Open access • Journal Article • DOI:10.1109/TPAMI.2015.2414429

Generalized Canonical Time Warping — Source link

Feng Zhou, Fernando De la Torre

Institutions: Carnegie Mellon University

Published on: 01 Feb 2016 - IEEE Transactions on Pattern Analysis and Machine Intelligence (IEEE)

Topics: Dynamic time warping, Image warping, Motion estimation and Motion capture

Generalized Canonical Time Warping

Feng Zhou and Fernando De la Torre

***Abstract*—Temporal alignment of human motion has been of recent interest due to its applications in animation, tele-rehabilitation and activity recognition. This paper presents generalized canonical time warping (GCTW), an extension of dynamic time warping (DTW) and canonical correlation analysis (CCA) for temporally aligning multi-modal sequences from multiple subjects performing similar activities. GCTW extends previous work on DTW and CCA in several ways: (1) it combines CCA with DTW to align multi-modal data (*e.g.*, video and motion capture data); (2) it extends DTW by using a more flexible path warping as combination of monotonic functions. Unlike exact DTW that has quadratic complexity, we propose a linear time algorithm to minimize GCTW. (3) GCTW allows simultaneous alignment of multiple sequences. Experimental results on aligning multi-modal data, facial expressions, motion capture data and video illustrate the benefits of GCTW. The code is available at** `http://humansensing.cs.cmu.edu/ctw`**.**

***Index Terms*—Multi-modal sequence alignment, Canonical correlation analysis, Dynamic time warping, Time warping.**

I. Introduction

Temporal alignment of multiple time series is an important problem with applications in many areas such as speech recognition [1], astronomy [2], computer graphics [3], computer vision [4], and bio-informatics [5]. In particular, alignment of human motion from sensory data has recently received increasing attention in computer vision and computer graphics to solve problems such as curve matching [6], temporal clustering [7], tele-rehabilitation [8], activity recognition [9] and motion synthesis [10], [11]. While algorithms for alignment of time series analysis have been commonly used in computer vision and computer graphics, a relatively unexplored problem has been the alignment of multi-dimensional and multi-modal time series that encode human motion. Fig. 1 illustrates the main problem addressed by this paper: how can we efficiently find the temporal correspondence between (1) the frames of a video, (2) the samples of motion capture data and (3) the samples of 3-axis accelerometers of different subjects performing a similar action (*e.g.*, kicking a ball)?

Several challenges contribute to the alignment of multi-dimensional, multi-modal time series of human motion. First, there is typically a large variation in the subjects' physical characteristics, motion style and speed of the activity. Second, large changes in view point also complicates the correspondence problem [12], [13]. Third, it is unclear how existing techniques can be used to align sets of time series that have different modalities (*e.g.*, video and motion capture data). While there is extensive literature on time series alignment and string matching [14], standard extensions of dynamic time warping (DTW) or Bayesian networks are not able to align multi-modal data. Moreover, it is unclear how existing technique can compensate for subject variability in style and morphology.

F. Zhou and F. De la Torre are with Robotics Institute, Carnegie Mellon University.

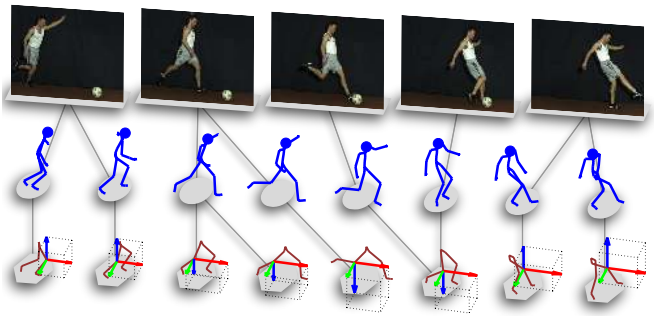

Fig. 1. Temporal alignment of three sequences recorded with different sensors (top row video, middle row motion capture and bottom row accelerometers) of three subjects kicking a ball.

To address these problems, this paper proposes generalized canonical time warping (GCTW), a technique to temporally align multi-modal time series of different subjects performing similar activities. GCTW is a spatio-temporal alignment method that temporally aligns two or more multi-dimensional and multi-modal time series by maximizing the correlation across them. GCTW can be seen as an extension of DTW or canonical correlation analysis (CCA). To accommodate for subject variability and take into account the difference in the dimensionality of the signals, GCTW uses CCA as measure of spatial correlation. GCTW extends DTW by incorporating a feature weighting mechanism to provide more importance to most correlated features and align signals of different dimensionality. It also extends DTW by parameterizing the warping path as combination of monotonic functions providing a more accurate alignment and faster optimization strategies. Unlike exact approaches based on DTW that have quadratic cost, GCTW uses a Gauss-Newton algorithm that has linear complexity in the length of the sequence. Preliminary versions of this paper were published in [15], [16].

The remainder of the paper is organized as follows. Section II reviews previous work on temporal alignment, CCA and DTW. Section III describes canonical time warping (CTW) and Section IV extends CTW to GCTW. Section V provides experimental results.

II. Previous work

This section describes previous work on temporal alignment, CCA and DTW.

A. Temporal alignment

This section discusses prior work on alignment of time series in the context of computer graphics, computer vision and data mining.

In the literature of computer graphics, temporal alignment of human motion has been commonly applied to solve problems such as content modeling [17], [18], and motion blending [3], [19], [20]. Hsu *et al.* [10] proposed iterative motion warping (IMW), a robust method that finds a spatio-temporal warping between two instances of motion captured data. Shapiro *et al.* [21] used independent component analysis to separate motion data into visually meaningful components called style components. Heloir *et al.* [22] introduced a multi-level dynamic time warping algorithm based on a weighted principal component analysis (PCA) [23]. Compared to these works, our method solves a more general problem of aligning human motion from multi-modal time series.

In the context of computer vision, temporal alignment of video captured with different cameras and view points has been a topic of interest. Rao *et al.* [24], [25] aligned trajectories of two moving points using constraints from the fundamental matrix. Junejo *et al.* [9] adopted DTW for synchronizing human actions with view changes based on a view-invariant descriptor. Compared to this work, our method simultaneously estimates the optimal spatial transformation and temporal correspondence to align video sequences. Recently, Gong and Medioni [26] proposed dynamic manifold warping to incorporate more complex spatial transformations through manifold learning. Nicolaou *et al.* [27] proposed a probabilistic extension of CTW for fusing multiple continuous expert annotations in tasks related to affective behavior.

In the field of data mining, there have been several extensions of DTW to align time series that differ in the temporal and spatial domain. Keogh and Pazzani [28], for example, used derivatives of the original signal to improve alignment with DTW. Listgarten *et al.* [2] proposed continuous profile models, a probabilistic method for simultaneously aligning and normalizing sets of time series in bio-informatics. Unlike these works, which were originally designed for aligning 1-D time series, our work addresses the more challenging problem of aligning multi-modal and multi-dimensional time series.

In the literature of manifold alignment, Ham *et al.* [29] aligned manifolds of images in a semi-supervised manner. The prior knowledge of pairwise correspondences between two sets was used to guide the graph embedding. Wang and Mahadevan [30] aligned manifolds based on an extension of the Procrustes Analysis (PA). A main benefit of this approach is that PA learns a mapping that can be applied to out-of-sample cases. In related work, Wang and Mahadevan [31] addressed the manifold alignment with no available correspondence information by learning a projection that simultaneously matches the local geometry and preserves the neighborhood relationship within each set. However, these models lack a mechanism to enforce temporal correspondence and continuity.

B. Canonical correlation analysis (CCA)

CCA [32] is a technique to extract common features from a pair of multi-variate data. Given two sets of n variables (see footnote for the notation[1]), $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_n] \in \mathbb{R}^{d_x \times n}$ and $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_n] \in \mathbb{R}^{d_y \times n}$, CCA finds the linear combinations of the variables in $\mathbf{X}$ that are most correlated with the linear combinations of the variables in $\mathbf{Y}$. Assuming zero-mean data ($\sum_i \mathbf{x}_i = \sum_j \mathbf{y}_j = 0$), regularized CCA finds a combination of the original features that minimizes the sum of the Euclidean distances between samples:

$$\min_{\{\mathbf{V}_x, \mathbf{V}_y\} \in \Phi} J_{cca} = \|\mathbf{V}_x^T \mathbf{X} - \mathbf{V}_y^T \mathbf{Y}\|_F^2 + \phi(\mathbf{V}_x) + \phi(\mathbf{V}_y), \tag{1}$$

where $\mathbf{V}_x \in \mathbb{R}^{d_x \times d}$ and $\mathbf{V}_y \in \mathbb{R}^{d_y \times d}$ denote the low-dimensional embeddings ($d \leq \min(d_x, d_y)$) for $\mathbf{X}$ and $\mathbf{Y}$ respectively. $\phi(\cdot)$ is a regularization function that penalizes the high-frequency of the embedding matrices:

$$\phi(\mathbf{V}) = \frac{\lambda}{1-\lambda} \|\mathbf{V}\|_F^2, \tag{2}$$

In order to avoid the trivial solution of $\mathbf{V}_x^T \mathbf{X}$ and $\mathbf{V}_y^T \mathbf{Y}$ being zero, CCA decorrelates the canonical variates (columns of $\mathbf{V}_x^T \mathbf{X}$ and $\mathbf{V}_y^T \mathbf{Y}$) by imposing the following orthogonal constraint on the embeddings:

$$\Phi = \Big\{ \{\mathbf{V}_x, \mathbf{V}_y\} \;\Big|\; \mathbf{V}_x^T \Big((1-\lambda)\mathbf{X}\mathbf{X}^T + \lambda \mathbf{I} \Big) \mathbf{V}_x = \mathbf{I}, \\ \mathbf{V}_y^T \Big((1-\lambda)\mathbf{Y}\mathbf{Y}^T + \lambda \mathbf{I} \Big) \mathbf{V}_y = \mathbf{I} \Big\}. \tag{3}$$

where $\lambda \in [0, 1]$ is a weight to trade-off between the least-square error and the regularization terms.

Optimizing Eq. 1 has a closed-form solution in terms of a generalized eigenvalue problem, *i.e.*, $[\mathbf{V}_x; \mathbf{V}_y] = \mathrm{eig}_d(\mathbf{A}, \mathbf{B})$, where:

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{X}\mathbf{Y}^T \\ \mathbf{Y}\mathbf{X}^T & \mathbf{0} \end{bmatrix}, \mathbf{B} = (1-\lambda) \begin{bmatrix} \mathbf{X}\mathbf{X}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{Y}\mathbf{Y}^T \end{bmatrix} + \lambda \mathbf{I}.$$

See [33] for a unification of several component analysis methods and a review of numerical techniques to efficiently solve generalized eigenvalue problems.

In computer vision, CCA has been used for matching sets of images in problems such as activity recognition from video [34] and activity correlation from cameras [35]. Recently, Fisher *et al.* [36] proposed an extension of CCA with parameterized warping functions to align protein expressions. The learned warping function is a linear combination of hyperbolic tangent functions with non-negative coefficients, ensuring monotonicity. Unlike our method, the warping function is unable to deal with feature weighting.

[1]Bold capital letters denote a matrix $\mathbf{X}$, bold lower-case letters a column vector $\mathbf{x}$. $\mathbf{x}_i$ and $\mathbf{x}^{(i)}$ represent the i^{th} column and i^{th} row of the matrix $\mathbf{X}$ respectively. x_{ij} denotes the scalar in the i^{th} row and j^{th} column of the matrix $\mathbf{X}$. All non-bold letters represent scalars. $\mathbf{1}_{m \times n}, \mathbf{0}_{m \times n} \in \mathbb{R}^{m \times n}$ are matrices of ones and zeros. $\mathbf{I}_n \in \mathbb{R}^{n \times n}$ is an identity matrix. $\|\mathbf{x}\|_p = \sqrt[p]{\sum |x_i|^p}$ denotes the p-norm. $\|\mathbf{X}\|_F^2 = \mathrm{tr}(\mathbf{X}^T\mathbf{X})$ designates the Frobenious norm. $\mathrm{vec}(\mathbf{X})$ denotes the vectorization of matrix $\mathbf{X}$. $\mathbf{X} \circ \mathbf{Y}$ is the Hadamard product of matrices. $\{i : j\}$ lists the integers, $\{i, i+1, \cdots, j-1, j\}$. $[\Rightarrow_i \mathbf{A}_i]$, $[\Downarrow_i \mathbf{A}_i]$, $[\searrow_i \mathbf{A}_i]$ are the horizontal, vertical, diagonal concatenation respectively. $\ominus$ denotes the tiled minus, *e.g.*, $\mathbf{A}_{6\times2} \ominus \mathbf{B}_{2\times2} = \mathbf{A}_{6\times2} - (\mathbf{1}_{3\times1} \otimes \mathbf{B}_{2\times2})$. $\mathrm{eig}_d(\mathbf{A}, \mathbf{B})$ denotes the top d eigenvectors $\mathbf{V}$ that solve the generalized eigenvalue problem $\mathbf{AV} = \mathbf{BV}\Lambda$.

