

# Palmprint Recognition using Principle Component Analysis Implemented on TMS320C6713 DSP Processor

Thulfiqar Hussein Mandeel<sup>1</sup>, Muhammad Imran Ahmad<sup>1</sup>, Mohd Nazrin Md Isa<sup>2</sup>, Ruzelita Ngadiran<sup>1</sup>

<sup>1</sup>*School of Computer and Communication Engineering*

<sup>2</sup>*School of Microelectronic Engineering,  
Universiti Malaysia Perlis, 02600 Arau, Perlis, Malaysia.  
m.imran@unimap.edu.my*

**Abstract**—This paper presents a human identification system using eigen-palm images. The proposed method consists of three main stages. The preprocessing stage computes the palmprint images to capture important information and produce a better representation of palmprint image data. The second stage extracts significant features from palmprint images and reduces the dimension of the palmprint image data by applying the principal component analysis (PCA) technique. Low-dimensional features in the feature space are assumed to be Gaussian. Thus, the Euclidean distance classifier can be used in the matching process to compare test image with the template. The proposed method is tested using a benchmark PolyU dataset. Experimental results show that the best achieved recognition rate is 97.5% when the palmprint image is represented using 34 PCA coefficients. Moreover, the Euclidean distance classifier is implemented on a digital signal processor (DSP) board. Implementing the proposed algorithm using the DSP processor achieves better performance in computation time compared with a personal computer-based system.

**Index Terms**—Biometrics; Palmprint Recognition; Principal Component Analysis; Feature Extraction.

## I. INTRODUCTION

Humans have used individual's characteristics, such as fingerprint, palmprint, gait and face to identify one another. A Biometric system is a modern and powerful tool used to recognize a person by using biological and behavioral characteristics [1]. In a biometric framework, verification is a process that validates a claim user and identification is a process that determines identity from the given biometric traits. Traditional methods based on tokens, passwords, and user identifiers have several limitations, such as being difficult to remember and easy to forge. Biometric technology has also been applied in criminal investigation, automatic access control, and security system. Various biometric traits have been proposed, such as face, fingerprint, palmprint, and iris, and all of these traits have their own advantages [3]. Several biometric processing techniques have been proposed to process behavioral or physiological characteristics. Human palms are similar to fingerprints because they contain valley and ridge patterns. Palmprints are considered to be more distinctive than fingerprints because the area of the palm is wider than that of a finger. The palm contains additional

features, such as wrinkles and principal lines, which can be used when comparing one palm with another [2].

## II. PROPOSED ALGORITHM

The proposed method includes preprocessing techniques, feature extraction based on linear projection technique, and classification process. Figure 1 shows the main part of the proposed algorithm. Each part is detailed subsequently in the order of implementation. The dimension reduction and matching processes are implemented on a computer and TMS320C6713 DSP board. In the proposed method, a training phase is implemented offline and the identification process is implemented in real time using a DSP processor. The proposed algorithm is tested using the benchmark PolyU palmprint dataset developed by Hong Kong Polytechnic University. The averaged palmprint representation method is used to represent palmprint data features and highlight the important features in the palmprint data. Raw palmprint images are high-dimensional data that consist of noise and redundant information. Thus, linear projection method such as PCA is a suitable tool to analyze and project data to lower-dimension feature space. After raw data are projected into a low-dimensional feature space, most of the information is preserved for the classification process.

### A. Image Preprocessing

The purpose of pre-processing is to enhance some image features that are important for further processing and minimize the overhead by cropping the region of interest (ROI) [3]. The cropped ROI would later be used for feature extraction.

#### i. Image Cropping

This approach is used in the proposed algorithm to extract the ROI of the palmprint image. Cropping is performed based on the principle of selecting a part of a palmprint that has rich texture patterns. This approach removes the unwanted region that is not beneficial for the recognition process. This region might correspond to fingers and the border of the palmprint image.

