 2 & 3 & 0 & -3 & -2 \\ 2 & 4 & 3 & 0 & -1 \\ 3 & 2 & 2 & 1 & 0 \end{bmatrix}$$

Sobel 135°

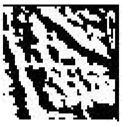


SPF 0°

SPF 45°



SPF 90°



SPF 135°

Fig.5. 5×5 Sobel Code Operator and Sobel-Palmprint features for 60×60 palmprint size

7×7 Sobel Code Operator

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 3 & 2 & 1 \\ 1 & 3 & 4 & 5 & 4 & 3 & 1 \\ 1 & 4 & 5 & 6 & 5 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -4 & -5 & -6 & -5 & -4 & -1 \\ -1 & -3 & -4 & -5 & -4 & -3 & -1 \\ -1 & -2 & -3 & -4 & -3 & -2 & -1 \end{bmatrix}$$

Sobel 0°

$$\begin{bmatrix} 4 & 3 & 2 & 1 & 1 & 1 & 0 \\ 3 & 5 & 4 & 3 & 4 & 0 & -1 \\ 2 & 4 & 6 & 5 & 0 & -4 & -1 \\ 1 & 3 & 5 & 0 & -5 & -3 & -1 \\ 1 & 4 & 0 & -5 & -6 & -4 & -2 \\ 1 & 0 & -4 & -3 & -4 & -5 & -3 \\ 0 & -1 & -1 & -1 & -2 & -3 & -4 \end{bmatrix}$$

Sobel 45°

$$\begin{bmatrix} 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 2 & 3 & 4 & 0 & -4 & -3 & -2 \\ 3 & 4 & 5 & 0 & -5 & -4 & -3 \\ 4 & 5 & 6 & 0 & -6 & -5 & -4 \\ 3 & 4 & 5 & 0 & -5 & -4 & -3 \\ 2 & 3 & 4 & 0 & -4 & -3 & -2 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \end{bmatrix}$$

Sobel 90°

$$\begin{bmatrix} 0 & -1 & -1 & -1 & -2 & -3 & -4 \\ 1 & 0 & -4 & -3 & -4 & -5 & -3 \\ 1 & 4 & 0 & -5 & -6 & -4 & -2 \\ 1 & 3 & 5 & 0 & -5 & -3 & -1 \\ 2 & 4 & 6 & 5 & 0 & -4 & -1 \\ 3 & 5 & 4 & 3 & 4 & 0 & -1 \\ 4 & 3 & 2 & 1 & 1 & 1 & 0 \end{bmatrix}$$

Sobel 135°

SPF 0°

SPF 45°

SPF 90°

SPF 135°

Fig.6. 7×7 Sobel Code Operator and Sobel-Palmprint features for 60×60 palmprint size

The Sobel-Palmprint features in Eq.(1) to Eq.(4) are used to obtain feature vector as in Eq.(5),

$$SPF0 = \text{Palmprint} * Sobel0^{o} \quad (1)$$

$$SPF1 = \text{Palmprint} * Sobel45^{o} \quad (2)$$

$$SPF2 = \text{Palmprint} * Sobel90^{o} \quad (3)$$

$$SPF3 = \text{Palmprint} * Sobel135^{o} \quad (4)$$

$$FV_i = [SPF0_i, SPF1_i, SPF2_i, SPF3_i] \quad (5)$$

where, *SPF* denotes Sobel-Palmprint features, Palmprint*$Sobel0^{o}$ signifies convolution of palmprint with Sobel operator of orientation 0°, *FV* is feature vector and $i$ can be 3×3, 5×5 and 7×7 Sobel Code operator.

### 3.2 CANNY EDGE DETECTION METHOD

The Canny edge detector is an edge detection method to find optimal edges in a palmprint image. The canny edge detection method can be implemented in following steps:

(a) First, the image is smoothed using a filter like Gaussian filter etc., which can remove noise from the original image, so that edges can be found out from the smoothened image.

(b) The next step is to find out the image gradient. The gradient magnitude or edge strength |G| can be calculated by the formula given below,

$$|G| = |G_x| + |G_y| \tag{6}$$

where, $G_x$ denotes the gradient in x-direction, $G_y$ denotes the gradient in y-direction.
The edge direction is given by the formula,

$$\text{Theta} = \text{invtan}(G_y / G_x). \tag{7}$$

After finding the edge direction using Eq.(7), the edge direction is related to a direction that can be marked in an image.

(c) Once the edge directions are known, the next step is to trace along the edge directions and suppress the pixel that is not at the maximum (not considered as edge).

(d) Finally, hysteresis is used to check out the remaining pixels that have not been suppressed in step 3. Here, two threshold values are used ie. $T_1$ and $T_2$. The magnitude value below $T_1$ is set to zero so as to make it non edge. The magnitude value above $T_2$, it is considered as an edge.

The extraction of Canny-Palmprint features from the palmprint image is shown in Fig.7.

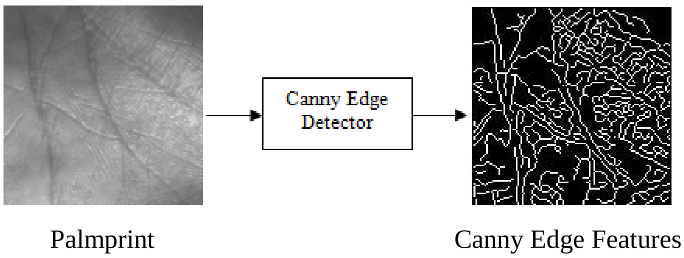



Fig.7. Feature Extraction by Canny Edge Detection method on palmprint image

Once a palm-print image is transformed by Canny edge detection method, the edge information is extracted and stored in feature vector. The feature vector *FV* is equal to Canny-Palmprint Features (CPF) which gives the line information (features),

$$FV = [CPF] \tag{8}$$

where, *FV* is feature vector.

## 3.3 PHASE CONGRUENCY EDGE DETECTION

Here, Line-feature extraction by phase congruency edge detector is proposed. There exist several line and gradient-based feature extraction methods like Sobel operators, Canny [14-15], line directional detectors that calculates the points of high intensity gradients to extract the line features in different directions. All these palmprint recognition methods are based on intensity gradients and therefore got affected by the image contrast and brightness. The proposed phase congruency model for line feature extraction is invariant to changes in image brightness and contrast. The extraction of Phase-Palmprint features from the palmprint image is shown in Fig.8.

