

A New Extension of Kernel Principal Component Analysis for Finger Vein Authentication

Sepehr Damavandinejadmonfared and Vijay Varadharajan

Advanced Cyber Security Research Centre
Dept. of Computing, Macquarie University
Sydney, Australia

Sepehr.damavandinejadmonfared@mq.edu.au - Vijay.varadharajan@mq.edu.au

Abstract

In this paper, we introduce a new method of data transformation for finger vein recognition system. Our proposed method uses kernel mapping functions to map the data before performing Principal Component Analysis. Kernel Principal Component Analysis (KPCA) is a well-known extension of PCA which is suitable for finding nonlinear patterns as it maps the data nonlinearly. In this work we develop an extension of KPCA which is both faster and more appropriate than KPCA for finger vein recognition system. The proposed method is called Feature Dependent Kernel Principal Component Analysis (FDKPCA). In FDKPCA the data is mapped differently from KPCA resulting in lower-dimension feature space where more important and valuable features are selected and extracted. Furthermore, extensive experiments reveal the significance of the proposed method for finger vein recognition systems.

Keywords: Finger vein Recognition, Kernel PCA, and Spectral data transformation.

1 Introduction

Data transformation has been a wide area for researchers as several challenges can be addressed by transferring data into another space where finding genuine patterns and features is desired. There is an extremely large amount of literature on data transformation algorithms and methods. Principal Component Analysis (PCA)(Abdi, Hervé 2010) is one of the well-known methods for dimensionality reduction and feature extraction. PCA(Beng & Rosdi 2011) is a fast method having several usages in multiple areas and application especially in pattern recognition. However, PCA is a linear method which may be inefficient when dealing with nonlinear patterns and data. To address the mentioned drawback of PCA, kernel PCA(Kim et al. 2002) was developed which is known as a very well-known and influential extension on PCA. In KPCA, PCA is performed in a kernel feature space which is nonlinearly related to the input data. More specifically, the whole input data space is mapped into another space (kernel space) having higher dimension than the input data dimension. It is enabled using a positive semi-definite (psd) kernel function computing the inner

products within the new space (kernel feature space). Therefore, constructing the so-called kernel matrix or the inner product matrix is vital. Then, using the top eigenvalues and their corresponding eigenvectors will lead to kernel PCA data transformation method. Kernel PCA has widespread use in many different areas namely, in machine learning algorithms, data classification, and data de-noising. Such methods have been used in biometrics systems such as face and finger vein recognition. In 2010, R. Jenssen proposed Kernel Entropy Component Analysis(Jenssen 2010) KECA as a new extension on kernel PCA. In 2012(Shekar et al. 2011), KECA was proposed in face recognition system. It is believed that kernel PCA and kernel ECA(Hu & Yang 2010) are more superior methods than PCA as the previous research shows these methods reach more accuracy rate and reliability in terms of data classification and image processing. Considering the way PCA, and KPCA are implemented on images for the purpose of classification and identification, where there are some samples available from each individual to train the system and the remaining samples to test, we propose FDKPCA to improve the performance of the mentioned algorithms. The mentioned methods have been proposed in both face recognition and finger vein recognition systems (Damavandinejadmonfared 2012).

In this paper, we develop a new spectral data transformation method, which can be more stable and faster than KPCA as in FDKPCA the dimension of the feature space is dependent on the dimension of the input data, not the number of input data. It means no matter how many data to analyse, the dimension of kernel matrix (kernel feature space) is fixed. One promising biometric(Delac et al. 2005) is finger vein authentication which has been given considerable attention recently. We have conducted experiments on finger vein(Wu & Liu 2011) database to be able to compare the outcome of the proposed method with KPCA. Experimental results show that not only the proposed method outperforms KPCA in finger vein system, but also it is more time efficient.

