ITIHand: A Real System for Palmprint Identification *

José García-Hernández, Roberto Paredes, Ismael Salvador and Javier Cano

Instituto Tecnológico de Informática
Universidad Politécnica de Valencia
Camino de Vera s/n, 46071 Valencia (Spain)

Abstract. In the networked society there are a great number of systems that need biometric identification, so it has become an important issue in our days. Biometrics takes advantage of a number of unique, reliable and stable personal physiological features, to offer an effective approach to identify subjects. This identification can be based on palmprint features. At present work is described a real biometric identification system based on palmprints that uses local features.

1 Introduction

The biometric automatic identification has become an important issue in our days. *Biometric identification methods* [1, 2] are those that allow us to recognise a subject using physiological or behavioural features. The pattern of palm hand lines is an example of physiological feature used in our days [3, 4]. As in fingerprints identification, twin brothers can be also distinguished because their palmprints are similar but not identical. Moreover, palm hand features are more difficult to hide than finger features by dirt or acidics.

In [4] we show a biometric identification method that uses *local features* of the palmprint. Using local features with nearest neighbour search and a direct voting scheme achieves excellent results for many image classification tasks [4–9].

In a typical classification scenario, each object is represented by a feature vector, and a classification procedure like the k-NN rule, Gaussian Mixtures, Neural Networks, etc is applied in order to formulate an hypothesis about the identity of a test vector that represents a whole object. In the local feature case, however, each image is represented by many feature vectors and a classification procedure is applied to formulate an hypothesis about the identity of each vector. As each feature vector could be classified into a different class, a decision scheme is required to finally decide the class of the test image.

In this work we present a real implementation of one of the methods shown in [4]. First, we present the theoretical approach followed. Then a real implementation, its hardware and software, is shown. Finally, some experiments and conclusions are reported.

* Work supported by the Spanish Project DPI2004-08279-C02-02

García-Hernández J., Paredes R., Salvador I. and Cano J. (2006).
ITIHand: A Real System for Palmprint Identification.
In *Proceedings of the 2nd International Workshop on Biosignal Processing and Classification*, pages 33-40
DOI: 10.5220/0001222900330040

2 Theoretical Approach

ITIHand classification method is based on the use of *local features*. As said, local representation implies that each image is scanned to compute many feature vectors belonging to different regions of the image. These regions correspond to the pixels with higher information content. There is not a unique method to select pixels from the image or to preprocess the image before pixels selection. In our work, we have used the local variance in a small window in order to select pixels and local equalisation to preprocess the image. Both methods are shown in [4].

2.1 Image Preprocessing

Image preprocessing is performed by a single *local equalisation* function. In this type of equalisation the image is cropped, starting in the upper-left corner, with a window of size v, such as $v \ll D$ and being $D \times D$ the image dimension. A given equalisation function is applied to the cropped image. This process is repeated by moving the crop all over the image and applying the equalisation for each one. In our case, the used *equalisation* function is called histogram equalisation. In this function the result is obtained using the cumulative density function of the image as a transfer function. The result of this process is that the histogram becomes approximately constant for all the gray values. For a given image of size $M \times N$ with G gray levels and cumulative histogram $H(g)$ this transfer function is given in equation (1).

$$T(g) = \frac{G-1}{MN} H(g) \tag{1}$$

In figure 1 an example of local equalisation using histogram equalisation is shown.