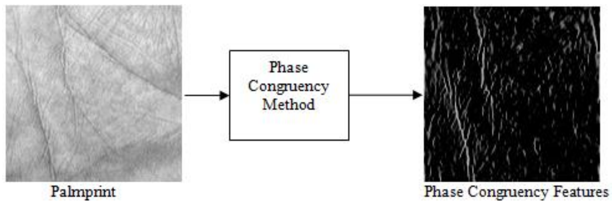
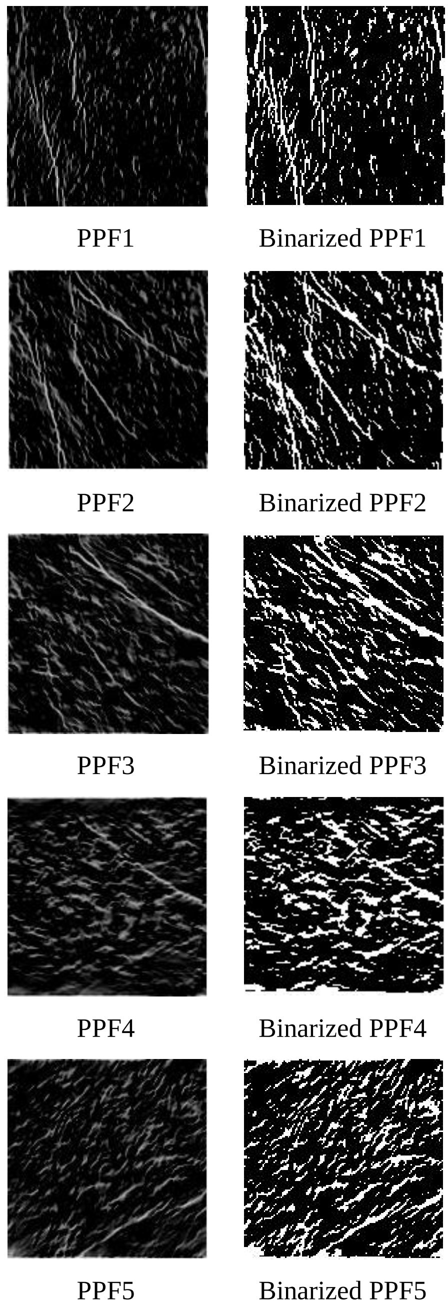



Fig.8. Feature Extraction by Phase Congruency method on palmprint image

The Phase Congruency edge detection method is applied on the palmprint image and Phase-Palmprint Features are extracted. The number of phase congruency features images depends on the number of orientations considered. Here, the number of orientation considered is six. The Phase-Palmprint Features (PPF) extracted and the binarized PPF images for six orientations are shown in Fig.9.



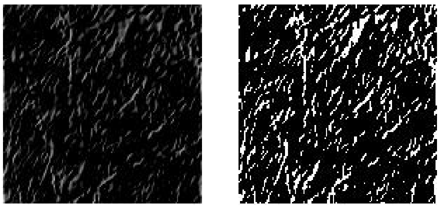



Fig.9. Phase-Palmprint Features and Binarized Phase- Palmprint features

Once a palm-print image is transformed by phase congruency model, the edge information is extracted and stored in feature vector. The feature vector *FV* is equal to Phase-Palmprint Features (PPF) which gives the line information (features),

$$FV_n = [PPF_n] \tag{9}$$

where, $FV_n$ is feature vector corresponding to $n$ orientations, $n$ = 1,2,…6.

## 4. FEATURE MATCHING BY HAMMING DISTANCE AND SLIDING WINDOW METHOD

A matching algorithm describes the degree of similarity between two feature vectors. In this paper, Palmprint features are matched by Hamming distance similarity measurement method that works on binary feature vectors. The line information (Palmprint features) extracted is binarized by the following Eq.(10),

$$PF\,(i,j) = \begin{cases} 1, PF\,(i,j) > 0 \\ 0, PF\,(i,j) \le 0 \end{cases} \tag{10}$$

where, *PF(i,j)*=Palmprint features corresponding to different edge detection methods, *i* and *j* are the rows and columns of the Palmprint features.

Hamming Distance calculates the difference between two binary feature vectors using XOR operation. The hamming distance similarity measurement for line feature extraction can be defined as,

$$HD = \sum_{i}^{60}\sum_{j}^{60}\left(FV\,(i,j) \oplus FV_{DB}(i,j)\right) \tag{11}$$

where, *HD* denotes the hamming distance, *i* and *j* is the row and column of the Palmprint feature vector, ⊕ is the exclusive OR operation, *FV* denotes the feature vector of the person to be matched, $FV_{DB}$ denotes the feature vector in database.

The Palmprint feature vectors are matched by Hamming distance similarity measurement using Sliding window approach. Sometimes during ROI extraction, it happens that the ROI of the same hand may be displaced by some rows or columns. To overcome this problem, Sliding Window method is used. In sliding window method the ROI is reduced by the window size and the window ((60–WS) × (60–WS)) slides over the rows and columns and minimum of the value is considered. The palmprint area of (60–WS) × (60–WS) pixels out of 60 × 60 pixels is considered for Hamming distance matching. The palmprint area of Palmprint feature vector is matched with the Palmprint feature vector in the database. Fig.10 shows the sliding window method using palmprint image.

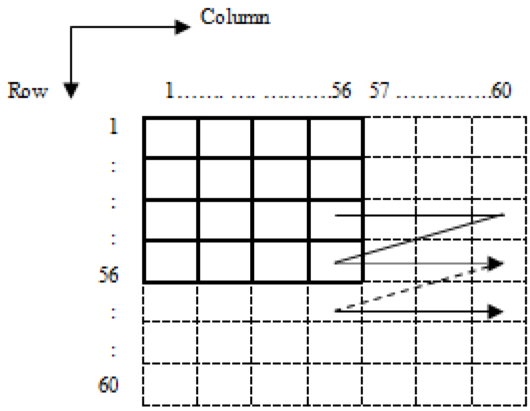



Fig.10. Sliding Window Approach with window size 4 and palmprint size 60×60

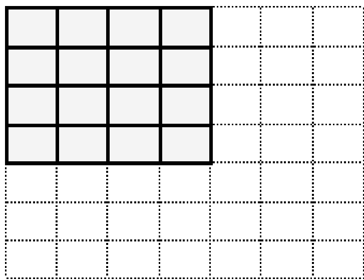
The hamming distance value at *0°* with window size *WS* is defined as,

$$HD_{WS} = \sum_{i}^{60-WS}\sum_{j}^{60-WS}\left(FV\,(i,j) \oplus FV_{DB}(i,j)\right) \tag{12}$$

