Pseudo Principal Components Analysis for Feature Extraction and Pattern Recognition of Time-Series data

Daewon An, and K. Wendy Tang
Electrical and Computer Engineering
Stony Brook University
Stony Brook, NY 11794-2350

Abstract - We proposed a novel method to extract a feature from time-series data by Principal Components Analysis (PCA) with time-delay embedding, and showed its usefulness to the pattern recognition. We first resampled from the original time series data and constructed a new data with time-delay embedding. Then we applied PCA to the new data to get a Pseudo Principal Component (PPC), which now represents the newly constructed data and hence the original time series data as well. The PPC was used as a feature vector for the original data, and the pattern classification of was performed upon PPC. In order to improve the performance of the classification, we incorporated with the Continuous Wavelet Transform (CWT) to the newly constructed data before we take the PPCs. The results showed that the new method is useful to classification tasks of time series data, and that the performance is improved when well combined with the CWT technique.

1. Introduction

Time-series or time-sequence data are a collection of the sequential measurements of some physical system over a certain period of time [4]. In general, the distance measure such as Euclidian, Mahalanobis, etc., are general methods for the classification problem of time-series datasets [13]. The feature vectors used in the distance measure can be any form of vector representing the original dataset, e.g. original data itself, mean, variance, and so on.

As a new feature extraction technique, we propose a novel method, Pseudo PCA, in which we resample from the original time series data and construct a new data with time-delay embedding, and then apply PCA on the new data. In this paper, we first review PCA, and then the proposed Pseudo PCA to extract a feature from the time-series data. Finally, we apply the Pseudo PCA to a practical datasets – the Synthetic Control Chart data [5], [7] and the Japanese Vowel data [14]. We also show that the performance of Pseudo PCA can be improved by incorporating the Continuous Wavelet Transform (CWT) technique.

2. Time-series data

Time-series or time-sequence data are a collection of the sequential measurements of some physical system over a certain period of time, and more examples and study about time-series data can be found in [4]. In general, those data are studied mostly to reduce the data size – data compress, or to predict the future values – function approximation or regression modeling [13], or to find similarities among sets of the data and classify the sets according to the similarities. We are interested in the latter known as the pattern recognition.

Let's consider a N-point sequence $X = \{x[1], x[2], \cdots, x[N]\}$ of an arbitrary variable, $x(t)$ generated from a certain machine or processor over time period $[t_1, t_2]$. The details about how to get these discrete signals from a continuous signal can be found in many textbooks, e.g. [8] and [9]. We may repeat observing the processor M times and get a set of M different sequences. These M different sequences or time series can be thought of M different estimations of the stochastic process, $x[t]$, as shown in Figure 1.

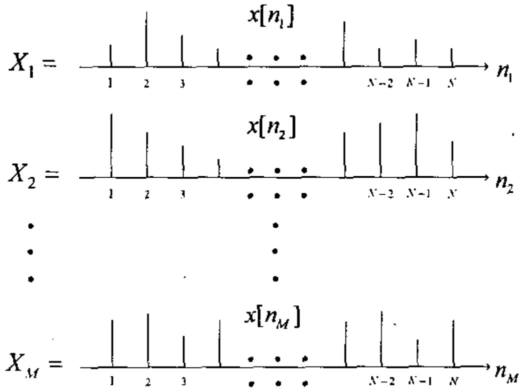

$x[n_1]$
$X_1 =$
1 2 3 • • • $N-2$ $N-1$ N
n_1
$x[n_2]$
$X_2 =$
1 2 3 • • • $N-2$ $N-1$ N
n_2
$x[n_M]$
$X_M =$
1 2 3 • • • $N-2$ $N-1$ N
n_M

Figure 1. Example of time-series data

For our experiment, we generated the Synthetic Control Chart data of length 60 or 60 points per each sequence, according the equations given in [2] and [5], and $M = 100$ sequences were generated as estimations of the 6 different predetermined pattern classes, namely A, B, C, D, E, and F, which can be then considered as 6 different stochastic processes. They are described as normal, cyclic, increasing trend, decreasing trend, upward shift, and downward shift. Each time sequence of length $N = 60$ was generated as follows for $1 \le t \le N$.

