Personal Verification and Identification Using Hand Geometry

Nongluk Covavisaruch, **Pipat Prateepamornkul**,
Puripant Ruchikachorn, and **Piyanaat Taksaphan**, Non-members

ABSTRACT

This research proposes a study of personal verification and identification using hand geometry. Hand geometry used in this research consists of the lengths and widths of fingers and the width of a palm. Users can place their hands freely without the need for pegs to fix the hand placement. In this research, systems using six different distance functions were tested and compared. Test data are from 96 users. Among the six different distance functions, S_1 gives the best results in both verification and identification.

Keywords: Biometric, Hand geometry, Verification, Identification

1. INTRODUCTION

At present, authentication plays an important role in the world's society and it is widely used in general security system. Conventional authentication systems are knowledge-based systems that users have to memorize passwords, and token-based systems that users have to own ID cards. The important drawbacks of these methods are as follows: (i) possessions can be lost, stolen or easily duplicated; (ii) knowledge can be forgotten; and (iii) both possessions and knowledge can be shared or stolen. Using physiological or behavioral characteristics, or biometric, provides a solution to these conventional problems and presents more secure and more reliable authentication systems.

Biometric is gaining more attention in recent years. There are many biometric systems based on different characteristics and different parts of the human body. Biometric systems that are widely used are based on iris, fingerprint, face, and hand. Each biometric has its strengths and weaknesses depending on its application and utilization. This research focuses on hand geometry which is one of many effective biometric systems. Hand geometry refers to the geometric structure of the hand that is composed of the lengths of fingers, the widths of fingers, and the width of a palm, etc. The advantages of a hand geometry system are that it is a relatively simple method that can use low resolution images and provides high efficiency with great users' acceptance. [1, 2]

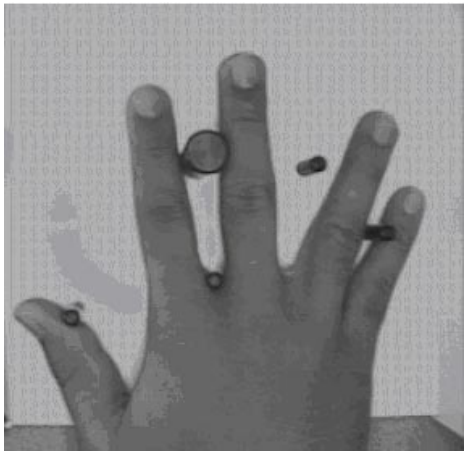

Fig.1: *Incorrect placement of a hand [3].*

Traditional hand geometry system always uses pegs to fix the placement of the hand [3, 4, 5, 6]. Two main weaknesses of using pegs are that pegs will definitely deform the shape of the hand silhouette and users might place their hands incorrectly [7] as shown in Fig. 1. These problems can certainly reduce the performance of the biometric system.

This research purposes on a study of a biometric system based on hand geometry without pegs to control the placement of the hand. Therefore, users can place their hands freely. This type biometric system is not complex and yields good performance.

2. METHODOLOGY

Hand geometry features are extracted from an image by 3 steps as follows: image acquisition, image pre-processing and feature extraction.

2.1 Image Acquisition

The image acquisition system comprises of a light source, a CCD digital camera, and a black flat surface used as a background. A user places one hand, pointing up, on the flat surface with the back of the hand touching the flat surface. The user can place a hand freely since there is no peg to fix the position of the hand. Then an image is acquired by using a CCD

Manuscript received on June 1, 2005 ; revised on February 15, 2006.

