Manifold Warping: Manifold Alignment over Time

Hoa T. Vu and **CJ Carey** and **Sridhar Mahadevan**
Computer Science Department
University of Massachusetts, Amherst
Amherst, Massachusetts, 01003
{hvu,ccarey,mahadeva}@cs.umass.edu

Abstract

Knowledge transfer is computationally challenging, due in part to the curse of dimensionality, compounded by source and target domains expressed using different features (e.g., documents written in different languages). Recent work on manifold learning has shown that data collected in real-world settings often have high-dimensional representations, but lie on low-dimensional manifolds. Furthermore, data sets collected from similar generating processes often present different high-dimensional views, even though their underlying manifolds are similar. The ability to align these data sets and extract this common structure is critical for many transfer learning tasks. In this paper, we present a novel framework for aligning two sequentially-ordered data sets, taking advantage of a shared low-dimensional manifold representation. Our approach combines traditional manifold alignment and dynamic time warping algorithms using alternating projections. We also show that the previously-proposed canonical time warping algorithm is a special case of our approach. We provide a theoretical formulation as well as experimental results on synthetic and real-world data, comparing manifold warping to other alignment methods.

Introduction

The advent of large, often high-dimensional, digital data sets has made automated knowledge extraction a critical research focus in the field of machine learning. Often, we find real-world sequential data sets that encode the same information with disparate surface feature representations, such as sensor network data, activity and object recognition corpora, and audio/video streams. In these cases, an automated technique for discovering correlations between sets will allow easy transfer of knowledge from one domain to another, avoiding costly or infeasible re-learning. In this paper, we present a framework that combines manifold alignment (Ham, Lee, and Saul 2005; Wang and Mahadevan 2009) and dynamic time warping (DTW) (Sakoe and Chiba 1978) for aligning two such sequential data sets. Temporal alignment of time series is an important research topic in bioinformatics, text analysis, computer vision, etc. Some specific applications include human motion recognition (Junejo et al. 2008), temporal segmentation (Zhou, Torre, and Hodgins 2008), or

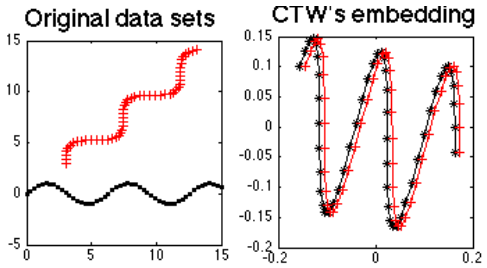

Figure 1: An illustration of two sinusoidal curves before and after applying CTW

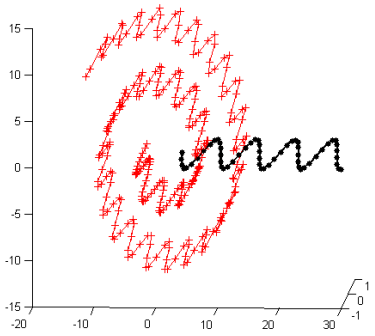

Figure 2: Sinusoidal curves on a plane and on a Swiss roll

building view-invariant representations of activities (Junejo et al. 2008). We also prove that the previously proposed method of canonical time warping (CTW) (Zhou and De la Torre 2009), which has been shown to outperform other state-of-the-art techniques based on DTW such as derivative dynamic time warping (Keogh and Pazzani 2001) or iterative time warping (Hsu, Pulli, and Popović 2005), is a special case of our approach. In addition, our approach performs better than CTW when data lie on different nonlinear manifolds.

Dynamic time warping has been used effectively for time-series alignment, but it requires an inter-set distance function, which usually implies that both input data sets must have the same dimensionality. DTW may also fail under arbitrary affine transformations of one or both inputs.

CTW aims to solve these two problems by alternating

between canonical correlation analysis (CCA) (Anderson 2003) and DTW until convergence, as illustrated in Figure 1. In the case where inputs are of different dimensions, CTW first projects both data sets into a shared space using principal component analysis (PCA) (Jolliffe 2002). The algorithm does not always converge to a global optimum, but CTW still improves the performance of the alignment when compared to applying DTW directly. However, CTW fails when the two related data sets require nonlinear transformations to uncover the shared manifold space. We illustrate such a case in Figure 2, in which two sequential data sets are two $sin(x)$ curves; one lying on a plane and the other on a Swiss roll. In this case, the CCA projection that CTW relies on will fail to unroll the second curve, making a good DTW alignment impossible.

We first provide a brief overview of manifold alignment and dynamic time warping before combining the two to formulate manifold warping.