A. Normal: $x(t) = m + r_1 s$ where $m = 30$, $s = 2$, and r_1 is a andom number between $[-3,3]$.

B. Cyclic: $x(t) = m + r_1 s + r_2 \sin(\frac{2\pi t}{r_3})$ where r_2 and r_3 are random numbers between $[10,15]$.

C. Increasing: $x(t) = m + r_1 s + r_4 t$ where r_4 is a random number between $[0.2, 0.5]$.

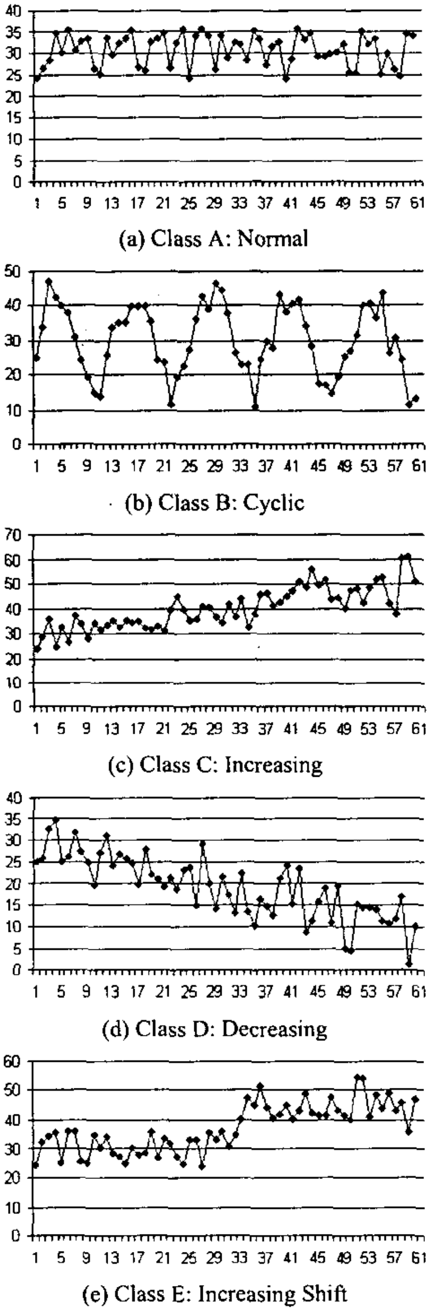
D. Decreasing: $x(t) = m + r_1 s - r_4 t$

0-7803-8639-6/04/$20.00 ©2004 IEEE.

E. Increasing Shift: $x(t) = m + r_1 s + r_5 k$ where r_5 is a random number between $[7.5, 20]$, and $k = 0$ for $1 \le t \le T$ and $k = 1$ for $T \le t \le N$ where T is a random number between $[N/3, 2N/3]$.

F. Decreasing Shift: $x(t) = m + r_1 s - r_5 k$

The characteristics of each class can be more understandable if it is plotted, and Figure 2 shows some examples of the time sequence of those 6 classes.



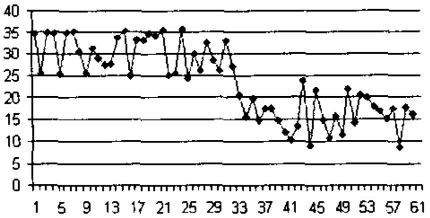
(a) Class A: Normal

(b) Class B: Cyclic

(c) Class C: Increasing

(d) Class D: Decreasing

(e) Class E: Increasing Shift



(f) Class F: Decreasing Shift

Figure 2. The Synthetic Control Chart data of length 60

Also, for the sake of comparison with existing methods, e.g. Hidden Markov Model, we tested our new method with the Japanese Vowel database created by M. Kudo et al. (see [14] for more details). The database has 640 time-series of 12 LPC cepstrum coefficients of the Japanese vowel sound '/ae/' taken from nine male speakers; 270 sequences for training and 370 sequences for testing whose length are from 7 to 29 depending upon the utterances. The first and second degree cepstrum coefficients are shown in Figure 3, and the dark lines are of the ninth speaker and the light lines are of the rest speakers.