The authors are with Department of Computer Engineering, Faculty of Engineering Chulalongkorn University, Bangkok 10330, Thailand. e-mail: nongluk.c@chula.ac.th, pipatp1@hotmail.com, puripant@hotmail.com and waew@hotmail.com

Fig.2: *Example images from image preprocessing process.*

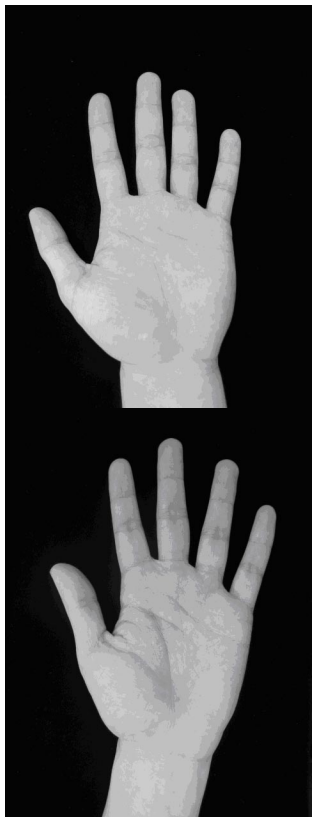

Fig.3: *Various poses of hand placement.*

digital camera. Users are only requested to make sure that their fingers do not touch one another and that the back of the hand lies flat and stays on the flat surface. In our experiments, only the left hand images of the users are acquired.

2.2 Image Preprocessing

Since the acquired image is a color image, it is converted to a grayscale image. Median filter is applied to remove noise in the image. Because of the black background, there is a clear distinct in intensity between the hand and the background. Therefore, the histogram of the image is bimodal. The image can be easily converted to a binary image by thresholding. The threshold value is automatically computed using Otsu method [8, 9]. Then the border of the hand silhouette is smoothed by using morphological opening and closing. The result is shown in Fig. 2.

2.3 Feature Extraction

Since there is no peg to fix the placement of hand, users can place their hands in various positions as shown in figure 3. Before extracting the hand features, the "landmark points" [7] have to be located. These landmark points include the fingertips and valley points that can be seen in Fig. 4.

Firstly, the reference position on a wrist, as shown in Fig. 4, must be found. By scanning the pixels at the bottom of the image from left to right, the left-most pixel of the hand image, $S1$, and the right-most pixel, $E1$ are located. The reference point is simply the middle point between $S1$ and $E1$.

The next step is to find all the fingertips and valley points of the hand. The distances between the reference point and each contour point of the hand, from $S1$ to $E1$, are measured by Euclidean distance as defined in equation 1.

$$D = \sqrt{(x - x_r)^2 + (y - y_r)^2} \tag{1}$$

where (x, y) is a point in the contour and (x_r, y_r) is the reference point.

Comparing the distances with those of other neighbor points' on the hand contour in some distances, the fingertips are the points that have the most distances, and the valley points, the least. The result positions of fingertips and valley points are marked as circles and shown in Fig. 4.

The extracted features used in our research are the *lengths of each finger*, the *widths of each finger* at 3 locations and the *width of the palm*. This results in 21 features all together. These features can be found as follows.

2.3.1 Finger Baselines

The finger baselines of a middle finger and a ring finger are obtained by connecting the valley points

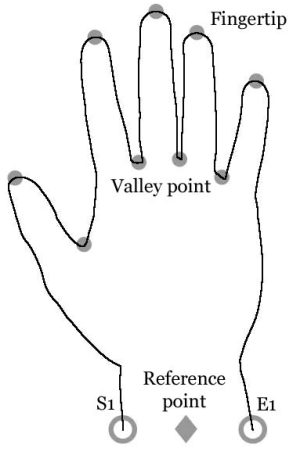

Fig.4: *Fingertips and valley points of a hand.*

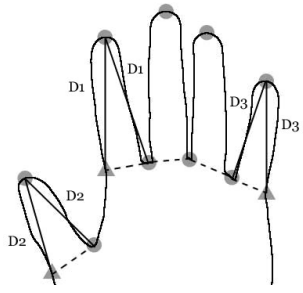

Fig.5: *Definitions of finger baselines.*

which are on both sides of that particular finger. However, for a thumb, an index and a little finger; each has only one adjacent valley point. Thus, in our research, the other valley points are assumed to be on the opposite side of the finger with the same distance from the fingertip to the existing valley point. For example, the located valley point of an index is on the right of the index contour with a distance $D1$ from the index fingertip as shown in Fig 5.