Manifold Alignment

In manifold alignment (Wang and Mahadevan 2009), we are given two data sets $X \in \mathbb{R}^{n_X \times d_X}$ and $Y \in \mathbb{R}^{n_Y \times d_Y}$ where $\mathbb{R}^{m \times n}$ denotes a m by n real matrix. In X and Y, each row is an *instance*. $W^{(X)} \in \mathbb{R}^{n_X \times n_X}$ and $W^{(Y)} \in \mathbb{R}^{n_Y \times n_Y}$ are similarity matrices that provide the similarities between instances in X and Y, respectively. These two matrices are usually constructed as adjacency matrices of nearest neighbor graphs, optionally applying a heat kernel function. In addition, we also have a warping matrix $W^{(X,Y)} \in \mathbb{R}^{n_X \times n_Y}$ that specifies the correspondences between instances in X and Y. Typically,

$$W_{i,j}^{(X,Y)} = \begin{cases} 1 & \text{if } X_i \text{ corresponds to } Y_j \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Suppose we have a mapping that maps X, Y to $F^{(X)} \in \mathbb{R}^{n_X \times d}, F^{(Y)} \in \mathbb{R}^{n_Y \times d}$ in a latent space with dimension $d \leq min(d_x, d_y)$. Note that in terms of the underlying matrix representation, any row i of X is mapped to row i of $F^{(X)}$, and a similar relation holds for Y and $F^{(Y)}$. We form the following loss function for the mapping as follows. The first term indicates that corresponding points across data sets should remain close to each other in the embedding. The last two terms specify that, within an input set, points close in the original space should remain close in the embedding. The factor μ controls how much we want to preserve inter-set correspondences versus local geometry.

$$\begin{aligned} L_1\left(F^{(X)}, F^{(Y)}\right) = \mu & \sum_{i \in X, j \in Y} ||F_i^{(X)} - F_j^{(Y)}||^2 W_{i,j}^{(X,Y)} \\ +(1-\mu) & \sum_{i,j \in X} ||F_i^{(X)} - F_j^{(X)}||^2 W_{i,j}^{(X)} \\ +(1-\mu) & \sum_{i,j \in Y} ||F_i^{(Y)} - F_j^{(Y)}||^2 W_{i,j}^{(Y)} \end{aligned} \quad (2)$$

The notation $i \in X$ simply means $1 \leq i \leq n_X$. We can combine $W^{(X)}$,$W^{(Y)}$, and $W^{(X,Y)}$ into a joint similarity matrix W:

$$W = \begin{bmatrix} (1-\mu)W^{(X)} & \mu W^{(X,Y)} \\ \mu W^{(Y,X)} & (1-\mu)W^{(Y)} \end{bmatrix} \quad (3)$$

Then, we combine $F^{(X)}, F^{(Y)}$ into F where $F = \begin{bmatrix} F^{(X)} \\ F^{(Y)} \end{bmatrix}$. Let $F_{i,k}$ denote the element (i,k) of F and $F_{\cdot,k}$ denote the kth column of F. Then the loss function can be rewritten:

$$\begin{aligned} L_1(F) = & \sum_{i,j} ||F_i - F_j||^2 W_{i,j} \\ = & \sum_k \sum_{i,j} ||F_{i,k} - F_{j,k}||^2 W_{i,j} \\ = & \ 2 \sum_k F_{\cdot,k}^T L F_{\cdot,k} = 2tr\left(F^T L F\right) \end{aligned} \quad (4)$$

, where L is the graph Laplacian of F. Let $D = \begin{bmatrix} D^{(X)} & 0 \\ 0 & D^{(Y)} \end{bmatrix}$ be the diagonal matrix in which each diagonal element is the degree of the corresponding vertex. The optimization problem becomes:

$$\underset{F}{\operatorname{argmin}}\,(L_1) = \underset{F}{\operatorname{argmin}}\left(tr\left(F^T L F\right)\right) \quad (5)$$

This matches the optimization problem of Laplacian Eigenmaps (Belkin and Niyogi 2001), except that in this case the similarity matrix is a joint matrix produced from two similarity matrices. As with Laplacian Eigenmaps, we add a constraint $F^T D F = I$ in order to remove an arbitrary scaling factor as well as to avoid a collapse to a subspace with dimension less than d. For example, this constraint prevents the trivial mapping to a single point. The solution $F = [f_1, f_2, ..., f_d]$ is given by d eigenvectors corresponding to the d smallest nonzero eigenvalues of the general eigenvalue problem: $Lf_i = \lambda D f_i$ for $i = 1, ..., d$.

