Principal Curve Time Warping

Umut Ozertem, *Member, IEEE*, and Deniz Erdogmus, *Senior Member, IEEE*

***Abstract*—Time warping finds use in many fields of time series analysis, and it has been effectively implemented in many different application areas. Rather than focusing on a particular application area we approach the general problem definition, and employ principal curves, a powerful machine learning tool, to improve the noise robustness of existing time warping methods. The increasing noise level is the most important problem that leads to unnatural alignments. Therefore, we tested our approach in low signal-to-noise ratio (SNR) signals, and obtained satisfactory results. Moreover, for the signals denoised by principal curve projections we propose a differential equation-based time warping method, which has a comparable performance with lower computational complexity than the existing techniques.**

***Index Terms*—Kernel density estimation (KDE), principal curves, signal denoising, time warping.**

I. Introduction

TIME series analysis is an important field in adaptive signal processing with numerous applications in econometric studies on stock market data, in biomedical and speech processing, or even in intelligent transportation systems analysis. A common problem in all these applications is to derive a suitable distance measure between time series signals. In many cases, although some pairs of signals demonstrate similar characteristics (for unsupervised scenarios) or belong to the same class (for supervised scenarios), the predominant structures of the signals do not align in the time axis. Dynamic time warping (DTW) is a technique to solve this alignment problem [1]. In general, DTW is a method that finds an optimal match between two sequences of feature vectors with certain restrictions—monotonicity, continuity, and boundary conditions. The problem is modelled as finding the minimum distance through a matrix of pairwise distances of the data samples, and DTW uses dynamic programming techniques to obtain the solution.

DTW algorithm is later referred to as the *DTW model*, since it was reinterpreted as a parametric model [2], [3]. Although it has found widespread use, DTW is susceptible to noise, and time warpings over low signal-to-noise ratio (SNR) signals may end up with many *singularities*. Singularities are defined as "unintuitive alignments where a single point on one time series maps onto a large subsection of another time series" [8]. Literature on DTW is rich on modifications to increase noise robustness. However, most of these are heuristic attempts which are not guaranteed to remove all singularities. Even worse, they may lead to suboptimal solutions. Techniques include using moving average filters on the time series signals to reduce the high frequency content, assuming that most of the high frequency content is the noise [4]–[6]. This can be considered as constraining the search space of *allowable* warpings, and although constraining the search space of allowable warpings may theoretically lead to suboptimal solutions for the time warping function, strong empirical evidence has also been reported that to this approach may increase the classification performance [7]. Most recent work in time warping literature includes derivative dynamic time warping (DDTW) [8] that uses the derivative of the signals rather than the original values, enhanced dynamic time warping (EDTW) [9] that brings a unifying view to DTW and hidden Markov models, and context dependent dynamic time warping (CDDTW) [10] that exploits application specific contextual characteristics of the signals to improve performance.

Our proposition to improve the noise sensitivity of the time warping problem is a principal curve-based preprocessing step, and we show that this approach allows one also to evaluate the time warping function in low SNR cases. Principal curves are defined by Hastie and Stuetzle [11], [12] as "*self-consistent finite length smooth curves passing through the middle of data*." Principal curve algorithms in the literature include Tibshirani's mixture-model expectation maximization approach [13], Sandilya and Kulkarni's bounded curvature approach [14], Kegl and colleagues' piecewise-linear bounded-length approach [15], [16], and Stanford and Raftery's outlier robust algorithm [17]. We recently proposed another definition, which describes the principal curve in terms of the gradient and the Hessian of the data probability density [18], and we will use our definition throughout this paper.

The contribution of this paper is twofold: i) Principal curve projection of the data is proposed as a preprocessing step for existing time warping algorithms. This projection can be performed either in the original signal space or any feature set derived from the data. Therefore, this denoising step can be coupled with any time warping approach in the literature. ii) We propose a simple differential equation-based method to find the time warping function. Although this method does not provide a robust approach for noisy cases, it brings a suitable computationally inexpensive alternative for the signals denoised by the principal curve projections.

II. Principal Curve Time Warping

To motivate the reader, we start with a brief discussion of our principal curve definition. Overall, principal curves generalize

Manuscript received November 09, 2007; accepted November 12, 2008. First published February 24, 2009; current version published May 15, 2009. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Pramod K. Varshney. This work was supported in part by the NSF by Grants ECS-0524835 and ECS-0622239.

U. Ozertem is with Yahoo! Labs, Sunnyvale, CA 95054 USA (e-mail: umut@yahoo-inc.com).

D. Erdogmus is with the Electrical and Computer Engineering Department, Northeastern University, Boston, MA 02215 USA (e-mail: derdogmus@ieee.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2009.2016268

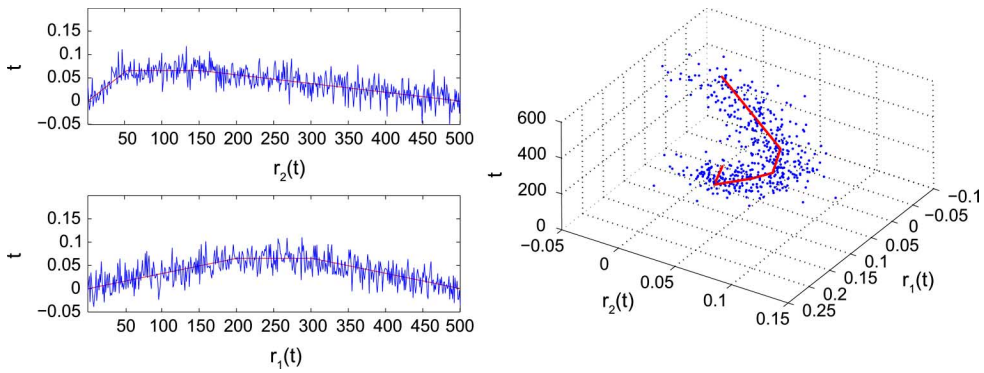

Fig. 1. Noisy signals ($\mathrm{SNR} = 3$ dB) r_1 and r_2 (blue) and the noiseless signals $s(t)$ and $s(f(t))$ (red) are presented (left). Same signals in the $\mathbf{r}$ space is also shown (right).

the self-consistency property of the principal line, the first principal component, into nonlinear structures, and they are used for capturing the predominant structure of the data. We define the principal curves in terms of the gradient and the Hessian of data probability density. We show that open ended problems in principal curve literature, like overfitting or smoothness constraints, can be answered by well studied results of density estimation literature. According to our definition, a point in the data feature space is on the principal curve *if and only if the gradient of the pdf is an eigenvector of the Hessian of the pdf, and the remaining eigenvectors have negative eigenvalues* [18], which defines the ridge of the pdf as the principal curve. This definition yields subspace constrained maximum likelihood algorithms [19], which can be developed in the spirit of the well-known mean shift algorithm [26], [27]. However, time warping problem yields a special principal curve and we provide a specialized derivation particularly for this problem. For further details of the general definition and the subspace constrained mean shift algorithm, please refer to our earlier work [18], [19].