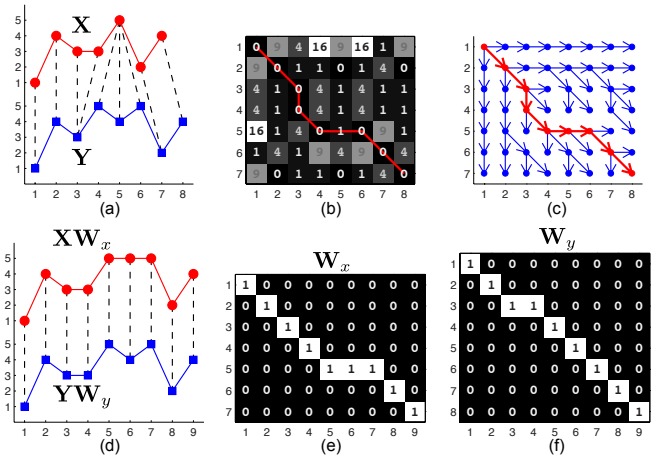

Fig. 2. An example of DTW for aligning time series. (a) Two 1-D time series ($n_x = 7$ and $n_y = 8$) and the optimal alignment between samples computed by DTW. (b) Euclidean distances between samples, where the red curve denotes the optimal warping path ($l = 9$). (c) DP policy at each pair of samples, where the three arrow directions, $\downarrow, \searrow, \rightarrow$, denote the policy, $\pi(\cdot,\cdot) \in \{[1,0],[1,1],[0,1]\}$, respectively. (d) A matrix-form interpretation of DTW as stretching the two time series in matrix products. (e) Warping matrix $\mathbf{W}_x$. (f) Warping matrix $\mathbf{W}_y$.

C. Dynamic time warping (DTW)

Given two time series, $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_{n_x}] \in \mathbb{R}^{d \times n_x}$ and $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_{n_y}] \in \mathbb{R}^{d \times n_y}$, DTW [1] is a technique to align $\mathbf{X}$ and $\mathbf{Y}$ such that the sum of the Euclidean distances between the aligned samples is minimized. DTW minimizes

$$\min_{\{\mathbf{p}_x, \mathbf{p}_y\} \in \Psi} J_{dtw} = \sum_{t=1}^{l} \|\mathbf{x}_{p_t^x} - \mathbf{y}_{p_t^y}\|_2^2, \tag{4}$$

where $l \geq \max(n_x, n_y)$ is the number of indexes to align the samples. The optimal l is automatically selected by the DTW algorithm. The warping paths, $\mathbf{p}_x \in \{1 : n_x\}^l$ and $\mathbf{p}_y \in \{1 : n_y\}^l$, denote the composition of alignment in frames. The i^{th} frame in $\mathbf{X}$ and the j^{th} frame in $\mathbf{Y}$ are aligned if there exists $p_t^x = i$ and $p_t^y = j$ at some step t.

In order to find a polynomial time solution, the warping paths have to satisfy three constraints:

$$\begin{aligned}
&\Psi = \Big\{ \{\mathbf{p}_x, \mathbf{p}_y\} \;\Big|\; \mathbf{p}_x \in \{1 : n_x\}^l \text{ and } \mathbf{p}_y \in \{1 : n_y\}^l, \\
&\textbf{Boundary:} \quad [p_1^x, p_1^y] = [1,1] \text{ and } [p_l^x, p_l^y] = [n_x, n_y], \\
&\textbf{Monotonicity:} \quad t_1 \geq t_2 \Rightarrow p_{t_1}^x \geq p_{t_2}^x \text{ and } p_{t_1}^y \geq p_{t_2}^y, \\
&\textbf{Continuity:} \quad [p_t^x, p_t^y] - [p_{t-1}^x, p_{t-1}^y] \in \{[0,1],[1,0],[1,1]\} \Big\}.
\end{aligned} \tag{5}$$

The choice of step size in the continuity constraint is not unique. For instance, replacing the step size by $\{[2,1],[1,2],[1,1]\}$ can avoid the degenerated case in which a single frame of one sequence is assigned to many consecutive frames in the other sequence. See [1] for an extensive review on several DTW's modifications to control the warping paths.

Although the number of possible ways to align $\mathbf{X}$ and $\mathbf{Y}$ is exponential in n_x and n_y, dynamic programming (DP) [37] offers an efficient approach with complexity of $O(n_x n_y)$ to minimize J_{dtw} using Bellman's equation:

$$J^*(p_t^x, p_t^y) = \min_{\pi(p_t^x, p_t^y)} \|\mathbf{x}_{p_t^x} - \mathbf{y}_{p_t^y}\|_2^2 + J^*(p_{t+1}^x, p_{t+1}^y),$$

where the cost-to-go value function, $J^*(p_t^x, p_t^y)$, represents the cost remaining starting at t^{th} step using the optimum policy π^*. The policy function, $\pi(\cdot,\cdot) : \{1 : n_x\} \times \{1 : n_y\} \rightarrow \{[1,0],[0,1],[1,1]\}$, defines the deterministic transition between consecutive steps, $[p_{t+1}^x, p_{t+1}^y] = [p_t^x, p_t^y] + \pi(p_t^x, p_t^y)$. Once the policy queue is known, the alignment steps can be recursively selected by backtracking, $p_l^x = n_x$ and $p_l^y = n_y$.

Fig. 2a shows an example of DTW for aligning two 1-D time series. Fig. 2b illustrates the Euclidean distance of each pair of samples. To compute the optimal warping path, DP efficiently enumerates all possible steps as in Fig. 2c from the upper-left corner to the bottom-right one. At the end, the optimal alignment (denoted as the red curve) can be computed by iteratively tracing back along the arrows.

Given two sequences of length n_x and n_y, exact DTW has a computational cost in space and time of $O(n_x n_y)$. In practice, various modifications [1] on the step size, local weights and global constraints have been proposed to speed up DTW computation as well as to better control the possible routes of the warping paths. In recent work [22], [38], [39], a multi-scale searching scheme has been shown to effectively generate a speedup of one to three orders of magnitude, compared to the classic DTW algorithm. More recently, Rakthanmanon *et al.* [40] have shown that DTW for mining 1-D sub-sequences can be scaled up to very large datasets using early-abandoning and cascading lower bounds. However, most of these works are originally designed for 1-D time series. Compared to these works, our method can be applied to handle more general multi-dimensional sequences and align signals of different dimensionality.

III. Canonical Time Warping (CTW)

DTW lacks a feature weighting mechanism and thus it cannot be directly used to align multi-modal sequences (*e.g.*, video and motion capture) with different features. To address this issue, this section presents CTW, a unified framework that combines DTW with CCA.

A. Least-squares formulation for DTW

In order to have a compact and compressible energy function for CTW, it is important to notice that the original objective of DTW (Eq. 4) can be reformulated in matrix form as

$$\min_{\{\mathbf{p}_x, \mathbf{p}_y\} \in \Psi} J_{dtw} = \|\mathbf{X}\mathbf{W}_x - \mathbf{Y}\mathbf{W}_y\|_F^2, \tag{6}$$

where $\mathbf{X} \in \mathbb{R}^{d \times n_x}$ and $\mathbf{Y} \in \mathbb{R}^{d \times n_y}$ denote the two time series needed to be aligned. $\mathbf{W}_x = \mathbf{W}(\mathbf{p}_x) \in \{0,1\}^{n_x \times l}$ and $\mathbf{W}_y = \mathbf{W}(\mathbf{p}_y) \in \{0,1\}^{n_y \times l}$ are two binary warping matrices associated with the warping paths by a non-linear mapping, $\mathbf{W}(\mathbf{p}) : \{1 : n\}^l \rightarrow \{0,1\}^{n \times l}$, which sets $w_{p_t,t} = 1$ for any step $t \in \{1 : l\}$ and zero otherwise. These warping matrices $\mathbf{W}_x$ and $\mathbf{W}_y$ can only replicate (multiple times) samples of the original sequences $\mathbf{X}$ and $\mathbf{Y}$. For instance, Fig. 2d illustrates that the DTW alignment in Fig. 2a can be equivalently interpreted as stretching

the two time series $\mathbf{X}$ and $\mathbf{Y}$ by multiplying the warping matrices $\mathbf{W}_x$ and $\mathbf{W}_y$ as shown in Fig. 2e-f, respectively. Observe that Eq. 6 is very similar to the CCA's objective (Eq. 1). CCA applies a linear transformation to combine the rows (features), while DTW applies binary transformations to replicate the columns (time).

B. Objective function of CTW

In order to accommodate for differences in style and subject variability, add a feature selection mechanism, and reduce the dimensionality of the signals, CTW adds a linear transformation ($\mathbf{V}_x \in \mathbb{R}^{d_x \times d}$ and $\mathbf{V}_y \in \mathbb{R}^{d_y \times d}$) as CCA to the least-square form of DTW (Eq. 6). Moreover, this transformation allows aligning temporal signals with different dimensionality (*e.g.*, video and motion capture). In a nutshell, CTW combines DTW and CCA by minimizing:

$$\min_{\{\mathbf{V}_x,\mathbf{V}_y\}\in\Phi,\{\mathbf{p}_x,\mathbf{p}_y\}\in\Psi} J_{ctw} = \|\mathbf{V}_x^T\mathbf{X}\mathbf{W}_x - \mathbf{V}_y^T\mathbf{Y}\mathbf{W}_y\|_F^2 + \phi(\mathbf{V}_x) + \phi(\mathbf{V}_y), \quad (7)$$

where $\mathbf{V}_x \in \mathbb{R}^{d_x \times d}$ and $\mathbf{V}_y \in \mathbb{R}^{d_y \times d}$ parameterize the spatial transformation and project the sequences into the same low-dimensional coordinate system. Constrained by Eq. 5, $\mathbf{W}_x$ and $\mathbf{W}_y$ warp the signal in time to achieve optimum temporal alignment. Similar to CCA, $\phi(\cdot)$ is a regularization term (Eq. 2) for $\mathbf{V}_x$ and $\mathbf{V}_y$. In addition, the projections have to satisfy the orthogonal constraints, *i.e.*,

$$\Phi = \Big\{\{\mathbf{V}_x,\mathbf{V}_y\} \;\Big|\; \mathbf{V}_x^T\Big((1-\lambda)\mathbf{X}\mathbf{W}_x\mathbf{W}_x^T\mathbf{X}^T + \lambda\mathbf{I}\Big)\mathbf{V}_x = \mathbf{I}, \; \mathbf{V}_y^T\Big((1-\lambda)\mathbf{Y}\mathbf{W}_y\mathbf{W}_y^T\mathbf{Y}^T + \lambda\mathbf{I}\Big)\mathbf{V}_y = \mathbf{I}\Big\},$$

where $\lambda \in [0,1]$ is a weight to trade-off between the least-square error and the regularization term.

Eq. 7 is the main contribution of this paper. CTW is a direct and clean extension of CCA and DTW to align two signals $\mathbf{X}$ and $\mathbf{Y}$ in space and time. It extends previous work on CCA by adding temporal alignment and on DTW by allowing a feature selection and dimensionality reduction mechanism for aligning signals of different dimensions.

C. Optimization of CTW

Optimizing J_{ctw} is a non-convex optimization problem with respect to the warping matrices and projection matrices. We take a coordinate-descent approach that alternates between solving the temporal alignment using DTW, and computing the spatial projections using CCA.

Given the warping matrices, the optimal projection matrices are the leading d generalized eigenvectors, *i.e.*, $[\mathbf{V}_x;\mathbf{V}_y] = \text{eig}_d(\mathbf{A},\mathbf{B})$, where:

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{X}\mathbf{W}_x\mathbf{W}_y^T\mathbf{Y}^T \\ \mathbf{Y}\mathbf{W}_y\mathbf{W}_x^T\mathbf{X}^T & \mathbf{0} \end{bmatrix},$$

$$\mathbf{B} = (1-\lambda)\begin{bmatrix} \mathbf{X}\mathbf{W}_x\mathbf{W}_x^T\mathbf{X}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{Y}\mathbf{W}_y\mathbf{W}_y^T\mathbf{Y}^T \end{bmatrix} + \lambda\mathbf{I}.$$

The dimension d can be selected to preserve a certain amount (*e.g.*, 90%) of the total correlation. Once the spatial transformation is computed, the temporal alignment is computed using standard approaches for DTW. Alternating between these two steps (spatial and temporal alignment) monotonically decreases J_{ctw}. J_{ctw} is bounded bellow and the proposed algorithm will converge to a critical point.