Therefore, the assumed other valley point of the index must be $D1$ distance on the left of the index contour as well. All valley points are located and shown in Fig. 5. Baselines are the lines connected between two valley points, also shown in Fig. 5 as dashed lines.

2.3.2 Finger Lengths

The "*finger lengths*" are obtained by measuring the distances from the fingertips to the middle points of the finger baselines. These *finger lengths* are shown in Fig. 6.

2.3.3 Finger Widths

In this research, the "*finger widths*" are the widths of a finger measured at 3 locations as shown in Fig. 6. The first one is measured at the middle of the finger length, the second one, at the one-third, and the last one, at the two-third of the finger length. All the finger widths are shown in Fig. 6.

2.3.4 Palm Width

The "*palm width*" is the distance from $b1$ to $b2$ in

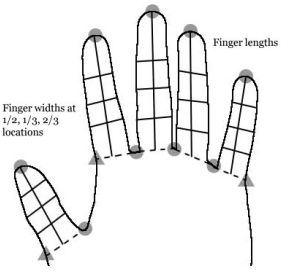

Fig.6: *Definitions of finger lengths and widths.*

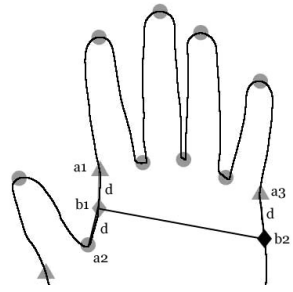

Fig.7: *Definitions of a palm width.*

Fig. 7. The point b1 is half way between a1 (which is the assumed valley point of an index) and $a2$ (which is the valley point of a thumb). The point $b2$ is defined to be equal distance from $a3$ (which is the assumed valley point of the little finger) as that of half the distance from $a1$ to $a2$. All the 21 features are as shown in Fig. 8.

3. PERSONAL VERIFICATION AND IDENTIFICATION

A biometric system is like other authentication systems in that an authorized user has to register oneself to the system before verification or identification can be accomplished. The extracted bio data of the registered person is stored as a template in a database. In order to authorize an individual, the system matches the claimer's bio data with the template(s) in the

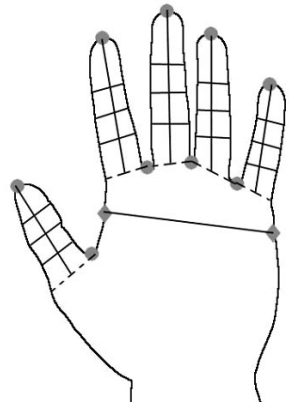

Fig.8: *Hand geometry features.*

database. The matching process can be divided into two types based on the application. They are verification and identification. Distance functions are utilized in the matching process to help differentiate the authorized and unauthorized persons. More details of this process are described in this section.

3.1 Verification and Identification

For a verification system, an individual claims as one of the authorized users previously registered to the system. The system confirms or denies the claimer by matching the individual's extracted data with those of the claimed person which is stored in a database. Therefore, a verification system does a one-to-one matching process. For an identification system, the claimer's extracted data are matched with those of all registered persons. The system then establishes or recognizes the identity of the individual. Identification is therefore a one-to-many matching process.

3.2 Distance Functions

As mentioned earlier, a personal verification system and an identification system compare the claimer's bio data with the templates in the database. Distance functions are used to decide whether the claimer is the claimed person or as whom the claimer is recognized. In this research, 6 distance functions are experimented as follows.