We can also restrict the mapping to be linear by instead solving the optimization problem:

$$\underset{\phi}{\operatorname{argmin}}\left(tr\left(\phi^T V^T L V \phi\right)\right) \text{ subject to } \phi^T V^T D V \phi = I \quad (6)$$

, where V is the joint data set:

$$V = \begin{bmatrix} X & 0 \\ 0 & Y \end{bmatrix} \quad (7)$$

and $\phi = \begin{bmatrix} \phi^{(X)} \\ \phi^{(Y)} \end{bmatrix}$ is the joint projection, L, D are the graph Laplacian and degree matrix of V respectively. The resultant linear embedding is then $X\phi^{(X)}$ and $Y\phi^{(Y)}$, instead of $F^{(X)}$ and $F^{(Y)}$. The solution for $\phi = [\phi_1, \phi_2, ..., \phi_d]$ is given by d eigenvectors corresponding to the d smallest nonzero eigenvalues of the general eigenvalue problem $V^T L V \phi_i = \lambda V^T D V \phi_i$ for $i = 0, ..., d$.

Dynamic Time Warping

We are given two sequential data sets $X = [x_1^T, \ldots, x_n^T]^T \in \mathbb{R}^{n \times d}$, $Y = [y_1^T, \ldots, y_m^T]^T \in \mathbb{R}^{m \times d}$ in the same space with a distance function $dist : X \times Y \to \mathbb{R}$. Let $P = \{p_1, ..., p_s\}$ represent an alignment between X and Y, where each $p_k = (i, j)$ is a pair of indices such that x_i corresponds with y_j. Since the alignment is restricted to sequentially-

ordered data, we impose the additional constraints:

$$p_1 = (1,1) \quad (8)$$

$$p_s = (n,m) \quad (9)$$

$$p_{k+1} - p_k = (1,0)\ or\ (0,1)\ or\ (1,1) \quad (10)$$

That is, an alignment must match the first and last instances and cannot skip any intermediate instance. This also yields the property that no two sub-alignments cross each other. Figure 3 is an example of a valid alignment. We can also represent the alignment in matrix form W where:

$$W_{i,j} = \begin{cases} 1 & \text{if } (i,j) \in P \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

To ensure that W represents an alignment which satisfies the constraints in Equations 8, 9, 10, W must be in the following form: $W_{1,1} = 1, W_{n,m} = 1$, none of the columns or rows of W is a 0 vector, and there must not be any 0 between any two 1's in a row or column of W. We call a W which satifies these conditions a *DTW matrix*. An optimal alignment is the one which minimizes the loss function with respect to the DTW matrix W:

$$L_2(W) = \sum_{i,j} dist\,(x_i, y_j)\, W_{i,j} \quad (12)$$

A naïve search over the space of all valid alignments would take exponential time; however, dynamic programming can produce an optimal alignment in $O(nm)$.

Manifold Warping

One-step algorithm

We now present a novel framework for aligning two sequentially-ordered data sets that share a common manifold representation. In our approach, we use the warping matrix produced by DTW as a heuristic correspondence matrix for manifold alignment. The proposed algorithm uses alternating projections, picking new correspondences with DTW and reprojecting both inputs using manifold alignment until the loss function is minimized. This presents an improvement over CTW in cases where nonlinear transformations are required to recover the underlying manifold structure of one or both input data sets. We introduce the following loss function for manifold warping:

$$\begin{aligned} &L_3\left(F^{(X)}, F^{(Y)}, W^{(X,Y)}\right) = \\ &\mu \sum_{i \in X, j \in Y} \|F_i^{(X)} - F_j^{(Y)}\|^2 W_{i,j}^{(X,Y)} \\ &+(1-\mu) \sum_{i,j \in X} \|F_i^{(X)} - F_j^{(X)}\|^2 W_{i,j}^{(X)} \\ &+(1-\mu) \sum_{i,j \in Y} \|F_i^{(Y)} - F_j^{(Y)}\|^2 W_{i,j}^{(Y)} \end{aligned} \quad (13)$$

The optimization becomes $\underset{F^{(X)}, F^{(Y)}, W^{(X,Y)}}{\operatorname{argmin}} (L_3)$ subject to $F^T D F = I$ where $F = \begin{bmatrix} F^{(X)} \\ F^{(Y)} \end{bmatrix}$ and $W^{(X,Y)}$ is a DTW matrix. Note that unlike manifold alignment, the correspondence matrix $W^{(X,Y)}$ is now an argument in the optimization problem. The intuition behind this loss function is similar to that of manifold alignment: the last two error terms ensure that the embedding preserves the local geometry of the inputs, and the first term promotes a high quality DTW alignment between two sequential data sets. Again, these goals are controlled by the parameter μ. We now propose an algorithm that minimizes L_3:

Input: X,Y: two time-series data sets
d: latent space dimension
k: number of nearest neighbors used
μ: preserving correspondence vs local geometry factor
Output: $F^{(X)}, F^{(Y)}$: the embeddings of X and Y in the latent space
$W^{(X,Y)}$: the result DTW matrix that provides the alignment of X and Y
begin
  $W^{(X)} \leftarrow$ KNNGraph(X,k)
  $W^{(Y)} \leftarrow$ KNNGraph(Y,k)
  Set $W_{1,1}^{(X,Y)} = W_{n_X,n_Y}^{(X,Y)} = 1$, and 0 everywhere else
  $t \leftarrow 0$
  repeat
    $W = \begin{bmatrix} (1-\mu)W^{(X)} & \mu W^{(X,Y),t} \\ \mu (W^{(X,Y),t})^T & (1-\mu)W^{(Y)} \end{bmatrix}$
    $F^{(X),t+1}, F^{(Y),t+1} \leftarrow$ MA$(F^{(X),t}, F^{(Y),t}, W, d, \mu)$
    $W^{(X,Y),t+1} \leftarrow$ DTW$(F^{(X),t+1}, F^{(Y),t+1})$
    $t \leftarrow t+1$
  until *convergence*;
  $F^{(X)} \leftarrow F^{(X),t}$; $F^{(Y)} \leftarrow F^{(Y),t}$;
  $W^{(X,Y)} \leftarrow W^{(X,Y),t}$
end

Algorithm 1: One-Step Manifold Warping

Figure 3: A valid time-series alignment

In Algorithm 1, MA(X,Y,W,d,μ) is a function that returns the embedding of X, Y in a d dimensional space using manifold alignment with the joint similarity matrix W and parameter μ described in the manifold alignment section. The function DTW(X,Y) returns a DTW matrix after aligning two sequences X, Y using dynamic time warping. The KNNGraph(X,k) function returns the k-nearest neighbors graph of a data set X. In some cases, it is useful to

replace the k-nearest neighbor graph approach with an ϵ-neighborhood graph (Belkin and Niyogi 2001).

Theorem 1. *Let $L_{3,t}$ be the loss function L_3 evaluated at $F^{(X),t}, F^{(Y),t}, W^{(X,Y),t}$ of Algorithm 1. The sequence $L_{3,t}$ converges to a mimimum as $t \to \infty$. Therefore, Algorithm 1 will terminate.*

Proof. In every iteration t, two steps are performed: using manifold alignment to solve for new projections $F^{(X),t+1}, F^{(Y),t+1}$, and using DTW to change the correspondences to $W^{(X,Y),t+1}$.

Recall that the loss function L_1 is just L_3 with fixed $W^{(X,Y)}$. In the first step, with fixed $W^{(X,Y),t}$, Algorithm 1 solves for new projections $F^{(X),t+1}, F^{(Y),t+1}$ using manifold alignment. In manifold alignment section, we showed that manifold alignment's mappings minimize the loss function L_3 when the correspondence matrix is fixed. Hence:

$$\begin{aligned} &L_3(F^{(X),t+1}, F^{(Y),t+1}, W^{(X,Y),t}) \\ &\leq L_3(F^{(X),t}, F^{(Y),t}, W^{(X,Y),t}) \end{aligned} \tag{14}$$

In the second step, the projections are fixed as $F^{(X),t+1}, F^{(Y),t+1}$. Algorithm 1 changes the correspondence matrix from $W^{(X,Y),t}$ to $W^{(X,Y),t+1}$ which does not affect last two terms in L_3. If we replace $dist(F_i^{(X)}, F_j^{(Y)})$ by $\mu||F_i^{(X),t+1} - F_j^{(Y),t+1}||^2$ in the loss function L_2 of DTW, we recover the first term in L_3 of manifold warping. Since $W^{(X,Y),t+1}$ is produced by DTW, it will minimize the first term of L_3. Therefore, we have:

$$\begin{aligned} &\mu \sum_{i \in X, j \in Y} ||F_i^{(X),t+1} - F_j^{(Y),t+1}||^2 W_{i,j}^{(X,Y),t+1} \\ &\leq \mu \sum_{i \in X, j \in Y} ||F_i^{(X),t+1} - F_j^{(Y),t+1}||^2 W_{i,j}^{(X,Y),t} \end{aligned} \tag{15}$$

Changing the correspondence matrix does not affect the last two terms of L_3, so:

$$\begin{aligned} &L_3(F^{(X),t+1}, F^{(Y),t+1}, W^{(X,Y),t+1}) \\ &\leq L_3(F^{(X),t+1}, F^{(Y),t+1}, W^{(X,Y),t}) \\ &\leq L_3(F^{(X),t}, F^{(Y),t}, W^{(X,Y),t}) \text{ from inequality 14} \\ &\Leftrightarrow L_{3,t+1} \leq L_{3,t} \end{aligned} \tag{16}$$

Therefore, $L_{3,t}$ is a decreasing sequence. We also have $L_{3,t} \geq 0$, so it is convergent. Therefore, Algorithm 1 will eventually terminate. □

Two-step algorithm

We now propose an algorithm that exploits the observation that if the local geometries of the two data sets are roughly the same, their similarity matrices will also be very similar to each other. (Wang and Mahadevan 2008) Thus, if we first perform a nonlinear projection on each input set independently, the embeddings are likely to be linearly alignable using either manifold warping or CTW.