In the pre-processing stage of this project, we find the central location of "busyness" and "edginess" of a palmprint image and create a suitable ROI, which includes this location.

Our aim is to implement a method that obtains a rectangular or square ROI with rich texture patterns. Figure 2 shows an example of the cropped ROI from a palmprint image.

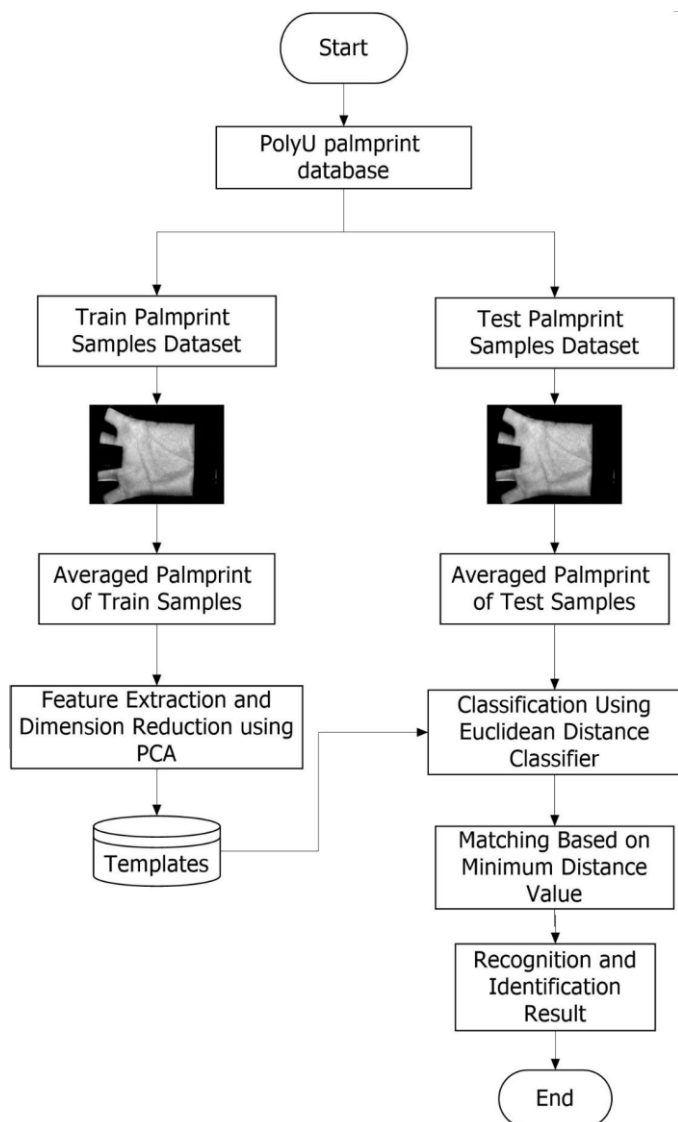


Figure 1: Overview of the Proposed Algorithm.

This concept cannot be directly applied to the set of tests and training palmprint images because the obtained ROIs are of unequal size and the boundaries of the strips on the right, left, bottom, and top are different. We use the intersection of these ROIs to ensure uniformity of the ROI size that is similarly located within all palms. However, this method provides a relatively small area and eliminates some important features.

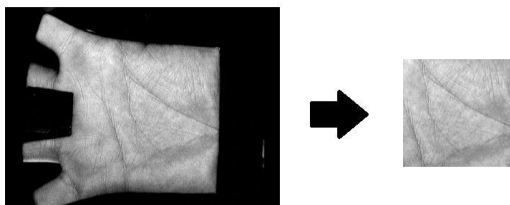


Figure 2: Cropping of palmprint image.

ii. Image resize

Given limited computing resources (processing speed, available memory and storage), we need to resize images to cope with these limitations. Smaller images consume less space and they are fast to process. Resizing the image to a smaller size than the original size can be performed in a scale ranging from 0.1 to 0.9.