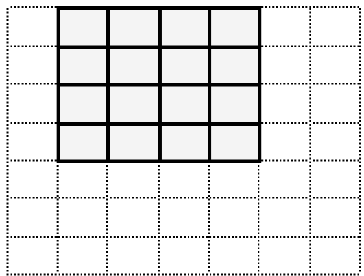
where $HD_{WS}$ denotes the hamming distance with window size *WS*, *i* and *j* is the row and column of the Palmprint feature vector, ⊕ is the exclusive OR operation, *WS* denotes the window size, *FV* denotes the feature vector of the person to be matched, $FV_{DB}$ denotes the feature vector in database. For window size *WS*, there will be *WS*×*WS* hamming distance values. For example, if window size is 4 there will be 4×4 = 16 hamming distance values. The minimum value out of 16 values of hamming distances is chosen as final hamming distance,

$$HD = \min\,(HD_1, HD_2, HD_3, \ldots\ldots HD_{16}) \tag{13}$$

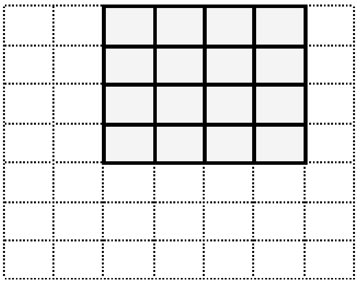
The various steps in sliding window method can be shown by the following images,



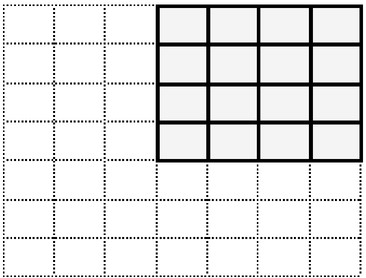
(a) Step 1



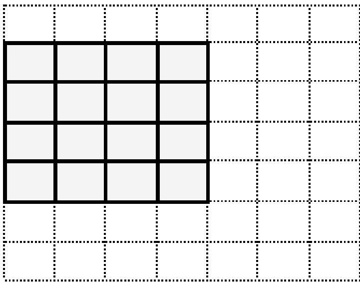
(b) Step 2



(c) Step 3

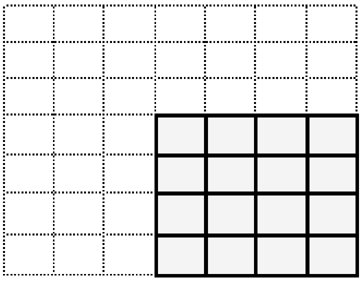


(d) Step 4



(e) Step 5

:
:



(f) Step 16

Fig.11. Various steps in Sliding window method

Hamming Distance value "1" signifies both feature vectors are exactly same and a value approaching "1" signifies both feature vectors belongs to same palm image. A value near to "1" is identified that is known as reference threshold. If matching score (or Hamming distance) of two feature vectors is less than reference threshold value, feature vectors are considered to be from same hands otherwise different hands. In this paper, a unique and effective way to identify reference threshold and threshold range for each hand is proposed. The proposed approach can improve overall system accuracy. The accuracy of the biometric authentication can be defined by following Eq.(14),

$$Accuracy\ (\%) = (100 - (FAR\ (\%) + FRR\ (\%)/2) \quad (14)$$

where, FAR is False Acceptance Rate, FRR is False Rejection Rate.

If either FAR or FRR is decreased, overall system accuracy is increased. The Min Max Threshold Range (MMTR) method can extremely decrease FAR that can result in stable authentication system.

## 5. ACCURACY IMPROVEMENT USING MIN MAX THRESHOLD RANGE (MMTR) APPROACH

In this paper, Min Max Threshold Range (MMTR) method is proposed that helps in increasing overall system accuracy by matching a person with multiple threshold values. In this technique, firstly the person is authenticated at global level using Reference threshold. Secondly, the person is authenticated at local level using range of Minimum and Maximum thresholds defined for a person. Generally, personal authentication is done using reference threshold but there are chances of false acceptance. So, by using the Minimum and Maximum Thresholds range of false accepted persons at personal level, a person is identified to be false accepted or genuinely accepted. MMTR is an effective technique to increase the accuracy of the palmprint authentication system by reducing the False Acceptance Rate (FAR).

In MMTR method, multiple hand image samples are required to find out reference threshold and min max threshold range of each hand. The hand image samples are divided into two groups G1 and G2.

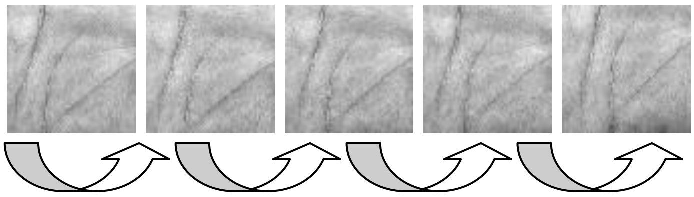


Fig.12. Matching of palmprints with each other

$G_1$ group:

$$P_1 = [I_1, I_2, \ldots\ldots\ldots\ I_{(M-1)}],\ P_2 = [I_1, I_2, \ldots\ldots\ldots\ I_{(M-1)}], \ldots\ldots\ldots\ P_N = [I_1, I_2, \ldots\ldots I_{(M-1)}] \quad (15)$$

$G_2$ group:

$$P_1 = [I_M],\ P_2 = [I_M], \ldots\ldots\ P_N = [I_M] \quad (16)$$

where, $P_i$ denotes $i^{th}$ person in group $G_1$, $G_2$, $I_j$ denotes the $j^{th}$ palm image in group $G_1$, $G_2$.

Table.1. Matching In Group $G_1$ among Person $P_1$

| $i$ / $j$ | 1 | 2 | 3 | | M-1 |
|---|---|---|---|---|---|
| 1 | X | $HD_{12}$ | $HD_{13}$ | ……… | $HD_{1(M-1)}$ |
| 2 | $HD_{21}$ | X | $HD_{23}$ | ……… | $HD_{2(M-1)}$ |
| : | : | : | : | : | : |
| : | : | : | : | : | : |
| M-1 | $HD_{(M-1)1}$ | $HD_{(M-1)2}$ | $HD_{(M-1)3}$ | | X |

In group G1, each hand feature vector in $P_1$ is matched with all other ($M$ – 1) hands feature vector by Hamming distance measurement method. The matching values are stored in threshold array.

$$TA_1 = \begin{bmatrix} HD_{12}, HD_{13}, ..HD_{1M-1}, HD_{21}, HD_{22}, .... \\ .HD_{2M-1}, ....HD_{(M-1)1}, HD_{(M-1)2}, ..HD_{(M-1)(M-2)} \end{bmatrix} \quad (17)$$

Similarly, all N hand image samples matching results are stored in Threshold array ($T_A$).

$$T_A = TA_1 + TA_2 + ........ + TA_N \quad (18)$$

The minimum and maximum of matching values are found out from the threshold array ($TA_1$, $TA_2$,……..$TA_N$) for each individual as shown in Eq.(19).

$$\left.\begin{aligned} T_{AiMIN} &= \min(T_{Ai}) \\ T_{AiMAX} &= \max(T_{Ai}) \end{aligned}\right\}_{i=1,....N} \quad (19)$$

The accuracy of the system is identified by matching group $G_2$ samples with group $G_1$ samples using threshold values stored in threshold array. Finally, a threshold value is chosen where FAR and FRR is minimum, this value is called Reference threshold.