In Algorithm 2, DimReduction(X, W, d) is a dimensionality reduction function which maps X with similarity matrix W to a lower dimensional space d. In this paper, we will use Laplacian Eigenmaps to be consistent with manifold alignment even though other methods such as LLE (Roweis and Saul 2000), Isomap (Tenenbaum, De Silva, and Langford 2000), etc. could be applied. LMA(X,Y,W,d,μ) is a function that performs linear manifold alignment described above on X and Y with the joint similarity matrix W, the target dimension d and returns the projection matrices $\phi^{(X)}$ and $\phi^{(Y)}$. We can think of DimReduction as a preprocessing step, then reformulate the loss function as:

Input: X,Y: two time-series data sets
d: latent space dimension
k: number of nearest neighbors used
μ: preserving correspondence/local geometry factor
Output: $F^{(X)}, F^{(Y)}$: the embeddings of X and Y in the latent space
$W^{(X,Y)}$: the result DTW matrix that provides the alignment of X and Y
begin
  $W^{(X)} \leftarrow$ KNNGraph(X,k)
  $W^{(Y)} \leftarrow$ KNNGraph(Y,k)
  $t \leftarrow 0$
  $F^{(X),t} \leftarrow$ DimReduction $\left(F^{(X)}, W^{(X)}, d\right)$
  $F^{(Y),t} \leftarrow$ DimReduction $\left(F^{(Y)}, W^{(Y)}, d\right)$
  repeat
    $W = \begin{bmatrix} (1-\mu)W^{(X)} & \mu W^{(X,Y),t} \\ \mu(W^{(X,Y),t})^T & (1-\mu)W^{(Y)} \end{bmatrix}$
    $\phi^{(Y),t+1}, \phi^{(X),t+1}$
    $\leftarrow$ LMA $\left(F^{(X),t}, F^{(Y),t}, W, d, \mu\right)$
    $F^{(X),t+1} \leftarrow F^{(X),t}\phi^{(X),t+1}$
    $F^{(Y),t+1} \leftarrow F^{(Y),t}\phi^{(Y),t+1}$
    $W^{(X,Y),t+1} \leftarrow$ DTW $\left(F^{(X),t+1}, F^{(Y),t+1}\right)$
    $t \leftarrow t+1$
  until *convergence*;
  $F^{(X)} \leftarrow F^{(X),t}$; $F^{(Y)} \leftarrow F^{(Y),t}$;
  $W^{(X,Y)} \leftarrow W^{(X,Y),t}$
end

Algorithm 2: Two-Step Manifold Warping

$$\begin{aligned} &L_4(\phi^{(X)}, \phi^{(Y)}, W^{(X,Y)}) \\ &= ((1-\mu) \sum_{i,j \in X} ||F_i^{(X)}\phi^{(X)} - F_j^{(X)}\phi^{(X)}||^2 W_{i,j}^{(X)} \\ &+(1-\mu) \sum_{i,j \in Y} ||F_i^{(Y)}\phi^{(Y)} - F_j^{(Y)}\phi^{(Y)}||^2 W_{i,j}^{(Y)} \\ &+\mu \sum_{i \in X, j \in Y} ||F_i^{(X)}\phi^{(X)} - F_j^{(Y)}\phi^{(Y)}||^2 W_{i,j}^{(X,Y)}) \end{aligned} \tag{17}$$

which is the same loss function as in linear manifold alignment except that $W^{(X,Y)}$ is now a variable. The two constraints are the constraint in Equation 6 of linear manifold alignment, and $W^{(X,Y)}$ must be a DTW matrix.

Theorem 2. *Let $L_{4,t}$ be the loss function L_4 evaluated at $\prod_{i=1}^{t} \phi^{(X),i}, \prod_{i=1}^{t} \phi^{(Y),i}, W^{(X,Y),t}$ of Algorithm 2. The sequence $L_{4,t}$ converges to a mimimum as $t \to \infty$. Therefore, Algorithm 2 will terminate.*

Proof. The proof is similar to that of theorem 1. At any iteration t, Algorithm 2 first fixes the correspondence matrix at $W^{(X,Y),t}$. Now let L_4' be like L_4 except that we replace

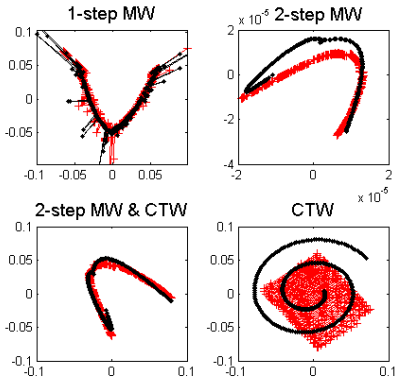

Figure 4: Embedding of two $sin(x^2)$ curves illustrated in Figure 2 onto 2D.