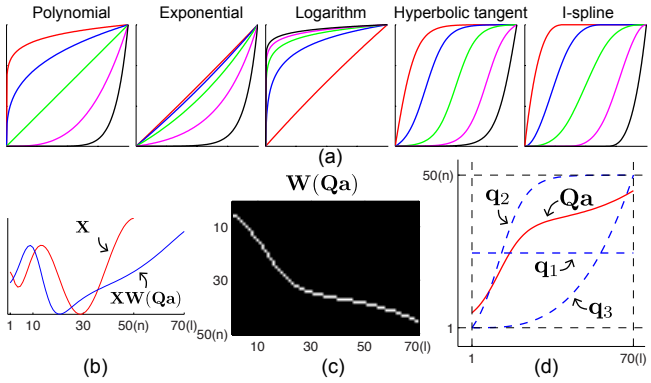

Fig. 3. Approximating temporal warping using monotone bases. (a) Five common choices for monotone bases. (b) An example of time warping $\mathbf{X}\mathbf{W}(\mathbf{Q}\mathbf{a}) \in \mathbb{R}^{1\times 70}$ of 1-D time series $\mathbf{X} \in \mathbb{R}^{1\times 50}$. (c) The warping matrix $\mathbf{W}(\mathbf{Q}\mathbf{a})$. (d) The warping function $\mathbf{Q}\mathbf{a}$ is a linear combination of three basis functions including a constant function ($\mathbf{q}_1$) and two monotonically increasing functions ($\mathbf{q}_2$ and $\mathbf{q}_3$).

IV. Generalized Canonical Time Warping (GCTW)

In the previous section, we described CTW for aligning two multi-modal sequences with different features. However, CTW has three main limitations inherited from DTW: (1) The exact computational complexity of DTW for multi-dimensional sequences is quadratic both in space and time; (2) CTW and its extensions address the problem of aligning two sequences, but it is unclear how to extend it to the alignment of multiple sequences; (3) The temporal alignment is computed using DTW, which relies on DP to find the optimal path. In some problems (*e.g.*, sub-sequence alignment) the warping path provided by DP is too rigid (*e.g.*, the first and the last samples have to match).

To address these issues, this section proposes GCTW, an efficient technique for spatio-temporal alignment of multiple time series. To accommodate for subject variability and to take into account the difference in the dimensionality of the signals, GCTW uses multi-set canonical correlation analysis (mCCA). To compensate for temporal changes, GCTW extends DTW by incorporating a more efficient and flexible temporal warping parameterized by a set of monotonic basis functions. Unlike existing approaches based on DP with quadratic complexity, GCTW efficiently optimizes the time warping function using a Gauss-Newton algorithm, which has linear complexity in the length of the sequence.

A. Objective function of GCTW

Given a collection of m time series, $\{\mathbf{X}_i\}_{i=1}^m$, GCTW aims to seek for each $\mathbf{X}_i = [\mathbf{x}_1^i, \cdots, \mathbf{x}_{n_i}^i] \in \mathbb{R}^{d_i \times n_i}$, a low-dimensional spatial embedding $\mathbf{V}_i \in \mathbb{R}^{d_i \times d}$ and a non-linear temporal transformation $\mathbf{W}_i = \mathbf{W}(\mathbf{p}_i) \in \{0,1\}^{n_i \times l}$ parameterized by $\mathbf{p}_i \in \{1 : n_i\}^l$, such that the resulting sequence $\mathbf{V}_i^T\mathbf{X}_i\mathbf{W}_i \in \mathbb{R}^{d\times l}$ is well aligned with the others in the least-squares sense. In a nutshell, GCTW minimizes

the sum of pairwise distances between the sequences:

$$\min_{\{\mathbf{V}_i\}_i\in\Phi,\{\mathbf{p}_i\}_i\in\Psi} J_{gctw} = \sum_{i=1}^{m}\sum_{j=1}^{m}\frac{1}{2}\|\mathbf{V}_i^T\mathbf{X}_i\mathbf{W}_i - \mathbf{V}_j^T\mathbf{X}_j\mathbf{W}_j\|_F^2 + \sum_{i=1}^{m}\Big(\phi(\mathbf{V}_i)+\psi(\mathbf{p}_i)\Big), \quad (8)$$

where $\phi(\cdot)$ is the regularization function penalizing the irregularity of the spatial transformation $\mathbf{V}_i$, *i.e.*,

$$\phi(\mathbf{V}_i) = \frac{m\lambda}{1-\lambda}\|\mathbf{V}_i\|_F^2,$$

where $\lambda \in [0,1]$ is a parameter to trade-off between the least-square error and the regularization term. Following the multi-set canonical correlation analysis (mCCA) [41], GCTW constrains the spatial embeddings as:

$$\Phi = \Big\{\{\mathbf{V}_i\}_i \;\Big|\; \sum_{i=1}^{m}\mathbf{V}_i^T\Big((1-\lambda)\mathbf{X}_i\mathbf{W}_i\mathbf{W}_i^T\mathbf{X}_i^T + \lambda\mathbf{I}\Big)\mathbf{V}_i = \mathbf{I}\Big\}.$$

$\psi(\cdot)$ and $\Psi(\cdot)$ defined in the following sections, are used to regularize and constrain the temporal transformation $\mathbf{p}_i$ respectively.

To be concise in notation, let us consider a single sequence $\mathbf{X} \in \mathbb{R}^{d\times n}$ and its temporal warping, $\mathbf{p} \in \{1:n\}^l$. While the possible composition of the temporal warping $\mathbf{p}$ is locally enforced by the original DTW constraints (Eq. 5), the global shape of any valid $\mathbf{p}$ must correspond to a monotonic and continuous trajectory in matrix $\mathbf{W} \in \{0,1\}^{n\times l}$ starting from the upper-left corner and ending at the bottom-right one. Since any positive combination of monotonic trajectories is guaranteed to be monotonic. GCTW parameterizes the warping path $\mathbf{p}$ as a linear combination of monotonic functions, that is:

$$\mathbf{p} \approx \sum_{c=1}^{k} a_c\mathbf{q}_c = \mathbf{Q}\mathbf{a}, \quad (9)$$

where $\mathbf{a} \in \mathbb{R}^k$ is the non-negative weight vector and $\mathbf{Q} = [\mathbf{q}_1,\cdots,\mathbf{q}_k] \in [1,n]^{l\times k}$ is the basis set composed of k pre-defined monotonically increasing functions. Fig. 3a illustrates five common choices for $\mathbf{q}_c$, including (1) polynomial function (ax^b), (2) exponential function ($\exp(ax+b)$), (3) logarithm function ($\log(ax+b)$), (4) hyperbolic tangent function ($\tanh(ax+b)$) and (5) I-spline [42]. Similar work by Fisher *et al.* [36] also used hyperbolic tangent functions as temporal bases, and the weights were optimized using a non-negative least squares algorithm. However, GCTW differs in three aspects: (1) GCTW allows aligning multi-dimensional time series that have different features, while [36] can only align one-dimensional time-series; (2) we use a more efficient eigen decomposition to solve mCCA and quadratic programming for optimizing the weights; and (3) we use a family of monotonic functions that allow for a more general warping (*e.g.*, sub-sequence alignment), and constraints to regularize the solution.

To approximate the DTW constraints (Eq. 5) on the warping path $\mathbf{p}$, we alternatively impose the following constraints on the weights $\mathbf{a}$.

Boundary conditions: We enforce the position of the first frame, $p_1 = \mathbf{q}^{(1)}\mathbf{a} \geq 1$, and the last frame, $p_l = \mathbf{q}^{(l)}\mathbf{a} \leq n$, where $\mathbf{q}^{(1)} \in \mathbb{R}^{1\times k}$ and $\mathbf{q}^{(l)} \in \mathbb{R}^{1\times k}$ to be the first and last rows of the basis matrix $\mathbf{Q}$ respectively. In contrast to DTW, which imposes a tight boundary (*i.e.*, $p_1 = 1$ and $p_l = n$), GCTW relaxes the equality with inequality constraints to allow $\mathbf{p}$ to index a sub-part of $\mathbf{X}$. This relaxation is useful in solving the more general problem of sub-sequence alignment. For instance, Fig. 4 illustrates an example of using GCTW for placing two 1-D time series in correspondence. In particular, the shorter blue sequence can only be partially matched to a sub-sequence of the longer red one. In this sub-sequence alignment problem, GCTW models the time warping $\mathbf{p}$ as a combination of a linear basis $\mathbf{q}_1$ and a constant one $\mathbf{q}_2$.

Monotonicity: We enforce $t_1 \leq t_2 \Rightarrow p_{t_1} \leq p_{t_2}$ by constraining the sign of the weight: $\mathbf{a} \geq \mathbf{0}$. Notice that constraining the weights is only a sufficient condition to ensure monotonicity but it is not necessary. See [43]–[45] for in-depth discussions on monotonic functions.

Continuity: To approximate the hard constraint on the step size (*e.g.*, $p_t - p_{t-1} \in \{0,1\}$), GCTW softly penalizes the curvature of the warping path, $\sum_{t=1}^{l}\|\nabla\mathbf{q}^{(t)}\mathbf{a}\|_2^2 \approx \|\mathbf{F}_l\mathbf{Q}\mathbf{a}\|_2^2$ where $\mathbf{F}_l \in \mathbb{R}^{l\times l}$ is the 1^{st} order differential operator. For instance, for any $\mathbf{X} \in \mathbb{R}^{d\times 4}$, the Matlab function, $gradient(\mathbf{X})$, computes the same value as $\mathbf{X}\mathbf{F}_4^T$, where:

$$\mathbf{F}_4 = \begin{bmatrix} 1 & \frac{1}{2} & 0 & 0 \\ -1 & 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{2} & 0 & 1 \\ 0 & 0 & -\frac{1}{2} & -1 \end{bmatrix}. \quad (10)$$

In summary, we constrain the warping path in Eq. 8 by the following constraints on $\mathbf{a}$ as:

$$\psi(\mathbf{a}) = \eta\|\mathbf{F}_l\mathbf{Q}\mathbf{a}\|_2^2, \quad \Psi = \{\mathbf{a} \mid \mathbf{L}\mathbf{a} \leq \mathbf{b}\}, \quad (11)$$
$$\text{where } \mathbf{L} = \begin{bmatrix} -\mathbf{I}_k \\ -\mathbf{q}^{(1)} \\ \mathbf{q}^{(l)} \end{bmatrix} \quad \text{and} \quad \mathbf{b} = \begin{bmatrix} \mathbf{0}_k \\ -1 \\ n \end{bmatrix}.$$

Therefore, given a basis set of k monotone functions, all feasible weights belong to a polyhedron in $\mathbb{R}^k$ parameterized by $\mathbf{L} \in \mathbb{R}^{(k+2)\times k}$ and $\mathbf{b} \in \mathbb{R}^{k+2}$. For instance, Fig. 3b illustrates an example of a warping function (red solid line) as a combination of three monotone functions (blue dotted lines).

B. Optimization of GCTW w.r.t. spatial basis

Minimizing J_{gctw} (Eq. 8) is a non-convex optimization problem with respect to the temporal transformation and the spatial projection. We optimize GCTW by alternating between solving for time warping using an efficient Gauss-Newton algorithm (discussed in the following section), and computing the spatial transformation using mCCA.

Assuming the time warping is fixed, mCCA computes the optimal $\{\mathbf{V}_i\}_i$ in closed form using a generalized eigen decomposition, *i.e.*, $[\Downarrow_i \mathbf{V}_i] = \text{eig}_d(\mathbf{A},\mathbf{B})$, where:

$$\mathbf{A} = [\Downarrow_i \mathbf{X}_i\mathbf{W}_i][\Rightarrow_j \mathbf{W}_j^T\mathbf{X}_j^T] - [\searrow_i \mathbf{X}_i\mathbf{W}_i\mathbf{W}_i^T\mathbf{X}_i^T],$$
$$\mathbf{B} = (1-\lambda)[\searrow_i \mathbf{X}_i\mathbf{W}_i\mathbf{W}_i^T\mathbf{X}_i^T] + \lambda\mathbf{I}.$$

These steps monotonically decrease J_{gctw}, and because the function is bounded below, the alternating scheme will converge to a critical point.

C. Optimization of GCTW w.r.t. temporal weights

By relaxing the warping path as a linear combination of monotonic paths, Eq. 9 provide a new model for temporal alignment and new methods for optimizing it. Given k basis functions[2], $\mathbf{Q} \in \mathbb{R}^{l\times k}$, optimizing Eq. 8 with respect to the warping paths $\{\mathbf{p}_i\}_i$ can be approximated as:

$$\begin{aligned}\min_{\{\mathbf{a}_i\}_i} J_a = &\sum_{i=1}^{m}\sum_{j=1}^{m}\frac{1}{2}\|\mathbf{V}_i^T\mathbf{X}_i\mathbf{W}(\mathbf{Q}\mathbf{a}_i) - \mathbf{V}_j^T\mathbf{X}_j\mathbf{W}(\mathbf{Q}\mathbf{a}_j)\|_F^2 \\ &+ \sum_{i=1}^{m}\eta\|\mathbf{F}_l\mathbf{Q}\mathbf{a}_i\|_2^2, \\ \text{s.t.}\quad &\mathbf{L}_i\mathbf{a}_i \le \mathbf{b}_i,\ \forall i \in \{1:m\},\end{aligned} \tag{12}$$

where $\mathbf{L}_i$ and $\mathbf{b}_i$ are used to constrain the monotonicity and boundary of the time warping for each sequence (Eq. 11).