In real time authentication system, if a person's hand is compared with the samples present in the database, the authenticity depends on the matching score. If matching score (Hamming Distance value T) is less than reference threshold ($R_T$), the person is considered to be genuine otherwise imposter as shown in Fig.13.

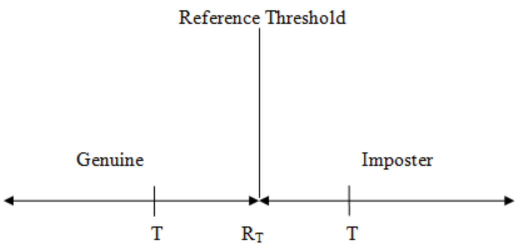



Fig.13. Criteria of authentication

It is possible that some wrong hand can be accepted as genuine if matching score fulfils the reference threshold criteria. Here, a second level of verification by min-max threshold range (MMTR) at hand level is proposed. For successful authentication matching score must be less than reference threshold and within the min-max threshold range of the person as shown in Fig.14. If the matching score of a person to be matched is in the $T_{MIN}$ to $R_T$ range, then the person will be considered as genuine otherwise imposter.

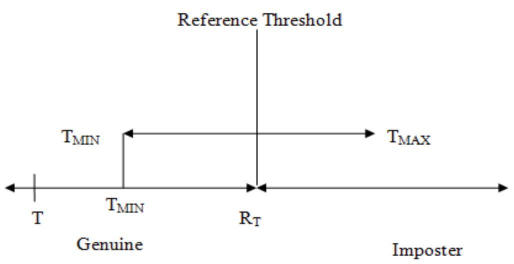



Fig.14. Criteria of authentication with MMTR method

MMTR method can be summarized as that each hand feature vector is matched with all other hands feature vector of the same person by hamming distance measurement method and the matching values are stored in threshold array as shown in Eq.(17) and Eq.(18). The min and max threshold values are identified from threshold array in Eq.(19) and stored in database. It is observed that different hands have different min and max range of threshold. So, the second level of verification within min and max range of threshold can reduce the chances of false acceptance.

## 6. EXPERIMENTAL RESULTS AND ANALYSIS

A database of 600 palm images from 100 palms with 6 samples for each palm is taken from PolyU palmprint database [27].

### 6.1 PALMPRINT AUTHENTICATION SYSTEM

The palmprint database is divided into two groups, first group ($G_1$) consists of 100 persons with each person having 5 palm sample images to train the system, and second group ($G_2$) contains 100 persons with each person having one palm image different from the first group images to test the system. The image size is 284×384 pixels. To reduce the computation complexity palmprint image is resized to 60×60 pixels.

$G_1$ group:

$P_1 = [I_1, I_2, I_3, I_4, I_5]$, $P_2 = [I_1, I_2, I_3, I_4, I_5]$, ………………..… $P_{100} = [I_1, I_2, I_3, I_4, I_5]$

In $G_1$ group each hand $P_i$ contains 5 sample image $I_{1-5}$.

$G_2$ group:

$P_1 = [I_6]$, $P_2 = [I_6]$, ………. $P_{100} = [I_6]$.

In $G_2$ group each hand $P_i$ contains only sample image $I_6$.

Image is pre-processed to get the region of interest. Pre-processing includes image enhancement, image binarization, boundary extraction, cropping of palmprint/ROI. The ROI size is 60×60 pixels. Sample of ROI is shown in Fig.15.

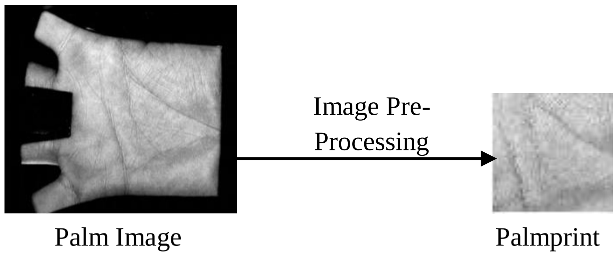



Fig.15. Sample of ROI

Feature extraction is done by Sobel Code method, Canny edge and phase congruency method to get the line feature. The line features extracted are stored in Palmprint feature vector for all hand images samples and stored in database.

Hamming distance method is used as a similarity measurement method for feature matching.

## 6.2 MIN MAX THRESHOLD RANGE (MMTR) APPROACH

In group G1, each hand feature vector in $P_1$ is matched with all other 4 hands feature vector by Hamming distance measurement method. The matching values are stored in threshold array. Similarly, for all 100 hand image samples, 2000 matching values are stored in Threshold array ($T_A$).

$$T_A = TA_1 + TA_2 + \ldots\ldots + TA_{100}$$

The minimum and maximum of matching values are found out from the threshold arrays ($TA_1$, $TA_2, \ldots\ldots TA_N$) for 100 individuals and are stored in the database.

$$\left.\begin{aligned} T_{AiMIN} &= \min\left(T_{Ai}\right) \\ T_{AiMAX} &= \max\left(T_{Ai}\right) \end{aligned}\right\}_{i=1,\ldots.100}$$

The maximum and minimum values are found out from threshold array ($T_A$) to calculate the reference threshold.

$$T_{AMIN} = \min(T_A)$$

$$T_{AMAX} = \max(T_A)$$

The minimum and maximum values of threshold array are divided into $T_H$ threshold values.

$$\Delta = (T_{AMAX} - T_{AMIN}) / T_H$$

$$\Delta 1 = T_{AMIN} + \Delta$$

$$\Delta 2 = T_{AMIN} + 2\Delta$$

Similarly, $\Delta T_H = T_{AMIN} + T_H\Delta$.

These $T_H$ threshold values are tested with group $G_2$ and group $G_1$ images. The FAR and FRR are values are plotted with respect to threshold range values. The value of reference threshold is chosen where FAR and FRR are minimum. Threshold values, respective FAR and FRR values and accuracy for the Sobel Code operator are tabulated in Table.2.