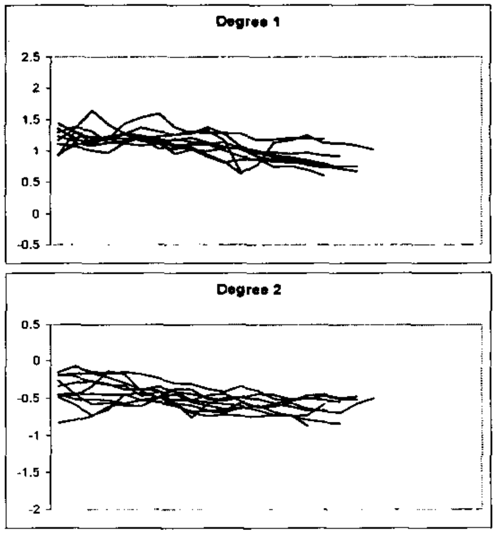


Figure 3. The first and second feature of the '/ae/' sound.

Our goal is to classify those sequences from the same classes into same group for the Synthetic Control Chart data, and to identify the speakers by the sequence of the cepstrum coefficients for the Japanese Vowel database.

3. Similarity and Feature Extraction

We naturally expect that estimations of a same process share some similar features that may discriminate themselves form others. Pattern Recognition (PR) is a scientific discipline that studies methods to classify or recognize those similar features. In general, PR approach uses so-called feature vectors, which one-to-one represent each element of the original dataset. Then, either the distance between vectors or the likelihood of the vectors (under some prior belief or probabilistic assumption) are computed as similarity or dissimilarity between two vectors. Finally, each vector in feature space is classified or grouped into certain classes according to so-called decision rules [13]. Usually, the feature vectors derived from the original data are in lower dimension than the original data, unless the original dimension is necessary. Once the feature space is established, one classifies or groups each feature vector in certain regions into given classes. Those regions are divided by hyper-planes formulated by so-called decision (or discriminant) functions [13]. In general the discriminant function is determined on the ground of the distance measure between feature vectors which uniquely represent the original data.

Principal Components Analysis (PCA) is a statistical tool, which is one of the most popular dimension-reducing methods [10, 15]. The linear transformation by the matrix of all principal components conserves the Mahalanobis distance between two feature vectors. We will propose a novel approach for feature extraction, built upon PCA.

4. Principal Components Analysis

PCA is a well known statistical method to reduce the dimension of a dataset or to find features in data of high dimension. PCA is to analyze the correlations or covariances between multiple variables [10], and hence requires the data to be two or more dimensional. However, time-series data are not always multi dimensional and PCA is not always applicable for the time-series data (of a single dimension). And yet, we propose a new method, where PCA is applicable and compatible with existing PR methods even to a single dimensional dataset.

PCA is a statistical tool to identify the variability of the multi-dimensional data, i.e., PCA analyzes the correlations between variables. We will shortly review the forward transform of PCA, which gives us the principal components or the eigenvector, and corresponding variances or eigenvalues. More details and advanced applications, such as data compression and data recovering, can be found in [10], [3] and [15].

In Figure 4, an exemplar dataset of two dimensions is depicted in two different coordinate systems, which are related by the equation as follows;

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x - \bar{X} \\ y - \bar{Y} \end{bmatrix}.$$

The variance of the dataset about the average point $(\bar{X}, \bar{Y})$ and along the direction of θ is proportional to V', the length of the projections onto the X' axis. In general, the projection of individual point is called Score of the point onto the X' axis, and V' is actually proportional to the variance of the Scores. By changing the projection angle θ, we can obtain the maximum V', and the angle θ_p at which the maximum variance is obtained is known as the principal angle.

The largest eigenvalue of the data, which is the variance of the Scores along the X' axis, is proportional to the maximum projection length V'. The eigenvector corresponding to the largest eignevalue is called the principal component and determines the principal angle θ_p and hence the X' axis.

PCA starts with a collection of multi-dimensional data. We then compute the covariance matrix from the data and compute the eigenvalues and corresponding unitary eigenvectors (e.g. $\mathbf{u}_{X'}$ and $\mathbf{u}_{Y'}$ in Figure 4) of the covariance matrix. Now the orthonormal eigenvectors can be used as the bases for a new space and each of the corresponding eigenvalue shows how much the corresponding eigenvector is related to the data, i.e. the variance of the Scores along the eigenvector.

