

# Dynamic Manifold Warping for View Invariant Action Recognition

Dian Gong and Gérard Medioni

Institute for Robotics and Intelligent Systems, University of Southern California  
Los Angeles, CA, 90089

{diangong|medioni}@usc.edu

## Abstract

We address the problem of learning view-invariant 3D models of human motion from motion capture data, in order to recognize human actions from a monocular video sequence with arbitrary viewpoint. We propose a Spatio-Temporal Manifold (STM) model to analyze non-linear multivariate time series with latent spatial structure and apply it to recognize actions in the joint-trajectories space. Based on STM, a novel alignment algorithm Dynamic Manifold Warping (DMW) and a robust motion similarity metric are proposed for human action sequences, both in 2D and 3D. DMW extends previous works on spatio-temporal alignment by incorporating manifold learning. We evaluate and compare the approach to state-of-the-art methods on motion capture data and realistic videos. Experimental results demonstrate the effectiveness of our approach, which yields visually appealing alignment results, produces higher action recognition accuracy, and can recognize actions from arbitrary views with partial occlusion.

## 1. Introduction

Recognizing human action is a key component in many vision applications, such as video surveillance, 3D human pose estimation and video indexing. Extracting this high level information from motion capture or video data is the problem we propose to address here. Although significant progress has been made in action recognition [2, 4, 10, 11, 24, 25], the problem remains inherently challenging due to significant intra-class variations, viewpoint change, partial occlusion and background dynamic variations. A key limitation of many action-recognition approaches is that their models are learned from single 2D view video features on individual datasets and thus unable to handle arbitrary view change or scale and background variations. Also, since they are not generalizable across different datasets, retraining is necessary for any new dataset.

Our approach is motivated by the requirement of view-invariant action recognition and the fact that the existing

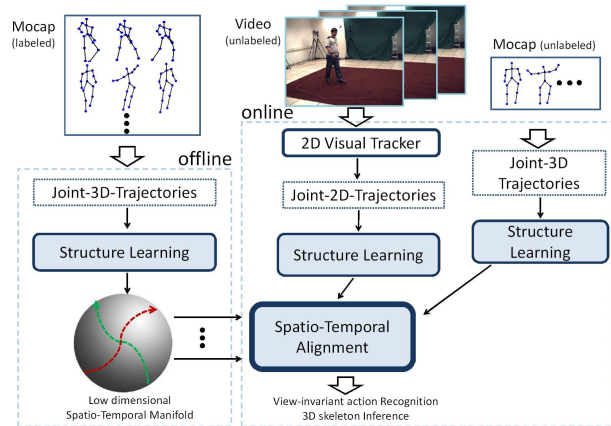


Figure 1. Flow chart of the proposed approach.

human motion capture (Mocap) data provides useful knowledge to understand the intrinsic motion structure. We address the problem of modeling and analyzing human motion in the joint-trajectories space. Our overall approach is sketched in Fig. 1, which has the following modules:

(1) Given a labeled Mocap sequence with  $M$  markers in 3D, which is a  $3M$ -dimensional sequential data ( $3MD+t$ ), the low dimensional manifold structure (i.e., tangent space, geodesics distance, etc) is learnt from the Spatio-Temporal Manifold (STM) Model (offline, as given in Sec. 3).

(2) For other unlabeled motion sequences in 3D, after the intrinsic structure learning (1), we can calculate a motion similarity score with each labeled motion sequence by the proposed approach Dynamic Manifold Warping (DMW) (Sec. 4), to do action recognition (online). DMW is designed as unsupervised learning algorithm, and can be applied to align multivariate time series in general.

(3) More interestingly, our system can recognize actions in videos (Sec. 5). To apply our approach to videos, we have a pre-processing step to extract joint 2D trajectories from image observations by a tracker. 2D tracking results from single view videos are often *noisy* and have *occlusions*, but our structure learning algorithms (1) remain the same and

our alignment approach (2) can naturally handle 2D input. Furthermore, the complete 3MD skeleton can be inferred from partially occluded tracks, as a by-product.