Table.2. Threshold Values, FAR, FRR, Accuracy values for Sobel Code, Canny Edge and Phase Congruency Method

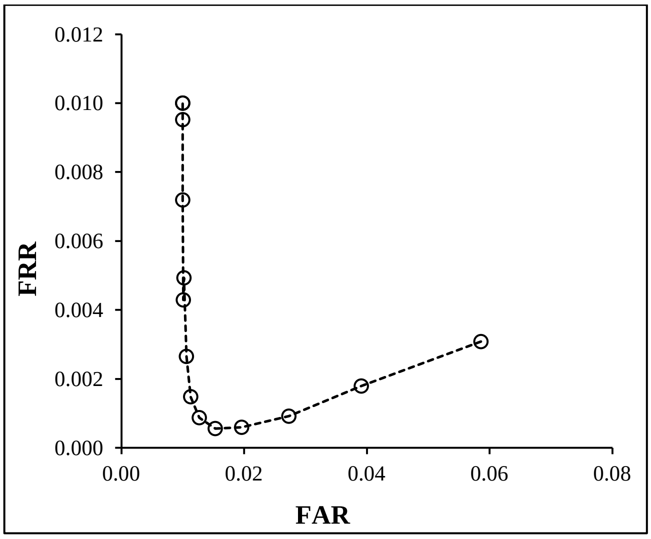
| Method | Reference Threshold | FAR | FRR | Accuracy |
|---|---|---|---|---|
| Sobel Code Method | **9.10E-01** | **5.86E-02** | **3.08E-03** | **96.9** |
| | 9.11E-01 | 3.91E-02 | 1.79E-03 | 98 |
| | 9.12E-01 | 2.73E-02 | 9.15E-04 | 98.6 |
| | 9.13E-01 | 1.96E-02 | 5.93E-04 | 99 |
| | 9.14E-01 | 1.53E-02 | 5.57E-04 | 99.2 |
| | 9.16E-01 | 1.27E-02 | 8.72E-04 | 99.3 |
| | 9.17E-01 | 1.13E-02 | 1.48E-03 | 99.4 |
| | 9.18E-01 | 1.06E-02 | 2.65E-03 | 99.3 |
| | 9.19E-01 | 1.02E-02 | 4.93E-03 | 99.2 |
| | 9.20E-01 | 1.01E-02 | 4.29E-03 | 99.3 |
| | 9.21E-01 | 1.00E-02 | 7.19E-03 | 99.1 |
| | 9.22E-01 | 1.00E-02 | 9.52E-03 | 99 |
| | 9.23E-01 | 1.00E-02 | 1.00E-02 | 99 |
| | 9.24E-01 | 1.00E-02 | 1.00E-02 | 99 |
| Phase Congruency Method | **8.64E-01** | **7.26E-02** | **4.26E-03** | **96.2** |
| | 8.65E-01 | 5.19E-02 | 2.70E-03 | 97.3 |
| | 8.67E-01 | 3.51E-02 | 1.47E-03 | 98.2 |
| | 8.69E-01 | 2.40E-02 | 9.85E-04 | 98.7 |
| | 8.70E-01 | 1.79E-02 | 8.63E-04 | 99.1 |
| | 8.72E-01 | 1.44E-02 | 6.26E-04 | 99.2 |
| | 8.73E-01 | 1.22E-02 | 1.00E-03 | 99.3 |
| | 8.75E-01 | 1.11E-02 | 1.68E-03 | 99.4 |
| | 8.76E-01 | 1.05E-02 | 3.03E-03 | 99.3 |
| | 8.78E-01 | 1.02E-02 | 5.45E-03 | 99.2 |
| | 8.79E-01 | 1.01E-02 | 4.74E-03 | 99.3 |
| | 8.81E-01 | 1.00E-02 | 7.94E-03 | 99.1 |
| | 8.83E-01 | 1.00E-02 | 9.71E-03 | 99 |
| | 8.84E-01 | 1.00E-02 | 1.01E-02 | 99 |
| | 8.86E-01 | 1.00E-02 | 1.00E-02 | 99 |
| | 8.87E-01 | 1.00E-02 | 1.02E-02 | 99 |
| Canny Edge Detection | **7.31E-01** | **5.18E-02** | **2.59E-03** | **97.3** |
| | 7.34E-01 | 3.40E-02 | 1.62E-03 | 98.2 |
| | 7.37E-01 | 2.42E-02 | 8.22E-04 | 98.7 |
| | 7.41E-01 | 1.79E-02 | 8.62E-04 | 99.1 |
| | 7.44E-01 | 1.45E-02 | 6.17E-04 | 99.2 |
| | 7.47E-01 | 1.23E-02 | 9.84E-04 | 99.3 |
| | 7.50E-01 | 1.12E-02 | 1.64E-03 | 99.4 |
| | 7.53E-01 | 1.05E-02 | 2.91E-03 | 99.3 |
| | 7.56E-01 | 1.02E-02 | 5.28E-03 | 99.2 |
| | 7.59E-01 | 1.01E-02 | 4.72E-03 | 99.3 |
| | 7.62E-01 | 1.00E-02 | 7.58E-03 | 99.1 |
| | 7.65E-01 | 1.00E-02 | 9.62E-03 | 99 |
| | 7.68E-01 | 1.00E-02 | 1.01E-02 | 99 |
| | 7.71E-01 | 1.00E-02 | 1.01E-02 | 99 |
| | 7.75E-01 | 1.00E-02 | 1.01E-02 | 99 |

Table.3 also shows the overall accuracy improvement after applying MMTR.