A direct optimization of J_a is difficult due to the non-linear function $\mathbf{W}(\cdot)$. Inspired by the Lucas-Kanade framework [46] for image alignment, we approximate J_a by solving a series convex problems using a Gauss-Newton method. More specifically, we linearize J_a by performing a first-order Taylor approximation on each term, $\mathbf{Z}(\mathbf{a}_i+\boldsymbol{\delta}_i) \doteq \mathbf{V}_i^T\mathbf{X}_i\mathbf{W}(\mathbf{Q}(\mathbf{a}_i+\boldsymbol{\delta}_i)) \in \mathbb{R}^{d\times l}$, with respect to the increment $\boldsymbol{\delta}_i \in \mathbb{R}^k$. To simplify the discussion, let us focus on, $\mathbf{z}_t(\mathbf{a}_i) \doteq [\mathbf{V}_i^T\mathbf{X}_i\mathbf{W}(\mathbf{Q}\mathbf{a}_i)]_t \in \mathbb{R}^d$, the t^{th} column of $\mathbf{V}_i^T\mathbf{X}_i\mathbf{W}(\mathbf{Q}\mathbf{a}_i) \in \mathbb{R}^{d\times l}$. Given the non-linear function $\mathbf{W}(\cdot)$ specified by the warping path $\mathbf{Q}\mathbf{a}_i \in \mathbb{R}^l$, $\mathbf{z}_t(\mathbf{a}_i)$ is the replication of j^{th} column of the signals $\mathbf{V}_i^T\mathbf{X}_i$, *i.e.*,

$$\mathbf{z}_t(\mathbf{a}_i) = [\mathbf{V}_i^T\mathbf{X}_i]_{\mathbf{q}^{(t)}\mathbf{a}_i}, \tag{13}$$

where $\mathbf{q}^{(t)}$ is the t^{th} row of $\mathbf{Q}$ and $\mathbf{q}^{(t)}\mathbf{a}_i$ is the t^{th} element of $\mathbf{Q}\mathbf{a}_i$. Based on this fact, we can break down the linear approximation of $\mathbf{Z}(\mathbf{a}_i+\delta_i)$ by each of its columns, *i.e.*,

$$\mathbf{z}_t(\mathbf{a}_i+\delta_i) \approx \mathbf{z}_t(\mathbf{a}_i) + \nabla\Big([\mathbf{V}_i^T\mathbf{X}_i]_{\mathbf{q}^{(t)}\mathbf{a}_i}\Big)\frac{\partial\mathbf{q}^{(t)}\mathbf{a}_i}{\partial\mathbf{a}_i}\delta_i. \tag{14}$$

Putting all the columns of $\mathbf{Z}(\mathbf{a}_i+\delta_i)$ together yields:

$$\text{vec}\Big(\mathbf{V}_i^T\mathbf{X}_i\mathbf{W}(\mathbf{Q}(\mathbf{a}_i+\boldsymbol{\delta}_i))\Big) \approx \mathbf{v}_i + \mathbf{G}_i\delta_i, \tag{15}$$

where $\mathbf{v}_i = \text{vec}\Big(\mathbf{V}_i^T\mathbf{X}_i\mathbf{W}(\mathbf{Q}\mathbf{a}_i)\Big)$, $\mathbf{G}_i = [\Downarrow_t \nabla\Big([\mathbf{V}_i^T\mathbf{X}_i]_{\mathbf{q}^{(t)}\mathbf{a}_i}\Big)\mathbf{q}^{(t)}]$.

Plugging Eq. 15 in Eq. 12 yields:

$$\begin{aligned}J_a \approx &\sum_{i=1}^{m}\sum_{j=1}^{m}\frac{1}{2}\|\mathbf{v}_i + \mathbf{G}_i\boldsymbol{\delta}_i - \mathbf{v}_j - \mathbf{G}_j\boldsymbol{\delta}_j\|_2^2 \\ &+ \sum_{i=1}^{m}\eta\|\mathbf{F}_i\mathbf{Q}(\mathbf{a}_i+\boldsymbol{\delta}_i)\|_2^2.\end{aligned} \tag{16}$$

Minimizing Eq. 16 with respect to all the weight increments $\boldsymbol{\delta} = [\Downarrow_i \boldsymbol{\delta}_i] \in \mathbb{R}^{mk}$ yields a quadratic programming problem:

$$\min_{\boldsymbol{\delta}} \quad \frac{1}{2}\boldsymbol{\delta}^T\mathbf{H}\boldsymbol{\delta} + \mathbf{f}^T\boldsymbol{\delta}, \quad \text{s.t.} \quad \mathbf{L}\boldsymbol{\delta} \le \mathbf{b} - \mathbf{L}\mathbf{a}, \tag{17}$$

whose components are computed as follows:

$$\begin{aligned}&\mathbf{H} = m[\searrow_i \mathbf{G}_i^T\mathbf{G}_i] - [\Downarrow_i \mathbf{G}_i^T][\Rightarrow_j \mathbf{G}_j] + [\searrow_i \eta\mathbf{Q}^T\mathbf{F}_i^T\mathbf{F}_i\mathbf{Q}], \\ &\mathbf{f} = [\searrow_i \mathbf{G}_i^T]\Big(m[\Downarrow_i \mathbf{v}_i] \ominus [\Rightarrow_i \mathbf{v}_i]\mathbf{1}_m\Big) + [\Downarrow_i \eta\mathbf{Q}^T\mathbf{F}_i^T\mathbf{F}_i\mathbf{Q}\mathbf{a}_i], \\ &\mathbf{a} = [\Downarrow_i \mathbf{a}_i],\ \mathbf{b} = [\Downarrow_i \mathbf{b}_i],\ \mathbf{L} = [\searrow_j \mathbf{L}_i].\end{aligned}$$

[2]Strictly speaking, sequences of different lengths (n_i) should be associated with different bases $\mathbf{Q}_i \in \{1:n_i\}^{l\times k}$. To be concise in notation, we remove the subscript i from $\mathbf{Q}_i$.

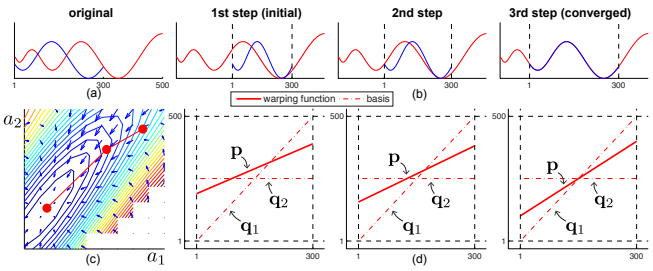

Fig. 4. An example of using Gauss-Newton for solving the sub-sequence alignment problem. (a) Two 1-D sequences. (b) The Gauss-Newton optimization procedure, the longer red sequence is warped to match the shorter blue sequence. (c) The contour of the objective function (J_a as defined in Eq. 12) with respect to the weights of two bases. (d) Warping function ($\mathbf{p}$) as a combination of a linear function ($\mathbf{q}_1$) and a constant function ($\mathbf{q}_2$) used for scaling and translation respectively.

Observe that the objective function of Eq. 17 is convex. Fig. 4 illustrates an example of aligning two 1-D time series (Fig. 4a) using this approach. To achieve sub-sequence alignment, we model the time warping path $\mathbf{p}$ as a combination of a linear basis $\mathbf{q}_1$ and a constant one $\mathbf{q}_2$ (Fig. 4d). As shown in Fig. 4b, Gauss-Newton takes three steps to find the optimal warping parameter in a 2-D space (Fig. 4c).

In all our experiments, we initialized $\mathbf{a}_i$ by uniformly aligning the sequences (the curve of GN-Init in Fig. 5b). The length of the warping path l is usually set to $l = 1.1\max_{i=1}^{m} n_i$. In practice, when the sequence length n_i is very large, an additional pre-conditioner should be used to obtain a numerically stable solution. For instance, a normalized version of Eq. 17 minimizes $\frac{1}{2}\boldsymbol{\delta}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{R}^{-1}\boldsymbol{\delta} + \mathbf{f}^T\mathbf{R}^{-1}\boldsymbol{\delta}$ subject to $\mathbf{L}\mathbf{R}^{-1}\boldsymbol{\delta} \le \mathbf{b} - \mathbf{L}\mathbf{a}$, where $\mathbf{R} = [\searrow_i n_i\mathbf{I}_k] \in \mathbb{R}^{mk\times mk}$ is the scaling matrix. After solving this new quadratic optimization problem, we need to rescale the result as $\boldsymbol{\delta} \leftarrow \mathbf{R}^{-1}\boldsymbol{\delta}$. The computational complexity of the algorithm is $O(dlmk + m^3k^3)$.

D. Comparison with other DTW techniques

As discussed in [1], [39], there are various techniques that have been proposed to accelerate and improve DTW. For instance, the Sakoe-Chiba band (DTW-SC) and the Itakura Parallelogram band (DTW-IP) reduce the complexity of the original DTW algorithm to $O(\beta n_x n_y)$ by constraining the warping path to be in a band of a certain shape, where $\beta < 1$ is the size ratio between the band and the original search space of DTW. However, using a narrow band (a small β) cuts off potential warping space, and may lead to a sub-optimal solution. For instance, Fig. 5a shows an example of two 1-D time series and the alignment results calculated by different algorithms. The results computed by DTW-SC and DTW-IP are less accurate than the ones computed using our proposed Gauss-Newton (GN). This is because both the SC and IP bands are over-constrained (Fig. 5b). See Fig. 6 for a detailed comparison on the number of free variables and computational complexity of different time warping methods.

To provide a quantitative evaluation, we synthetically generated 1-D sequences at 15 scales. For DTW-SC, we set

the band width as $\beta = 0.1$. For GN, we varied k among $6, 10, 14$ to investigate the effect of the number of bases. For each scale, we randomly generated 100 pairs of sequences. The error is computed using Eq. 18 and shown in Fig. 5c-d. DTW obtains the lowest error but takes the most time to compute. This is because DTW exhaustively searches the entire parameter space to find the global optima. Both DTW-SC and DTW-IP need less time than DTW because they need to search a smaller space constrained by different bands. Empirically, DTW-IP is more accurate than DTW-SC for our synthetic dataset. This is because the global optima is more likely to lie in the IP band than the SC band. Compared to DTW, DTW-SC and DTW-IP, GN is more computationally efficient because it has linear complexity in the length of the sequence. Moreover, increasing the number of bases monotonically reduces the error.

V. Experiments

This section compares CTW and GCTW against state-of-the-art methods for temporal alignment of time series in six experiments. In the first experiment, we compared the performance of CTW and GCTW against DTW, DDTW [28], IMW [10] and their multi-sequence extensions in the problem of aligning synthetic time series of varying complexity. In the second experiment, we aligned videos of different subjects performing a similar activity; each video is represented using different types of visual features. In the third experiment, we aligned facial expressions across subjects on videos with naturally occurring facial behavior. In the fourth experiment, we showed how GCTW can be applied to large-scale alignment. We aligned approximately $50,000$ frames of motion capture data of two subjects cooking the same recipe. In the fifth experiment, we showed how GCTW can be used to localize common subsequences between two time series. The last experiment shows how GCTW is able to align three sequences of different subjects performing a similar action recorded with different sensors (motion capture data, accelerometers and video). In the first three experiments, the ground truth was known and we provided quantitative evaluation of the performance. In the other experiments, we evaluated the quality of the alignment visually. The code for our method and the baseline methods is available at `http://humansensing.cs.cmu.edu/ctw`.

A. Evaluation methods

In the experiments, we compared CTW and GCTW with several state-of-the-art methods for temporal alignment of time series. Below, we provide a brief description of the techniques that we used for comparison.

DTW and mDTW: DTW is solved using a standard dynamic programming algorithm that minimizes Eq. 4. To evaluate the performance of temporal alignment of multiple sequences, we extended the concept of Procrustes analysis [47] to time series. That is, given $m(>2)$ time series, multi-sequence DTW (mDTW) aims to seek for a set of warping paths $\{\mathbf{p}_i\}_i$ that minimizes:

$$\min_{\{\mathbf{p}_i \in \Psi\}_i} J_{mdtw} = \sum_{i=1}^{m}\sum_{j=1}^{m} \frac{1}{2}\|\mathbf{X}_i\mathbf{W}_i - \mathbf{X}_j\mathbf{W}_j\|_F^2.$$

After some linear algebra, it can be shown that J_{mdtw} can be equivalently computed as:

$$J_{mdtw} = m\sum_{i=1}^{m}\|\mathbf{X}_i\mathbf{W}_i - \underbrace{\frac{1}{m}\sum_{j=1}^{m}\mathbf{X}_j\mathbf{W}_j}_{\bar{\mathbf{X}}}\|_F^2.$$

Based on the above fact, mDTW alternates between independently solving each $\mathbf{p}_i$ using an asymmetrical DTW[3] and updating the mean sequence $\bar{\mathbf{X}}$ by averaging $\{\mathbf{X}_i\mathbf{W}_i\}_i$.