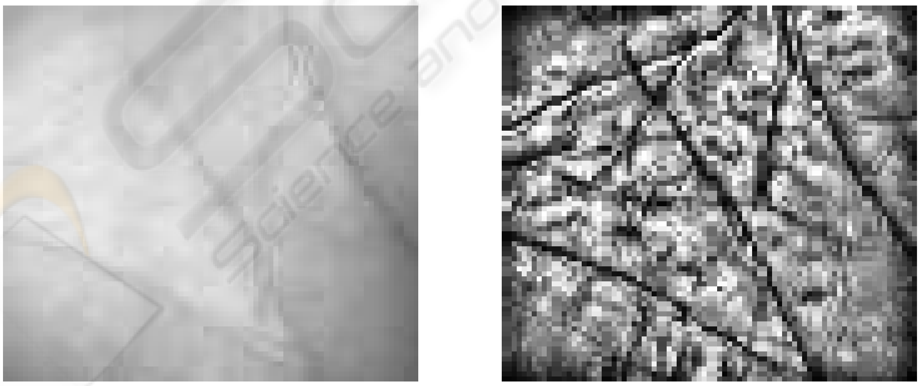

Fig. 1. Equalisation example. Left: original image. Right: locally equalised image.

2.2 Local Features Extraction

As local features object codification method represents each image by many feature vectors, once the image is preprocessed we select the n vectors with higher information content. For this purpose, we have chosen a simple and fast method: the local variance in a small window is measured for each pixel and the n pixels with a greater variance are selected. In this work we have used a window with a fixed size of 5×5.

For each selected pixel, a w^2-dimensional vector of grey values is obtained from the preprocessed image by application of a $w \times w$ window around it, such as $w \ll D$ and being $D \times D$ the image dimension. The dimension of the resulting vector is then reduced from w^2 to 30 using *Principal Component Analysis* (PCA), thus obtaining a compact local representation of the $w \times w$ window. A value of 30 for the dimension has been chosen because this value provides the best performance in most previous classification tasks. The process is illustrated in figure 2.

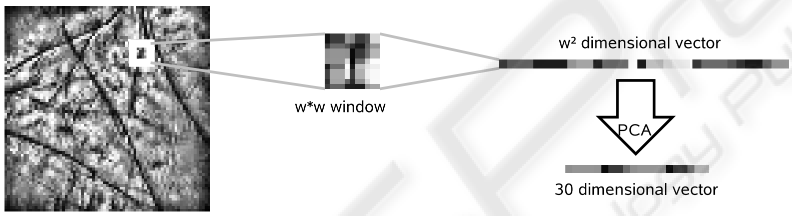

Fig. 2. Local features extraction process.

2.3 Classification through a k-NN based Voting Scheme

In a global classifier, each object is represented by a feature vector, and a discrimination rule is applied to classify a test vector that also represents one object. As discussed before, local representation, however, implies that each image is scanned to compute many feature vectors. Each one can be classified into a different class, and therefore a decision scheme is required to finally decide a single class for a test image.

Let Y be a test image. Following the conventional probabilistic framework, Y can be optimally classified in a class $\hat{c}$ having the maximum posterior probability among C classes. By applying the feature extraction process described in the previous section to Y, a set of m_Y feature vectors, $\{\mathbf{y}_1, \ldots, \mathbf{y}_{m_Y}\}$ is obtained. An approximation to $P(c_j|Y)$ can be obtained using the so called "*sum rule*" and then, the expression of $\hat{c}$ becomes:

$$\hat{c} = \arg\max_{1 \le j \le C} \sum_{i=1}^{m_Y} P(c_j|\mathbf{y}_i) \tag{2}$$

In our case, posterior probabilities are directly estimated by k-Nearest Neighbours. Let k_{ij} the number of neighbours of $\mathbf{y}_i$ belonging to class c_j. Using this estimate in (2), our classification rule becomes:

$$\hat{c} = \arg\max_{1 \leq j \leq C} \sum_{i=1}^{m_Y} k_{ij} \tag{3}$$

That is, a class $\hat{c}$ with the largest number of *"votes"* accumulated over all the feature vectors belonging to the test image is selected. This justifies why techniques of this type are often referred to as *"voting schemes"*.