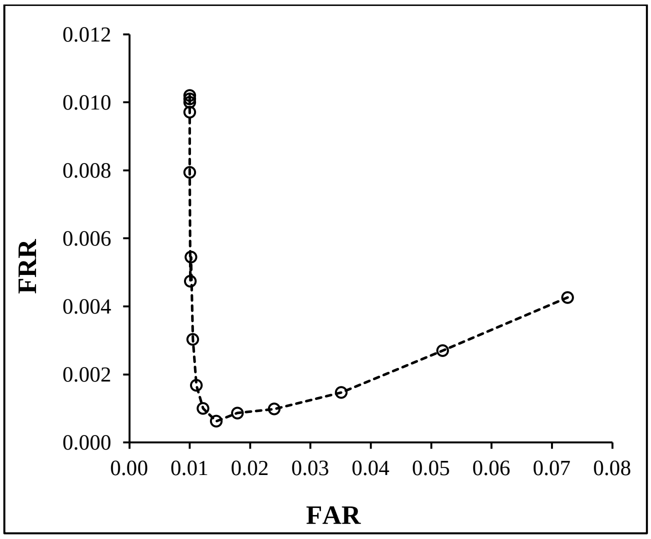
Table.3. Threshold Values, FAR, FRR, Accuracy Values After MMTR

| Method | Reference Threshold | FAR | FRR | Accuracy | FAR with MM TR | FRR with MM TR | Accuracy with MMTR |
|---|---|---|---|---|---|---|---|
| Sobel Code Operator | 9.10E-01 | 5.86E-02 | 3.08E-03 | 96.9 | 1.52E-02 | 4.57E-04 | 99.2 |
| Phase Congruency | 8.64E-01 | 7.26E-02 | 4.26E-03 | 96.2 | 2.26E-02 | 8.73E-04 | 98.8 |
| Canny Edge Detector | 7.31E-01 | 5.18E-02 | 2.59E-03 | 97.3 | 1.59E-02 | 3.53E-04 | 99.2 |

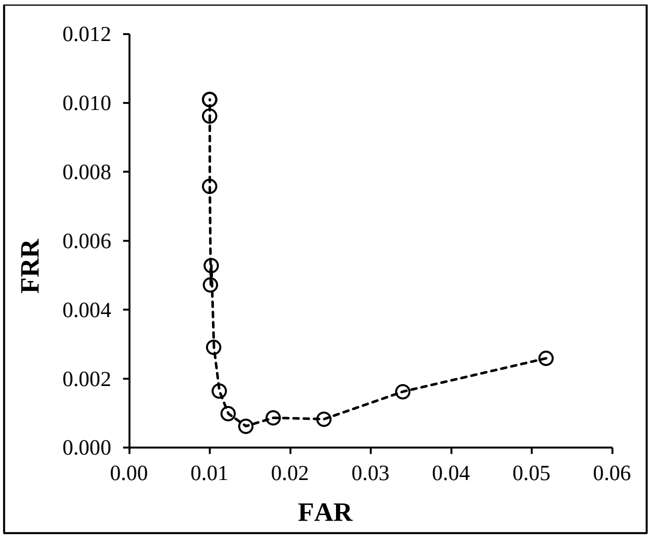
The Canny edge method has performed better with an accuracy of 97.3% than other edge detection methods. The accuracy after applying MMTR has been improved to 99.2%. FAR values with respect to FRR values are plotted in Fig. 16 for all the feature extraction methods. Accuracy values with respect to threshold values for three different feature extraction methods are plotted in Fig.17.




(a) Sobel Code Method

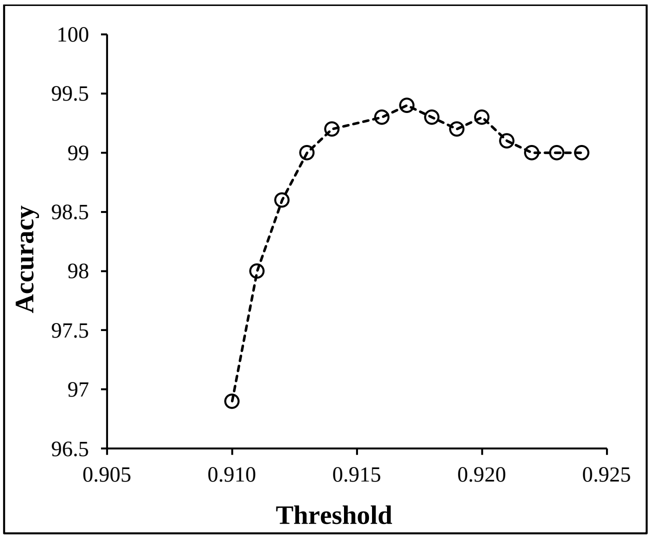



(b) Phase Congruency Method

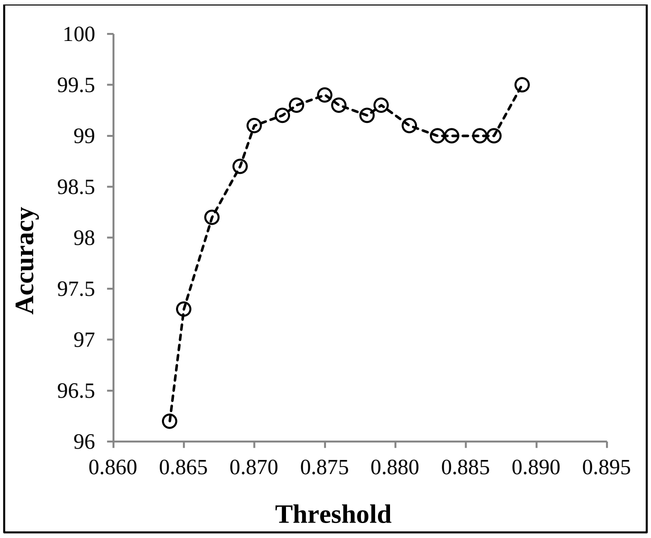



(c) Canny Edge Detection Method

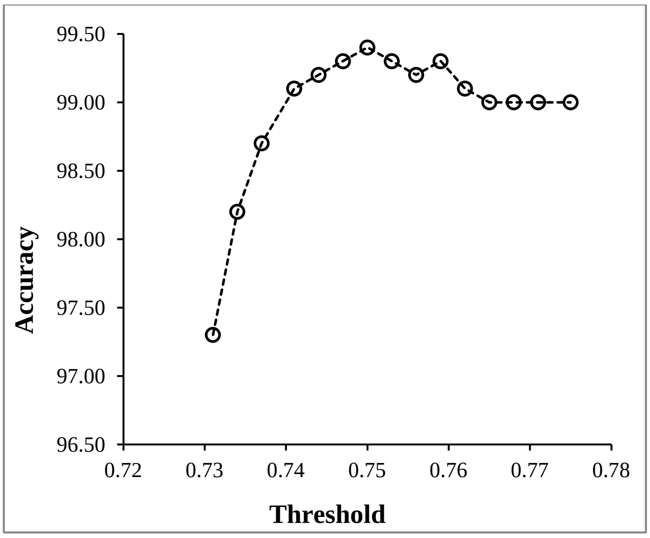
Fig.16. FAR Vs FRR




(a) Sobel Code Method




(b) Phase Congruency Method




(c) Canny Edge Detection Method

Fig.17 Accuracy Vs Threshold

## 6.3 METHODS COMPARISON

In this paper, we have compared the accuracy performance of Edward et. al [18, 19] with the proposed approach. We have also tested the performance with Directional operator [20] and DLEF [21] with our proposed approach. Table.4 shows the comparison of feature extraction methods [18, 19] with our proposed approach with MMTR.

