Fast Complex Gabor Wavelet Based Palmprint Authentication

Jyoti Malik jyoti_reck@yahoo.com
Electrical Department
National Institute of Technology
Kurukshetra, 136119, India

Ratna Dahiya ratna_dahiya@yahoo.co.in
Electrical Department
National Institute of Technology
Kurukshetra, 136119, India

G Sainarayanan sai.jgk@gmail.com
Maples ESM Technologies
Chennai, India

Abstract

A biometric system is a pattern recognition system that recognizes a person on the basis of the physiological or behavioral characteristics that the person possesses. There is increasing interest of researchers in the development of fast and accurate personal recognition systems. In this paper, Sliding window method is used to make the system fast by reducing the matching time. The reduction in computation time indirectly reduces the overall comparison time that makes the system fast. Here, 2-D Complex Gabor Wavelet method is used to extract features from palmprint. The extracted features are stored in a feature vector and matched by hamming distance similarity measurement using sliding window approach. Reduction of 74.12% and 90.32% in comparison time is achieved using Sliding window methods. The improvement in time is indicated by experimental results that makes a system rapid.

Keywords: Palmprint Authentication, Complex Gabor Wavelet, Similarity Measurement, Sliding Window Method.

---

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 [1-3]. In this paper, palmprint biometric is selected for personal authentication as it is unique and relatively low resolution images (less than 100 dpi) are sufficient to extract its unique features [4][5].

Palmprint features include line features, geometry features, point features, texture features and statistical features. In this paper, line features are extracted using Complex Gabor Wavelet Transform method. A complex Gabor wavelet is defined as the product of a Gaussian kernel times a complex sinusoid. The line features extracted by complex gabor wavelet [6-15] at various values of theta is stored in the feature vector. The feature vector is matched by Hamming Distance similarity measurement using sliding window method.

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 Gabor-Palmprint features is prepared. Reference threshold values are also identified and stored in database. These values will be later used by Authentication system.

In Authentication system, the authenticity of a person being genuine or imposter is identified with the help of Reference threshold values stored in Pre-authentication system database.

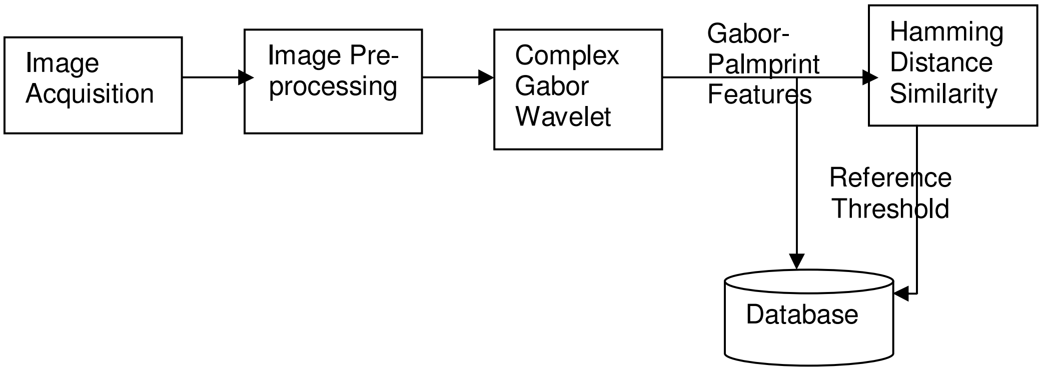


FIGURE 1: Palmprint Pre-Authentication system

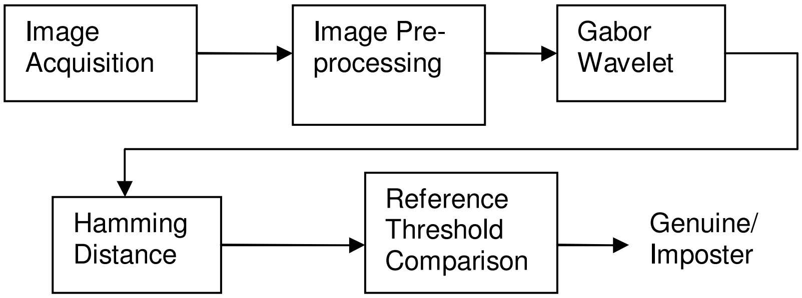


FIGURE 2: Palmprint Authentication System

In Palmprint-based personal recognition system matching time plays very important role in making the real time authentication system. In this paper, we have proposed fast palmprint authentication system. The reduction in matching time is done using Sliding window methods and explained later in section 4.

In the following sections, Section 2 defines palmprint feature extraction by our proposed Complex gabor wavelet method. Section 3 explains the feature matching method by hamming distance using sliding window method. Section 4 explains Sliding window method (SWM), SWM1, SWM2 and Section 5 defines the reference threshold calculation. Section 6 discusses the experimental setup with the results. Section 7 concludes the conclusion.