3 The ITIHand Hardware

The ITIHand hardware is built by using an ABS-plastic box, whose size is $250 \times 160 \times 150$ mm (figure 3). An Unibrain Fire-i^{TM} Digital Camera [1] is installed inside. Although it is a color camera, we use it in gray-scale mode. High resolution images are not required in this task. The camera captures the palmprint image that is seen through a squared window in the top of the box. The image of the palmprint is lighted by 8 white led diodes. The ITIHand hardware has a FireWire interface, a DC 12V input and a light switch. Its use is as simple as placing the hand over the window, as it is shown in figure 3.

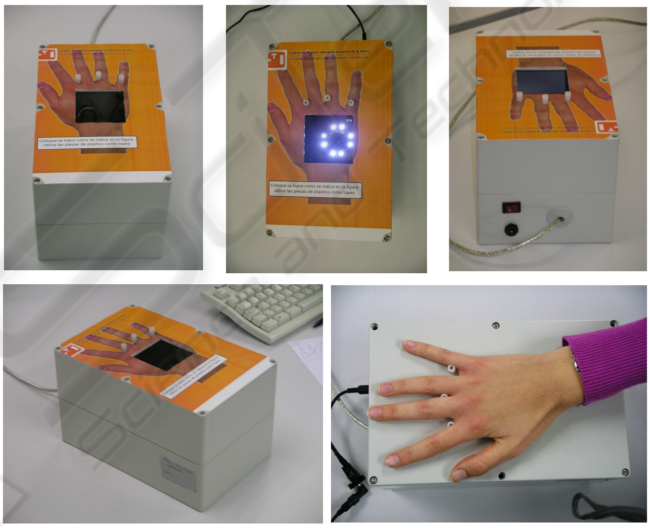

Fig. 3. ITIHand device from different views and its use.

[1] http://www.unibrain.com/1394_products/firei_dig_cam/digital_camera_pc.htm

4 The ITIHand Software

The ITIHand software is the implementation of the method shown in section 2 joined with a graphical interface (figure 4). It uses $GTK2.0$ and runs on *Linux* operating system. Its use is very intuitive and similar to other window-interface applications. On the left of the interface we can see the image shown by the camera.

The software has 2 function modes: training (by clicking ”Entrenar Identificador” button) and identification (by placing the hand as in figure 3 and clicking “Identificar” button). Identification mode gives the name of the user who has placed the hand if he/she is in the database previously acquired. The user is rejected by the system if he/she is not. Train image acquisitions are performed by an external application that does not use a graphical interface.

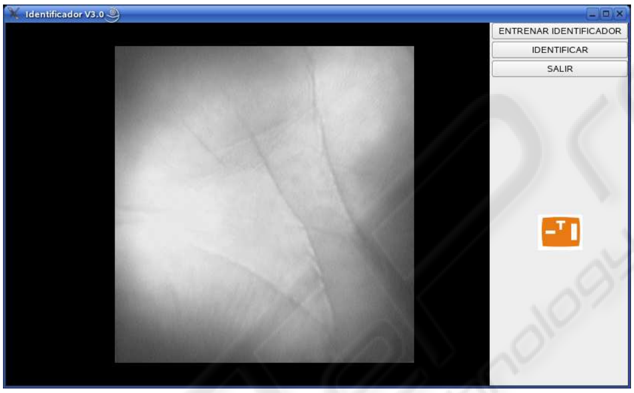

Fig. 4. ITIHand software interface.

5 Experiments

The experiments were designed to estimate the performance of the ITIHand and the procedure shown in section 2 that the ITIHand uses. So, this procedure is used in all the experiments shown in this section. The experimental method was verification by the *True Imposter Protocol* [10]. In this method the used database is split in two set. While the samples included in the first one are used as *clients* the samples included in the second one are used as *impostors*.

In the experiments we have used two databases: ITIHand database and PolyU Palmprint Database [11], in order to evaluate the performance of the system with two different and independent data bases.