The reminder of this paper is organized as follows:

Section 2 illustrates some examples of spectral data transformation methods of importance. In section 3, Feature Dependent Kernel Principal Component Analysis (FDKPCA) is introduced. In section 4, Image acquisition and ROI extraction algorithms are explained. A finger vein recognition algorithm is proposed in section 5. Experimental results are presented in section 6. Finally, section 7 concludes the paper.

2 Spectral Data Transformation

In this section, we explain the fundamentals of PCA, and KPCA with examples to comprehend spectral basic data transformation methods.

2.1 Principal Component Analysis (PCA)

A well-known spectral data transformation method is PCA. Let $X = [x_1, \dots, x_N]$, where $x_t \in \mathbb{R}^d$ and $t = [1, \dots, N]$. As PCA is a linear method, the following transformation is sought assuming A is $[d \times d]$ such that $y_t \in \mathbb{R}^d$ and $t = [1, \dots, N]$:

$Y_{pca} = AX$ where $Y_{pca} = [Y_1, \dots, Y_N]$. Therefore, the sample correlation matrix of Y_{pca} equals to:

$$\frac{1}{N} Y_{pca} Y_{pca}^T = \frac{1}{N} AX (AX)^T = A \frac{1}{N} XX^T A^T \quad (1)$$

The sample correlation matrix of X is $\frac{1}{N} XX^T$. Determining A such that $\frac{1}{N} Y_{pca} Y_{pca}^T = I$ is the goal. Considering eigen-decomposition, we will have $\frac{1}{N} XX^T = V \Delta V^T$, where Δ is a diagonal matrix of the eigenvalues $\delta_1, \dots, \delta_d$ in descending order having the corresponding eigenvectors v_1, \dots, v_d as the columns of V . Substituting into (1), it can be clearly observed that $A = \Delta^{-\frac{1}{2}} V^T$ leads to the goal such that $Y_{pca} = \Delta^{-\frac{1}{2}} V^T X$.

Performing a dimensionality reduction from d to $l \leq d$ is often achieved by the projection of data onto a subspace spanned by the eigenvectors (principal axes) corresponding to the largest top l eigenvalues. Hence, it is also well-known that l -dimensional Y_{pca_l} preserves the maximum amount of second order statistics in the dimensionality reduced data in comparison with the original d -dimensional data.

2.2 Kernel Principal Component Analysis (KPCA)

Kernel PCA is a non-linear version of PCA operating in a new feature space called *kernel feature space*. This space is non-linearly related to the input space. The nonlinear mapping function (kernel function) is given $\Phi: \mathcal{R}^d \rightarrow F$ such that $x_t = \Phi(x_t), t = 1, \dots, N$ and $\Phi = [\phi(x_1), \dots, \phi(x_N)]$. After performing such mapping in input data, PCA if implemented in F , we need an expression for the projection of $P_{u_i} \Phi$ of Φ onto a subspace of feature space principal axes, for example, top l principals. It can be given by a positive semi-definite kernel function or Mercer kernel, $k_\sigma: \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ computes an inner product in the Hilbert space F :

$$k_\sigma(x_t, x_{t'}) = \langle \phi(x_t), \phi(x_{t'}) \rangle. \quad (2)$$

The $(N \times N)$ kernel matrix K is defined such that element (t, t') of the kernel matrix equals to $k_\sigma(x_t, x_{t'})$. Therefore, $K = \Phi^T \Phi$ is the inner product matrix (Gram matrix) in F . Then, Eigen-decomposing the kernel matrix we have $K = EDE^T$ where E is the eigenvectors e_1, \dots, e_N column wise and their corresponding eigenvalues are in $D = \lambda_1, \dots, \lambda_N$.

Williams in (C.K.I. Williams 2002) discussed that the equivalence between PCA and KPCA holds in KPCA as well (kernel feature space). Hence, we have:

$$\Phi_{pca} = P_{u_i} \Phi = D_l^{\frac{1}{2}} E_l^T \quad (3)$$

Where D_l is the top large l eigenvalues of K and E_l is their corresponding eigenvectors stored in columns. It means that projecting Φ onto spanned feature space (principal axes) is given by $P_{u_i} \Phi = \sqrt{\lambda_i} e_i^T$.