2. FEATURE EXTRACTION

The desired line features are extracted from the palmprint using Complex Gabor wavelet method which extracts line features from the input palm-print image. The Gabor wavelet is basically a Gaussian (with variances sx and sy along x and y-axes respectively) modulated by a complex sinusoid (with centre frequencies U and V along x and y-axes respectively) described by the following equation:

$$G_i(x,y)=\frac{1}{2*pi*sx*sy}*\exp\left(\left[\frac{-1}{2}\left\{\left(\frac{x}{sx}\right)^2+\left(\frac{y}{sy}\right)^2\right\}\right]\right)*M_i(x,y,f) \qquad (1)$$

$i=1,2$

$$M_1(x,y,f) = \cos\left[2 * pi * f * \sqrt{\left(x^2 + y^2\right)}\right]$$
$$M_2(x,y,f) = \cos\left[2 * pi * f * \left(x * \cos\theta + y * \sin\theta\right)\right]$$

Where sx and sy are the variances along x and y -axes respectively, f is the frequency of the sinusoidal function, θ is the orientation of Gabor wavelet, G_1 and G_2 are the output Gabor wavelets. The sample of Gabor wavelet convolution with the palmprint image is shown in Fig. 3.

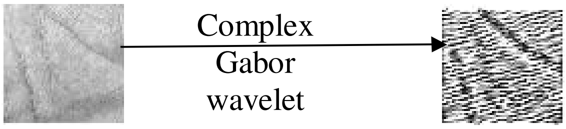


FIGURE 3: Feature extraction by Complex Gabor Wavelet

The feature vector contains features extracted by Complex Gabor wavelet at different orientations i.e. 0°, 30°, 60°, 90°, 120°, 150° and 180° as shown by figure 4.

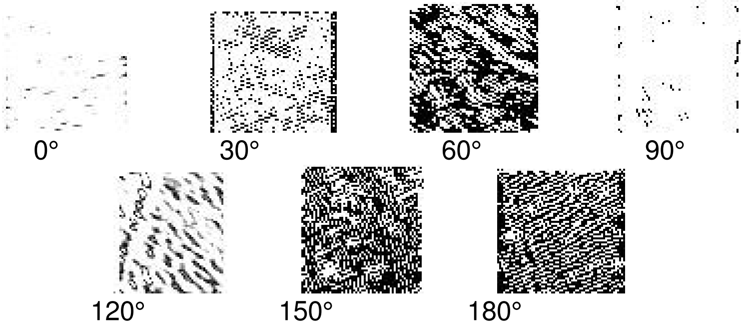


FIGURE 4: Complex Gabor wavelet features images

The feature vector contains 7 elements at corresponding to each orientation. The Feature vector matrix is given by (2)

$$CGWF = \left[CGWF_0, CGWF_1, CGWF_2, CGWF_3, CGWF_4, CGWF_5, CGWF_6\right] \tag{2}$$

3. FEATURE MATCHING BY HAMMING DISTANCE AND SLIDING WINDOW METHOD

The complex gabor wavelet line feature vectors (line information) are matched by Hamming distance similarity measurement method but firstly the line information (complex gabor wavelet features) extracted is binarized by the following equation (3):

$$CGWF_n(i,j) = \begin{cases} 1, CGWF_n(i,j) > 0 \\ 0, CGWF_n(i,j) \le 0 \end{cases} \tag{3}$$

where, $CGWF_n(i,j)$= complex gabor wavelet features corresponding to n^{th} orientation, $n = 0,1,2,....6$, *i* and *j* are the rows and columns of the complex gabor wavelet features.

Hamming Distance calculates the difference between two binary feature vectors using EX-OR operation and can be defined as in (4):

$$HD_n = \sum_{i}^{60} \sum_{j}^{60} \left(FV\ (i,j) \oplus FV_{DB}(i,j)\right) \tag{4}$$

where, HD_n denotes the Hamming distance at orientation *n*, $n = 0,1,2,....6$, *i* and *j* is the row and column of the complex gabor wavelet feature vector, $\oplus$ is the exclusive OR operation, *FV* denotes the feature vector of the person to be matched, FV_{DB} denotes the feature vector in database. Here, *FV* is same as *CGWF*.

The hamming distance value is calculated by using sliding window method. In sliding window method, the ROI is reduced by the window size (WS) and the window of ((60–WS)×(60–WS)) slides over the rows and columns out of 60×60 pixels considered for Hamming distance matching. The minimum value of the hamming distance values is considered. Fig. 5 shows the sliding window approach.

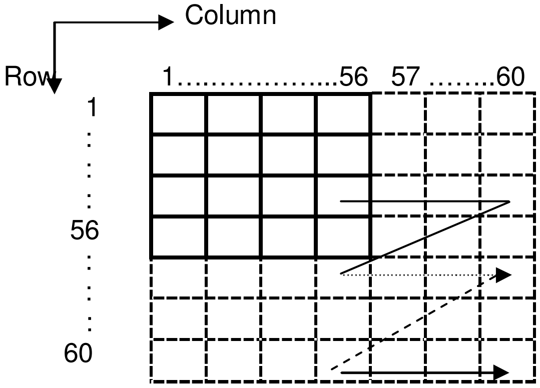


FIGURE 5: Sliding Window Approach with window size 4 and palmprint size 60×60

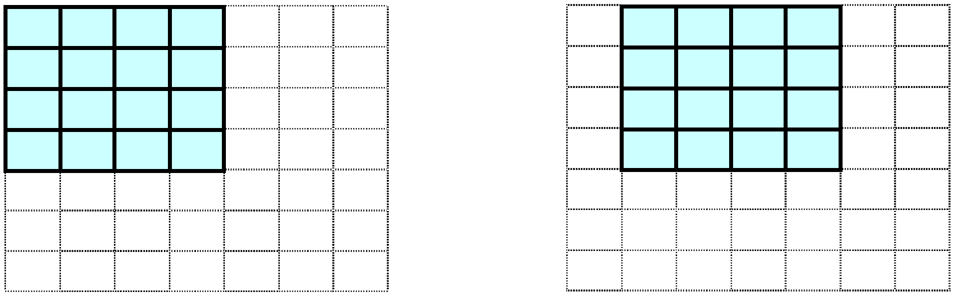
The modified Hamming distance value at *n* orientation with window size *WS* is defined in (5) as:

$$HD_{WSn} = \sum_{i}^{60-WS} \sum_{j}^{60-WS} \left(FV\,(i,j) \oplus FV_{DB}(i,j) \right) \tag{5}$$

where, HD_{WSn} denotes the Hamming distance with window size *WS* and at an orientation *n*, $n = 0,1,2,....6$, *i* and *j* is the row and column of the Complex gabor wavelet 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 minimum value out of 16 (For window size *WS=4,* 4×4 = 16) values of Hamming distances is chosen as Hamming distance as calculated in (6).

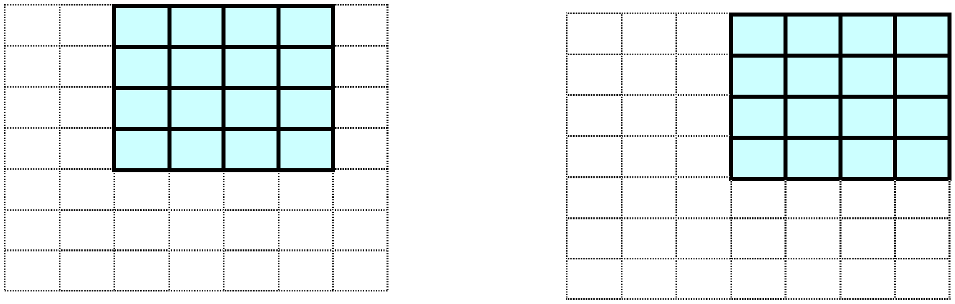
$$HD_n = \min\left(HD_{1\_n}, HD_{2\_n}, HD_{3\_n}, ...HD_{16\_n}\right) \tag{6}$$

The various steps in sliding window method are shown in figure 6.

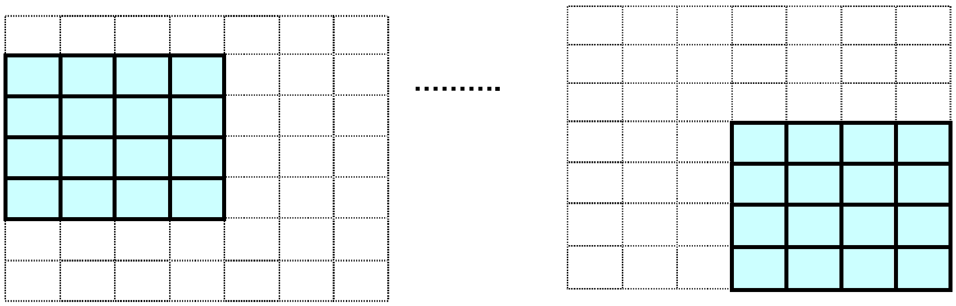


(a) Step 1

(b) Step 2



(c) Step 3 (d) Step 4



(e) Step 5 (f) Step 16

FIGURE 6: Various steps in sliding window method

The average of all Hamming distances for n orientations is calculated as shown in (7).

$$AHD = \left(\frac{\sum HD_n}{n} \right)_{n=0,1,2,....6} \quad (7)$$

where, HD_n denotes the hamming distance for n^{th} orientation, *AHD* denotes the average value of hamming distance. The average value will help in finding the reference threshold value.

If, the Hamming distance value of two feature vectors is less than reference threshold value, feature vectors are considered to be from same hands otherwise different hands.

4. COMPARISON TIME IMPROVEMENT USING FAST SLIDING WINDOW METHOD

The sliding window method is the basic approach of feature matching and an accurate method but slow because of window size. The window size is chosen in such a manner to avoid alignment problem in the palmprint images. Here, two sliding window approaches are proposed. Sliding window method 1 and Sliding window method 2 is about using palmprint segment in such a manner so that the matching operation time is reduced. In this paper, complexity of the algorithm of SWM is reduced in SWM1 and SWM2.

4.1 Sliding Window Method

SWM is the basic technique used for matching two feature vectors. In feature matching, full image is matched with the image stored in the database. If the image is not aligned properly then the concept of window size comes into picture. According to (5), the number of calculations increases by a factor of $(WindowSize)^2$. The complexity calculation algorithm of SWM is explained below.

$$ComplexitySWM = M \times N \times (WS)^2 \times O \times T_1 \quad (8)$$

where, $M = (No.ofRows - WS) = (60 - WS)$, $N = (No.ofColumns - WS) = (60 - WS)$

$WS = WindowSise = 1to4$, $O = No.ofOrientations = 7$, $T_1 = EX - ORTime$

According to (5), if WS = 4 and time taken for each EX-OR operation is T_1 as shown in (9), then total time taken for hamming distance calculation is given by:

$$(FV\ (i,j) \oplus FV_{DB}(i,j)) = T_1 \text{ time} \tag{9}$$

Complexity or total sliding window time (T_{SWM}) taken for 6 orientations will be given by (10):

$$T_{SWM} = (60-4)\times(60-4)\times(4\times4)\times7\times T_1 = 56\times56\times16\times7\times T_1 = 351232T_1 \tag{10}$$

where, T_{SWM} specifies time taken to compare feature vectors of two palmprints.