A. The Feature Space and Principal Curve Projections

In template matching or hypothesis testing algorithms, the test signals are compared with the noiseless template signal. Here we consider the more realistic case of two noise corrupted signals, assuming a noiseless template signal may not be available in all applications. We write the noisy signals as

$$\begin{aligned} r_1(t) &= s(f(t)) + n_1(t), \quad t = t_1 < \cdots < t_N \\ r_2(t) &= s(t) + n_2(t), \quad t = t_1 < \cdots < t_N \end{aligned} \tag{1}$$

where $f(t)$ is the time warping function and $n_1(t)$ and $n_2(t)$ are unimodal additive noise. Here we assume that the signals $r_1(t)$ and $r_2(t)$ have the same length, and we build the principal curve feature space as

$$\mathbf{r}_i = \begin{bmatrix} r_1(t_i) \\ r_2(t_i) \\ t_i \end{bmatrix}, \quad t = t_1 < \cdots < t_N \tag{2}$$

Fig. 1 shows two realizations of $\mathbf{r}$ along with the corresponding noisy signal pairs $r_1(t)$, and $r_2(t)$ and noiseless signal pairs $s(t)$, and $s(f(t))$. Fig. 1(a) presents piecewise linear signals $s(t)$, and $s(f(t))$ (red) and their noisy versions $r_1(t)$, and $r_2(t)$ (blue). The data structure in $\mathbf{r}$ given in Fig. 1(b) demonstrates the pairwise signal characteristics, as a perturbation around a predominant shape in time; again red and blue show the noiseless and noisy signals, respectively. As the noise level increases, the amount of perturbation around the predominant shape increases. Here, we propose to use the principal curve projections of the data samples to approximate the noiseless signal characteristics in $\mathbf{r}$ domain. *Projecting the signal samples onto the principal curve can be regarded as a nonparametric nonlinear filtering.* This yields

$$\widetilde{\mathbf{r}}_i = \begin{bmatrix} \widetilde{s}(f(t_i)) \\ \widetilde{s}(t_i) \\ t_i \end{bmatrix}, \quad t = t_1 < \cdots < t_N \tag{3}$$

where $\widetilde{\mathbf{r}}_i$ is the projection of $\mathbf{r}_i$ onto principal curve.

To implement the principal curve projections, one can directly use the approaches we proposed earlier [18], [19]. However, particularly for the time warping application, we have a much easier scenario due to the following:

1) Only the samples of the principal curve at time indices $t = t_1 < \cdots < t_N$ are sufficient. Higher time resolution or seeking for the portion of the principal curve that lies outside the given time interval are unnecessary.
2) Under the unimodal additive noise assumption there will be no *branching* in the principal curve, since $s(t)$ and $s(f(t))$ are functions of time. Thus, for every subspace of $\mathbf{r}$ defined by $t = t_i t_i \in [0, 1]$ there is a single point on the principal curve.
3) Unlike the general case of random vectors, the third dimension of $\mathbf{r}$ is deterministic; in the case of uniform sampling, we can model this density as being uniform for theoretical analysis.

One can select the initialization of the algorithm and the constrained space of the projection using above simplifications. At this point, starting from the data samples themselves, and selecting the constrained space as the $t = t_i$ for each data sample is our choice for the two following reasons:

1) Selecting constrained space orthogonal to time index guarantees that there is only one denoised signal value at all time indices.
2) One important observation here is that the peak of the pdf in each constrained space $t = t_i$ is very close to principal curve.

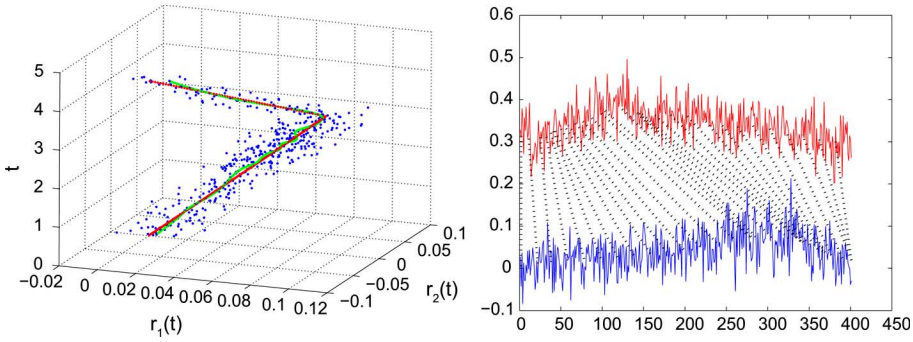

Fig. 2. The noisy (blue), the noiseless data (red) characteristics in $\mathbf{r}$, along with the principal curve estimated from the noisy data (green) is shown (left), and the corresponding alignments are shown for the same signals (right).

TABLE I
PRINCIPAL CURVE DENOISING

1) Build the feature space $\mathbf{x}$, select the kernel bandwidth σ of the Gaussian kernel.
2) Evaluate the mean shift update using (7).
3) Project the mean shift update as in (8), so that the mean shift procedure remains in the constrained space.
4) If convergence not achieved, go to step 2, if convergence is achieved go to the next step. If convergence achieved, the following steps are optional to guarantee the convergence onto the principal curve.
5) For every trajectory evaluate the mean shift update in (7).
6) Evaluate the gradient, the Hessian, and perform the eigendecomposition of $\mathbf{\Sigma^{-1}}(\mathbf{x}(k)) = \mathbf{V\Gamma V}$.
7) Let $\mathbf{v}$ be the leading eigenvector of $\mathbf{\Sigma^{-1}}$.
8) $\tilde{\mathbf{x}} = \mathbf{v}\mathbf{v}^T\mathbf{m}(\mathbf{x}(k))$
9) If $|\mathbf{g}^T\mathbf{H}\mathbf{g}|/\|\mathbf{g}\|\|\mathbf{H}\mathbf{g}\| > threshold$ then *stop*, else $\mathbf{x}(k+1) \leftarrow \tilde{\mathbf{x}}$.
10) If convergence is not achieved, increment k and go to step 6.