Considering the analogy in (3), $\Phi_{pca} = D_l^{\frac{1}{2}} E_l^T$ is the solution to the following optimization problem:

$$\Phi_{pca} = D_l^{\frac{1}{2}} E_l^T : \min_{\lambda_i, e_i^T, \dots, \lambda_N, e_N^T} 1^T (K - K_{pca})^2 \cdot 1. \quad (4)$$

Where $K_{pca} = \Phi_{pca}^T \Phi_{pca} = E_l D_l E_l^T$. Therefore, this procedure minimizes the norm of $K - K_{pca}$.

3 Feature Dependent Kernel Principal Component Analysis (FDKPCA)

Generally, in spectral data transformation methods, finding the most valuable principal axes (appropriate directions in the feature space) is of most importance. In PCA, for example, it is extracted linearly from the principal feature space. In KPCA, however, these axes are extracted from kernel feature space as discussed in previous subsection. We define Feature Dependent Kernel PCA as a k -dimensional data transformation method obtained by projecting input data onto a subspace spanned by principal kernel axes contributing to the feature dependent kernel space. Feature dependent kernel space is defined as follows:

Let $X = [x_1, \dots, x_N]$, where $x_t \in \mathbb{R}^d$ and $t = [1, \dots, N]$.

The nonlinear mapping function is given $\Phi: \mathcal{R}^d \rightarrow F^d$ such that $x'_t = \Phi(x'_t), t = 1, \dots, d$ where x'_t is an N dimensional vector including all of the t_{th} features from N input data. Explaining this, we have $\Phi = [\phi(x'_1), \dots, \phi(x'_d)]$. The use of a positive semi-definite

kernel function or Mercer kernel computes an inner product in the new space F^d :

$$k_{\sigma}(x'_i, x'_j) = \langle \phi(x'_i), \phi(x'_j) \rangle. \quad (5)$$

The $(N \times N)$ kernel matrix-we define that as K_{FDKPCA} is now defined such that element (t, t') of the kernel matrix is $k_{\sigma}(x'_t, x'_{t'})$. Therefore, K_{FDKPCA} is the Gram matrix or the inner product matrix in F^d . The next stage in FDKPCA is to perform PCA on K_{FDKPCA} . Note that the kernel matrix given in FDKPCA feature space (K_{FDKPCA}) is totally different from that of KPCA. Firstly, we explain KPCA feature space for the sake of clarity and then, FDKPCA feature space is introduced.

Figure. 1. illustrates a brief flow diagram of reaching kernel feature space from the scratch. As it is shown in Figure. 1, N input data are first mapped into kernel space by ϕ and then the Gram matrix (kernel matrix) is calculated using inner product. Note that the dimension of kernel matrix is equal to the number of input data- N . Eigen-decomposition is the next step where all eigenvalues and their corresponding eigenvectors are extracted and reordered in a descending manner from the greatest to the smallest value. After finding the kernel axes in this space, the kernel matrix, which represents the input data, is projected onto the kernel feature vectors (eigenvectors). The drawback to KPCA is that the dimension of feature space and kernel matrix could become too high and as a result data transformation could be computationally expensive. In addition, finding the most optimized sub-space in kernel feature space could be challenging and sometimes inefficient.

In FDKPCA feature space, the input data is projected onto a subspace spanned by principal kernel axes contributing to the feature dependent kernel space. In FDKPCA all features having the same dimension from all input data are firstly considered in separate vectors, and then mapped into kernel space which is called FDKPCA feature space. Finally, the kernel matrix (Gram matrix) using inner products which is a d -dimensional space is computed. Note that the input data has the dimension of d which means there is no growth of dimension while computing FDKPCA feature space. Having d -dimensional FDKPCA feature space, the eigenvectors and their corresponding eigenvalues are decomposed in this step. The original input data is projected onto a subspace of FDKPCA feature vectors for the purpose of transformation and dimensionality reduction. In the FDKPCA feature space, the non-linear relations between data inputs are extracted in a feature wise manner which results in having both more efficiency and higher speed.