iii. Image conversion from 2D to 1D

Image conversion is conducted to prepare the training and testing palmprint images to compute the principal components of the feature information. In this approach, each image is converted from a 2D matrix to a single vector with high dimension by creating a chain from rows of pixels in the original image done by concatenating the rows of pixels. For example, a 100 x 100 pixel image produces 10,000 dimensions of a single vector. Single vectors from each image are stacked to form a 2D matrix ( $T$ ), and each row of the matrix represents a single image for each person. We assume that we have nine classes and each class has  $n$  dimension vectors. A matrix to represent all images is as follows:

$$T = \begin{bmatrix} A_{1,1} & \dots & A_{1,n} \\ \vdots & \ddots & \vdots \\ A_{9,1} & \dots & A_{9,n} \end{bmatrix}$$

B. Feature Extraction and Dimension Reduction

Feature extraction refers to the gathering of specific features from the raw data to minimize the data size by removing noise and redundant features. Feature extraction is important in the classification process because of its high discrimination power in a feature space. The number of extracted features is limited as much as possible with regard to the classification accuracy and measurement cost. Small feature sets consume less memory and have low computational time, which is important for real-time application [4].

i. Principal Component Analysis (PCA)

Principal component analysis (PCA) is one of the statistical algorithms that convert a group of observations of possibly correlated variables into a group of values of linearly uncorrelated variables called principal components by employing orthogonal transformation. PCA projects the entire dataset onto a different feature subspace, where the desired goal is to minimize the dimensions of a  $n$ -dimensional dataset by projecting it onto a  $m$ -dimensional subspace (where  $m \ll n$ ) to increase computational efficiency while retaining most of the information [5].

The eigenvectors and eigenvalues of a covariance matrix represent the important parameters in PCA. The eigenvectors (principal components) determine the directions of the new feature space, and the eigenvalues determine their magnitude. In other words, the eigenvalues explain the variance of the data along the new feature axes. [6].

Eigenvectors are the most principal components derived from a dataset and a selection of these is used in a projection matrix. Each of those eigenvectors is associated with an eigenvalue, which can be interpreted as the length or

magnitude of the corresponding eigenvector. Some eigenvalues that have a significantly larger magnitude than others indicate that the reduction of the dataset via PCA onto a smaller dimensional subspace is conducted by using only those eigenpairs (eigenvalue and eigenvectors) and dropping the less informative eigenpairs.

ii. *Computation of Principal Components*

The PCA technique reduces the dimensionality of palmprint data through the use of linear projection based on the Karhunen–Loeve approach. This method maximizes the scatter of all projected palmprint image samples. Dimensional feature reduction increases discrimination power and reduces the computation time and memory cost. The linear projection method is utilized in the PCA framework to project high-dimensional data onto a low-dimensional subspace.

To explain the computation of principal components, consider a set of  $N$  sample palmprint images  $\{\gamma_1, \gamma_2, \dots, \gamma_n\}$  gathered in an  $n$ -dimensional image space and assume that each palmprint image belongs to a different class of  $c$  number of classes  $\{X_1, X_2, \dots, X_c\}$ . Assume linear transformation that transforms the original high-dimension palmprint image space with  $n$ -dimensions into a new low-dimension feature space with  $m$ -dimensions, where  $m \ll n$ . The generated feature vectors,  $y_k = R^m$ , can be represented by Equation (1), which is the linear transformation equation.

$$y_k = W^T \gamma_k \quad (1)$$

where  $k = 1, 2, \dots, N$ .