The second observation here is based on the additive unimodal noise assumption and the fact that there is no branching in the principal curve. Although it is close, the maximum in the constrained space $t = t_i$ is not exactly on the principal curve. Therefore, after convergence in $t = t_i$ subspace, we will use the eigendecomposition of the Hessian matrix to select the constrained space to ensure the projection onto the principal curve. Using Kernel density estimation (KDE) to estimate the density of $\mathbf{r}$ one can write

$$p(\mathbf{r}) = \frac{1}{N}\sum_{i=1}^{N} K_\Sigma(\mathbf{r} - \mathbf{r}_i) \tag{4}$$

where $K_\Sigma(\cdot)$ is the kernel function with covariance Σ. We use the typical Gaussian kernel, and we will discuss the selection of the covariance of the Gaussian later in detail. Substituting the Gaussian kernel function into (4), the KDE of $\mathbf{r}$, and its gradient are given as

$$\begin{aligned} p(\mathbf{r}) &= N^{-1}\sum_{i=1}^{N} G_\Sigma(\mathbf{r} - \mathbf{r}_i) \\ \mathbf{g}(\mathbf{r}) &= -N^{-1}\sum_{i=1}^{N} \Sigma^{-1}(\mathbf{r} - \mathbf{r}_i) G_\Sigma(\mathbf{r} - \mathbf{r}_i). \end{aligned} \tag{5}$$

Using $\mathbf{g}(\mathbf{r}) = 0$, one obtains

$$\sum_{i=1}^{N} \Sigma^{-1}(\mathbf{r} - \mathbf{r}_i) G_\Sigma(\mathbf{r} - \mathbf{r}_i) = 0$$

$$\sum_{i=1}^{N} \Sigma^{-1}\mathbf{r} G_\Sigma(\mathbf{r} - \mathbf{r}_i) - \sum_{i=1}^{N} \Sigma^{-1}\mathbf{r}_i G_\Sigma(\mathbf{r} - \mathbf{r}_i) = 0.$$

$$\mathbf{r}\sum_{i=1}^{N} \Sigma^{-1} G_\Sigma(\mathbf{r} - \mathbf{r}_i) - \sum_{i=1}^{N} \Sigma^{-1}\mathbf{r}_i G_\Sigma(\mathbf{r} - \mathbf{r}_i) = 0. \tag{6}$$

Solving for $\mathbf{r}$ yields the well-known mean shift update rule [26], [27]

$$\mathbf{r} \leftarrow \mathbf{m}(\mathbf{r}) = \left(\sum_{i=1}^{N} \mathbf{r}_i \Sigma^{-1} G_\Sigma(\mathbf{r} - \mathbf{r}_i)\right)^{-1} \times \sum_{i=1}^{N} \Sigma^{-1} G_\Sigma(\mathbf{r} - \mathbf{r}_i). \tag{7}$$

To implement the constraint on the space, assuming the sampling times t_i are noisefree, one should modify (7) by projecting the update onto the initial $t = t_0$ plane. This can simply be implemented using a matrix multiplication from left.

$$\mathbf{r} \leftarrow \mathbf{A}\mathbf{m}(\mathbf{x}) + \mathbf{b}. \tag{8}$$

The projection matrix $\mathbf{A}$, and translation vector $\mathbf{b}$ are given by

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 0 \\ 0 \\ t_0 \end{bmatrix} \tag{9}$$

where t_0 is the time index of the initial sample $\mathbf{r}_i$. Iterating (8) until convergence, one obtains the conditional maximum of the pdf in the constrained space $t = t_i$. Although the maximum in the $t = t_i$ space is very close to the principal curve, to ensure the convergence onto the principal curve, one should use the eigendecomposition of the Hessian of the pdf as we proposed earlier in SCMS algorithm [19]. Here we skip the derivation of the SCMS algorithm; however, to provide a complete implementation, details of the SCMS algorithm is included in Table I. Overall, the proposed denoising scheme has a complexity of $O(N^2)$.

Fig. 2(a) shows the found principal curve (green) for the signal pairs presented in Fig. 2(b) along with their time alignments. We will provide a more detailed analysis on the accuracy of the approximation given in (3) in the experimental results section by presenting the distance between the noiseless signal structure and the principal curve of the noisy signal.

B. Solving for the Time Warping Function

Regarding the principal curve projections as an independent denoising preprocessing filter, proposed principal curve projec-

tions can be coupled with any time warping algorithm in the literature. For this purpose, one can simply use the original DTW algorithm or any of its derivatives in the literature. Here we present a differential equation-based method as a computationally efficient alternative.

Instead of directly working on the signal samples, we use the derivative of the signals to ensure the continuity and monotonicity properties of the time warping function. We define a nonnegative and bounded derivative by construction to guarantee these properties. Using the approximation in (3), one can write the derivative of $\widetilde{\mathbf{r}}$ as

$$\widetilde{\mathbf{r}}_i' \simeq \begin{bmatrix} s'(f(t_i))f'(t_i) \\ s'(t_i) \\ 1 \end{bmatrix}, \quad t_1 < \cdots < t_N. \tag{10}$$

Using $\widetilde{\mathbf{r}}$ one can write a differential equation to find the derivative of $f(t)$[1]. This yields

$$f'(t) = \frac{\widetilde{r}_1'(t)}{\widetilde{r}_2'(f(t))}. \tag{11}$$

Selecting the initial condition of the differential equation according to the boundary condition $f(t_1) = t_1$, and integrating it with a suitable step size Δ one can evaluate the time warping function. For convenience, we drop the subscript of the time warping function for the rest of the paper.