If we consider 100 palmprints the total matching time will be given by:

$$((351232T_1 \times 100) = 35123200T_1)$$

It is observed that the number of operations is large in number in sliding window method and will take lot of time. Comparison time need to be reduced by improving sliding window method.

4.2 Sliding Window Method 1 (SWM1)

In this method, a part of the actual palmprint area is considered and it is named as palmprint segment. The chosen palmprint area can be any of the palmprint segment mentioned in the Fig. 7.

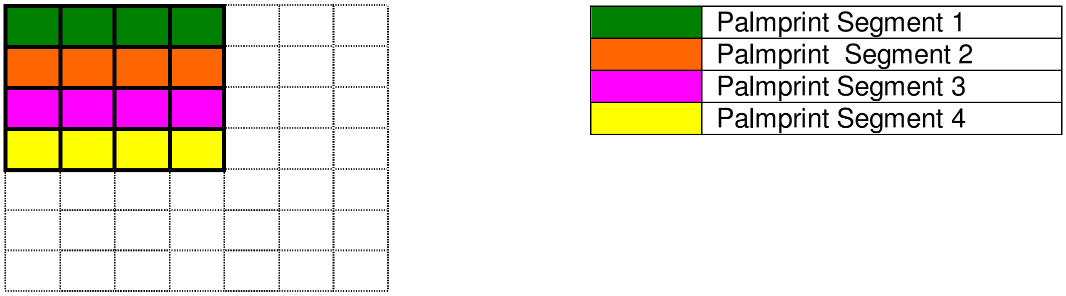


FIGURE 7: The segmented palmprint.

The complexity calculation algorithm of SWM1 is explained below.

$$ComplexitySWM1 = (M_1 \times N \times (WS)^2 \times T_1 + M \times N \times T_1) \times O \tag{11}$$

where, $M_1 = \left(\frac{No.ofRows}{4} - WS\right) = \left(\frac{60}{4} - WS\right) = (15 - WS)$,

$M = (No.ofRows - WS) = (60 - WS)$, $N = (No.ofColumns - WS) = (60 - WS)$

$WS = WindowSise = 1to4$, $O = No.ofOrientations = 7$, $T_1 = EX - ORTime$

In (6) minimum hamming distance value using sliding window method is calculated. In SWM1 method, hamming distance value for *o* orientations is found out using sliding window method as shown in (5). The (6) signifies minimum hamming distance in sliding window as the closest matching between two palmprints.

$$\min\_index\_HD_{WS} = index(\min(HD_{WS})) \tag{12}$$

Total time for 6 orientations and window size 4 will be given by (11):

$$T_{SWM1} = (((11\times56\times16\times T_1 + 56\times56T_1)\times7) = 90944T_1).$$

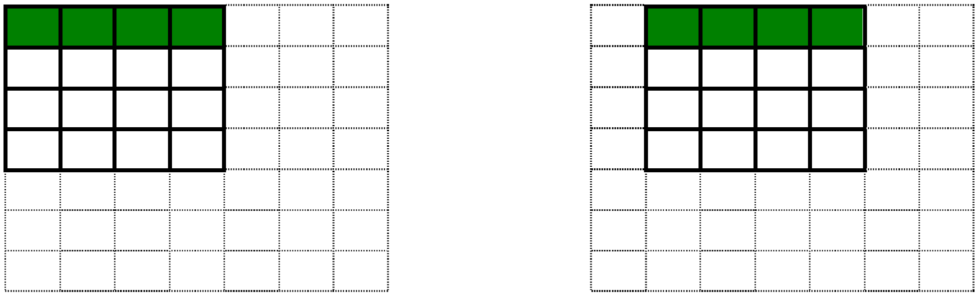
If we consider 100 palmprints the total matching time will be $((90944T_1 \times 100) = 9094400T_1)$.

The reduction in matching time is given by (13).

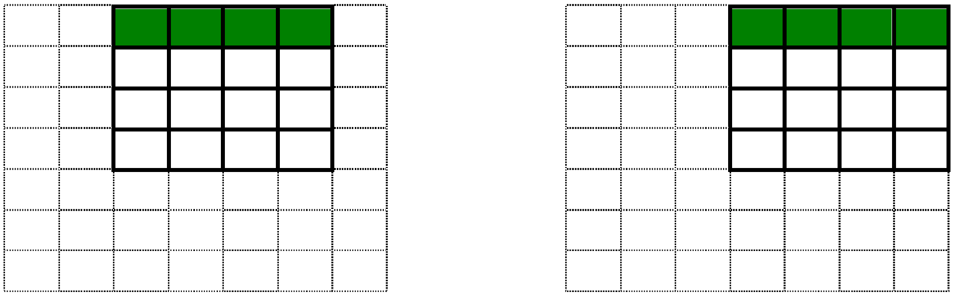
$$\frac{(351232T_1 - 90944T_1)}{351232T_1}\times100 = 74.11\% \tag{13}$$

There is 74.11% reduction in matching time using SWM1 than SWM. The reduced matching time signifies fast authentication system.