Table.4. Comparison of Feature Extraction Methods with Proposed Approach

| Method | Accuracy |
|---|---|
| David Zhang et. al [15] | 98.5 |
| Edward et. al [18] | 97.35 |
| Edward et. al [19] | 94.84 |
| Directional operator [20] | 97.81 |
| DLEF [21] | 97.50 |
| Proposed Approach Accuracy | **99.2** |

We have found that our proposed approach has performed better than other methods. This shows that by using proposed with MMTR, accuracy of the system improves because MMTR offers two level of authentication.

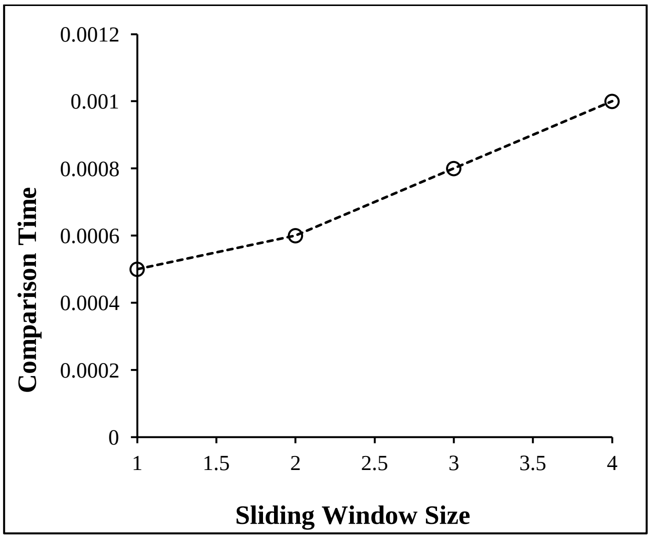
## 6.4 COMPARISON TIME AND DATABASE PREPARATION TIME

The comparison time and DB preparation time for sliding window size 2 is tabulated below in Table.5.

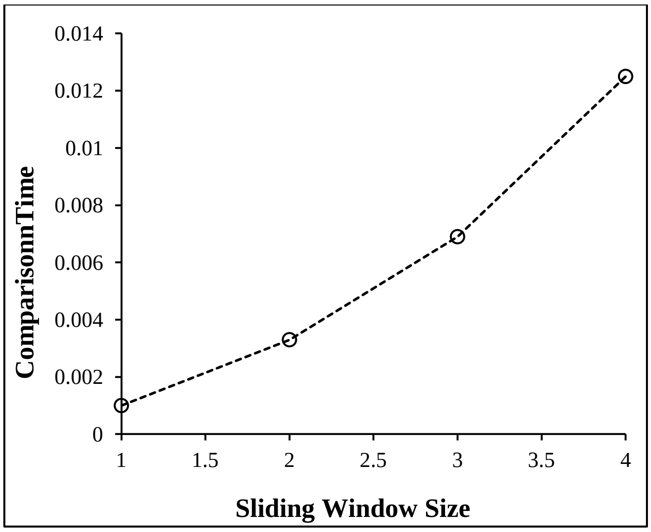
Table.5. Comparison of DB Preparation Time and Comparison Time

| Method | Comparison Time | DB Preparation Time |
|---|---|---|
| Sobel Code Operator | 5.46E-04 | 2.96E-02 |
| Phase Congruency | 3.32E-05 | 2.61E-01 |
| Canny Edge Detector | 3.18E-05 | 1.10E-01 |

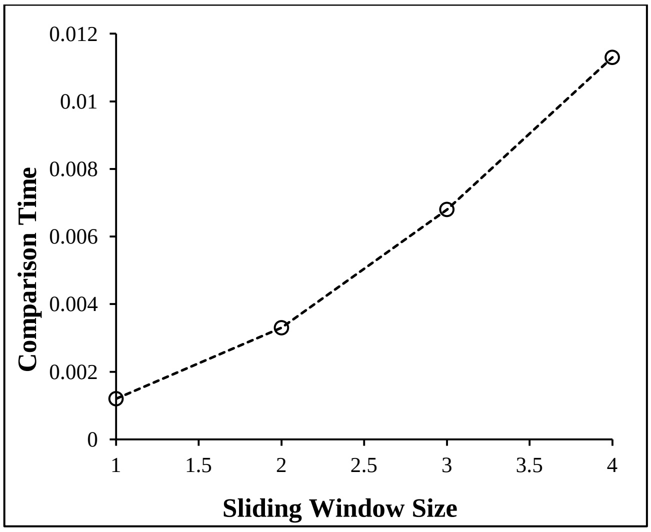
The comparison time and DB preparation time for Sobel, Canny and Phase congruency method are shown in Fig.18 and Fig.19.




(a) Comparison Time (Sobel Method)




(b) Comparison Time (Canny Edge)




(c) Comparison Time (Phase Congruency)

Fig.18 Comparison Time Vs Sliding Window size

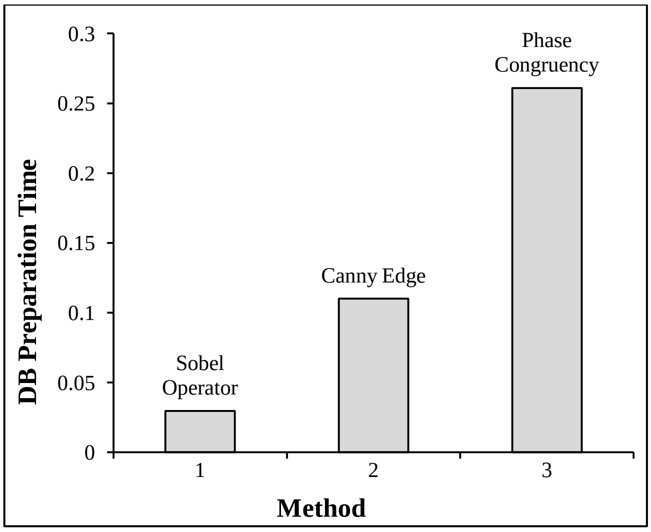



Fig.19 DB Preparation Time

The comparison time is less for Canny Edge method and DB Preparation time is less for Sobel Code operator.