The entire scatter matrix,  $S_T$ , is represented by Equation (2).

$$S_T = \sum_{k=1}^N (\gamma_k - \mu)(\gamma_k - \mu)^T \quad (2)$$

where  $N$  is the number of all sample palmprint images and  $\mu \in R^m$  is the mean vector of all palmprint image samples computed in the training process. By performing linear transformation with  $W^T$ , the output of  $y_k$  is  $W^T S_T W$ . To maximize the determinant of the scatter matrix obtained from the projected palmprint image, projection  $W_{opt}$  is selected to maximize the determinant of the total scatter matrix of the projected samples, that is:

$$W_{opt} = \arg \max_{\omega} |W^T S_T W| = [\omega_1 \ \omega_2 \ \dots \ \omega_m] \quad (3)$$

where  $\{\omega_1, \omega_2 \dots \omega_m\}$  represent the eigenvectors with  $n$ -dimensions obtained from  $S_T$  that are selected based on the corresponding largest eigenvalues. The generated eigenvectors are called eigen palmprint images because they have the same dimension as the original palmprint images. After gathering the eigen palmprint images, the classifier algorithm can be implemented in a reduced feature space for palmprint recognition with minimal computation cost and time.

iii. *Linear Projection of a Raw Image to a Low-Dimensional Feature Space*

The projection framework from a raw image to a low-dimensional feature space consists of several stages, such as image normalization, covariance matrix computation, determination of eigenvectors and eigenvalues, ranking of the principal components, and projection. These stages are explained in the following sections.

iv. *Image Normalization*

Image normalization is an important part of every biometric recognition system and is widely applied to improve the quality of a raw image. The purpose of normalizing the individual components of the extracted feature vectors is to obtain (normalized) vectors that are more convenient to use in the classification process. Image normalization is conducted by calculating standard scores (distribution mean) of the feature matrix. The mean vector is calculated by adding all values in each column of the feature matrix and dividing it by the number of image vectors, as in Equation (4).

$$\bar{X} = \frac{\sum_{i=1}^n x_i}{n} \quad (4)$$

where  $\bar{X}$  is the mean value,  $n$  is the number of image vectors, and  $x_i$  are the values in each column of the feature matrix.

After calculating the mean of the training data matrix, the zero-mean training data matrix deviation is calculated by subtracting the mean row vector of the training data matrix from each image vector (row) in the training data matrix. The 2D zero-mean training data matrix is created by applying Equation (5).

$$Z(m) = T(m) - \bar{X} \quad (5)$$

where  $Z$  is the 2D zero-mean training data matrix,  $(\bar{X})$  is the mean row vector of the training data matrix,  $T$  is the averaged palmprint image vector of the training palmprint of a specific subject, and  $(m)$  is the vector number.

v. *Eigenvalues and Eigenvectors from the Covariance Matrix*

The covariance matrix is a matrix whose element in the  $n, m$  position is the covariance between the  $n$ th and  $m$ th elements of a normalized image vector. The covariance matrix generalizes the concept of variance to multiple dimensions. The equation of covariance matrix is:

$$S = \frac{1}{n} \sum_{i=1}^n (x_i)(x_i)^T \quad (6)$$

where  $S$  is the 2D covariance matrix,  $x$  is the 2D zero-mean training data matrix, and  $x^T$  is the transpose matrix of the 2D zero-mean training data matrix.

The covariance matrix is symmetric. Eigenvalues and eigenvectors are calculated from the covariance matrix to

decrease the computations required by project data to lower dimensional feature space. The eigenvectors and eigenvalues can be calculated by Equation (7) as follows:

$$S \cdot e = \lambda \cdot e \quad (7)$$

$$|\lambda \cdot I - S| = 0 \quad (8)$$

where  $S$  is the 2D covariance matrix,  $e$  is the 2D eigenvector matrix, and  $\lambda$  is the eigenvalue diagonal matrix.

vi. Selection of Principal Components

To decide which eigenvectors can be removed without losing too much information in our lower-dimensional subspace, we have to inspect the corresponding eigenvalues. The eigenvectors with the lowest eigenvalues carry the least information about the distribution of the data and can therefore be dropped. The common approach to choose the top  $k$  eigenvectors is to rank the eigenvalues from highest to lowest [7].