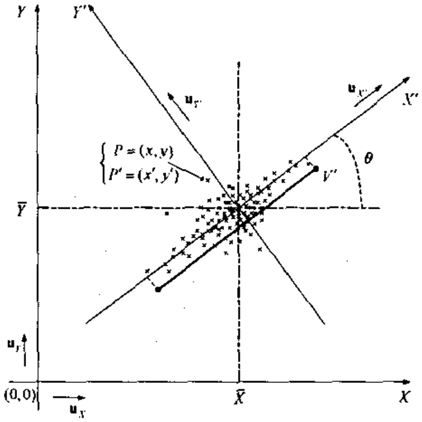


Figure 4. Example data of two dimensions.

As the eigenvector of a larger eigenvalue is more related to the data and contains more statistical information about the data, the largest eigenvalue that we now call the "Pseudo" Principal Component (PPC) of the data plays an important role for feature extraction and pattern recognition, as we will discuss in Section 6.

5. Eigenvalues and eigenvectors

As we have seen in Section 4, eigenvalues and eigenvectors is very important in PCA. Now let us discuss the eigenproblem, i.e., how to find the eigenvalues and eigenvectors of a matrix. The theory and computing mechanisms about eigenproblem is well established in many textbook e.g. [1] and [12].

A square matrix $\mathbf{A}$ can be Schur-factored with QR algorithm and Householder transformation [12] into $\mathbf{A} = \mathbf{STS}'$ where $\mathbf{T}$ is an upper Hessenberg matrix and $\mathbf{S}$ is Schur vectors and $\mathbf{S}'$ is a transpose of $\mathbf{S}$. Then the diagonal elements of the matrix $\mathbf{T}$ are the eigenvalues of the matrix $\mathbf{A}$. For the corresponding eigenvectors, we first compute the dominant (or the largest) eigenvalue by the power iteration [1], and then the rests using the deflation algorithm [1]. In short, if we have an eigenvalue λ_1 of n-dimensional matrix $\mathbf{A}$ then the additional eigenvalues $\lambda_2,\cdots,\lambda_n$ can be obtained through the following steps. First, we transform $\mathbf{A}$ into

$$\mathbf{HAH'}=\begin{bmatrix}\lambda_1 & \mathbf{b'}\\ 0 & \mathbf{B}\end{bmatrix}$$

where $\mathbf{H}$ is a Householder matrix and $\mathbf{B}$ is a matrix of order n-1 containing eigenvalues $\lambda_2,\cdots,\lambda_n$. Second, we compute eigenvalue λ_2 and the corresponding eigenvector $\mathbf{g}_2$ of $\mathbf{B}$. Third, the eigenvector $\mathbf{G}_2$ of $\mathbf{A}$ corresponding λ_2 is given by the equations;

$$\mathbf{G}_2=\mathbf{H'}\begin{bmatrix}a\\ \mathbf{g}_2\end{bmatrix} \text{ and } a=\frac{\mathbf{b'g}_2}{\lambda_2-\lambda_1}$$

We repeat these until we get all the eigenvectors of $\mathbf{A}$.

6. Pseudo Principal Component Analysis

PCA requires multi-dimensional square covariance matrix, and hence we cannot directly apply PCA to the time-series dataset of a single dimension. Instead, we construct new datasets from the original times-series dataset and apply PCA to the newly constructed datasets. In this section, we will discuss about time-delay embedding [11] for establishing new datasets and how to extract features from the new datasets. Our objective in this section is to find the statistical relationship among individual instant values of time-series data and extract a feature (and now a pseudo principal component) for each of the dataset.

Our methodology rests on transforming a single dimensional dataset into datasets of multi dimension. We, first, resample from the original dataset by a sampling filter kernel with a certain time delay and set up new sets of data of multi dimensions. We used a non-causal MA (moving average) filter as a sampling kernel of a form of FIR (Finite Impulse Response), however theoretically there is no limitation for the kernel. Let's consider the data $\{x[n]\}_1$ in Figure 5. We set up data $\{d_1[n]\}$ and $\{d_2[n]\}$ using an MA of kernel size K and delay time $\tau=1$ as shown in Figure 5.