## 7. CONCLUSION

Accuracy is the main and important part of real time palmprint authentication. In this paper, three different feature extraction methods are used to extract the features from palmprint image. In addition, MMTR approach is proposed that can drastically improve the accuracy of the system. PolyU database palm images are used to prepare the database of 600 palm images. Palm images are enhanced and pre-processed to get the region of interest (ROI). Multi-scale (3×3, 5×5 and 7×7) Sobel Code operators, Canny edge detection method, Phase congruency methods are applied to the palmprint image. The palmprint feature vector is compared with other feature vector in the database using Hamming distance similarity measurement method. 99.2% of accuracy is obtained using feature extraction methods and MMTR approach.

## REFERENCES

[1] P. Jonathon Phillips, Alvin Martin, C.L. Wilson and Mark Przybocki, "An Introduction to Evaluating Biometric Systems", *Proceedings of IEEE Computer society*, Vol. 33, No. 2, pp.56-63, 2000.

[2] Jain, A. Ross, and S. Prabhakar, "An Introduction to Biometric Recognition," *IEEE Transactions on Circuits Systems Video Technology*, Vol. 14, No. 1, pp. 4–20, 2004.

[3] Sharath Pankanti, Ruud M. Bolle and Anil Jain, "Biometrics: The Future of Identification", Proceedings of *IEEE Computer society*, Vol. 33, No. 2, pp. 46-49, 2000.

[4] Maylor K.H. Leung, A.C.M. Fong, and Siu Cheung Hui Nanyang, "Palmprint Verification for Controlling Access to Shared Computing Resources", *Proceedings of IEEE Computer Society*, *Pervasive Computing*, Vol. 6, No. 4, pp.40-47, 2007.

[5] Xiangqian Wu, David Zhang, and Kuanquan Wang, "Palm Line Extraction and Matching for Personal Authentication", *IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans*, Vol. 36, No. 5, pp. 978-987, 2006.

[6] N. Duta, A. Jain, and K. Mardia, "Matching of palmprint," *Pattern Recognition Letters*, Vol. 23, No. 4, pp. 477–485, 2001.

[7] D. Zhang, W. Kong, J. You, and M. Wong, "Online Palmprint Identification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 25, No. 9, pp. 1041-1050, 2003.

[8] Han, H. Chen, C. Lin, and K. Fan, "Personal authentication using palmprint features," *Pattern Recognition*, Vol. 36, No. 2, pp. 371–381, 2003.

[9] Xiangqian Wu, David Zhang, Kuanquan Wang and Bo Huang, "Palmprint classification using principal lines", *Pattern Recognition*, Vol. 37, pp. 1987-1998, 2004.

[10] Kumar, D. Wong, H. Shen, and A. Jain, "Personal verification using palmprint and hand geometry biometric,"

*in Lecture Notes in Computer Science, Berlin, Germany: Springer-Verlag*, Vol. 2688, pp. 668– 678, 2003,.

[11] Nirupama Srinivasan and Evangelia Micheli-Tzanakou, "Palm Print Recognition: A New Algorithm For Corner Detection Using Palm Anatomy Features", *IEEE International Workshop on Measurement Systems for Homeland Security, Contraband Detection and Personal Safety Alexandria*, 2006.

[12] P. S. Wu and M. Li, "Pyramid edge detection based on Stack filter", *Pattern Recognition letter*, Vol. 18, No. 4, pp. 239-248, 1997.

[13] Xiangqian Wu, Kuanquan Wang, and David Zhang, "Palmprint texture analysis using derivative of gaussian filters", *International Conference on Computational Intelligence and Security*, Vol. 1, pp. 751-754, 2006.

[14] W. Li, D. Zhang, and Z. Xu, "Palmprint Identification by Fourier Transform", *International Journal on Pattern Recognition and Artificial Intelligence*, Vol. 16, No. 4, pp. 417-432, 2002.

[15] X.Q. Wu, K.Q. Wang and D. Zhang, "Wavelet Based Palm print Recognition", *Proceedings of First International Conference on Machine Learning and Cybernetics*, Vol. 3, pp. 1253-1257, 2002.

[16] Kovesi P, "Image features from phase congruency", *Videre J. Comput. Vis. Res*., Vol. 1, No. 3, pp. 1–26, 1999.

[17] Wong K. Y. E., G. Sainarayanan and Ali Chekima, "Palmprint Identification Using SobelCode" *Malaysia-Japan International Symposium on Advanced Technology, Kuala Lumpur, Malaysia*, 2007.

[18] Wong K. Y. E., Jamal A. Dargham, Ali Chekima and G. Sainarayanan, "Palmprint Identification Using 5 x 5 Sobel Operator," *Proceedings of First Seminar on Engineering and Information Technology, Sabah Malaysia*, pp. 208-211, 2008.

[19] Wong K. Y. E., Ali Chekima, Jamal A. Dargham and G. Sainarayanan, "Palmprint Identification using Sobel Operator," *10th international Conference on Control, Automation, Robotics and Vision 2008, ICARCV 2008, Hanoi, Vietnam*, pp. 1338-1341, 2008.

[20] Wu X.Q, Wang K.Q and Zhang D, 'Palm line extraction and matching for personal authentication'. *IEEE Transactions on Systems, Man and Cybernatics*, Vol. 36, No. 5, pp 978-986, 2006.

[21] Wu X.Q, Wang K.Q and Zhang D, "Palmprint Recognition using Directional Line Energy feature", *Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04)*, 2004.

[22] Edward Wong G. Sainarayanan and Ali Chekima, "Palmprint Authentication using Relative Geometric Features", *Proceedings of the Third International Conference on Artificial Intelligence in Engineering & Technology, Kota Kinabalu, Sabah, Malaysia*, 2006.

[23] Eddins, S. L, Gonzalez, R.C and Woods, R. E, "*Digital Image Processing Using Matlab*", Upper Saddle River, New Jersey: Person Education, Inc., 2009.

[24] Poon, D.C.M. Wong and H.C. Shen, "A New Method in Locating and Segmenting Palmprint into Region-of-interest", *Proceedings of the 17th International Conference on Pattern Recognition*, Vol. 4, pp. 533 – 536, 2004.

[25] W. Shu, G. Rong, Z. Bian and D. Zhang, "Automatic palmprint verification", *International Journal on Image and Graphics*, Vol. 1, pp. 135-151,2001.

[26] J. Canny, "A computational approach to edge detection," *IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. PAMI-8*, No. 6, pp. 679–698, 1986.

[27] PolyU Palmprint Database: http:// www. comp.polyu. edu. hk/~biometrics/

[28] Jyoti Malik, G. Sainarayanan and Ratna Dahiya, "Min Max Threshold Range (MMTR) Based Approach In Palmprint Authentication By Sobel Code Method", *Procedia Computer Science, Proceedings of the International Conference and Exhibition on Biometric Technology*, Vol. 2, pp 149-158, 2010.