On the one hand, the first group of experiments was done by using our own database, which is called *ITIHand database* and was built by using the device. It comprises images of the right palm of 53 ITI staff and collaborators, 3 samples per user. We split the users in two sets, 25 were used as clients and 28 as impostors. For clients, we use 1 sample for training and 2 for testing. The number of local features obtained per image

(n) has been ranged from 100 to 500 with steps of 50. For clarifying we only show the results using $n = 250$. This value is the smallest which we obtain the best result for. Besides, the experiments were carried out with different values of local window dimension ($w \times w$) and equalisation window dimension ($v \times v$).

The results are shown in table 1 for each combination of w and v. As can be seen, ITIHand achieves a very good performance. For instance, for $w = 19$ and $v = 12$, False Positive and False Negative Rates of 0% are achieved with a verification threshold of 0.16 with this database.

Table 1. Verification results of database made by use the ITIHand device. (FP=False Positive.FN=False Negative.ER=Error Rate.THRES=Threshold).

w	v	FP (%)	FN (%)	ER (%)	THRES
11	8	0.76	4.00	2.38	0.108001
15	8	0.14	0.00	0.07	0.140002
19	8	0.38	0.00	0.19	0.160002
11	10	2.38	0.00	1.19	0.092001
15	10	0.00	0.00	0.00	0.140002
19	10	0.05	0.00	0.02	0.152002
11	12	0.86	2.00	1.43	0.104001
15	12	0.05	0.00	0.02	0.136002
19	12	0.00	0.00	0.00	0.160002

On the other hand, the second group of experiments was done by using the *PolyU Palmprint Database*, created by the Biometric Research Center of Hong Kong [11]. This database contains 600 grayscale palmprint images from 100 different palms, 6 images for each one. For clients we have selected 3 images for training and 3 for testing. The database is more detailed in [11] and a bigger version is used and described in [3].

As in the previous database, the number of local features obtained per image (n) has been ranged from 100 to 500 with steps of 50. For clarifying and an easier comparison with the other database results, we only show the results using $n = 250$. Besides, the experiments were also carried out with different values of local window dimension ($w \times w$) and equalisation window dimension ($v \times v$). As this database is bigger than the first one, we have done 3 different experiment by varying the users selected as clients and as impostors. First, we have used 50 users as clients and 50 as impostors. Second, we have used 75 users as clients and 25 as impostors. Finally, we have used 25 users as clients and 75 as impostors. The results are shown, respectively, in tables 2, 3 and 4 for each combination of w and v.

As can be seen, with this database the used method achieves a very good performance too. For instance, for $w = 19$ and $v = 12$, False Positive Rate of 0.51% and False Negative Rate of 0.67% are achieved with a verification threshold of 0.10 by using 50 clients as users and 50 as impostors. In our previous work [4], more experiments with this second database are shown.

Table 2. Verification results with the *PolyU Palmprint Database* and 50 users as clients and 50 as impostors.(FP=False Positive.FN=False Negative.ER=Error Rate.THRES=Threshold).

w	v	FP (%)	FN (%)	ER (%)	THRES
11	8	1.666667	4.000000	2.833333	0.060000
15	8	2.100000	0.000000	1.050000	0.068001
19	8	0.140000	0.000000	0.070000	0.132002
11	10	1.040000	4.000000	2.520000	0.067901
15	10	0.673333	1.333333	1.003333	0.091901
19	10	0.680000	0.000000	0.340000	0.100001
11	12	1.346667	3.333334	2.340000	0.063901
15	12	1.280000	0.666667	0.973333	0.076001
19	12	0.513333	0.666667	0.590000	0.103901

Table 3. Verification results with the *PolyU Palmprint Database* and 75 users as clients and 25 as impostors. (FP=False Positive.FN=False Negative.ER=Error Rate.THRES=Threshold).

w	v	FP (%)	FN (%)	ER (%)	THRES
11	8	0.577778	2.222222	1.400000	0.051900
15	8	1.040000	0.000000	0.520000	0.052000
19	8	0.044444	0.000000	0.022222	0.108001
11	10	0.515556	2.222222	1.368889	0.051900
15	10	0.657778	0.444444	0.551111	0.056000
19	10	0.231111	0.000000	0.115556	0.084001
11	12	0.862222	1.777778	1.320000	0.044000
15	12	0.266667	0.888889	0.577778	0.067901
19	12	0.595556	0.000000	0.297778	0.064001

6 Conclusions and Future Work

A real implementation to automatically identify subjects by using the palm print features has been presented. The hardware is made by using cheap and easy to find components. It is very easy to build in a little time with common and simple tools.