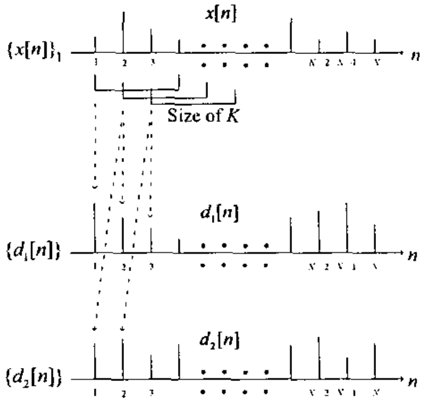


Figure 5. Re-sampling and new datasets

Now we get 2 dimensional data of $\{d_1[n]\}$ and $\{d_2[n]\}$ with length N from a single dimensional data $\{x[n]\}_1$ with length N. We compute covariance matrix $CM^{2\times 2}$ of the new data of 2nd order as follows;

$$CM^{2\times 2}=\begin{bmatrix}\mathrm{cov}(d_1,d_1) & \mathrm{cov}(d_1,d_2)\\ \mathrm{cov}(d_2,d_1) & \mathrm{cov}(d_2,d_2)\end{bmatrix}$$

$$\mathrm{cov}(d_i,d_j)=\frac{1}{N}\sum_{m=1}^{m=N}(d_i[m]-\overline{d_i})(d_j[m]-\overline{d_j})$$

$$\overline{d_i}=\frac{1}{N}\sum_{m=1}^{m=N}d_i[m]$$

From $CM^{2\times 2}$, we compute the Pseudo Principal Component (PPC_1) of the new data and this PPC_1 represents the feature of the original times-series data $\{x[n]\}_1$. Figure 6 shows each step to the feature extraction discussed above.

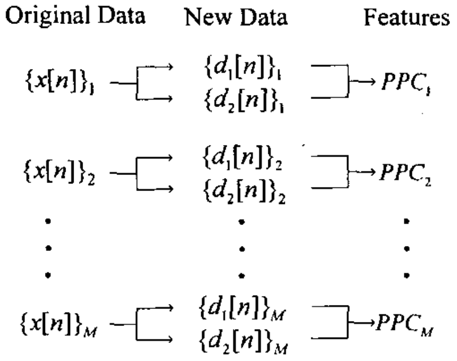


Figure 6. Feature extraction

7. Continuous Wavelet Transform

In addition to Pseudo PCA, we incorporated with Continuous Wavelet Transform (CWT) that is a well known technique to analyzing localized frequencies overcoming the disadvantage of the Fourier Transform [16, 17]. Figure 7 shows how CWT contributes to feature extraction with Pseudo PCA.

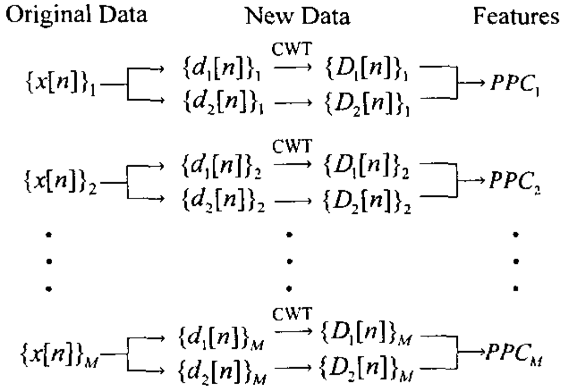


Figure 7. Pseudo PCA with CWT

For arbitrary signal $s(t)$, the equation of CWT or the continuous wavelet coefficients is given by,

$$C(\alpha,\tau) = \frac{1}{\sqrt{\alpha}}\langle s(t), \Psi(\frac{\tau - t}{\alpha})\rangle$$
$$= \frac{1}{\sqrt{\alpha}}\int_{-\infty}^{\infty} s(t)\Psi(\frac{\tau - t}{\alpha})dt$$

where $\langle\bullet,\bullet\rangle$ is an inner product, α is a scaling factor, τ is a translation factor, and $\Psi(\bullet)$ is a mother wavelet (function) [6]. We used the Daubechies-2 mother wavelet for our application, and it is shown in Figure 8.