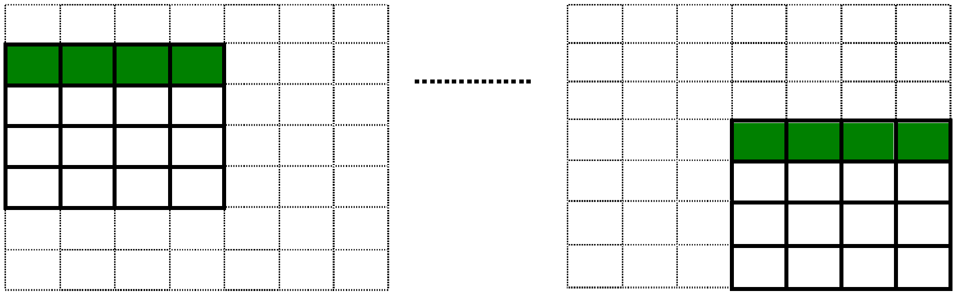
The sliding window method on the chosen palmprint area can be shown diagrammatically as in Fig. 8.



(a) Step 1 (b) Step 2



(c) Step 3 (d) Step 4



(e) Step 5 (f) Step 16

FIGURE 8: Various steps in improved Sliding window method

As we can see from the above diagrams that the area of palmprint segment in sliding window method has reduced considerably and it leads to reduction in matching time.

4.3 Sliding Window Method 2 (SWM2)

In this method, the minimum hamming distance value is not calculated for all the orientations using sliding window method. The index of minimum hamming distance value is calculated for one orientation and same index value will be used to calculate hamming distance for other orientations. The complexity calculation algorithm of SWM2 is explained below.

$$ComplexitySWM2 = \left(\left(\left(M_1 \times N \times (WS)^2 \times M \times N\right) + (M \times N) \times (O-1)\right) \times T_1\right) \qquad (14)$$

where, $M_1 = \left(\frac{No.ofRows}{4} - WS\right) = \left(\frac{60}{4} - WS\right) = (15 - WS)$,

$M = (No.ofRows - WS) = (60 - WS), N = (No.ofColumns - WS) = (60 - WS)$

$WS = WindowSise = 1to4$, $O = No.ofOrientations = 7$, $T_1 = EX - ORTime$

The total time taken given by (14) for window size 4, $T_{SWM2} = ((16 \times 11 \times 56 \times T_1) + (56 \times 56 \times T_1) + (56 \times 56T_1) \times 6) = 31808T_1$. If we consider 100 palmprints the total matching time will be $(31808T_1 \times 100 = 3180800T_1)$.

$$\frac{(351232T_1 - 31808T_1)}{351232T_1} \times 100 = 90.94\% \tag{15}$$

There is 90.94% reduction in matching time using SWM2 than SWM. The reduced matching time signifies fast authentication system.

All the sliding window methods are summarized in table 1.

Method used	**Number of operations**
1. SWM: Hamming distance calculation with sliding window method with window size *WS*. $HD_{WS} = \sum_{i}^{60-WS} \sum_{j}^{60-WS} \left(FV\ (i,j) \oplus FV_{DB}(i,j)\right)$	$\underbrace{56 \times 56}_{} \times 16 \times 7 = 351232$ Operations The (56×56) implies the number of comparisons of the palmprint with the palmprint in the database. $(56 \times 56 \times 16)$ signifies sliding window method comparisons for window size 4, so $(4 \times 4) = 16$ comparisons. The total number of comparisons takes place for seven orientations is 351232.
2. SWM1: Hamming distance calculation with sliding window method (15×60) with window size 4. $HD = \sum_{i}^{15-WS} \sum_{j}^{60-WS} \left(FV\ (i,j) \oplus FV_{DB}(i,j)\right)$ The palmprint segment size is (15×60).	$(\underbrace{11 \times 56}_{} \times 16 + \underbrace{56 \times 56}_{}) \times 7 = 90944$ Operations The (11×56) implies the number of comparisons of the palmprint segment with the palmprint segment in the database. $(11 \times 56 \times 16)$ signifies to sliding window method comparisons for window size 4, so $(4 \times 4) = 16$ comparisons. (56×56) signifies the comparison of palmprint at the minimum index value. The sum of $(11 \times 56 \times 16)$ and (56×56) give the number of comparison for each orientation. The total number of comparisons takes place for seven orientations is 90944 Operations.
Assumption: If time taken to do each matching operation is T₁.	Theoretical time improvement $\frac{(351232T_1 - 90944T_1)}{351232T_1} \times 100 = 74.11\%$
3. SWM2: Hamming distance calculation with sliding window method (15×60) with window size *WS*. $HD = \sum_{i}^{15-WS} \sum_{j}^{60-WS} \left(FV\ (i,j) \oplus FV_{DB}(i,j)\right)$	$(\underbrace{11 \times 56}_{} \times 16 + \underbrace{56 \times 56}_{}) + (\underbrace{56 \times 56}_{}) \times 6 = 31808$ Operations The assumption here is that the minimum hamming distance values for other orientations will also be at the same index as it is for one orientation. $(11 \times 56 \times 16)$ applies to sliding window method comparisons for window size 4, so

	$(4\times 4=16)$ comparisons. (56×56) signifies the comparison of palmprint and finding the minimum index value. The sum of $(11\times 56\times 16)$ and (56×56) give the number of operations for one orientation. The number of comparisons for another 6 orientation is at the minimum index value as assumed for this method. So, additional $(56\times 56)\times 6$ comparisons will be added. The total number of comparisons is 31808.
Assumption: If time taken to do each matching operation is T_1.	Theoretical time improvement $\frac{(351232T_1-31808T_1)}{351232T_1}\times 100=90.94\%$

TABLE 1: Reduction in matching operations using various Sliding window method

The number of operations, matching operation time with respect to sliding window size is tabulated in table 2.