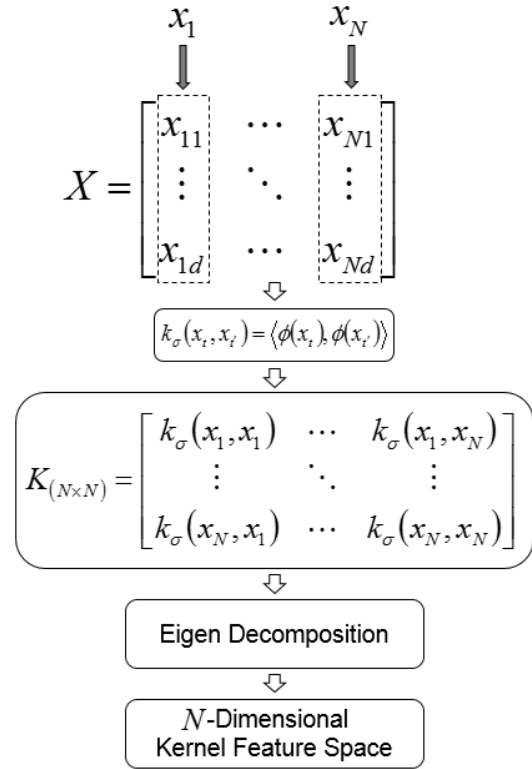


Figure. 1. Flow diagram of reaching Kernel Feature Space

4 Image Acquisition and Region of Interest (ROI) Extraction Algorithm

Based on the proven scientific fact that the light rays can be absorbed by deoxygenated hemoglobin in the vein, absorption coefficient (AC) of the vein is higher than other parts of finger. In order to provide the finger vein images, four low cost prototype devices are needed such as an infrared LED and its control circuit with wavelength 830nm, a camera to capture the images, a micro-computer unit (MCU) to control the LED array, and a computer to process the images. The web-cam has an IR blocking filter; hence, it is not sensitive to the infrared (IR) rays. To solve this problem an IR blocking filter is used to prevent the infrared rays from being blocked.

Three major steps are used to crop images optimally: first one is detecting the edge. In order to perform the cropping part, two horizontal lines are determined by finding the horizontal edges in original images. Two conditions should be satisfied to find the appropriate lines by edge detection algorithm: (1) the pairs of detected points should be located between 35% and 65% of the height of the captured image, and (2) among the detected pairs, the pair of the edge that are the widest will be chosen. Last step is to crop the images from 5% from right border and 5% percent from left border vertically. An example is shown in Figure. 2 a) and b).

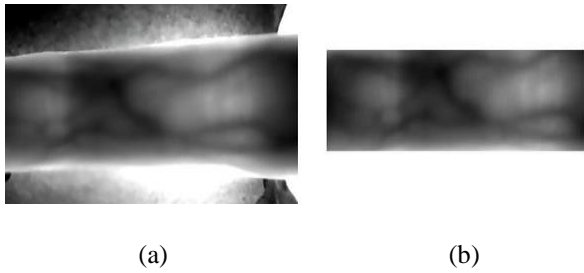


Figure. 2. a) original image, b) cropped image

5 Proposed Finger Vein Recognition Algorithm

The flow diagram of the proposed finger vein recognition algorithm is shown in Figure. 3. First step is to extract the region of interest from the samples which was explained in section 4. After extraction the region of interest, the proposed FDKPCA is conducted on the data to find the optimal axes (eigenvectors) to project the data onto. Actually, based on the dimension of the input data, the number of experiments is assigned. It is because of the nature of PCA based algorithm as there are as many different dimensions as the dimension of the input data. for instance, if the dimension of the input images is 100, there are 100 different implementations on the same data using 100 different feature vectors to reduce the dimension and extract the features. And finally, Nearest Neighbor classifier is taken into account to classify the extracted features and make the final decision. Figure. 3. Indicated the flow diagram of the clustering algorithm.