1. Absolute Distance

$$D_a = \sum_{i=1}^{d} |y_i - f_i| \tag{2}$$

2. Weighted Absolute Distance

$$D_{wa} = \sum_{i=1}^{d} \frac{|y_i - f_i|}{\sigma_i} \tag{3}$$

3. Euclidean Distance

$$D_e = \sqrt{\sum_{i=1}^{d} (y_i - f_i)^2} \tag{4}$$

4. Weighted Euclidean Distance

$$D_{we} = \sqrt{\sum_{i=1}^{d} \frac{(y_i - f_i)^2}{\sigma_i^2}} \tag{5}$$

5. D_1 Distance

$$D_{d_1} = \sum_{i=1}^{d} \frac{|y_i - f_i|}{y_i + f_i} \tag{6}$$

6. S_1 Distance

$$D_{s_1} = \frac{1}{n} \sum_{i=1}^{d} \frac{min\,(y_i, f_i)}{max\,(y_i, f_i)} \tag{7}$$

where $F = \langle f_1, f_2, \ldots, f_d \rangle$ is the feature vector with d dimension of a registered user in the database, and

$Y = \langle y_1, y_2, \ldots, y_d \rangle$ is the feature vector of an unknown or a claimer, and

σ_i^2 is the feature variance of the i^{th} feature.

After calculating the distance, the system compares the result with a predefined threshold and classifies the claimer. The system accepts the claimer *if and only if* the calculated distance is lower than the threshold, and it rejects the claimer *if and only if* the calculated distance is higher than the threshold.

4. EXPERIMENTS AND RESULTS

In our research, we divide the tests into 2 operation modes, a verification mode and an identification mode. Six different distance functions, as shown in section 3.2, are used in the feature matching process. The data used in the experiments are described in section 4.1 and the experiments and results of a verification mode and an identification mode are illustrated in sections 4.2 and 4.3 respectively.

4.1 Data Used in Our Experiments

There are 96 test users in our experiments. Ten left-hand images are acquired from each user. These images are divided into 2 groups. The first group consists of the images of all 96 users, 5 images from each user. They are used for the enrolment process to define the users' templates, or feature vectors. The features are extracted as mentioned earlier in section 2.2. For each user, the average and the variance of each extracted feature are kept as the user's template in a database. The rest of the images form the second image group. These images are used for testing the system performance.

4.2 Experiments and Results from Verification Mode

The system performance can be measured from the errors of the system. There are 2 types of errors; FRR (False Rejection Rate) and FAR (False Acceptance Rate). The FRR is the percent error of a system that rejects genuine users as imposters while FAR is the percent error of a system that accepts imposters as genuine users.

The FRR is obtained from testing the system by matching the extracted features of the same person. In other words, the test image data of a claimer must be extracted and matched with the template of the same person. Distance (D_x - one of the six distances in this research) between the extracted feature vector of the claimer and the template of the same person from the database is measured. The system decides whether it will accept or reject the claimer by comparing the distance to a predefined threshold. The FRR is computed by equation (8) as follows:

$$FRR = \frac{\sum_{i=1}^{N} f(x_i)}{N},$$
$$f(x_i) = \left\{ \begin{matrix} 1, \ D_x(F_i, Y_{C_i}) > T \\ 0, \ Otherwise \end{matrix} \right\} \quad (8)$$

where F_i is the feature vector of the test image of the i^{th} user.
F_{C_i} is the feature vector template of the claimed identity that, in this case, the same person as the claimer.
T is a predefined threshold.
$f(x_i)$ is the function that equals one when the distance is higher than the threshold.
$D_x(F_i, Y_{C_i})$ is the distance measured from matching the feature vector with the template Y_{C_i}.
N is the total number of test claimers' images.

In contrast with the FRR, the FAR is obtained by testing the system by matching the extracted features of a claimer with the templates of other registered persons'. In this research, the templates of other registered users' are randomly selected for matching. Distance (D_x) from the matching process is measured and compared with a predefined threshold. In order to make the FAR and FRR comparable, the predefined thresholds used for the processes of finding FRR and FAR must be set equally. The FAR is calculated by equation (9) as follows:

$$FAR = \frac{\sum_{i=1}^{N} f(x_i)}{N},$$
$$f(x_i) = \left\{ \begin{matrix} 1, \ D_x(F_i, Y_{C_i}) < T; \ i \neq j \\ 0, \ Otherwise \end{matrix} \right\} \quad (9)$$

where Y_{C_i} is the feature vector template of a different user.