$$\begin{aligned} f_{k+1} &= f_k + \Delta f'(k\Delta) \\ f_{k+1} &= f_k + \Delta \left(\frac{r_1'(k\Delta)}{r_2'(f(k\Delta))} \right). \end{aligned} \tag{12}$$

To approximate the derivative of r_1 and r_2 we use the same step size Δ. This yields

$$\begin{aligned} r_1'(k\Delta) &= \frac{r_1((k+1)\Delta) - r_1(k\Delta)}{\Delta} \\ r_2'(f(k\Delta)) &= \frac{r_2(f(k\Delta)) - r_2(f(k\Delta))}{\Delta}. \end{aligned} \tag{13}$$

Substituting (13) into (12), one can obtain the differential equation for the solution of the time warping function, the result of which can be evaluated by integrating the differential equation throughout the time interval $[0, 1]$ with a suitable step size Δ. The default selection should be $\Delta = 1/N$ so that the solution of the differential equation can be carried out over the available signal samples. As Δ increases, the computational complexity decreases. This is not much different than subsampling the signals to end up with a smaller cost matrix for DTW. Therefore, we will use $\Delta = 1/N$ in our experiments.

Let us briefly review the required properties of the time warping function.

Boundary Conditions: To satisfy boundary conditions one should satisfy $f(t_1) = t_1,\ f(t_N) = t_N$.

Monotonicity: Given $r_1'(k\Delta)/r_2'(f(k\Delta)) \geq 0$, the time warping function has a nonnegative derivative; if this condition is not satisfied due to numerical errors we modify the update equation as $f_{k+1} = f_k$ to ensure the resulting time warping function is nondecreasing.

Continuity: Given as a solution of a differential integration problem, the time warping function is continuous by construction.

[1]Note that a best-fit solution that satisfies (11) could also be obtained by solving an error minimization problem with initial and final value constraints.

C. Selecting the Covariance of the Kernel Function

A significant practical consideration for the implementation of the algorithm is the selection of the bandwidth of the Gaussian kernel function. As with many other kernel-based approaches, PCTW cannot provide satisfactory results for improper bandwidth selections. Fortunately, literature on density estimation and kernel machines present many reliable methods for selecting the kernel bandwidth [20]–[25]. These techniques extend from local neighborhood distances-based heuristic approaches [20] to maximum likelihood-based principled methods [21]. For instance, Silverman's rule [23], and Comaniciu's data driven approach [24] are among the most commonly used bandwidth selection techniques. In many kernel machine applications, spherically symmetric Gaussian kernels suffice. However, this cannot be the case for the feature space we define in (3). The variance around the principal curve in the first two dimensions depends on the noise power of the two compared signals, whereas time index in the third dimension may have any arbitrary scale depending on the sampling rate.

Furthermore, the bandwidth of the Gaussian kernel can be adjusted by exploiting the actual physical meaning of data feature space. In many signal processing applications, the noise power can be estimated very reliably. In such cases, the estimate of the noise distribution can be used as the kernel function. Assuming n_1 and n_2 are independent Gaussian noise, one can write the covariance of the Gaussian kernel function as

$$\Sigma = \begin{bmatrix} \sigma_{n_1} & 0 & 0 \\ 0 & \sigma_{n_2} & 0 \\ 0 & 0 & N \end{bmatrix} \tag{14}$$

where N controls the amount of smoothness to be introduced along the time axis. Generally, this choice is not optimal; still, it eliminates tedious kernel optimization efforts, and yields satisfactory results. Specific selections regarding to our experiments will be mentioned in the experimental results section.

III. Experimental Results

We will start the experimental results on the accuracy of the denoising approximation [that is (3)], and present the error between the noiseless signal structure and the principal curve for different noise levels. Afterwards, we will present signal pairs and corresponding time warping functions for synthetic and real data examples. These experiments use the true signal or the true time warping function to report results for different noise levels. Eventually, the aim of time warping is to define a distance measure between time series signal pairs to increase the clustering/classification performance. Therefore we also present results for the time series clustering and time series classification.

A. Noiseless Signal Versus Principal Curve of the Noisy Signal

The principal curve denoising results presented in this subsection exclude the solution of the time warping function to give

the reader the chance to evaluate the principal curve denoising step independently. Therefore, here we present results for the approximation we make in (3), and give the integrated error between the noiseless signal structure and the principal curve of the noisy signal for different noise levels. In this experiment, we will use the following synthetic signal to be able to repeat the experiment for different noise levels.

$$s_{\text{synthetic}}(t) = \begin{cases} \frac{t}{t_1} & : 0 \le t \le t_1 \\ 1 & : t_1 \le t \le t_2 \\ \frac{t-1}{t_2-1} & : t_1 \le t \le 1 \end{cases} \tag{15}$$

where t_1 is uniformly distributed between 0.05 and 0.45, and t_2 is uniformly distributed between 0.55 and 0.95[2]. We generate realizations of this random signal $s_{\text{synthetic}}(t)$ of length $N = 200$ for each noise level, and add white Gaussian noise to obtain the samples of the noisy signal $r_{\text{synthetic}}(t)$ for 10, 5, 3, and 2 dB. Hence, the noisy signal is

$$r_{\text{synthetic}}(t) = s_{\text{synthetic}}(t) + n_{\text{Gaussian}}(t). \tag{16}$$

For 100 pairs of random realizations of $r_{\text{synthetic}}(t)$, we build the feature space as described in (2) and use the iterative scheme given in Table I to project the data onto the principal curve. The kernel function is selected by using the known SNR levels of the signals, as described in Section III-C. We evaluate the integrated error between the noiseless structure using pairs of $s_{\text{synthetic}}(t)$, and the approximation provided by principal curve for different noise levels.

Fig. 3 shows the mean and the variance, the error bar demonstrates ± 2 standard deviation from the mean, of the integrated error of these 100 Monte Carlo simulations at noise levels 10, 5, 3, and 2 dB. The accuracy of the principal curve approximation decreases with increasing noise level. Still, the method is able to provide reliable approximations for noise levels as high as 2 dB.

B. Solutions of PCTW in Different Noise Levels

In this section, we present the solutions of PCTW on synthetic and real data for different noise levels. First we will present how denoising actually affects the final results. Afterwards, we will compare results of the original DTW algorithm and the proposed computationally inexpensive differential equation method along with their computational times.