It uses local features and the classification method previously presented in [4]. The graphical interface is made by using a well known graphical tool as $GTK2.0$. In this moment, the ITIHand system is being used for acquiring more samples for the database. For this purpose, more right palms of ITI staff and collaborators are being acquired. On the other hand, the device must be understood as a prototype. We are working on a smaller one in order to include it in an entry-phone access control.

Future work will be focus in three main areas. First an evaluation of other preprocessing methods, like gabor filters. Second, to improve local features classification: applying global constrains over the relative placement of the local features and/or using other local features selection methods (more discriminative). Third, to use other illumination source, for instance infrared light, and/or to place the light source in a different plane with respect to the palm print to obtain more contrast of the ridges and valleys of the palm print.

Table 4. Verification results with the *PolyU Palmprint Database* and 25 users as clients and 75 as impostors. (FP=False Positive.FN=False Negative.ER=Error Rate.THRES=Threshold).

w	v	FP (%)	FN (%)	ER (%)	THRES
11	8	7.626667	1.333333	4.480000	0.083901
15	8	2.320000	1.333333	1.826667	0.120001
19	8	0.373333	0.000000	0.186667	0.184003
11	10	1.902222	9.333333	5.617778	0.111901
15	10	2.844444	0.000000	1.422222	0.112001
19	10	1.244444	0.000000	0.622222	0.144002
11	12	4.017778	5.333333	4.675556	0.092001
15	12	1.751111	2.666667	2.208889	0.120001
19	12	0.951111	0.000000	0.475556	0.144002

References

1. Delac, K., Grgic, M.: A survey of biometric recognition methods. In: 46th International Symposium Electronics in Marine, ELMAR-2004, Zadar, Croatia (2004)
2. Bolle, R., Pankanti, S.: Biometrics, Personal Identification in Networked Society. Kluwer Academic Publishers (1998)
3. Zhang, D., Kong, W.K., You, J., Wong, M.: Online palmprint identification. IEEE Transaction on Pattern Analysis and Machine Learning **25** (2003) 1041–1050
4. García-Hernández, J., Paredes, R.: Biometric identification using palmprint local features. In: 3nd COST 275 Workshop. Biometrics on the Internet Fundamentals, Advances and Applications. (2005) 11–14
5. Paredes, R., Vidal, E.: Learning prototypes and distances (lpd). a prototype reduction technique based on nearest neighbor error minimization. In: In ICPR 2004. (2004) 442–445
6. Paredes, R., Pérez, J.C., Juan, A., Vidal, E.: Face Recognition using Local Representations and a direct Voting Scheme. In: Proc. of the IX Spanish Symposium on Pattern Recognition and Image Analysis. Volume I., Benicàssim (Spain) (2001) 249–254
7. Paredes, R., Pérez, J.C., Juan, A., Vidal, E.: Local Representations and a direct Voting Scheme for Face Recognition. In: Proc. of PRIS 01. (2001)
8. Keysers, D., Paredes, R., Ney, H., Vidal, E.: Combination of tangent vectors and local representations for handwritten digit recognition. In: In SPR 2002. (2002)
9. Paredes, R., Keysers, D., Lehmann, T., Wein, B.B., Ney, H., Vidal, E.: Classification of medical images using local representations. In: BVM 2002, Bildverarbeitung für die Medizin 2002. (2002)
10. Li, S.Z., Jain, A.K., eds.: Handbook of Face Recognition. Springer (2004)
11. : Polyu palmprint database. (http://www.comp.polyu.edu.hk/∼biometrics/)