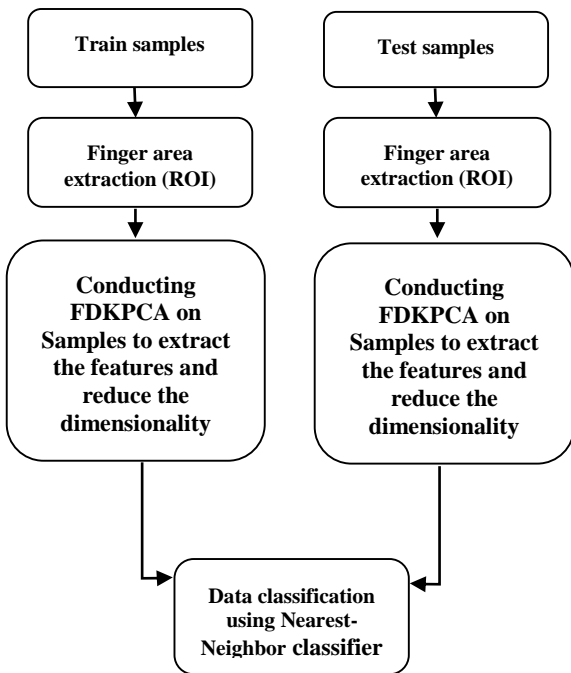


Figure. 3. Flow diagram of the proposed clustering algorithm for finger vein recognition

6 Experimental Results

In this section, the experiments are conducted to corroborate the performance of Feature Dependent Kernel Principal Component Analysis (FDKPCA) over Kernel Principal Component Analysis (KPCA). We used Gaussian kernel function as the mapping function in both KPCA and FDKPCA. Finger vein database used in the experiments consists of 500 images from 50 individuals; 10 samples from each subject were taken. In this experiment 4, 5, and 6 randomly selected samples are used to train and the remaining 6, 5, and 4 samples are used to test respectively. In each experiment, the accuracy is calculated using the first 200 components of the extracted features meaning that each experiment is repeated 200 times using the first 200 features to project the data onto, and also the dimension is reduced from 100% to 0% in different experiments. The results are shown in Figure 4, 5, and 6.

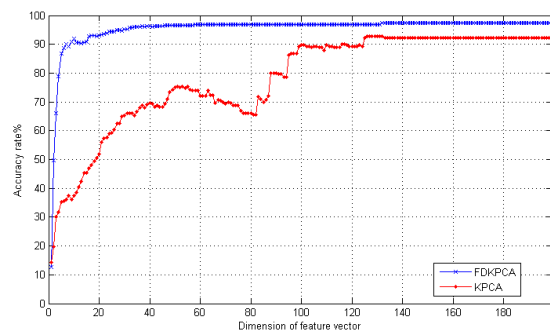


Figure. 4 Comparison of accuracies between KPCA and FDKPCA obtained using 4 images to train and 6 to test

As it was expected, the more the number of training samples gets, the higher accuracy rate goes. It is observed that no matter how many samples to train and test and no matter how high the dimension of the feature vector is, using FDKPCA results in a higher accuracy rate in all experiments and all different feature vectors. The discrepancy between the obtained accuracies is very dramatic in the first half of the graphs.