Sliding Window Size	Number of Operations			Matching operation Time			Percentage reduction in Matching time		
	SWM	SWM 1	SWM2	SWM	SWM1	SWM2	SWM	SWM1	SWM2
1	24367	24367	24367	4.48E-03	4.45E-03	4.43E-03	NA	0.67	1.12
2	94192	44660	26564	1.75E-02	8.34E-03	4.98E-03	NA	52.34	71.54
3	204687	65835	28899	3.89E-02	1.23E-02	5.53E-03	NA	68.38	85.78
4	351232	90944	31808	6.53E-02	1.69E-02	6.32E-03	NA	74.12	90.32

TABLE 2: Percentage reduction in matching operation time

From table 2 it is concluded that there is maximum of 74.12% and 90.32% practical reduction in matching operation time. So it can be concluded from table 1 and table 2 results that with two different sliding window methods, the authentication system becomes very fast. Figure 9 shows that with SWM1 and SWM2 the comparison time reduces significantly.

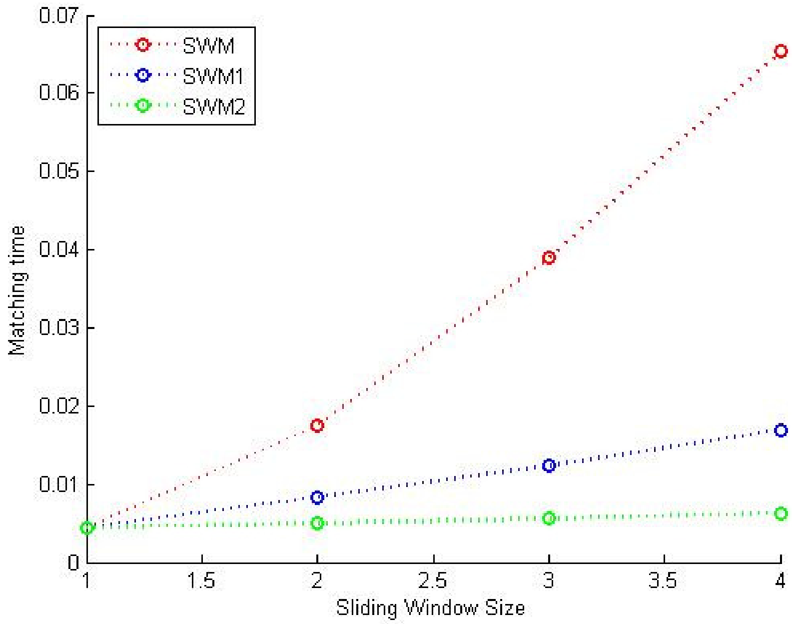


FIGURE 9: Matching time Vs Sliding Window size

5. REFERENCE THRESHOLD

The hand image samples are divided into two groups G1 and G2.

G_1 group

$$P_1 = \left[I_1, I_2, ...I_{(M-1)}\right],\ P_2 = \left[I_1, I_2, ...I_{(M-1)}\right], ...\ P_N = \left[I_1, I_2, ...I_{(M-1)}\right] \tag{16}$$

G_2 group

$$P_1 = \left[I_M\right],\quad P_2 = \left[I_M\right], ... P_N = \left[I_M\right] \tag{17}$$

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.

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

TABLE 3: Matching in group G_1 among person P_1

In group G1, each hand feature vector in P_1 is matched with all other (m-1) hands feature vector by Hamming Distance similarity measurement method. The matching values are approaching “0” and are stored in threshold array.

$$TA_1 = \begin{bmatrix} HD_{12}, HD_{13}, ..HD_{1M-1}, HD_{21}, HD_{23}, ...HD_{2M-1}, ... \\ ....HD_{(M-1)1}, HD_{(M-1)2}, ..HD_{(M-1)(M-2)} \end{bmatrix} \tag{18}$$

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

$$T_A = TA_1 + TA_2 + ... + TA_N \tag{19}$$

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 equation (20).

$$\left.\begin{aligned} T_{AiMIN} &= \min(T_{Ai}) \\ T_{AiMAX} &= \max(T_{Ai}) \end{aligned}\right\}_{i=1,\dots N} \tag{20}$$

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.

6. EXPERIMENTAL RESULTS

The effectiveness of the proposed Complex Gabor Wavelet method for feature extraction technique was assessed on the PolyU database [16]. The database consists of 600 palmprint images from 100 palms with 6 samples for each palm.

6.1 Palmprint Authentication System

For the verification experiments the database has to be partitioned into two non-overlapping groups. The first group (G1) represents the training group, consisting of 100 persons with each person having 5 palm sample images whereas the second group (G2) represents the testing group containing 100 persons with each person having one palm image different from the first group images. The image size is 284×384 pixels.