For the synthetic dataset, we select the same noiseless signal used in the previous experiment for simplicity. To show how the denoising step affects the final results, for this experiment we compare results of DTW algorithm using (i) the original noisy signals and (ii) the principal curve projections. Fig. 4 present the synthetic signal pairs, the noiseless signal pairs, and the solution of time warping function for different noise levels. Also

[2]In fact, this is the same noiseless signal presented in Fig. 1(a).

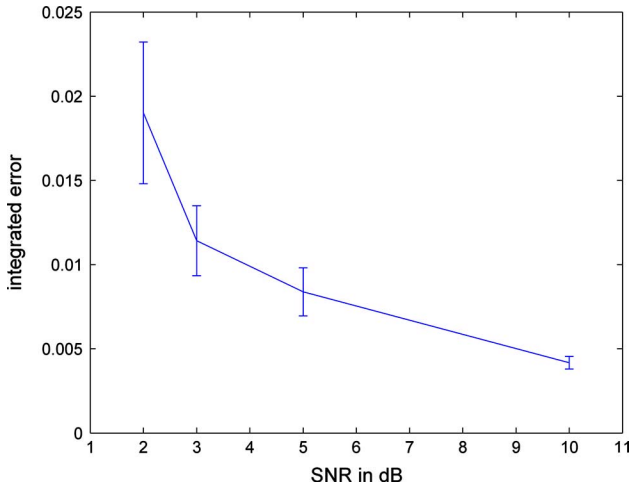

Fig. 3. SNR versus integrated error between the noiseless signal structure, and the approximation given by the principal curve. Mean error and ± 2 standard deviations are given for 100 Monte Carlo simulations.

TABLE II
PERFORMANCE EVALUATIONS USING THE NOISY DATA

	error	running time
DTW 10dB	0.0264	0.89 unit
diff.eqn. 10dB	0.0342	0.68 unit
DTW 5dB	0.0564	1 unit
diff.eqn. 5dB	0.0589	0.72 unit
DTW 3dB	0.2640	0.94 unit
diff.eqn. 3dB	0.3626	0.75 unit
DTW 2dB	0.4184	0.91 unit
diff.eqn. 2dB	0.4966	0.79 unit

note that for this simple piecewise linear case the correct time warping function is known, and presented along with the solutions. As the noise level increases, the principal curve denoising still leads to reasonably good results, whereas for the original noisy signal, the performance drops significantly.

To present results on real life data, we will use inertial measurement unit (IMU) readings collected from a wrist-worn sensor during physical exercise. Evidence has indicated that following a simple exercise routine improves quality of life for elderly individuals, and could also help to slow the progression of dementia. Providing individualized exercise direction to elders is expensive when people are the ones providing the direction. Our aim is to provide an automated interactive exercise routine for elders. The ultimate goal is being able to have an in-home automated exercise program that can detect if the subjects are performing the proper exercise they have been instructed to, and possibly if they are performing it well or not.

The IMU device measures the angular velocity and linear acceleration in x, y and z directions. Since the subjects are going to perform all these moves differently (for example, move up fast, wait for a while, and go down versus move up slowly go down fast without waiting, etc.) this is a natural application for time warping. Fig. 5 shows the particular exercise sequence that we used in the experiments.[3] Fig. 6 shows a pair of 3-channel accelerometer recordings for this exercise. In this high SNR signal

[3]The data used in this paper is not collected from real subjects of this study. This preliminary data presented here is collected by A. Ozertem while visiting our laboratory as a summer intern, by deliberately performing parts of the exercise at different speeds.

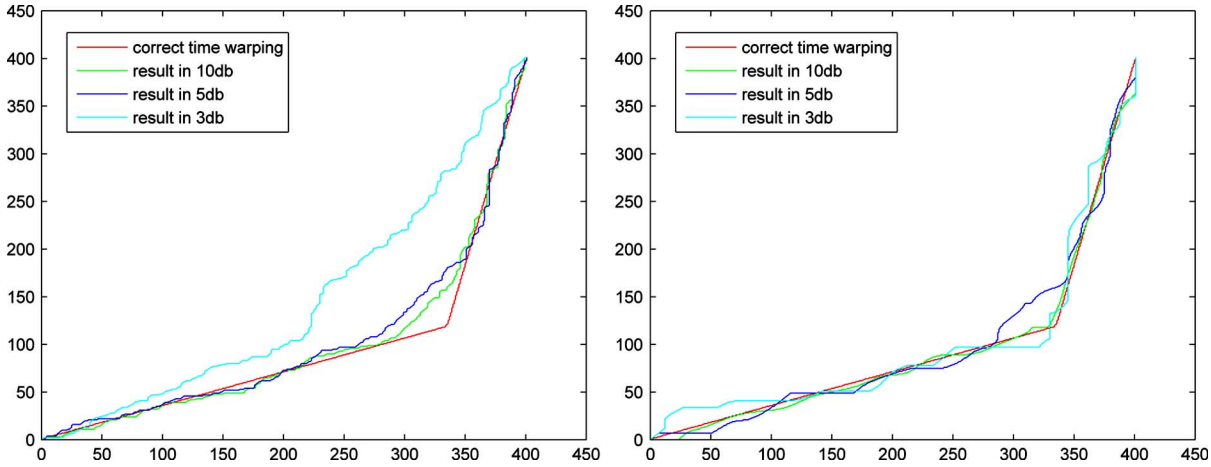

Fig. 4. Correct time warping function and solutions in different noise levels using the original data (left) and the principal curve projections (right).

Fig. 5. The exercise sequence used in the experiments.

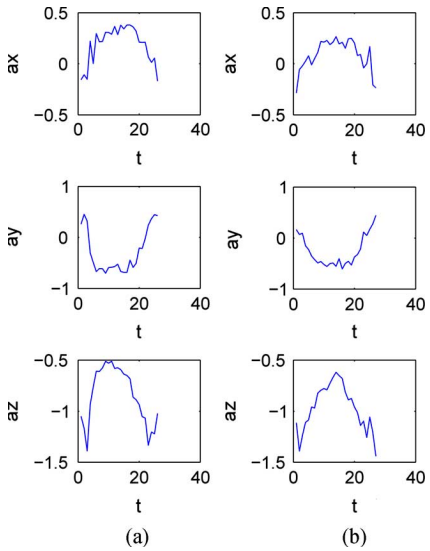

Fig. 6. IMU data for two realizations of the exercise sequence in Fig. 7.