For a verification system, the optimal performance of the system is where the FRR equals the FAR. Figure 9 illustrates the results from our experiments in graphs of FAR and FRR versus thresholds. Therefore, the system's optimal point is the intersection of the FAR and the FRR. This point is called CER (Crossover Error Rate) or EER (Equal Error Rate).

Table 1: *Percent error from verification mode*

Distance function	CER (%)
Absolute	3.374
Weighted Absolute	3.208
Euclidean	4.374
Weighted Euclidean	4.083
D_1	3.625
S_1	2.999

In general, CER or EER is used to compare the system's optimal performance. The system with lower CER has better performance than the system with higher CER. Therefore, the performances of our systems with six different distance functions are compared by CER as shown in Table 1. The experimental results reveal that the system using S_1 distance function has the best performance.

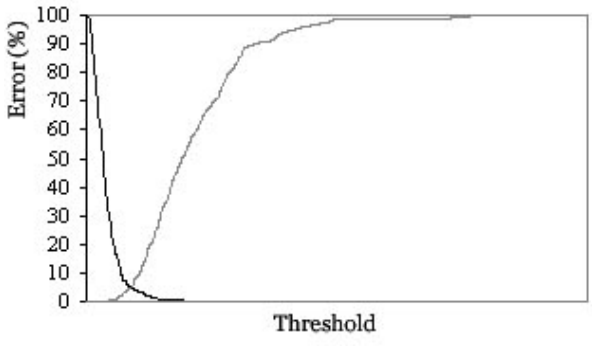

Fig.9: *Graph FAR and FRR with vary threshold.*

Nevertheless, it is not necessary to set the threshold to achieve the system's optimal performance in practice. Setting the system threshold actually depends more on the applications. For example, a good high security system should not reject a genuine user and, at the same time, it should not accept an imposter either. Therefore, this kind of system needs a threshold that yields high FRR and low FAR. On the other hand, some applications might prefer lower security level to gain users' acceptance. The reason is that genuine users might be annoyed if they are rejected. The threshold for such a system can be set to a value that lower FRR and thus, higher the FAR. However, this type of system must be able to take some risks from imposters.

The performance of a system can be shown with ROC (Receiver Operating Characteristics) curve. It is a curve plotted between FAR and FRR, or sometimes, the FRR is changed to GAR (Genuine Acceptance Rate) which is calculated by equation (10).

$$GAR = 1 - FRR \quad (10)$$

Figure 10 shows the ROC curves of the systems from our study. It can be seen from our experiments that, at the same FAR in the range of fewer than 10%, the system with S_1 distance function gives the highest GAR, and hence highest security, while the system using Euclidean distance function provides the least GAR.

4.3 Experiments and Results from Identification Mode

For the identification mode, a claimer is matched with all the registered identities. If all matching distances are higher than a predefined threshold, it

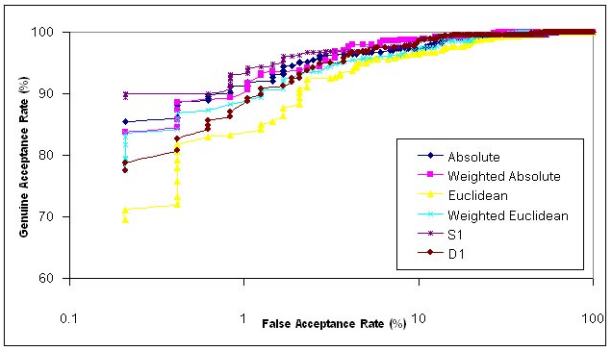

Fig.10: *The ROC of six different distance functions*

means that the claimer or the test user, which is actually one of the registered users, is classified as an imposter by the system. If there exists one or more matching distances lower than threshold, the system identifies the claimer as the registered user that gives the least distance.