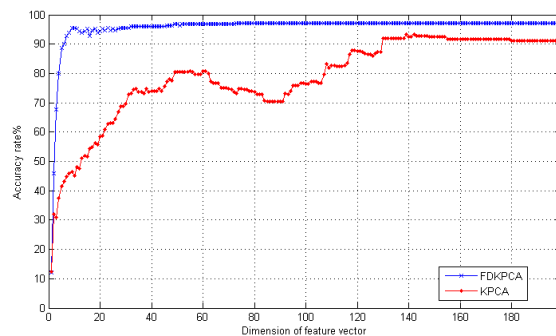


Figure. 5 Comparison of accuracies between KPCA and FDKPCA obtained using 5 images to train and 5 to test

It can be explained by the nature of the FDKPCA which is able to find more valuable feature vectors in the

lower dimension than KPCA. For example, FDKPCA reaches the accuracy of around 98% to 100% using less than 50 vectors to transfer the data while KPCA gets its highest accuracy which is around 90 % to 93% using the feature vectors with the dimension of more than 120.

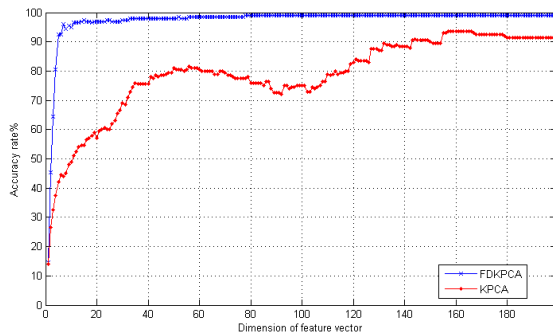


Figure. 6 Comparison of accuracies between KPCA and FDKPCA obtained using 6 images to train and 4 to test

Experimental results reveal that not only is the proposed system (FDKPCA) more superior than KPCA for finger vein recognition, but also it can be considered much faster than KPCA as it reaches its peak in much lower dimension than KPCA.

7 Conclusion

In this research, we proposed a new method of dimensionality reduction and feature extraction (FDKPCA) which is a combination of the well-known Principal Component Analysis (PCA) and Kernel Principal Component Analysis (KPCA), which was proven to be faster and more accurate than PCA and KPCA in terms of finger vein recognition. We also proposed a new finger vein recognition algorithm using the FDKPCA method to extract the most valuable features from the samples and reduce the dimension of the data. In the proposed system, the images are automatically cropped first and then FDKPCA is performed for the purpose of feature extraction. Finally, Nearest Neighbour classifier is conducted to classify the extracted feature and make the final decision. Extensive experiments on our finger vein data reveal the significance of the proposed method in comparison with the traditionally used methods.

8 References

- Abdi, Hervé, and L.J.W., 2010. Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4), pp.433–459.
- Beng, T.S. & Rosdi, B.A., 2011. Finger-vein identification using pattern map and principal component analysis. *2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pp.530–534.
- C.K.I. Williams, 2002. On a Connection between Kernel PCA and Metric Multidimensional Scaling. *Machine Learning*, 46, pp.11–19.

Damavandinejadmonfared, S. et al., 2012. Evaluate and Determine the Most Appropriate Method to Identify Finger Vein. *Procedia Engineering*, 41(Iris), pp.516–521.

Delac, K., Grgic, M. & Grgic, S., 2005. Independent comparative study of PCA, ICA, and LDA on the FERET data set. *International Journal of Imaging Systems and Technology*, 15(5), pp.252–260.

Hu, P. & Yang, A., 2010. Indefinite Kernel Entropy Component Analysis. *Science And Technology*, (3), pp.0–3.

Jenssen, R., 2010. Kernel entropy component analysis. *IEEE transactions on pattern analysis and machine intelligence*, 32(5), pp.847–60.

Kim, K.I., Jung, K. & Kim, H.J., 2002. Face Recognition using Kernel Principal Component Analysis. *Signal Processing*, 9(2), pp.40–42.

Shekar, B.H. et al., 2011. Face recognition using kernel entropy component analysis. *Neurocomputing*, 74(6), pp.1053–1057.

Wu, J.-D. & Liu, C.-T., 2011. Finger-vein pattern identification using principal component analysis and the neural network technique. *Expert Systems with Applications*, 38(5), pp.5423–5427.