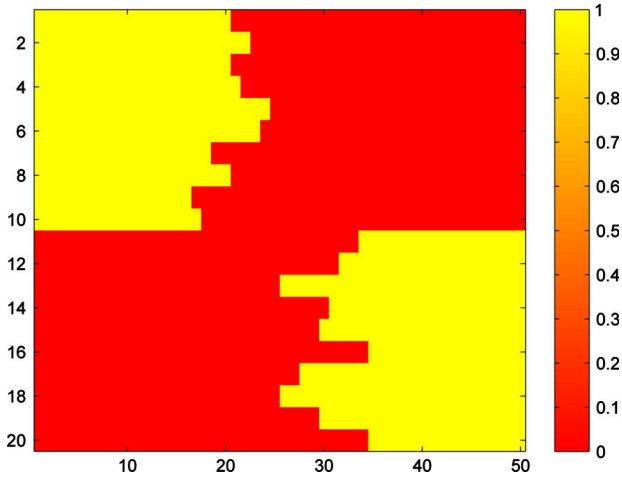

Fig. 7. Time series clustering dataset.

pair, one can see that both signals demonstrate similar characteristics, but the structure does not align in the time axis.

In this experiment, we generate 20 of these IMU recordings for the same exercise and perform the principal curve projections for each data pair—here we have a total of 190 pairs. We solve for the time warping function using both DTW and the differential equation given in Section III-B at different noise levels—we artificially add Gaussian noise to the data to present results at different SNR levels. Here we take the time warping function regarding to the noiseless IMU measurement as the ground truth and measure the performance using the integrated error between the found time warping functions and the ground truth for different noise levels.

Table II shows the results using original features along with the average computation times. Table III presents the same comparisons using the results of principal curve denoising. Here the integrated errors are normalized with the norm of the correct warping function, and the times are normalized with the longest computation time. As compared to the proposed differential equation-based solution, DTW or its derivatives in the literature provide more robust tools for evaluating the time warping function. This is obvious as one can observe from Table II, where the data becomes noisy the performance of the proposed differential equation-based method drops significantly. However, when combined with the principal curve projections as the preprocessing stage, the proposed differential equation method is able to provide similar performance in less computation time.

The reasoning behind the differential equation-based approach follows directly by observing the characteristics in $\mathbf{r}$; the derivative of the time warping function can be written in terms of the partial derivatives of the principal curve. Frankly,

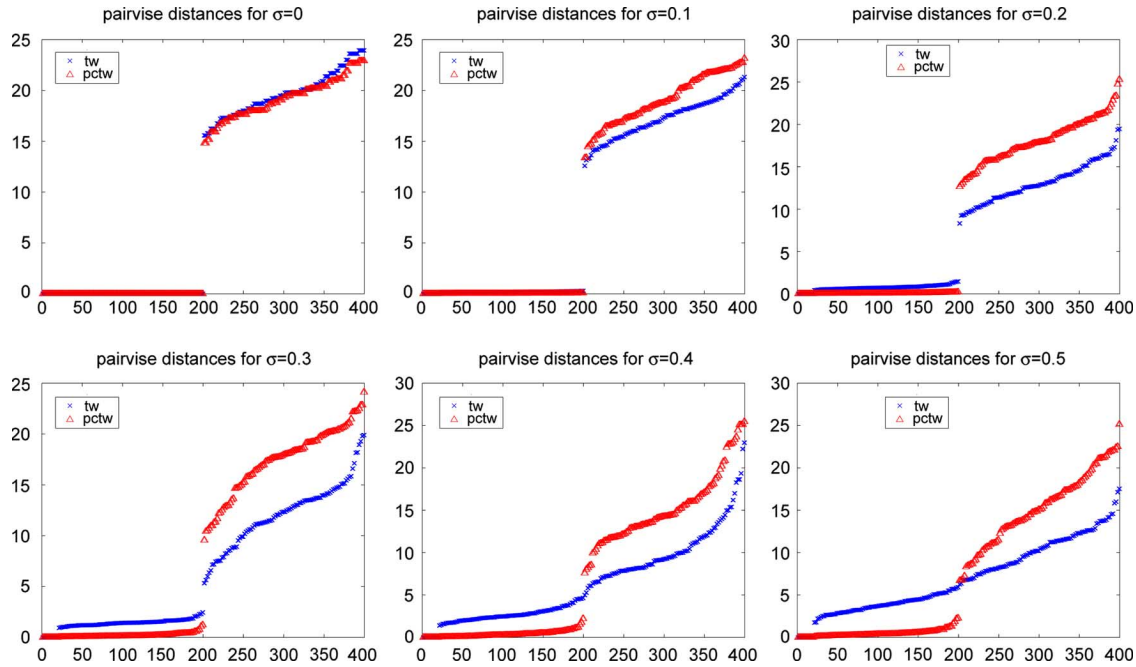

Fig. 8. Time series clustering results, where the first 200 distances are in-cluster distances and the remaining 200 are intercluster distances. Distances for DTW are presented with (red) and without (blue) principal curve denoising for different noise levels.

TABLE III
PERFORMANCE EVALUATIONS USING PRINCIPAL CURVE PROJECTIONS

	error	running time
DTW 10dB	0.0020	1.04 unit
diff.eqn. 10dB	0.0044	0.65 unit
DTW 5dB	0.0133	0.99 unit
diff.eqn. 5dB	0.0187	0.77 unit
DTW 3dB	0.1017	1.09 unit
diff.eqn. 3dB	0.1126	0.82 unit
DTW 2dB	0.1271	1.05 unit
diff.eqn. 2dB	0.1602	0.81 unit

since it suffers from error accumulation, using a differential equation-based solution may not be the best tool to implement this idea. For example, fitting a parametric curve (for example a spline) to the denoised samples may be an alternative that allows one to take the derivative using the parametric model of the curve. We leave the investigation of these ideas as future work.

C. Time Series Clustering Results

We will compare time series classification results for a synthetic dataset using DTW, with and without the principal curve denoising. To show how much the principal curve denoising changes the final classification performance, we will provide the pairwise distances for two clusters.