$$e_{projected} \leq e \quad (9)$$

where  $e_{projected}$  is the eigenvector matrix after projection, and  $e$  is the original eigenvector matrix.

vii. Projection Matrix

The feature vector matrix of the training data is calculated by multiplying the zero-mean training data matrix with the eigenvector matrix after projection. The resulting feature vector matrix is finally used for classification and is defined by Equation (10).

$$F = Z * e_{projected} \quad (10)$$

where  $F$  is the feature vector matrix of the training data,  $Z$  is the zero-mean training data matrix, and  $(e_{projected})$  is the eigenvector matrix after projection.

C. Matching Process

During the matching process, the test template is passed through a matching module, such as a Euclidean distance classifier, to calculate the distance between features in the template and recognition according to the smallest distances.

In a Cartesian coordinate system, we assume that  $q = (q_1, q_2, \dots, q_n)$  and  $p = (p_1, p_2, \dots, p_n)$  are two points in the Euclidean  $n$ -space. Then, by using the Pythagorean formula, we can find the Euclidean distance ( $EU$ ) from  $p$  to  $q$ , or from  $q$  to  $p$ , as given by:

$$EU(p, q) = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad (11)$$

The Euclidean distance classifier is used to measure the distance between feature vectors for the training and test images to determine the closeness between these vectors. The smallest distance indicates that the two feature vectors of the training and test data belong to the same person.

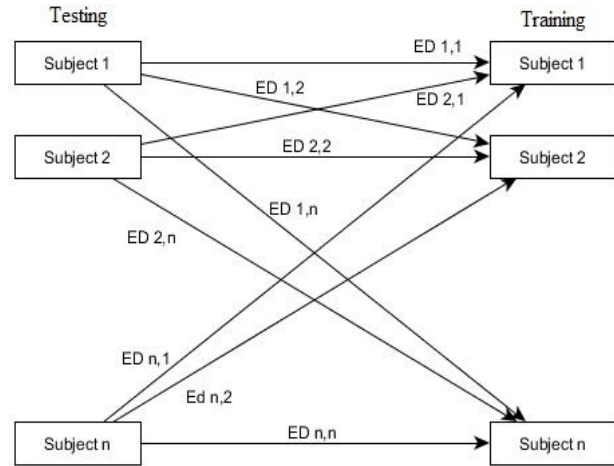


Figure 3: Implementation of Euclidean Distance between Subjects.

Figure 3 shows the implementation of the Euclidean distance between subjects to calculate the distance between the test palmprint image and all the train palmprint images in the database to find the user identity.

III. POLYU MULTISPECTRAL PALMPRINT DATASET

The PolyU palmprint dataset includes 7,752 images from 386 different users. Users supply either their left or right hand, but not both. Normally, each user has 20 samples, which are gathered in two sessions. The PolyU palmprint dataset is widely used in research on biometric recognition based on palmprints [8][9][10][11][12].

IV. TMS320C6713 DSP STARTER KIT

The TMS320C6713 floating-point DSP is the heart of the board. The architecture of the C6x DSP is well suited for numerically intensive calculations. Based on VLIW architecture, the speed of the TMS320C6713 DSP is 225 MHz with 4 K-byte L1P program cache, 4 K-byte L1D data cache, and 256 K-byte L2 (outside the DSP chip) memory [13].

The vectors of the normalized training and test images, as well as the selected PCA coefficients obtained through an offline process in a personal computer, are transferred to the TMS320C6713 DSK board by loading them into a memory as an array of one-dimensional vectors. Linear projection and classification are then performed on the TMS320C6713 DSK board. Offline training is performed using MATLAB software running on the computer. Based on the parameter obtained in the training process, the classification process is then implemented in real time on the TMS320C6713 DSK board. The output of the classification process is shown in the CCS IDE, as presented in Figure 4. The normalized training and test images that have been transferred include 8 palmprint

images for training and 1 palmprint image for a test of 40 subjects. Thirty-four PCA coefficients are used in the projection conducted in the design environment of the TMS320C6713 DSK board.