DDTW and mDDTW: In order to make DTW invariant to translation, derivative dynamic time warping (DDTW) [28] uses the derivatives of the original features and minimizes:

$$\min_{\{\mathbf{p}_x,\mathbf{p}_y\}\in\Psi} J_{ddtw} = \|\mathbf{X}\mathbf{F}_{n_x}^T\mathbf{W}_x - \mathbf{Y}\mathbf{F}_{n_y}^T\mathbf{W}_y\|_F^2,$$

where $\mathbf{F}_{n_x}$ and $\mathbf{F}_{n_y}$ are the 1^{st} order differential operators defined similarly as Eq. 10. To align multiple sequences, multi-sequence DDTW (mDDTW) extends DDTW in the Procrustes framework by optimizing:

$$\min_{\{\mathbf{p}_i\in\Psi\}_i} J_{mddtw} = \sum_{i=1}^{m}\sum_{j=1}^{m}\frac{1}{2}\|\mathbf{X}_i\mathbf{F}_{n_i}^T\mathbf{W}_i - \mathbf{X}_j\mathbf{F}_{n_j}^T\mathbf{W}_j\|_F^2.$$

IMW and mIMW: Similar to CTW, iterative motion warping (IMW) [10] alternates between time warping and spatial transformation to align two sequences. Assuming the same number of spatial features between $\mathbf{X}\in\mathbb{R}^{d\times n_x}$ and $\mathbf{Y}\in\mathbb{R}^{d\times n_y}$, IMW translates and re-scales each feature in $\mathbf{X}$ independently to match with $\mathbf{Y}$. Written in a simple matrix form, IMW minimizes:

$$\begin{aligned}\min_{\mathbf{p}_x\in\Psi,\mathbf{A}_x,\mathbf{B}_x} \quad & J_{imw} = \|(\mathbf{X}\circ\mathbf{A}_x + \mathbf{B}_x)\mathbf{W}_x - \mathbf{Y}\|_F^2 \\ & + \lambda_a\|\mathbf{A}_x\mathbf{F}_{n_x}^T\|_F^2 + \lambda_b\|\mathbf{B}_x\mathbf{F}_{n_x}^T\|_F^2,\end{aligned}$$

where $\mathbf{A}_x, \mathbf{B}_x \in \mathbb{R}^{d\times n_x}$ are the scaling and translation parameter respectively. λ_a and λ_b are the weights to trade-off between the least-square error and the regularization terms. The regularization terms are used to enforce a smooth change in the columns of $\mathbf{A}_x$ and $\mathbf{B}_x$. In the experiments, we set them as $\lambda_a = \lambda_b = 1$. IMW takes a coordinate-descent approach to optimize the time warping, the scaling and translation. Given the warping matrix $\mathbf{W}_x$, the optimal spatial transformation can be computed in closed-form. To align multiple sequences, we extended IMW as multi-sequence IMW (mIMW) by minimizing:

$$\begin{aligned}&\min_{\{\mathbf{p}_i\in\Psi,\mathbf{A}_i,\mathbf{B}_i\}_i} J_{mimw} \\ &= \sum_{i=1}^{m}\sum_{j=1}^{m}\frac{1}{2}\|(\mathbf{X}_i\circ\mathbf{A}_i + \mathbf{B}_i)\mathbf{W}_i - (\mathbf{X}_j\circ\mathbf{A}_j + \mathbf{B}_j)\mathbf{W}_j\|_F^2 \\ &\quad + \sum_{i=1}^{m}\Big(\lambda_a\|\mathbf{A}_i\mathbf{F}_{n_i}^T\|_F^2 + \lambda_b\|\mathbf{B}_i\mathbf{F}_{n_i}^T\|_F^2\Big).\end{aligned}$$

mCTW: CTW was originally proposed to align two multi-modal sequences. Similar to other DTW-based methods, we extended CTW in multi-sequence CTW (mCTW)

[3]Given two sequences $\mathbf{X}$ and $\mathbf{Y}$, asymmetric DTW is used to only warp $\mathbf{X}$ towards $\mathbf{Y}$ and no time warping is computed for $\mathbf{Y}$.

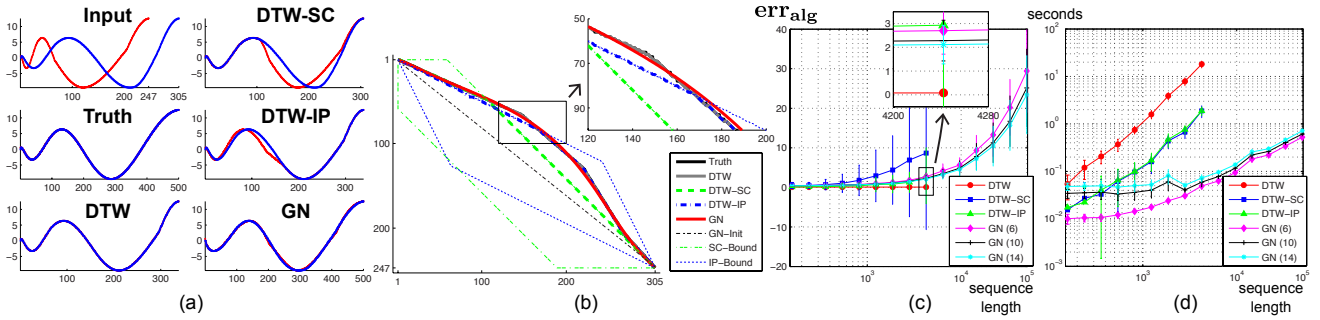

Fig. 5. Comparison between Gauss-Newton and variants of DTW for temporal alignment. (a) An example of two 1-D time series (with 247 and 305 frames respectively) and the alignment results calculated using the ground truth, DTW, DTW constrained in the Sakoe-Chiba band (DTW-SC), DTW constrained in the Itakura Parallelogram band (DTW-IP) and Gauss-Newton (GN). (b) Comparison of different warping paths. GN-Init denotes the initial warping used for GN. SC-Bound and IP-Bound denote the boundaries of SC band and IP band respectively. (c) Comparison of alignment errors. (d) Statistics of computational costs.

for aligning multiple time series using the Procrustes Analysis framework. mCTW optimizes the same objective (Eq. 8) as GCTW does. The main difference between mCTW and GCTW comes from the temporal alignment step. mCTW alternates between warping each time series using asymmetric DTW and updating the mean sequence, while GCTW uses Gauss-Newton for jointly optimizing over all weights of the monotonic bases.

Fig. 6 compares the temporal alignment methods in terms of the number of variables as well as the computational complexity. The comparison is divided into two cases, one for alignment of two sequences and another for alignment of more than two sequences. In the first case, given two time series, $\mathbf{X} \in \mathbb{R}^{d_x \times n_x}$ and $\mathbf{Y} \in \mathbb{R}^{d_x \times n_y}$, DTW and DDTW require the same complexity $O(n_x n_y)$ for finding the optimal l-length warping path. IMW additionally solves d least-squares problems for each row of $\mathbf{A}_x$ and $\mathbf{B}_x$ independently. Similarly, CTW relies on DTW to optimize the time warping, resulting in a complexity of $O(n_x n_y)$ in both space and time. However, CTW uses CCA to accommodate the feature difference by solving a generalized eigen-decomposition of two $(d_x + d_y)$-by-$(d_x + d_y)$ matrices. CTW has fewer variables than IMW and thus is less likely to overfit the data. Compared to CTW, GCTW has the same complexity for computing the spatial embedding. The main advantage of GCTW is its Gauss-Newton component, which optimizes a small-scale QP with $2k$ variables for the time warping.

In the second case, given m sequences, $\{\mathbf{X}_i \in \mathbb{R}^{d_i \times n_i}\}_{i=1}^m$, a direct generalization of the DTW is infeasible due to the combinatorial explosion of possible warpings, incurring a complexity of $O(\prod_{i=1}^m n_i)$. In the experiment, mDTW is used as an approximation of the exact DTW optimization. However, mDTW and other DTW-based methods (mDDTW, mIMW and mCTW) still have quadratic complexity. Instead, GCTW approximates the combinatorial problem of time warping as a continuous optimization that can be more efficiently optimized by solving a small-scale QP with mk variables.

	Method	Degree-of-Freedom		Complexity $O(\cdot)$	
		Embedding	Warping	Embedding	Warping
Two-sequence alignment $\mathbf{X} \in \mathbb{R}^{d_x \times n_x}$ $\mathbf{Y} \in \mathbb{R}^{d_y \times n_y}$	DTW	–	$2l$	–	$n_x n_y$
	DTW-SC	–	$2l$	–	$\beta n_x n_y$
	DTW-IP	–	$2l$	–	$\beta n_x n_y$
	DDTW	–	$2l$	–	$n_x n_y$
	IMW	$2dn_x$	$2l$	$dLS(2n_x)$	$n_x n_y$
	CTW	$d(d_x + d_y)$	$2l$	$eig(d_x + d_y)$	$n_x n_y$
	GCTW	$d(d_x + d_y)$	$2k$	$eig(d_x + d_y)$	$QP(2k)$
Multi-sequence alignment $\{\mathbf{X}_i \in \mathbb{R}^{d_i \times n_i}\}_i$	DTW	–	ml	–	$\Pi_i n_i$
	mDTW	–	ml	–	$l \sum_i n_i$
	mDDTW	–	ml	–	$l \sum_i n_i$
	mIMW	$2l \sum_i d_i$	ml	$\sum_i d_i LS(2l)$	$l \sum_i n_i$
	mCTW	$d \sum_i d_i$	ml	$eig(\sum_i d_i)$	$l \sum_i n_i$
	GCTW	$d \sum_i d_i$	mk	$eig(\sum_i d_i)$	$QP(mk)$

Fig. 6. Comparison of temporal alignment algorithms as a function of degrees-of-freedom (number of free variables) and complexity. l is the length of warping path computed by the algorithm. $LS(n)$, $QP(n)$ and $eig(n)$ denote the complexity of solving a least-squares of n variables, a QP of n variables and a generalized eigenvalue problem with two n-by-n matrices, respectively.

B. Evaluation metrics

The alignment result of m sequences will be denoted by a set of time warping paths, $\mathbf{P}_{alg} = [\mathbf{p}_1^{alg}, \cdots, \mathbf{p}_m^{alg}] \in \mathbb{R}^{l_{alg} \times m}$, where $\mathbf{p}_i^{alg} \in \mathbb{R}^{l_{alg}}$ is the time warping path for the i^{th} sequence. To evaluate the error of the time warping paths given by different methods, we computed its difference from the ground-truth, $\mathbf{P}_{tru} = [\mathbf{p}_1^{tru}, \cdots, \mathbf{p}_m^{tru}] \in \mathbb{R}^{l_{tru} \times m}$, where the number of warping steps (l_{alg} and l_{tru}) could be different. To better understand the error, let us consider each warping path $\mathbf{P} \in \mathbb{R}^{l \times m}$ as a curve in $\mathbb{R}^m$ with l points (rows of $\mathbf{P}$). For instance, Fig. 7c and Fig. 7g compare the warping paths as 2-D and 3-D curves respectively. The error can be hence defined as the normalized distance between the curves $\mathbf{P}_{alg}$ and $\mathbf{P}_{tru}$, *i.e.*,

$$error = \frac{dist(\mathbf{P}_{alg}, \mathbf{P}_{tru}) + dist(\mathbf{P}_{tru}, \mathbf{P}_{alg})}{l_{alg} + l_{tru}}, \qquad (18)$$

$$\text{where} \quad dist(\mathbf{P}_1, \mathbf{P}_2) = \sum_{i=1}^{l_1} \min_{j=1}^{l_2} \|\mathbf{p}_1^{(i)} - \mathbf{p}_2^{(j)}\|_2.$$

The term, $\min_{j=1}^{l_2} \|\mathbf{p}_1^{(i)} - \mathbf{p}_2^{(j)}\|_2$, measures the shortest distance between the point $\mathbf{p}_1^{(i)}$ and any point on the curve $\mathbf{P}_2$, where $\mathbf{p}_1^{(i)} \in \mathbb{R}^{1 \times m}$ and $\mathbf{p}_2^{(j)} \in \mathbb{R}^{1 \times m}$ are the i^{th} row of $\mathbf{P}_1$ and j^{th} row of $\mathbf{P}_2$ respectively.