In this experiment we use the following two-cluster dataset

$$r_{\text{cluster1}}(t) = \begin{cases} 1 & : 0 \le t \le t_1 \\ 0 & : t_1 \le t \le 1 \end{cases} \quad t_1 \in U(0.3, 0.4)$$
$$r_{cluster2}(t) = \begin{cases} 0 & : 0 \le t \le t_2 \\ 1 & : t_2 \le t \le 1 \end{cases} \quad t_2 \in U(0.5, 0.6) \quad (17)$$

where t_1 and t_2 are uniformly distributed as shown above. Fig. 7 shows a collection of 20 signals, 10 signal for each cluster. Each realization of the signal has 50 samples.

Fig. 8 shows the pairwise distances between these signals, evaluated using DTW, with (red) and without (blue) the principal curve denoising for different noise levels. Here we present the sorted intracluster and intercluster distances, a total of 400 pairwise distances. Hence, the first 200 pairwise distances are between sample pairs of same cluster, and the second 200 represent the distances between sample pairs of different cluster. Principal curve denoising preserves the gap between intra- and intercluster distances for noisy cases as well, which is essentially what is required for good clustering results.

D. Time Series Classification Results

Very similar to previous section, we will present time series classification results using a publicly available process control dataset [28], which has 300 training and 300 test samples. In this experiment, we present the ratio of samples from the correct class among the K-nearest neighbors using Euclidean distance, and DTW distance with and without the principal curve denoising. We add Gaussian noise to both training and testing data and repeat the experiment for different noise levels and report results for different values of K.

Fig. 9 presents the times series classification results using Euclidean distance (red $\diamond$), and DTW distance with (green $\triangle$) and without (blue $\times$) the proposed principal curve denoising. One can see that for the original dataset in Fig. 9(a), principal curve denoising does not change the results significantly. We repeat the same experiment by adding Gaussian noise to the data. Results are given in Fig. 9(b) and (c). One important observation is that the proposed projections slightly degrades performance in the original dataset Fig. 9(a), which presumably due to oversmoothing of the data. However, for the noisy cases, the principal curve denoising improves the results significantly.

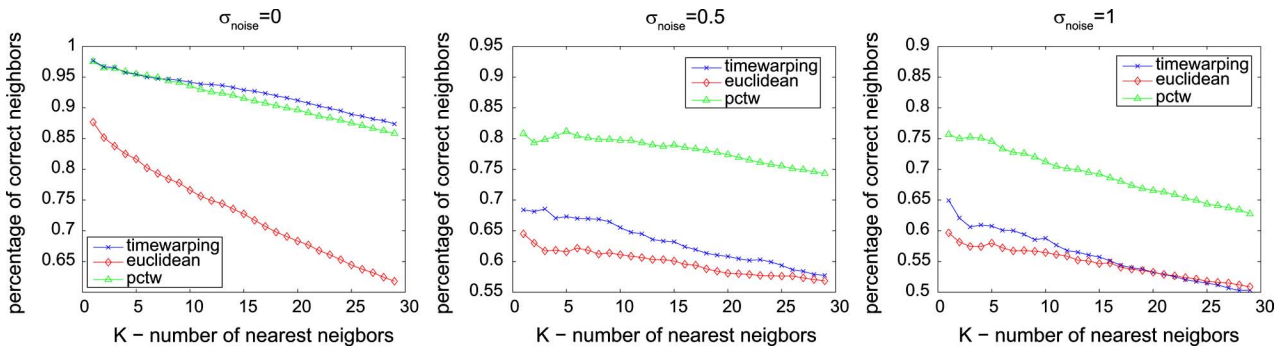

Fig. 9. Time series classification results. The percentage of correct nearest neighbors are presented for Euclidean distance (blue), and result of DTW with (green) and without (red) principal curve denoising is presented for different noise levels.

Considering the results presented in time series classification context, note that our aim is not to improve upon the performance of any particular existing time warping application. There are many techniques that use more elaborate application dependent features to achieve better results; however, such optimizations are out of the scope of this paper.

IV. Discussion

We present a robust time warping strategy based on principal curves. The principal curve projections implement a nonlinear nonparametric noise reduction filter. We derive the principal curve-based denoising under unimodal additive noise assumption. Since the proposed principal curve projection converges to the maxima of the pdf in the constrained space, unimodal noise assumption is required for theoretical analysis to have a single maxima point, hence, a single denoised signal value in the pdf for all time indices. In practice, we implement the principal curve projection by a constrained mean shift algorithm that uses Gaussian kernels. The effect of the multimodal noise can be removed by increasing the bandwidth of the Gaussian kernel. However, this may potentially lead to oversmoothing of the signal and should not be preferred without any knowledge on the frequency content of the signal.

The nonparametric implementation of the principal curve time warping technique is based on KDE. At this point, note that the definition of the principal curve is not coupled with a specific density estimation method. Hence, one can employ other density estimation techniques, if a particular method yields more advantageous characteristics in specific applications.

For example, if there are too few number of samples or if the noise level is extremely high, one obvious shortcoming of using KDE will be the possible ill-conditioning of the Hessian matrix. One can tackle this problem by imposing more structure to the density model (such as a Gaussian mixture, etc.) to reduce the statistical variance on the density estimate and fix the ill-conditioning of the Hessian. Given our KDE-based implementation, deriving the required algorithms for different density estimation methods is fairly straightforward by working out the principal curve projection directly from the definition [18]. Overall, the improvement on noise robustness not only increases the stability of existing time warping applications, but also may trigger new application areas where the signals that need to be compared are buried in noise.

Acknowledgment

The authors would like to thank K. A. Ozertem for collection and preprocessing of the ETAC exercise data.

References

[1] H. Sakoe and S. Chiba, "A dynamic programming approach to continuous speech recognition," in *Proc. Int. Congress Acoust.*, Budapest, Hungary, 1971.

[2] M. R. Sabnur and L. R. Rabiner, "A statistical decision approach to the recognition of connected digits," *IEEE. Trans. Acoust., Speech, Signal Process.*, vol. ASSP-24, pp. 550–558, 1976.