G_1 group

$$P_1 = [I_1, I_2, I_3, I_4, I_5], \quad P_2 = [I_1, I_2, I_3, I_4, I_5], \ldots.. \; P_{100} = [I_1, I_2, I_3, I_4, I_5]$$

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

G_2 group

$$P_1 = [I_6], P_2 = [I_6] \ldots\ldots\ldots\ldots\ldots P_{100} = [I_6]$$

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. Feature extraction is done by Complex Gabor Wavelet Transform to get the imaginary wavelet coefficients. The Complex Gabor coefficients are calculated at seven different orientations respectively. The feature vector contains features extracted by Complex Gabor wavelet at different orientations. Feature vector of all hand images samples is calculated and stored in database. Feature vector matrix is $CGWF = [CGWF_0, CGWF_1, CGWF_2, CGWF_3, CGWF_4, CGWF_5, CGWF_6]$. Sample of ROI is shown in Fig.10.

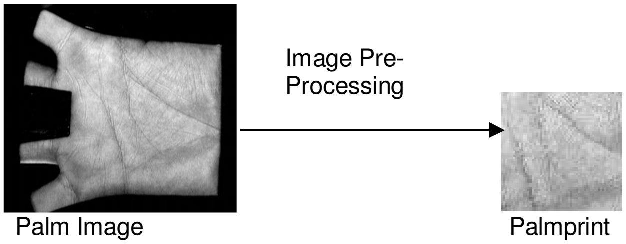


FIGURE 10: Sample of ROI.

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

6.2 Reference Threshold Calculation

In group G1, each hand feature vector in P_1 is matched with all other 4 hands feature vector by Hamming Distance similarity measurement method. Similarly, all 100 hand image samples 2000 matching values are stored in Threshold array (TA).

$$T_A = TA_1 + TA_2 + ... + TA_{100}$$

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

The maximum and minimum values are found out from threshold array.

These 25 threshold values are tested with group G_2 and group G_1 images. The 25 threshold values, the respective FAR and FRR values and accuracy at frequency sixteen and theta zero are shown in table 4.

Reference Threshold	FAR	FRR	Accuracy
5.46E-01	1.23E-01	6.69E-03	9.35E+01
5.52E-01	1.27E-01	4.91E-03	9.34E+01
5.58E-01	1.16E-01	2.67E-03	9.40E+01
5.65E-01	9.25E-02	1.33E-03	9.53E+01
5.71E-01	**6.95E-02**	**6.94E-03**	**9.62E+01**
5.77E-01	4.92E-02	3.74E-04	9.75E+01
5.84E-01	3.59E-02	2.75E-04	9.82E+01
5.90E-01	2.71E-02	3.13E-04	9.86E+01
5.96E-01	2.16E-02	1.85E-04	9.89E+01

TABLE 4: Threshold Values, FAR, FRR, Accuracy Values for SWM.

Method	FAR	FRR	Accuracy
SWM	6.95E-02	6.94E-03	9.62E+01
SWM1	6.94E-02	7.13E-03	9.62E+01
SWM2	7.07E-02	6.93E-03	9.61E+01

TABLE 5: FAR, FRR, Accuracy values for sliding window size 2

From the table 5 it can be concluded that by using SWM1 and SWM2 techniques the accuracy of the system is not affected.

The FAR and FRR are values are plotted with respect to 25 threshold range values. From the graph the value of reference threshold is chosen where FAR and FRR are minimum. Plot of FAR and FRR is shown in figure 11. The plot between accuracy and threshold is shown in figure 11.

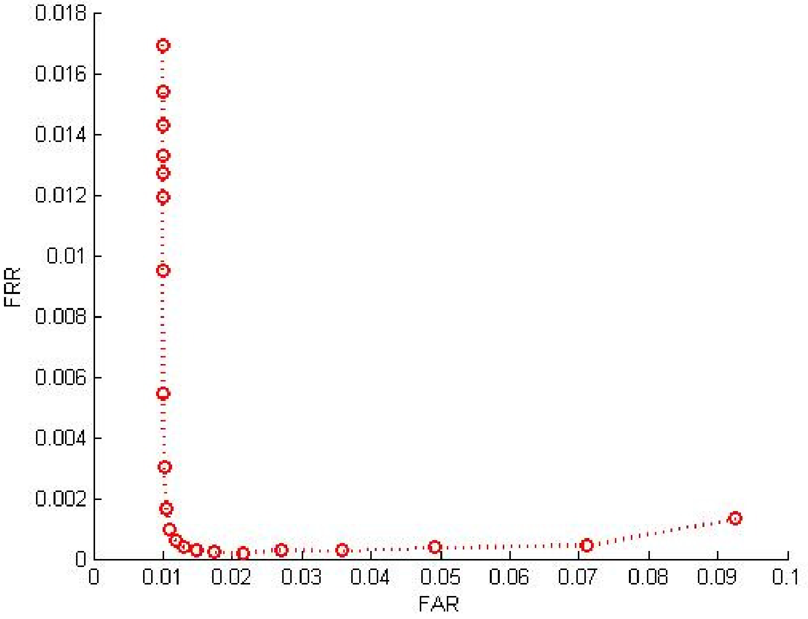


FIGURE 11: FAR Vs FRR

Accuracy values with respect to threshold values are plotted in Fig. 12.