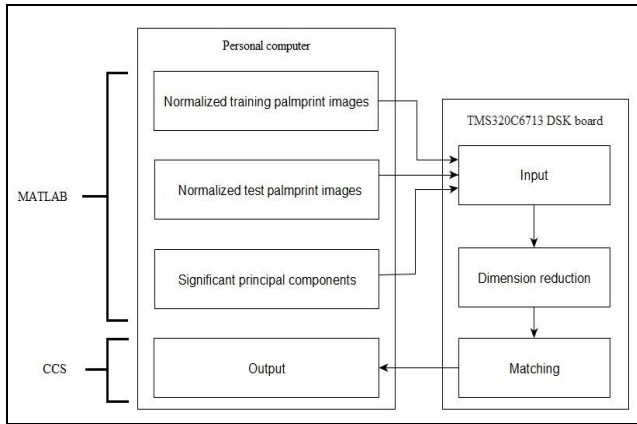


Figure 4: Overview of Proposed System.

### V. RESULT AND DISCUSSION

In this section, the performance of the proposed palmprint recognition using an eigen-palmprint image is validated using the benchmark PolyU palmprint dataset. This dataset is widely used by other researchers to measure recognition performance in previously published studies. A considerable number of analyses are conducted to examine the parameter in the algorithm that is able to achieve the most accurate result. Although tradeoffs among accuracy, speed, and storage exist, we believe that accuracy is one of the most important factors to be considered. Thus, most of the analyses in this section focus on accuracy instead of other factors. Discussions on the issues faced by the system and the overall results are included.

PolyU palmprint dataset comprises images from 386 classes. In this study, we randomly choose 40 subjects to reduce memory requirement and computational time. For each subject, nine palmprint images are used for training and testing. All possible combinations formed by matching the palmprint images of 40 subjects are used to obtain the subject identity.

Several parameters that may affect recognition accuracy have been examined. These parameters may be divided into three groups: effect of image size, effect of number of training images, and effect of number of PCA coefficients.

#### A. Image Preprocessing

In this phase, the palmprint images are preprocessed by reducing the image size to several scales. Image resizing can significantly affect computational time and memory requirement. Using a very small image results in loss of significant information, whereas using a very large image increases computational time and storage requirement, especially when operated on DSP board. Thus, the best resize scale must be identified to facilitate implementation.

At this stage, the cropped ROI from a palmprint image is resized at different scales, and the scale that offers the best identification rate is determined. The image resizing

percentage must be  $\leq 40\%$  of the original size to fit the available memory and achieve superior performance.

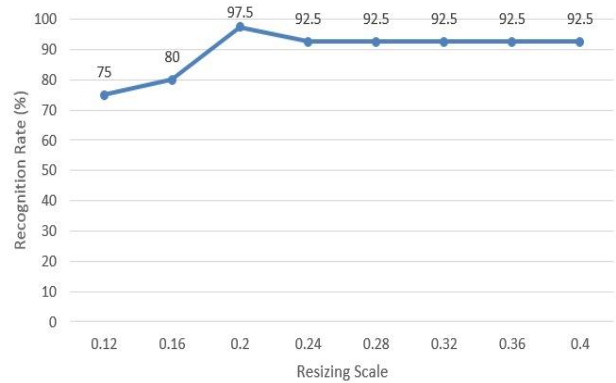


Figure 5: Recognition Rates at Different Resizing Scales.

Figure 5 shows the recognition rates at different resizing scales. According to the figure, the 0.2 resizing scale provides the best recognition rate (97.5%), has the smallest size ( $31 \times 31$ ), and requires the least time for identification, as shown in Figure 6. Using a resizing scale smaller than 0.2 deteriorates performance because many details are lost when the image size is too small. Using a resizing scale larger than 0.2 also deteriorates performance. Thus, 0.2 is the best scaling size to reduce memory usage.