[3] J. S. Bridle, "Stochastic models and template matching: Some important relationships between two apparently different techniques in speech processing," *Proc. Inst. Acoust.*, vol. 6, pp. 452a–452h, 1984.

[4] L. Rabiner, A. Rosenberg, and S. Levinson, "Consideration in dynamic time warping algorithms for discrete word recognition," *IEEE Trans. Acoust., Speech Signal Process.*, vol. ASSP-26, pp. 575–582, 1978.

[5] D. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," in *Workshop on Knowledge Discovery in Databases*, 1994.

[6] C. Myers, L. Rabnier, and A. Rosenberg, "Performance tradeoffs in dynamic time warping algorithms isolated word recognition," *IEEE Trans. Acoust., Speech Signal Process.*, vol. ASSP-28, pp. 623–635, 1980.

[7] C. A. Ratanamahatana and E. Keogh, "Everything you know about dynamic time warping is wrong," in *Proc. Third Workshop on Mining Temporal and Sequential Data*, Seattle, WA, 2004.

[8] E. J. Keogh and M. J. Pazzani, "Derivative dynamic time warping," in *Proc. First SIAM Int. Conf. Data Mining*, 2001.

[9] E. Yaniv and D. Burshtein, "An enhanced dynamic time warping model for improved estimation of DTW parameters," *IEEE Trans. Speech Audio Process.*, vol. 11, 2003.

[10] A. Pikrakis, S. Theodoridis, and D. Kamarotos, "Recognition of isolated musical patterns using context dependent dynamic time warping," *IEEE Trans. Speech Audio Process.*, vol. 11, 2003.

[11] T. Hastie, "Principal curves and surfaces," Ph.D., Stanford Univ. , Stanford, CA, 1984.

[12] T. Hastie and W. Stuetzle, "Principal curves," *J. Amer. Statist. Assoc.*, vol. 84, pp. 502–516, 1989.

[13] R. Tibshirani, "Principal curves revisited," *Statist. Computat.*, vol. 2, pp. 183–190, 1992.

[14] S. Sandilya and S. R. Kulkarni, "Principal curves with bounded turn," *IEEE Trans. Inf. Theory*, vol. 48, no. 10, pp. 2789–2793, 2002.

[15] B. Kegl, A. Kryzak, T. Linder, and K. Zeger, "Learning and design of principal curves," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 3, pp. 281–297, 2000.

[16] B. Kegl and A. Kryzak, "Piecewise linear skeletonization using principal curves," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 1, pp. 59–74, 2002.

[17] D. C. Stanford and A. E. Raftery, "Finding curvilinear features in spatial point patterns: Principal curve clustering with noise," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 6, pp. 601–609, 2000.

[18] D. Erdogmus and U. Ozertem, "Self-consistent locally defined principal surfaces," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2007, vol. 2, pp. II549–II552.

[19] U. Ozertem and D. Erdogmus, "Local conditions for critical and principal manifolds," in *IEEE Int. Conf. Acoust., Speech Signal Process.*, 2008, pp. 1893–1896.
[20] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," *Adv. Neural Inf. Process. Syst.*, vol. 17, 2004.
[21] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: Wiley, 2000.
[22] R. Jenssen, T. Eltoft, D. Erdogmus, and J. C. Principe, "Some equivalences between kernel methods and information theoretic methods," *J. VLSI Signal Process. Syst.*, vol. 45, no. 1–2, pp. 49–65, 2006.
[23] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*. London, U.K.: Chapman and Hall, 1986.
[24] D. Comaniciu, "An algorithm for data-driven bandwidth selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 2, pp. 281–288, 2003.
[25] E. Parzen, "On the estimation of a probability density function and the mode," *Ann. Math. Statist.*, vol. 32, pp. 1065–1076, 1962.
[26] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 8, pp. 790–799, 1995.
[27] D. Comaniciu and P. Meer, "Mean shift: A robust approach towards feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, pp. 603–619, 2002.
[28] E. Keogh, X. Xi, L. Wei, and C. A. Ratanamahatana, The UCR Time Series Classification/Clustering Homepage Publicly [Online]. Available: www.cs.ucr.edu/~eamonn/time_series_data/

Umut Ozertem (M'09) received the B.S. degree in electrical engineering in 2003 from the Middle East Technical University, Ankara, Turkey. He received the M.S. and Ph.D. degrees in electrical engineering, in 2006 and 2008, respectively, from Oregon Health and Science University, Portland.

Prior to joining Yahoo! Labs, Sunnyvale, CA, he was with Intel Corp. between January-July 2007, and was a Graduate Research Assistant with the Oregon Health and Science University. His research interests include nonparametric machine learning, adaptive and statistical signal processing, information theory and its applications to signal processing, and adaptive learning algorithms.

Dr. Ozertem serves as a reviewer for Elsevier's *Signal Processing*, *Neurocomputing*, *Pattern Recognition Letters*, the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the IEEE TRANSACTIONS ON NEURAL NETWORKS, the IEEE SIGNAL PROCESSING LETTERS, the IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING, and numerous conferences.

Deniz Erdogmus (SM'08) received the B.S. degree in electrical engineering and the B.S. degree in mathematics in 1997, and the M.S. degree in electrical engineering, in 1999, all from the Middle East Technical University, Ankara, Turkey. He received the Ph.D. degree in electrical and computer engineering (ECE) from the University of Florida, Gainesville, in 2002.

He was a Postdoctoral Research Associate with the University of Florida until 2004. Prior to joining Northeastern University, Boston, MA, in 2008, where he is currently an Assistant Professor of ECE, he held an Assistant Professor position jointly with the CSEE and BME Departments of the Oregon Health and Science University, Portland. His expertise is in information theoretic and nonparametric machine learning and adaptive signal processing, specifically focusing on cognitive signal processing including brain interfaces and technologies that collaboratively improve human performance in a variety of tasks.

Dr. Erdogmus has been serving as an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the IEEE TRANSACTIONS ON NEURAL NETWORKS, the IEEE SIGNAL PROCESSING LETTERS, Elsevier's *Neurocomputing*, *Neural Processing Letters*, and Hindawi's *Computational Intelligence and Neuroscience*. He is a member of the IEEE-SPS Machine Learning for Signal Processing Technical Committee. He is a member of TBP and HKN.