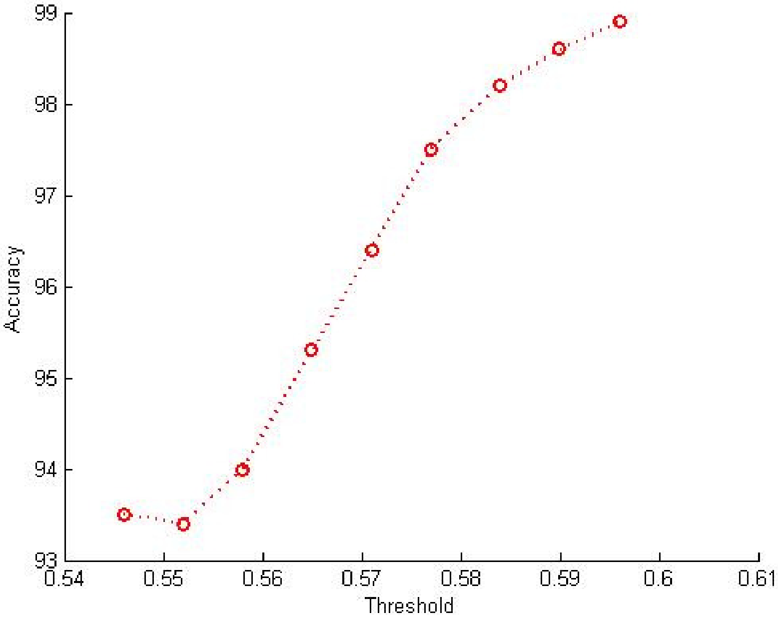


FIGURE 12: Accuracy Vs Threshold

6.3 Speed Performance

Table 1 shows the reduction in comparison time by sliding window method 1 (SWM1) and sliding window method 2 (SWM2). It can be observed that with palmprint segments, the comparison time reduces and speed to verify the person is improved.

7. CONCLUSION

The Complex Gabor Wavelet transform is applied to each palmprint image in the database to get imaginary wavelet coefficients. The features are computed at different orientations. The feature vector generated from complex gabor features are matched by hamming distance using sliding window method. Modified sliding window methods SWM1 and SWM2 are also used to make the matching system fast. Experimental results clearly show that our proposed Complex Gabor Wavelet method can discriminate palmprints and is fast authentication system.

8. REFERENCES

[1] Jain A.K., Ross A., Prabhakar S.: ‘An introduction to biometric recognition’, IEEE Trans. Circuits Syst. Video Technol., 14, (1), pp. 4–20, 2004.

[2] Pavesˇlc´ N., Ribaric´ S., Ribaric ´ D.: ‘Personal authentication using hand-geometry and palmprint features – the state of the art’. Proc. Workshop: Biometrics – Challenges Arising from Theory to Practice, Cambridge, pp. 17–26, 2004.

[3] Kumar and D. Zhang, “Combining fingerprint, palmprint and handshape for user authentication,” In *Proceedings of ICPR*, vol.4, pp.549- 552.

[4] Kumar A., Zhang D.: ‘Personal authentication using multiple palmprint representation’, Pattern Recognit., 38, (10), pp. 1695–1704, 2005.

[5] Zhang, D., Kongi, W., You, J., and Wong, M.: “Online palmprint identification”, IEEE Trans. Pattern Anal. Mach. Intell., vol. 25, no. 9, pp. 1041–1049, 2003.

[6] B.S. Manjunath, “Gabor Wavelet Transform and Application to Problems in Early vision”, IEEE, pp. 796-800.

[7] Wu, X., Zhang, D., and Wang, K.: “Fisherpalms based palmprint recognition”, Pattern Recognit. Letters, vol. 24, no. 15, pp. 2829–2838, 2003.

[8] Hu, D., Feng, G., and Zhou, Z.: “Two dimensional locality preserving projections with its applications to palmprint recognition”, Pattern Recognit, 40, pp. 339–402, 2007.

[9] Lu, G., Zhang, D., and Wang, K.: “Palmprint recognition using eigenpalms features”, Pattern Recognit. Lett., vol. 24, no.9–10, pp. 1463–1467, 2003.

[10] Liu, C.: “Gabor-based kernel PCA with fractional power polynomial models for face recognition”, IEEE Trans. Pattern Anal. Mach. Intell., vol. 26, no. 5, pp. 572–581, 2004.

[11] J. Daugman, “Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression,” IEEE Trans. Acousr. Speech Signal Proc., vol. 36, no. 7, pp. 1169-1 179, 1988.

[12] Sylvain Fischer and Gabriel Cristdbal, “Minimum Entropy Transform Using Gabor Wavelets For Image Compression”, IEEE, pp 428-433, 2001.

[13] Jain and F. Farrokhnia, “Unsupervised texture segmentation using gabor filters. Pattern Recognition, 24(12):1167– 1186, 1991.

[14] Say Song Goh, Amos Ron, Zuowei Shen “Gabor and Wavelet Frames”.

[15] Muwei Jian, Haoyan GUO, Lei Liu, “Texture Image Classification Using Visual Perceptual Texture Features and Gabor Wavelet Features”, Journal Of Computers, Vol. 4, No. 8, Academy Publisher, pp 763-770,Aug. 2009.

[16] The PolyU Palmprint Database: http://www4.comp. polyu.edu.hk/biometrics/