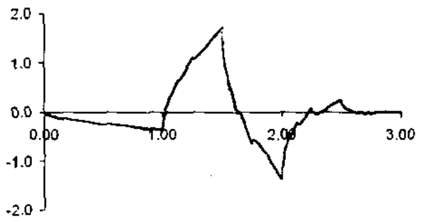
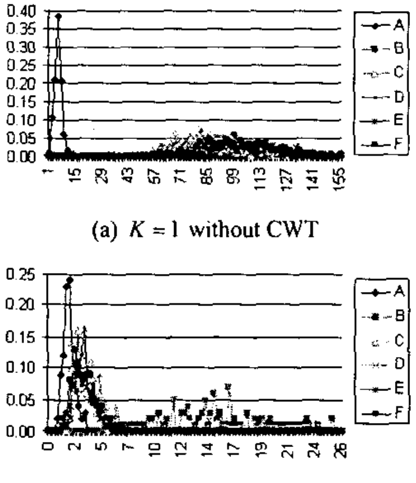


Figure 8. The mother wavelet of Daubechies-2.

8. Pattern recognition – Classification

Each of the extracted features represents the corresponding data and will be used to classify the patterns. Moreover, if we are given unknown future data, we can predict which class the data should be associated to. In this section we will discuss how the Pseudo Principal Component of each data can be used to classify and recognize the embedded patterns.

Some of the probability mass functions (PMF) of the pseudo principal components are shown in Figure 9 with different kernel size K and wavelet scale factor α. The vertical axis is the probability mass and the horizontal axis is the principal components of the data. As indicated in Figure 9, it turned out that the datasets of the same class can be grouped and separated from the others. For example, the Class A is separated from the others in Figure 9-a. The Class B was hard to separate from the others in Figure 9-a, however, we can separate Class B if we consider Figure 9-b. Once we have classification information, we can predict the class for future data as follows. We set a range where the principal components of the dataset of a specific class as a classification criteria. If the principal component of the test data is within the range, we associate the data to the corresponding class. We repeat this predictions L times with different kernel size K and wavelet scale factor α, then the majority of the results will be the final prediction for the data. Figure 10 shows how this prediction process works.



(a) $K = 1$ without CWT

(b) $K = 1, \alpha = 1$ with CWT

Figure 9. PMF of the Synthetic Control Chart data

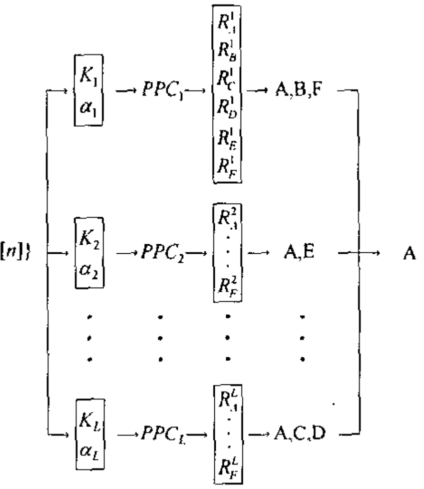


Figure 10. Prediction mechanism

9. Results

We generated a total of 1200 sequences of the Synthetic Control Chart data in order to learn the embedded patterns (or to get the classification criteria). Each of the class set was consisted of $M = 200$ time-series data of length $N = 60$ (see Figure 1 and 2). We first used 60 different sampling kernels of size $K = \{1, 2, \ldots, 60\}$ without CWT. After learning the patterns, we verified the prediction with new testing data. A total of 600 new data was tested with $M = 100$ per each class and the results are shown in Table 1. As we can see in Table 1, the pseudo principal components computed by the process that we have described above results in a good separation for Class A, B, C, and D. However, Class E and F were predicted a little bit worse than the others by the pseudo principal components. At this point, we introduce the CWT technique. As we can see in Figure 9, feature extraction with CWT gives better grouping and separation. The test results with both PCA and CWT are shown in Table 2. It turns out that a proper selection of kernel size K and wavelet scale factor α leads us to a good recognition results.

Table 1 Test results with only Pseudo PCA

Class	A	B	C	D	E	F
Error (%)	1.0	1.0	2.0	1.0	27.0	31.0

$K = \{1, 2, \ldots, 60\}$

Table 2 Test results with both Pseudo PCA and CWT

Class	A	B	C	D	E	F
Error (%)	0.0	1.0	2.0	0.0	9.0	6.0

$(K,\alpha) = \{(1,1),(1,31),(21,31),(41,41),(51,5)\}$

For the sake of performance comparison with the state of the arts for speech recognition problem, we downloaded the Japanese Vowel database from the URL, http://kdd.ics.uci.edu. The database is created by M. Kudo, et al., and is thankfully open to the public for research. As we can see in Table 3, the performance of our novel method was compatible with the state of the art methods, although we see the needs for more study to improve the performance. In Table 3, MDC is multi-dimensional classifier; 5-NN HMM is 5 nearest neighbor hidden Markov model; and PPCA is pseudo PCA.