$F_i^{(X)}, F_i^{(Y)}$ by $F_i^{(X),t}, F_i^{(Y),t}$ and Algorithm 2 minimizes L'_4 over $\phi^{(X),t+1}, \phi^{(Y),t+1}$ using linear manifold alignment. Thus,

$$\begin{aligned} & L'_4(\phi^{(X),t+1}, \phi^{(Y),t+1}, W^{(X,Y),t}) \\ & \leq L'_4(I, I, W^{(X,Y),t}) \\ & = L_4(\textstyle\prod_{i=1}^{t} \phi^{(X),i}, \prod_{i=1}^{t} \phi^{(Y),i}, W^{(X,Y),t}) \\ & = L_{4,t} \end{aligned} \tag{18}$$

since $F^{(X),t} = F^{(X),0} \prod_{i=1}^{t} \phi^{(X),i}$ and $F^{(Y),t} = F^{(Y),0} \prod_{i=1}^{t} \phi^{(X),i}$. We also have:

$$\begin{aligned} & L'_4(\phi^{(X),t+1}, \phi^{(Y),t+1}, W^{(X,Y),t}) \\ & = L_4(\textstyle\prod_{i=1}^{t+1} \phi^{(X),i}, \prod_{i=1}^{t+1} \phi^{(Y),i}, W^{(X,Y),t}) \\ & \leq L_{4,t} \end{aligned} \tag{19}$$

Algorithm 2 then performs DTW to change $W^{(X,Y),t}$ to $W^{(X,Y),t+1}$. Using the same argument as in the proof of Theorem 1, we have:

$$\begin{aligned} & L_4(\textstyle\prod_{i=1}^{t+1} \phi^{(X),i}, \prod_{i=1}^{t+1} \phi^{(Y),i}, W^{(X,Y),t+1}) \\ & \leq L_4(\textstyle\prod_{i=1}^{t+1} \phi^{(X),i}, \prod_{i=1}^{t+1} \phi^{(Y),i}, W^{(X,Y),t}) \\ & \leq L_{4,t} \\ & \Leftrightarrow L_{4,t+1} \leq L_{4,t}. \end{aligned} \tag{20}$$

So, the convergence follows. □

Furthermore, when we set $\mu = 1$, the loss function L_4 will become similar to that of CTW. We can also substitute CTW in place of the loop in the algorithm.

Experimental Results

Synthetic data sets

We compare the performance of CTW and manifold warping by trying to align two $sin(x^2)$ curves: one is on the flat plane, the another is projected onto the Swiss roll as illustrated in Figure 2. Some duplicate points are added along the curves to create many-to-one correspondences in the alignment.

As shown in Figure 4, manifold warping produced similar embeddings for two curves based on their local geometry while CTW linearly collapsed the Swiss roll curve onto the plane.

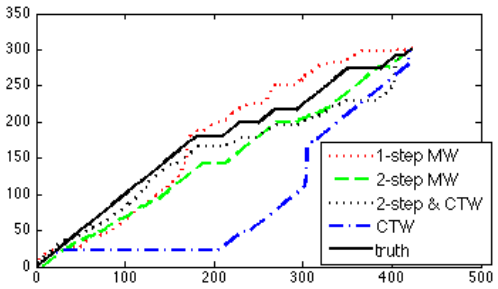

Figure 5: Result warping path of each algorithm's alignment and the ground truth warping path for the two sine curves illustrated in Figure 2

Figure 6: Samples from pairs of COIL-100 image series

As a result, the warping path (that is, the alignment path) produced by manifold warping stays closer to the true warping path than that produced by CTW. The error is calculated by the area between the result path and the ground truth path as suggested in (Zhou and De la Torre 2009). We also normalize the error by dividing by the whole plot area, $n_X \times n_Y$.

The warping paths and the calculated errors, shown in Figure 5 and Table 1, show that manifold warping yields a smaller error than CTW.