C. Aligning synthetic sequences

In the first experiment, we synthetically generated spatio-temporal signals (3-D in space and 1-D in time) to evaluate

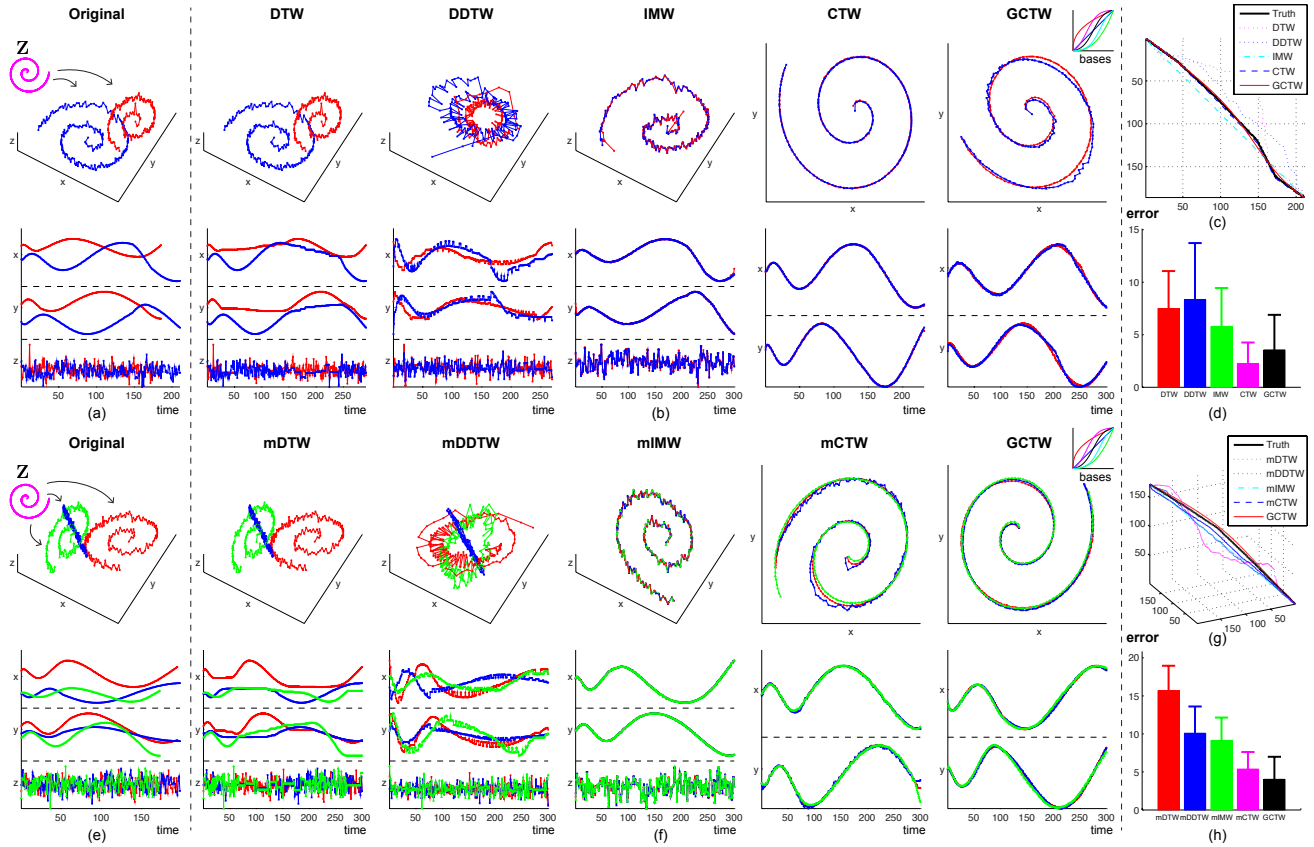

Fig. 7. Comparison of temporal alignment algorithms on the synthetic dataset. (a) An example of two synthetic time series ($\mathbf{X}$ and $\mathbf{Y}$) generated by performing a random spatio-temporal transformation of a 2-D latent sequence $\mathbf{Z}$ and adding Gaussian noise in the 3^{rd} dimension. (e) An example of three synthetic time series. (b)(f) The alignment results. (c)(g) Comparison of time warping paths. (d)(h) Mean and variance of the alignment errors.

the performance of CTW, mCTW and GCTW. As shown in Fig. 7a, the signals were a randomly generated by spatially and temporally transforming a latent 2-D spiral, $\mathbf{Z} \in \mathbb{R}^{2\times l}, l = 300$ as follows:

$$\mathbf{X} = \begin{bmatrix} \mathbf{U}^T(\mathbf{Z} + \mathbf{b}\mathbf{1}^T)\mathbf{M} \\ \mathbf{e}^T \end{bmatrix} \in \mathbb{R}^{3\times n},$$

where $\mathbf{U} \in \mathbb{R}^{2\times 2}$ and $\mathbf{b} \in \mathbb{R}^2$ were randomly generated projection matrix and translation vector respectively. To synthesize the temporal distortion, a binary selection matrix $\mathbf{M} \in \{0,1\}^{l\times n}$ was generated by randomly choosing $n \leq l$ columns from the identity matrix $\mathbf{I}_l$. The third spatial dimension $\mathbf{e} \in \mathbb{R}^n$ was added with a zero-mean Gaussian noise. In this experiment, we considered two types of alignment problems. In the first setting, we compared CTW and GCTW with DTW, DDTW and IMW for aligning two time series, while in the second one, mCTW and GCTW were compared with mDTW, mDDTW and mIMW for aligning three time series. In both settings, the ground-truth alignment was known and the performance of each method was evaluated in terms of the alignment errors defined in Eq. 18. We repeated the above process 100 times with random numbers. In each trial, we initialized all iterative methods by uniformly aligning the sequences, *i.e.*, the initial warping path for $\mathbf{X}$ was computed as $\mathbf{p}_0 = round(linspace(1, n, l))'$, where $round(\cdot)$ and $linspace(\cdot)$ are MATLAB functions. For CTW, mCTW and GCTW, d was selected to preserve 90% of the total correlation and the regularization weight λ was set to zero. For GCTW, we selected three hyperbolic tangent and three polynomial functions as monotonic bases (the last column in Fig. 7f) and we set $\eta = 1$.

Figs. 7a-d compare the methods for aliging two time series. Fig. 7b shows the spatial-temporal warping estimated by each algorithm. More specifically, the five columns in Fig. 7b plot $\mathbf{XW}$ for DTW, $\mathbf{XF}^T\mathbf{W}$ for DDTW, $(\mathbf{X} \circ \mathbf{A} + \mathbf{B})\mathbf{W}$ for IMW, and $\mathbf{V}^T\mathbf{XW}$ for both CTW and GCTW, respectively. Fig. 7c compares the alignment paths computed by different methods and Fig. 7d shows the average error for 100 generated time series. DDTW performs poorly with this example because the feature derivatives are not able to sufficiently capture the structure of the sequence. IMW warps one sequence towards the other by translating and re-scaling each frame in each dimension. As summarized in Fig. 6, IMW has more parameters ($2dn_x$) than CTW ($d(d_x + d_y)$) for feature weighting, and hence IMW is more prone to overfitting. Furthermore, IMW tries to fit the third noisy dimension, biasing alignment in time, whereas CTW has a feature selection mechanism that effectively cancels the third dimension. In aligning the two time series, CTW achieved better performance than GCTW because GCTW employed an approximation of DTW in the temporal alignment step.

Figs. 7e-h illustrate the comparison of previous methods for aligning multiple time series. Fig. 7f shows the spatio-

temporal warping estimated mDTW, mDDTW, mIMW, mCTW and GCTW, respectively. Fig. 7g compares the warping paths computed by different methods as 3-D curves. Fig. 7h shows the error for 100 randomly generated time series. Both mDTW and mDDTW performed poorly in this case since they do not have a feature weight mechanism to adapt the spatial transformation of the sequences. mIMW warps sequences towards others by translating and re-scaling each frame in each dimension. Moreover, mIMW has more parameters ($2l\sum_i d_i$) than mCTW and GCTW ($d\sum_i d_i$), and hence mIMW is more prone to over-fitting. Furthermore, mIMW tries to fit the noisy dimension (3^{rd} spatial component) biasing alignment in time, whereas both mCTW and GCTW had a feature selection mechanism that effectively cancels out the third dimension. Compared to mCTW, GCTW achieves better performance aligning more than two sequences because GCTW jointly optimizes over all the possible time warpings for each time series, while mCTW takes a greedy approach by warping each sequence towards the mean sequence independently. In addition, GCTW can be more efficiently optimized than other DTW-based approaches in large-scale cases.

D. Aligning videos with different features

In the second experiment, we used CTW, mCTW and GCTW to align video sequences of different people performing a similar action. Each video was encoded using different visual features. The video sequences were taken from the Weizmann database [48], which contains nine people performing ten actions. To represent dynamic videos, we subtracted the background (the top rows in Fig. 8a and Fig. 8e) and computed three popular shape features (the bottom rows of Fig. 8a and Fig. 8e) for each 70-by-35 re-scaled mask image, including (1) binary image, (2) Euclidean distance transform [49], and (3) solution of Poisson equation [50]. In order to reduce the dimension of the feature space (2450), we picked the top 123 principal components that preserved 99% of the total energy. We split this experiment into two settings so that we could evaluate the performance of aligning two and three sequences separately. In the first setting, we randomly selected two walking sequences, each of which was manually cropped into two cycles of human walking. The ground-truth alignment was approximated by using DTW using the same feature (Euclidean distance transform), and it provided an accurate visual temporal alignment. In the second setting, we randomly selected three sequences and estimated the ground-truth using mDTW for aligning the sequences with same feature. In both settings, GCTW was initialized with uniform alignment, and we set $\lambda = 0.1$. We used five hyperbolic tangent and five polynomial functions as the monotonic bases (Fig. 8b upper-top).

Fig. 8d and Fig. 8h show the error for 10 randomly generated sets of videos in the first and second setting respectively. Note that neither DTW, and DDTW were able to align the videos because they are not able to handle alignment of signals of different dimensions. Their multiple sequence counterparts, mDTW and mDDTW failed similarly. IMW and mIMW register the top three components well in space; however, both overfit and compute a biased time warping path. In contrast, CTW, mCTW and GCTW warp the sequences accurately in both space and time.

E. Aligning facial expression sequences

In the third experiment, we compared CTW and GCTW in the task of aligning unscripted facial expression sequences. The facial videos were taken from the RU-FACS database [51], which consists of digitized videos of 29 young adults. They were recorded during an interview (approximately two-minutes long) in which they either lied or told the truth in response to an interviewer's questions. Pose orientation was mostly frontal with small to moderate out-of-plane head motion. The action units (AUs) in this database have been manually coded, and we randomly cropped video segments containing AU12 (smiling) to run our experiments. Each event of AU12 is coded at its peak position. We used a person-specific active appearance model [52] to track 66 landmarks on the face. For the alignment of AU12, we used only 18 landmarks that correspond to the outline of the mouth. See Fig. 9a for example frames aligned by GCTW where the mouth outlines are plotted.

The performance of CTW and GCTW were compared with DTW, DDTW and IMW. We initialized IMW, CTW and GCTW using the same uniform warping. Fig. 9b shows the alignment result obtained by different methods, where the three dimensions correspond to the first three principal components of the original signals. As an approximate ground-truth, the position of the peak frame of each AU12 event is indicated as the red and blue points on the curves in Fig. 9b and the intersection of the two dotted lines in Fig. 9c. As we can observe from Fig. 9b-c, the two peaks in the low-dimensional projection found by CTW and GCTW are closer to the manually labeled peak than the ones in the original space used for DTW and DDTW. Finally, the distance between the peak point and the warping path is computed to quantitatively measure the performance. Fig. 9d shows the average error as the distance normalized by the sequence lengths over 20 random repetitions. Here, CTW and GCTW achieved better performance than other state-of-the-art methods.