In identification mode, the system's least percent error is used for system performance evaluation. The system's least percent error can be found by varying the threshold values until the system yields the least error. Table 2 reports the percent errors of a system with different distance functions from our experiments.

From both verification and identification experiments, it is found that S_1 distance function gives the best performance. This is due to the fact that each feature distance is normalized before they are averaged. Weighted absolute distance and weighted Euclidean distance result in better system performance than the absolute distance and Euclidean distance. It is because each feature distance is adjusted with its variance so that they can classify the difference between users more effectively and hence, resulting in better system's performance.

Table 2: *Percent error of identification*

Distance function	Error (%)
Absolute	12.50
Weighted Absolute	11.04
Euclidean	18.54
Weighted Euclidean	12.29
D_1	6.04
S_1	5.83

5. CONCLUSIONS

This research purposes a study of personal verification and identification using hand geometry. Users can place their hands freely without need of pegs to fix the placement of a hand. The features used for matching are the lengths of the fingers, the widths of the fingers at one-third, half and two-thirds of the finger length, and finally, the width of the palm.

In our study, systems with 6 different distance functions are tested in verification and identification modes. The images used for enrolment and testing are acquired from 96 users. In the verification mode, S_1 distance gives the best performance, with 3% CER. In identification mode, S_1 distance also yields the best system performance with 94% accuracy and approximately 6% error.

References

[1] A. K. Jain, A. Ross, and S. Prabhakar,"An Introduction to Biometric Recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, Special Issue on Image- and Video-Based Biometrics, Vol. 14, No. 1, pp. 4-20, Jan. 2004.

[2] John Chirillo, and Scott Blaul,Implementing Biometric Security, John Wiley & Sons, Apr. 2003.

[3] A. K. Jain, A. Ross, and S. Pankanti,"A Prototype Hand Geometry-based Verfication System," *2nd International Conference on Audio- and Video-based Biometric Person Authentication*, pp. 166-171, Mar. 1999.

[4] A. K. Jain and N. Duta,"Deformable Matching of Hand Shapes for Verfication," *IEEE International Conference on Image Processing*, pp. 857-861, Oct. 1999.

[5] R. Sanchez-Reillo,"Hand Geometry Pattern Recognition Through Gaussian Mixture Modeling," *15th International Conference on Pattern Recognition*, Vol. 2, pp. 937-940, Sep. 2000.

[6] R. Sanchez-Reillo, C. Sanchez-Avila, and A. Gonzalez-Marcos,"Biometric Identification Through Hand Geometry Measurements," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 10, pp. 1168-1171, 2000.

[7] Alexandra L.N. Wong and Pengcheng Shi,"Peg-Free Hand Geometry Recognition Using Hierarchical Geometry and Shape Matching," *IAPR Workshop on Machine Vision Applications*, Nara, Japan, pp. 281-284, Dec. 2002.

[8] N. Otsu,"A Threshold Selection Method From Gray-scale Histogram," *IEEE Transaction Syst., Man, Cybern.*, Vol. 8, pp. 62-66, 1978.

[9] Linda G. Shapiro, and George C. Stockman, Computer Vision, Prentice Hall, Jan. 2001.

Nongluk Covavisaruch received an M.S. degree in Electrical Engineering from University of Missouri-Columbia and an M.A. (Language and International Trade) from Eastern Michigan University. She is an Assistant Professor of the Department of Computer Engineering at Chulalongkorn University. She has joined the department since 1990 and teaches both undergraduate and graduate courses in digital image processing. She served as the head of Computer Graphics and Computer Imaging Laboratory from 2001-2005. Her research interests include image and vision computing techniques, biometrics, biomedical image processing, image processing and computer vision applications and colors in computers.

Pipat Prateepamornkul received B. Eng. degree in computer engineering from King Mongkut's Institute of Technology - Ladkrabang, Thailand, in 2003, and M. Eng. degree in computer engineering from Chulalongkorn University, Thailand, in 2005. His research interests include biometric and image processing.