COIL-100 data set

We also test these algorithms on a real-world vision data set from the Columbia Object Image Library (COIL100) (S. A. Nene 1996). The corpus consists of different series of images taken of different objects on a rotating platform. Each series has 72 images, each 128×128 pixels. We try to align two series of images of two different objects, with differences in shape and brightness producing very different high-dimensional representations. To demonstrate our algorithm's ability to work with data sets of different dimensionality, we compress one image series to a smaller resolution (64×64 pixels). Additionally, some duplicate images are added to each series, to ensure that the correct mapping is not trivially one-to-one.

In both experiments, manifold warping methods achieve alignments with a much smaller error than CTW. The depiction in Figure 7 provides an intuitive picture of the manifold warping algorithm. In the first projection to two dimensions, both image series are mapped to circles. The next several iterations rotate these circles to match the first and last points, then the points in between. For the case of one-step Manifold Warping (where all mappings are nonlinear), we pick a small μ to prioritize preserving local geometry of each se-

ries. This avoids over-fitting the embedding to a potentially bad intermediate DTW correspondence.

We perform the experiment with two pairs of COIL image series, illustrated in Figure 6.

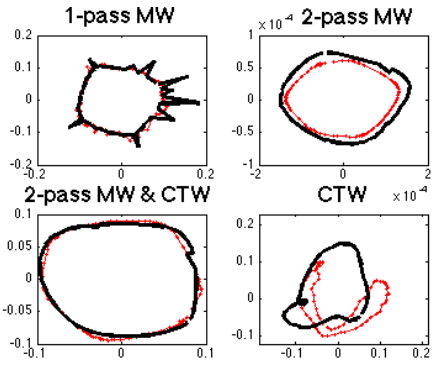

Figure 7: 2D embedding of dog/cat toy image series (Figure 6).

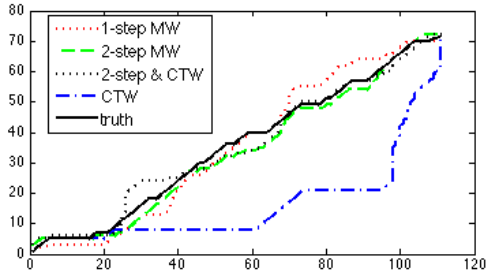

Figure 8: Warping paths for dog/cat toy images

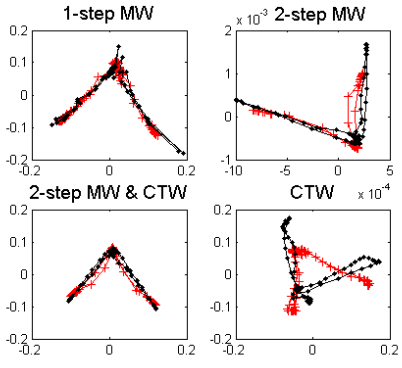

Figure 9: 2D embedding of cup image series (Figure 6).

Kitchen data set

Our last experiment uses the kitchen data set (De la Torre et al. 2008) from the CMU Quality of Life Grand Challenge, which records human subjects cooking a variety of dishes. Here, we attempt nonlinear alignments between the same subject and task, across different sensors.

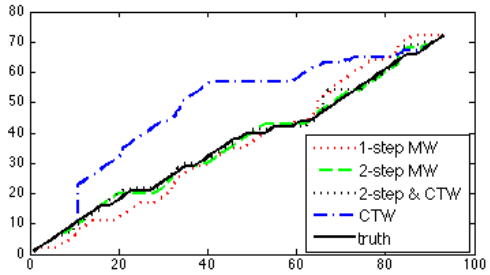

Figure 10: Warping paths for cup images

	Synthetic	Dog +Cat	Cups	Kitchen Brownie	Kitchen Egg
1-step MW	0.0768	0.0447	0.0464	**0.0257**	**0.0267**
2-step MW	0.0817	**0.0282**	**0.0125**	0.0396	0.0469
2-step MW	**0.0652**	0.0298	0.0143	0.0772	0.0479
CTW	0.2784	0.2656	0.1668	0.0966	0.0510

Table 1: Alignment error across algorithms and data sets

Our experiment considers two separate views of the same moment in time, during which the subject prepares a brownie and an egg. The two views are 9-dimensional inertial measurement unit (IMU) readings and 87-dimensional motion capture suit coordinates (MOCAP). Aligning two views of the same task provides a straightforward evaluation metric, because the time stamps on each reading yield ground-truth correspondence information. To make the problem computationally feasible, we subsampled the original data sets. Each manifold warping method performs better than CTW, based on the results shown below in Figure 11, Figure 12, and Table 1.