F. Aligning large-scale motion capture sequences

This experiment illustrates the benefits of using GCTW for aligning two large-scale motion capture sequences. The two sequences were taken from the CMU-Multimodal Activity Dataset [53], which contains multi-sensor recordings (video, audio, motion capture data and accelerometers) of naturalistic behavior of 40 subjects cooking five different recipes. The two sequences used in this experiment contain 44387 and 48724 frames of two subjects cooking brownies. See Fig. 10c for several key-frames of these two sequences. For each motion capture frame, we computed the quaternions for the four joints on the right hand, resulting in a 12-D feature vector that describes the body configuration. In

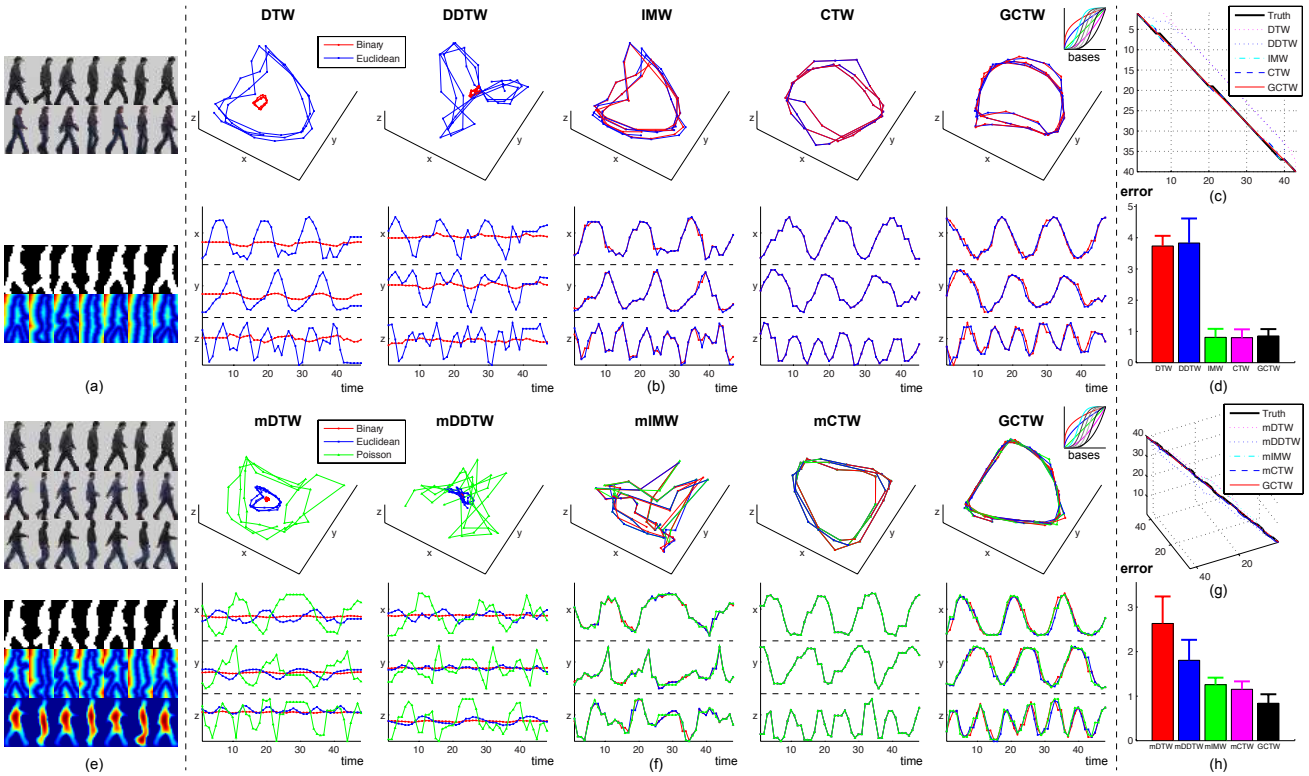

Fig. 8. Comparison of temporal alignment algorithms for aligning multi-feature video data. (a) An example of two video sequences aligned by GCTW. The top two sequences are the key-frames after background subtraction, while the bottom two are the binary images and the Euclidean distance transforms. (e) An example of three aligned videos by GCTW. The top three sequences are the original frames after background subtraction, while the bottom three are the binary images, the Euclidean distance transforms and the solutions of the Poisson equation. (b)(f) The alignment results. (c)(g) Comparison of time warping paths. (d)(h) Mean and variance of the alignment errors.

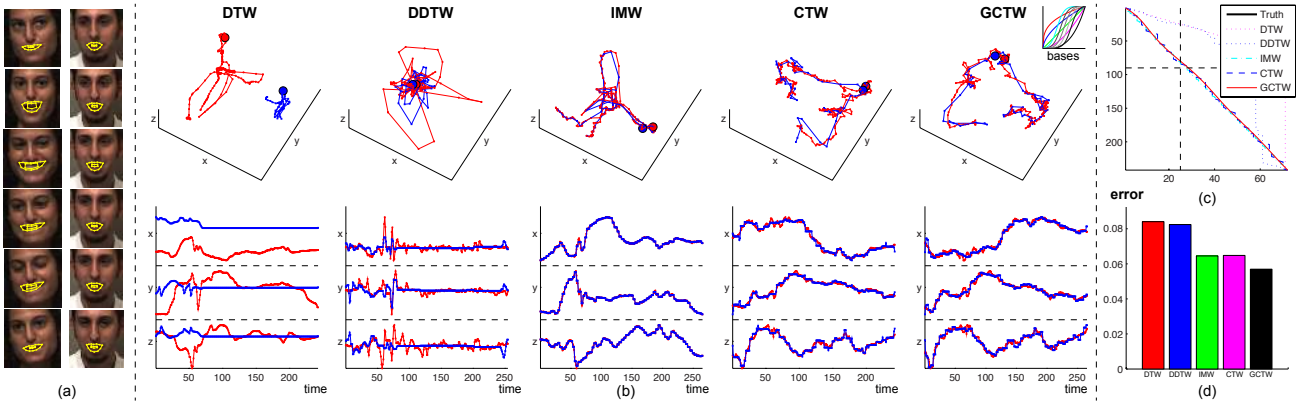

Fig. 9. Comparison of temporal alignment algorithms for aligning facial expression sequences between different people. (a) An example of two smiling expression sequences aligned by GCTW. The features of the two sequences are computed as the 18 landmark coordinates of the mouth given by a face tracker. (b) The alignment results, where the expression peaks are indicated by points on the curves in the top row. (c) Comparison of time warping paths, where the expression peaks are indicated by the intersection of the two dotted lines. (d) Mean of the alignment errors.

this experiment, we only tested the performance of GCTW on aligning large-scale sequences, and we did not optimize GCTW over its spatial component. We used five polynomial functions and five $\tanh(\cdot)$ functions as monotonic bases for the time warping function (upper-right corner in Fig. 10b).

To avoid local minima in the alignment, we used a temporal coarse-to-fine strategy for the Gauss-Newton optimization in GCTW. As shown in Fig. 10a, the coarse-to-fine strategy proceeds in two steps: (1) In the pre-processing step, we obtained a three-level pyramid for each time series by recursively applying Gaussian smoothing with $\sigma = 200$. For instance, the first row of Fig. 10b illustrates the two sequences in three levels, where the ones in the first level correspond to the original signals, while the ones in the third level contain less detailed but much smoother signals. (2) In the optimization step, GCTW was first used to align the two sequences on the third level instead of the first level. The computed time warping result was then used to initialize GCTW on the second level. We repeated the same procedure to compute the final time warping result of the original sequences on the first level.

For this large-scale example, DTW is too slow and ex-

pensive to compute. However, GCTW is able to efficiently find the temporal correspondence between the sequences in just a few seconds using Matlab on a regular laptop with a 2.5GHz Intel CPU. Since the ground-truth is unknown, we qualitatively evaluate the performance of GCTW by showing the aligned key frames in Fig. 10c. Although the two subjects spent different amounts of time and followed different procedures to cook the same recipe, GCTW was able to align similar body poses.

G. Detection and alignment of similar sub-sequences

A major problem of DTW-type of techniques for aligning long sequences is that they require an exact matching between the first frame and the last one, see boundary conditions (Eq. 5). These boundary conditions are impractical and very restrictive when only a subset of the input sequence is similar to the sequence to be aligned. The first row of Fig. 11 illustrates this problem: how can we align four motion capture signals composed by different walking cycles? This problem is related to sub-sequence DTW [54], the longest common sub-sequence [14] and temporal commonality discovery [55]. A major limitation of these methods is that they are not computationally feasible when handling more than two sequences. This experiment shows how can we used GTW for multiple sub-sequence alignment in the context of aligning motion capture data.

We selected four walking and running sequences from the CMU motion capture database[4]. For each motion capture frame, we computed the quaternions for 14 joints on the body, resulting in a 42-D feature vector that describes the human pose. The first and second row of Fig. 11a illustrate the first three principal components of the walking and running sequences respectively. To allow for sub-sequence alignment, the warping path in GCTW is represented by a combination of a constant function and a linear one as the monotonic bases (see upper-left corner of Fig. 11c). Both GCTW and the baseline mDTW method are initialized by uniformly aligning the sequences.

A visual comparison between mDTW and GCTW is illustrated in Fig. 11. Without any manual cropping, most of the conventional DTW-based methods, such as mDTW, aligned the sequences by matching the first and the last frame, which results in incorrect alignments, see Fig. 11b. Some parts (noted by arrows) of the sequences with fewer cycles have to be stretched into flat lines in order to match the other sequences with more cycles. Unlike conventional DTW-based methods built on dynamic programming, GCTW uses the Gauss-Newton method, which allows for a more flexible time warping. By incorporating a constant function in the set of bases, GCTW can naturally be generalized to deal with the sub-sequence alignment problem across multiple sequences. As shown in Fig. 11c-d, GCTW is not only able to align the sequences in time, but also locate the boundaries of the sub-sequences that contain similar motions. This experiment demonstrates the benefits of GCTW in controlling the warping path when using a specific time warping bases.

[4] `http://mocap.cs.cmu.edu`

H. Aligning multi-modal sequences

This experiment uses GCTW to align sequences of different people performing a similar activity recorded with different sensors. We selected one motion capture sequence (Subject 12, Trial 29) from the CMU motion capture database, one video sequence (`Eli_jacking.avi`) from the Weizmann database [48], and we collected the accelerometer signal of a subject performing jumping jacks. Some instances of the multi-modal data can be seen in Fig. 12d. Note that to make the problem more challenging, the two subjects in the mocap (top row) and video (middle row) are performing the same activity, but in the accelerometer sequence (bottom row) the subject only moves one hand and not the legs. Even in this challenging scenario, GCTW is able to solve for the temporal correspondence that maximizes the correlation between signals.

For the mocap data, we computed the quaternions for the 20 joints, resulting in a 60 dimensional feature vector that describes the body configuration. In the case of the Weizmann dataset, we computed the Euclidean distance transform as described earlier. The X, Y and Z axis accelerometer data was collected using an X6-2 mini USB accelerometer (Fig. 12a) at a rate of 40Hz. GCTW was initialized by uniformly aligning the three sequences. We used five hyperbolic tangent and five polynomial functions as monotonic bases. Fig. 12b shows the first components of the three sequences projected separately by PCA. As shown in Fig. 12c, GCTW found an accurate temporal correspondence between the three sequences. Unfortunately, we do not have ground-truth for this experiment. However, visual inspection of the video suggests that the results are consistent with human labeling. Fig. 12d shows several frames that have been put in correspondence by GCTW.

VI. Conclusions

This paper proposes CTW and GCTW, two new techniques for spatio-temporal alignment of multiple multi-modal time series. CTW extends DTW by adding a feature selection mechanism and enabling alignment of signals with different dimensionality. CTW extends CCA by adding temporal alignment and allowing temporally local projections. To improve the efficiency of CTW, allow a more flexible time-warping, and align multiple sequences, GCTW extends CTW by parameterizing the warping path as a combination of monotonic functions. Inspired by existing work on image alignment, GCTW is optimized using a coarse-to-fine Gauss-Newton updates, which allows for efficient alignment of long sequences.

Although CTW and GCTW have shown promising preliminary results, there are still unresolved issues. First, the Gauss-Newton algorithm used in GCTW for time warping converges poorly in the area where the objective function is non-smooth. Second, both CTW and GCTW are subject to local minima. The effect of local minima can be partially

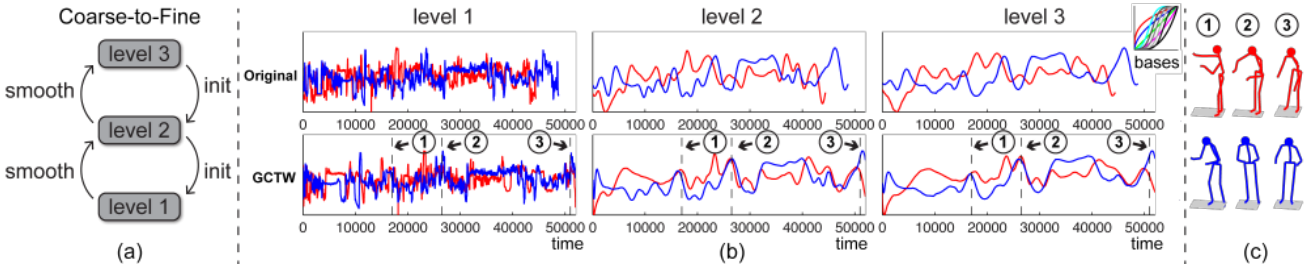

Fig. 10. Example of aligning two large-scale motion capture sequences using GCTW. (a) A coarse-to-fine strategy for improving the optimization performance of GCTW. (b) The first row shows the first principal components of the original sequences for three levels of the temporal pyramids. The second row corresponds to the aligned sequences using GCTW. (c) Key frames of similar body poses aligned by GCTW.