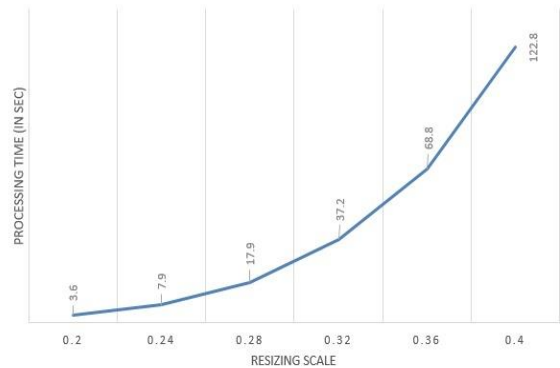


Figure 6: Processing Time in Seconds at Different Resizing Scales.

#### B. Effect of Number of Principal Components

PCA utilizes an orthogonal transformation to convert a group of observations of possibly correlated variables into a group of values of linearly uncorrelated variables called principal components. The principal components with high eigenvalues are selected to represent the palmprint images. These principal components have major variances in the feature space. Considering measurement cost and classification accuracy, the number of principal components used should be kept as small as possible. The best recognition rate of 97.5% is achieved by using 34 PCA coefficients, as shown in Figure 7. Using few PCA coefficients decreases performance, whereas using numerous PCA coefficients reduces discrimination power. In this analysis, 34 PCA coefficients result in maximum discrimination power in a feature space.

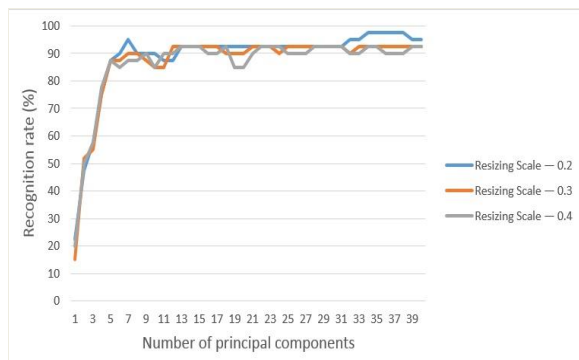


Figure 7: Recognition Rates with Different Numbers of Principal Components and Resizing Scales.

### C. Comparison of Running the Proposed Algorithm on Both Personal Computer and TMS320C6713 DSP Processor

The identification and projection conducted on both personal computer and TMS320C6713 DSK board generate different computational times at the same recognition rate. On the one hand, the TMS320C6713 floating-point DSP is based on VLIW architecture with 225 MHz of speed, 4 K-byte L1P program cache, 4 K-byte L1D data cache, and 256 K-byte L2 memory, and is suitable for DSP algorithms[13]. On the other hand, the general-purpose personal computer has an Intel CPU (1.80 GHz) 2 Core(s) and 4 GB of physical memory (RAM). The analysis is performed offline and online, and the computational time results are compared. Some of the processing tasks, such as preprocessing, image normalization, feature extraction, dimension reduction, and matching, are conducted offline. Real-time processing includes projection and classification processes. Table 1 shows the performance of both systems in terms of identification rate and consumed time.

Table 1  
Performance of Proposed Algorithm on Different Platforms.