Discussion and Future Work

Due to the lack of a linearity constraint, manifold warping consistently performs better than CTW when the inputs lie

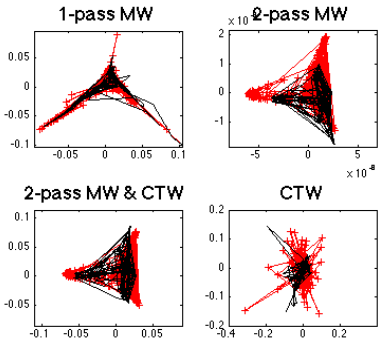

Figure 11: The embeddings of two sensors measurements series IMU and MOCAP for a kitchen task involving brownie.

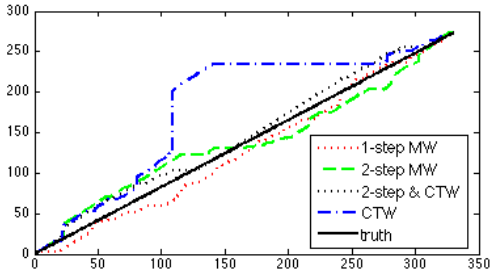

Figure 12: Result warping path of each algorithm's alignment for the kitchen data set - brownie experiment

on manifolds that are not accessible via linear transformations. Even in the linear case, the alignment quality of manifold warping is at just as good as CTW.

Importantly, this improved alignment quality does not impose significant runtime overhead. Both algorithms rely on the same DTW step, and tuned implementations of manifold alignment are comparable in runtime to the CCA step used in CTW. We found that while each manifold warping iteration is marginally slower than a similar CTW iteration, manifold warping tends to converge with fewer steps. Speedups may be possible by using a relaxed variation of DTW for the first few iterations, and parallelizing the initial alignments of the two-step algorithm.

Both the one-step and two-step manifold warping algorithms are natural extensions of canonical time warping on manifolds, and the results presented in this paper indicate that this added information has the potential to significantly improve alignment quality.

Several variants of DTW and manifold alignment may prove beneficial within the manifold warping framework. Future work will explore multiscale and local alignment strategies, enabling broader applications for manifold warping.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant Nos. NSF CCF-1025120, IIS-0534999, and IIS-0803288.

References

Anderson, T. 2003. *An introduction to multivariate statistical analysis*. Wiley series in probability and mathematical statistics. Probability and mathematical statistics. Wiley-Interscience.

Belkin, M., and Niyogi, P. 2001. Laplacian eigenmaps and spectral techniques for embedding and clustering. *Advances in neural information processing systems* 14:585–591.

De la Torre, F.; Hodgins, J.; Bargteil, A.; Martin, X.; Macey, J.; Collado, A.; and Beltran, P. 2008. Guide to the carnegie mellon university multimodal activity (cmu-mmac) database.

Ham, J.; Lee, D.; and Saul, L. 2005. Semisupervised alignment of manifolds. In *Proceedings of the Annual Conference on Uncertainty in Artificial Intelligence, Z. Ghahramani and R. Cowell, Eds*, volume 10, 120–127.

Hsu, E.; Pulli, K.; and Popović, J. 2005. Style translation for human motion. In *ACM Transactions on Graphics (TOG)*, volume 24, 1082–1089. ACM.

Jolliffe, I. 2002. *Principal component analysis*. Springer series in statistics. Springer-Verlag.

Junejo, I.; Dexter, E.; Laptev, I.; Pérez, P.; et al. 2008. Cross-view action recognition from temporal self-similarities.

Keogh, E., and Pazzani, M. 2001. Derivative dynamic time warping. In *First SIAM international conference on data mining*, 5–7.

Roweis, S., and Saul, L. 2000. Nonlinear dimensionality reduction by locally linear embedding. *Science* 290(2323–232).

S. A. Nene, S. K. Nayar, H. M. 1996. Columbia object image library (coil-100). *Technical Report CUCS-006-96*.

Sakoe, H., and Chiba, S. 1978. Dynamic programming algorithm optimization for spoken word recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions on* 26(1):43–49.

Tenenbaum, J.; De Silva, V.; and Langford, J. 2000. A global geometric framework for nonlinear dimensionality reduction. *Science* 290(5500):2319–2323.

Wang, C., and Mahadevan, S. 2008. Manifold alignment using procrustes analysis. In *Proceedings of the 25th international conference on Machine learning*, 1120–1127. ACM.

Wang, C., and Mahadevan, S. 2009. A general framework for manifold alignment. In *AAAI Fall Symposium on Manifold Learning and its Applications*.

Zhou, F., and De la Torre, F. 2009. Canonical time warping for alignment of human behavior. *Advances in Neural Information Processing Systems (NIPS)* 1–9.

Zhou, F.; Torre, F.; and Hodgins, J. 2008. Aligned cluster analysis for temporal segmentation of human motion. In *Automatic Face & Gesture Recognition, 2008. FG'08. 8th IEEE International Conference on*, 1–7. IEEE.