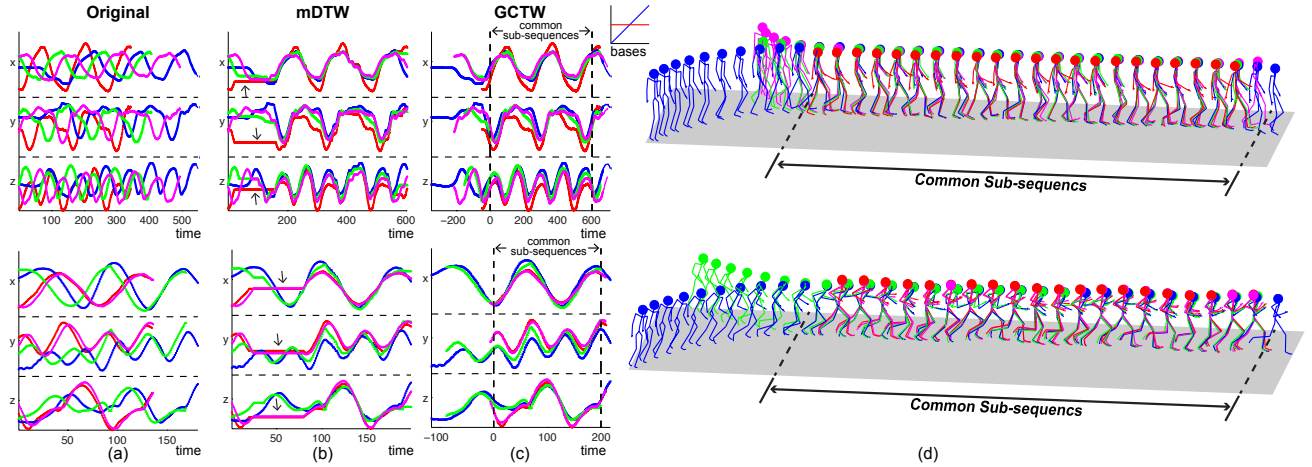

Fig. 11. Example of locating and aligning similar sub-sequences of four walking (1^{st} row) and four running (2^{nd} row) motion capture signals. (a) Original features of four mocap walking sequences. (b) Alignment achieved by mDTW. mDTW tries to align the sequences end-to-end and stretch some parts of the sequences (flat lines indicated by arrows). (c) Alignment by GCTW. GCTW efficiently aligns the sub-sequences and also finds the boundaries of the sub-sequences containing similar motions. (d) Key frames aligned by GCTW.

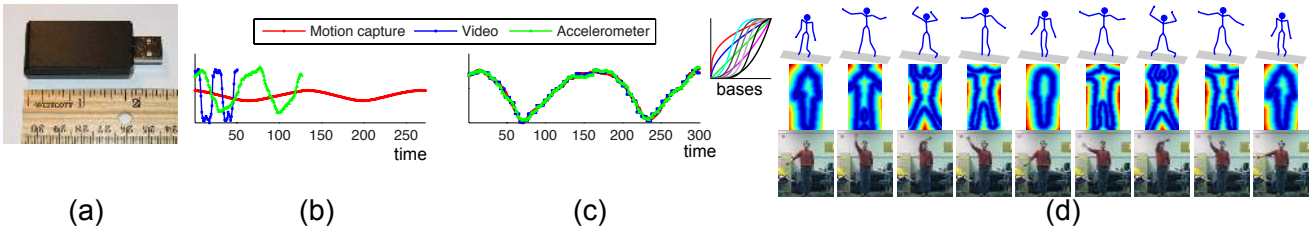

Fig. 12. Example of aligning multi-modal sequences. (a) Accelerometer. (b) Projection onto the first principal component for the motion capture data, video and accelerometers respectively. (c) GCTW. (d) Key frames aligned by GCTW. Notice that the similar hand gestures have been aligned. On the top row we show mocap data, in the middle row video, and in the bottom the images of the accelerometer data.

alleviated using a temporal coarse-to-fine approach as in the case of image alignment. In future work, we plan to explore better initialization strategies. Third, although the experiments show good results using manually designed bases, we plan to learn a set of monotonic bases that are adapted to the particular alignment problem.

Acknowledgments This work was partially supported by the National Science Foundation under Grant No. EEEC-0540865, RI-1116583 and CPS-0931999. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] L. Rabiner and B. Juang, *Fundamentals of speech recognition*. Prentice Hall, 1993.

[2] J. Listgarten, R. M. Neal, S. T. Roweis, and A. Emili, "Multiple alignment of continuous time series," in *Proc. Neural Information Processing Systems*, 2005.

[3] A. Bruderlin and L. Williams, "Motion signal processing," in *ACM SIGGRAPH*, 1995, pp. 97–104.

[4] Y. Caspi and M. Irani, "Aligning non-overlapping sequences," *Int'l J. Computer Vision*, vol. 48, no. 1, pp. 39–51, 2002.

[5] J. Aach and G. M. Church, "Aligning gene expression time series with time warping algorithms," *Bioinformatics*, vol. 17, no. 6, pp. 495–508, 2001.

[6] T. B. Sebastian, P. N. Klein, and B. B. Kimia, "On aligning curves," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 1, pp. 116–125, 2003.

[7] F. Zhou, F. De la Torre, and J. K. Hodgins, "Hierarchical aligned cluster analysis for temporal clustering of human motion," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 35, no. 3, pp. 582–596, 2013.

[8] J. M. Winters and Y. Wang, "Wearable sensors and telerehabilitation," *IEEE Engineering in Medicine and Biology Magazine*, vol. 22, no. 3, pp. 56–65, 2003.

[9] I. N. Junejo, E. Dexter, I. Laptev, and P. Pérez, "View-independent

action recognition from temporal self-similarities," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33, no. 1, pp. 172–185, 2011.

[10] E. Hsu, K. Pulli, and J. Popovic, "Style translation for human motion," *ACM Trans. Graphics*, vol. 24, no. 3, pp. 1082–1089, 2005.

[11] W. Pan and L. Torresani, "Unsupervised hierarchical modeling of locomotion styles," in *Proc. Int'l Conf. Machine Learning*, 2009.

[12] Y. Sheikh, M. Sheikh, and M. Shah, "Exploring the space of a human action," in *Proc. IEEE Int'l Conf. Computer Vision*, 2005.

[13] A. Veeraraghavan, A. Srivastava, A. K. R. Chowdhury, and R. Chellappa, "Rate-invariant recognition of humans and their activities," *IEEE Trans. Image Processing*, vol. 18, no. 6, pp. 1326–1339, 2009.

[14] D. Gusfield, *Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology*. Cambridge University Press, 1997.

[15] F. Zhou and F. De la Torre, "Canonical time warping for alignment of human behavior," in *Proc. Neural Information Processing Systems*, 2009.

[16] ——, "Generalized time warping for alignment of human motion," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2012.

[17] M. Brand and A. Hertzmann, "Style machines," in *ACM SIGGRAPH*, 2000, pp. 183–192.

[18] G. W. Taylor, G. E. Hinton, and S. T. Roweis, "Modeling human motion using binary latent variables," in *Proc. Neural Information Processing Systems*, vol. 19, 2007, p. 1345.

[19] C. Rose, M. F. Cohen, and B. Bodenheimer, "Verbs and adverbs: Multidimensional motion interpolation," *IEEE Comput. Graph Appl.*, vol. 18, no. 5, pp. 32–40, 1998.

[20] L. Kovar and M. Gleicher, "Flexible automatic motion blending with registration curves," in *Proc. ACM SIGGRAPH / Eurographics Symp. Computer Animation*, 2003, p. 224.

[21] A. Shapiro, Y. Cao, and P. Faloutsos, "Style components," in *Graphics Interface*, 2006, pp. 33–39.

[22] A. Heloir, N. Courty, S. Gibet, and F. Multon, "Temporal alignment of communicative gesture sequences," *J. Visualization and Computer Animation*, vol. 17, no. 3-4, pp. 347–357, 2006.

[23] K. Forbes and E. Fiume, "An efficient search algorithm for motion data using weighted PCA," in *Proc. ACM SIGGRAPH / Eurographics Symp. Computer Animation*, 2005.

[24] C. Rao, A. Gritai, M. Shah, and T. F. Syeda-Mahmood, "View-invariant alignment and matching of video sequences," in *Proc. IEEE Int'l Conf. Computer Vision*, 2003.

[25] A. Gritai, Y. Sheikh, C. Rao, and M. Shah, "Matching trajectories of anatomical landmarks under viewpoint, anthropometric and temporal transforms," *Int'l J. Computer Vision*, 2009.

[26] D. Gong and G. G. Medioni, "Dynamic manifold warping for view invariant action recognition," in *Proc. IEEE Int'l Conf. Computer Vision*, 2011.

[27] M. A. Nicolaou, V. Pavlovic, and M. Pantic, "Dynamic probabilistic cca for analysis of affective behaviour," in *Proc. European Conf. Computer Vision*, 2012.

[28] E. J. Keogh and M. J. Pazzani, "Derivative dynamic time warping," in *SIAM Int'l Conf. on Data Mining*, 2001.

[29] J. Ham, D. Lee, and L. Saul, "Semisupervised alignment of manifolds," in *Int'l Conf. on Artificial Intelligence and Statistics*, 2005.

[30] C. Wang and S. Mahadevan, "Manifold alignment using Procrustes analysis," in *Proc. Int'l Conf. Machine Learning*, 2008.

[31] ——, "Manifold alignment without correspondence," in *Int'l Joint Conf. on Artificial Intelligence*, 2009, pp. 1273–1278.

[32] T. W. Anderson, *An introduction to multivariate statistical analysis*. Wiley-Interscience, 2003.

[33] F. De la Torre, "A least-squares framework for component analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 34, no. 6, pp. 1041–1055, 2012.

[34] T. K. Kim and R. Cipolla, "Canonical correlation analysis of video volume tensors for action categorization and detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, pp. 1415–1428, 2009.

[35] C. C. Loy, T. Xiang, and S. Gong, "Time-delayed correlation analysis for multi-camera activity understanding," *Int'l J. Computer Vision*, vol. 90, no. 1, pp. 106–129, 2010.

[36] B. Fischer, V. Roth, and J. Buhmann, "Time-series alignment by non-negative multiple generalized canonical correlation analysis," *BMC Bioinformatics*, vol. 8, no. 10, 2007.

[37] D. P. Bertsekas, *Dynamic programming and optimal control*. Athena Scientific, 1995.

[38] S. Chu, E. J. Keogh, D. Hart, and M. J. Pazzani, "Iterative deepening dynamic time warping for time series," in *SIAM Int'l Conf. on Data Mining*, 2002.

[39] S. Salvador and P. Chan, "Toward accurate dynamic time warping in linear time and space," *Intelligent Data Analysis*, vol. 11, no. 5, pp. 561–580, 2007.

[40] T. Rakthanmanon, B. J. L. Campana, A. Mueen, G. E. A. P. A. Batista, M. B. Westover, Q. Zhu, J. Zakaria, and E. J. Keogh, "Searching and mining trillions of time series subsequences under dynamic time warping," in *ACM Conf. Knowledge Discovery and Data Mining*, 2012.

[41] M. A. Hasan, "On multi-set canonical correlation analysis," in *Int'l Joint Conf. on Neural Networks*, 2009.

[42] J. O. Ramsay and B. W. Silverman, *Functional Data Analysis*, 2nd ed. Springer, 2005.

[43] J. O. Ramsay, "Estimating smooth monotone functions," *J. Royal Statistical Society: Series B Statistical Methodology*, vol. 60, no. 2, pp. 365–375, 1998.

[44] T. Robertson, F. T. Wright, and R. L. Dykstra, *Order restricted statistical inference*. Wiley, 1988.

[45] I. W. Wright and E. J. Wegman, "Isotonic, convex and related splines," *Annals of Statistics*, pp. 1023–1035, 1980.

[46] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Int'l Joint Conf. on Artificial Intelligence*, 1981.

[47] I. L. Dryden and K. V. Mardia, *Statistical shape analysis*. Wiley, 1998.

[48] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, no. 12, pp. 2247–2253, 2007.

[49] C. R. Maurer, R. Qi, and V. V. Raghavan, "A linear time algorithm for computing exact Euclidean distance transforms of binary images in arbitrary dimensions," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, pp. 265–270, 2003.

[50] L. Gorelick, M. Galun, E. Sharon, R. Basri, and A. Brandt, "Shape representation and classification using the Poisson equation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 1991–2005, 2006.

[51] M. S. Bartlett, G. C. Littlewort, M. G. Frank, C. Lainscsek, I. Fasel, and J. R. Movellan, "Automatic recognition of facial actions in spontaneous expressions," *J. Multimedia*, vol. 1, no. 6, pp. 22–35, 2006.

[52] I. Matthews and S. Baker, "Active appearance models revisited," *Int'l J. Computer Vision*, vol. 60, no. 2, pp. 135–164, 2004.

[53] F. De la Torre, J. K. Hodgins, J. Montano, and S. Valcarcel, "Detailed human data acquisition of kitchen activities: the CMU-multimodal activity database (CMU-MMAC)," CMU, Tech. Rep. RI-TR-08-22, 2009.

[54] P. Tormene, T. Giorgino, S. Quaglini, and M. Stefanelli, "Matching incomplete time series with dynamic time warping: an algorithm and an application to post-stroke rehabilitation," *Artificial Intelligence in Medicine*, vol. 45, no. 1, pp. 11–34, 2009.

[55] W.-S. Chu, F. Zhou, and F. De la Torre, "Unsupervised temporal commonality discovery," in *Proc. European Conf. Computer Vision*, 2012.