Platform	Identification Rate	Processing Time (in Sec)
PC	97.50%	3.6
DSP board	97.50%	1.7

## VI. CONCLUSION

The proposed algorithm implements PCA, which is considered as one of the most successful methods to extract feature vectors in subspace-based approaches. In the proposed algorithm, the image is projected to a different feature space to reduce the dimensions of the dataset by using PCA. The Euclidean distance algorithm is implemented to find the distances between test palmprints and training palmprints, and the user identity is obtained by finding the smallest Euclidean distance. The efficiency of the proposed algorithm for palmprint recognition is tested using the PolyU palmprint dataset. Experimental results show that the algorithm

demonstrates convincing high-recognition performance when tested using the PolyU palmprint dataset, with 40 palmprints for training and 40 palmprints for test, ROI (151 pixels  $\times$  151 pixels), and 34 eigenvectors. The best performance of the proposed algorithm is 97.5% recognition rate. The proposed algorithm shows superior recognition performance when using the Euclidean distance classifier with a lower number (i.e., 34) of eigenvectors, and 20% resizing from the original palmprint image size. The average processing time is an important factor especially in real-time scenarios. The processing times for the projection and Euclidean distance classification using the TMS320C6713 DSK board exhibit better speed compared with using the PC for processing the same functions, the speed of which can be improved to a ratio of 0.5.

## ACKNOWLEDGMENT

This work is supported by the Ministry of Education Malaysia, under FRGS Grant (Grants No: 9003-00383).

## REFERENCES

- [1] Jain, A. K., Ross, A., Prabhakar, S. (2004). An Introduction To Biometric Recognition. *Ieee Transactions On Circuits And Systems For Video Technology*, 14(1), Pp. 4-20.
- [2] Zhang, D., Kong, W. K., You, J., & Wong, M. (2003). Online Palmprint Identification. *Ieee Transactions On Pattern Analysis And Machine Intelligence*, 25(9), Pp. 1041-1050.
- [3] Duta, N., Jain, A. K., & Mardia, K. V. (2002). Matching Of Palmprints. *Pattern Recognition Letters*, 23(4), Pp. 477-485.
- [4] Liu, L., & Zhang, D. (2005, September). Palm-Line Detection. In *Image Processing, 2005. Icip 2005. Ieee International Conference On* Vol. 3, Pp. 269-272.
- [5] Wang, L., Jin, W., Liu, Z., He, Y., & Li, G. (2012). A Method Of Palmprint Recognition Integrated By 2d-Pca And Sparse Representation. *Opto-Electronic Engineering*.
- [6] Jolliffe, I. (2002). *Principal Component Analysis*. John Wiley & Sons, Ltd.
- [7] Mu, M., Ruan, Q., & Shen, Y. (2010, February). Palmprint Recognition Based On Discriminative Local Binary Patterns Statistic Feature. *International Conference On Signal Acquisition And Processing, 2010. Icsap'10*. Pp. 193-197
- [8] Iitsuka, S., Miyazawa, K., & Aoki, T. (2009, November). A Palmprint Recognition Algorithm Using Principal Component Analysis Of Phase Information. *16th Ieee International Conference On Image Processing (Icip)*, Pp. 1973-1976.
- [9] Guo, Z., Wu, G., Chen, Q., & Liu, W. (2011, November). Palmprint Recognition By A Two-Phase Test Sample Sparse Representation. *International Conference On Hand-Based Biometrics (Icbh), 2011*, Pp. 1-4.
- [10] Ghandehari, A., Anvaripour, M., & Soltanpour, S. (2012, March). Palmprint Verification And Identification Using Pyramidal Hog Feature And Fast Tree Based Matching. *5th Iapri International Conference On Biometrics (Icb), 2012*, Pp. 421-426.
- [11] Li, W., Zhang, B., Zhang, L., & Yan, J. (2012). Principal Line-Based Alignment Refinement For Palmprint Recognition. *Ieee Transactions On Systems, Man, And Cybernetics, Part C: Applications And Reviews*, , 42(6), Pp. 1491-1499.
- [12] Badrinath, G. S., & Gupta, P. (2009). Palmprint Based Verification System Robust To Rotation, Scale And Occlusion. *12th International Conference On Computers And Information Technology, 2009. (Iccit'09)*. Pp. 408-413.
- [13] Chassaing, R. (2004). *Digital Signal Processing And Applications With The C6713 And C6416 Dsk* (Vol. 16). John Wiley & Sons.