Table 3 Test results for the Japanese Vowel

Method	MDC	5-NN HMM	PPCA
Correct (%)	94.1	96.2	92.2

10. Conclusion

We proposed a novel method, Pseudo PCA, to extract a feature from time-series data, by constructing new datasets with time delay embedding and then computing their principal components by PCA. We experimented the novel method with the pattern recognition of the Synthetic Control Chart data. Also, we combined the CWT technique to improve the performance of the proposed method. The results showed that PCA with time-delay embedding is useful to classification tasks of time-series data, and that a proper selection of kernel size K and wavelet scale factor α leads us to a good recognition results.

The comparison with the state of the art methods for the Japanese Vowel database also showed that the novel method is comparable in performance. Also more study is needed to improve the performance, and show the computational efficiency, e.g. compared to the Hidden Markov Models.

10. References

[1] Michael T. Heath, *Scientific Computing: An Introductory Survey, Second Edition*, McGraw-Hill, New York; 2002.

[2] R. J. Alcock and Y. Manolopoulos, "Time-Series Similarity Queries Employing a Feature Based Approach", *in 7th Hellenic Conference on Informatics*, Ioannina, Greece, August 27-29, 1999.

[3] Michael E. Tippong and C. M. Bishop, "Probabilistic Principal Components Analysis", *Journal of the Royal Statistical Society*, Series B, 61, Part 3, 1999, pp 611-622.

[4] A. S. Weigend and N. A. Gershenfeld et al., *Time Series Prediction: Forecasting the future and understanding the past, Proc. of the NATO advanced research workshop*, Santa Fe, New Mexico, May 14-17, 1998.

[5] D. T. Pham and A. B. Chan, "Control Chart Pattern Recognition Using a New Type of Self-Organising Neural Network", *Proceedings of Institution of Mechanical Engineers*, vol. 212, No. 1, 1998, pp 115 - 127.

[6] M. J. Vrhel, C. Lee, and M. Unser, "Rapid Computation of the Continuous Wavelet Transform by Oblique Projections", *IEEE Trans. on Signal Processing*, vol. 45, No. 4, 1997, pp 891-900.

[7] D.T. Pham and E. Oztemel, "Control chart pattern recognition using combinations of multi-layer perceptrons and learning-vector-quantization neural networks", *Proceedings of Institution of Mechanical Engineers*, vol. 207, Part I, 1993, pp 113-118.

[8] C.T. Chen, *Analog & Digital Control System Design: Transfer-function, State-space, & Algebraic Methods*, Saunders College Publishing; 1993.

[9] Alen V. Oppenheim and Ronald W. Schafer, *Discrete-Time Signal Processing*, Englewood-Cliffs, New Jersey, Prentice-Hall; 1989.

[10] R. W. Preisendorfer, *Developments in Atmospheric Science 17: Principal Components Analysis in Meteorology and Oceanography*, Elseveir Science Publishers B. V.; 1988.

[11] F. Takens, "Detecting strange attractors in turbulence," *Dynamical Systems and Turbulence, Lecture Notes in Mathematics*, vol. 898, Springer-Verlag, Berlin, pp 366-381, 1981.

[12] G. H. Golub and C. F. Van Loan, *Matrix Computations. Third Edition*, The Johns Hopkins Univ. Press; 1996

[13] J. P. Marques de Sá, *Pattern Recognition: Concepts, Methods and Applications*, Springer, 2001.

[14] M. Kudo, J. Toyama and M. Shimbo, "Multidimensional Curve Classification Using Passing-Through Regions". *Pattern Recognition Letters*, Vol. 20, No. 11--13, pp 1103-1111, 1999.

[15] J. E. Jacson, *A User's Guide to Principal Components*, Wiley Series in Probability and Statistics, 1991.

[16] I. Daubechies, Ten Lectures on Wavelets, SIAM, 1992.

[17] O. M. Nielsen, *Wavelets in Scientific Computing*, PhD Dissertation, Dept. of Mathematical Modeling, Technical University of Denmark, 1998.