#### Contributions.

- **One or very few examples** are required in each action category in the training stage, compared to 100s for many learning approaches.
- **View invariance:** low dimensional motion manifolds are learnt from 3D Mocap data, and our alignment algorithms can handle 2D input (arbitrary viewpoint); these two features enable our system to recognize actions regardless of the viewpoint.
- **Transfer learning:** when applying our approach to a video dataset, there is no training process on this dataset and people in these videos do not necessarily appear in the labeled Mocap sequences. Thus, our approach can be considered as a *transfer learning* framework, i.e., the knowledge from labeled Mocap data can be *adapted* to any action video.
- **Intra and inter-person variations:** a person repeating an action differently, or two people performing an action with differences in both pose style and motion dynamic, are handled by combining temporal and spatial alignment together.
- **Occlusion handling:** in order to recognize actions from videos, key points trajectories need to be tracked. Instead of  $M$  key points, often only  $K$  visible points can be tracked during the action ( $K \leq M$ ), such as a side view video of a walking man. Our system can handle these 2D noisy tracked trajectories, even with occlusion ( $2KD+t$ ).
- **Alignment:** considering alignment algorithm only, we introduce DMW to align two human motion sequences and provide a robust motion similarity score. As the first attempt to incorporate non-linear latent variable model and spatio-temporal alignment, DMW outperforms state-of-the-art competing approaches in our applications.

## 2. Related Work

Action recognition is a multifaceted field, our discussion focuses on view-invariant methods, and readers can refer to a recent review [13] for more details.

**View Invariant Recognition.** Hidden Markov Model (HMM) is built on 3D joint-trajectories (Mocap) to capture the dynamic information of human motion [7]. The claimed advantage of the 3D HMM model is that the dependence on view point and illumination is removed. However, HMM requires large amount of training data in relatively high dimensional space (e.g. 67) and the HMM model structure must be adaptively designed for specific application domains. These may be potential factors that make the recognition performance unsatisfactory, and AdaBoost is used to improve the accuracy [7]. View-independence is also addressed in [9] by rendering Mocap data of vari-

ous actions from multiple viewpoints, which is a time and storage consuming process. Another class of methods relies on recovering 3D poses information from silhouettes. In [23], 3D models are projected onto 2D silhouettes with respect to different view point, and [25] detects 2D feature first and then back-projects them to action features based on a 3D visual hull. These methods require a computationally expensive search process over model parameters to find the best match between 2D features and 3D model. Very recently, in [24], a 3D HoG descriptor was proposed to handle view point change, and this approach requires the multiple view camera settings for training data to achieve the view-invariant recognition. Departing from these methods, our recognition process *does not* require 2D pose rendering or parameters search. Our trajectory features are located at body skeleton’s key locations, with explicit semantic meaning, allowing our system to be directly applied to arbitrary scene without datasets dependent training.

**Dynamic Manifold Model.** Non-linear manifold learning and latent variable modeling (LVM) is prominent in machine learning research in the past decade [19, 22]. In particular, some probabilistic latent variable frameworks, i.e., GP-LVM, GPDM and its variants [5, 21], focus on motion capture data and try to capture the intrinsic structure of human motion, which is further applied to 3D monocular people tracking [20].

**Motion Sequence Matching.** Canonical Component Analysis (CCA) [1], proposed for learning the shared subspace between two high dimensional features, has been used as the spatial matching algorithm for activity recognition from video [3]. Video synchronization is addressed as a temporal alignment problem in [14, 18], which uses dynamic time warping (DTW) or its variants. [12] uses linear transformation model to solve the spatio-temporal alignment problem for multiple unsynchronized video sequences. Very recently, as an elegant extension of CCA and DTW, Canonical Time Warping (CTW) is proposed for spatio-temporal alignment of two multivariate time series and applied to align human motion sequences between two subjects [26]. DMW is different from all previous approaches, e.g, DTW and CTW, and a detailed analysis is given in Sec. 4.

## 3. Spatio-Temporal Manifold Model for Human Motion Data

Suppose there is a  $d$ -dimensional submanifold  $\mathfrak{M}$  embedded in an ambient space of dimensionality  $D \gg d$ . We use a latent variable model (LVM) to represent  $\mathfrak{M}$  as a mapping between the intrinsic space and the ambient space:  $f : \mathbb{R}^d \rightarrow \mathbb{R}^D$  and  $\mathbf{x} = f(\boldsymbol{\tau}) + \boldsymbol{\epsilon}$ , where  $\mathbf{x} \in \mathbb{R}^D$  is the observation variable,  $\boldsymbol{\tau} \in \mathbb{R}^d$  is the latent variable and  $\boldsymbol{\epsilon} \in \mathbb{R}^D$  is the noise. To incorporate the temporal dimension into standard LVM, we propose a novel model as follows.

*Definition: a spatio-temporal manifold (STM) is a di-*

rected traversing path  $\mathfrak{M}_p$  (with boundary or compact) on a spatial-manifold  $\mathfrak{M}$ , and further embedded in  $\mathbb{R}^D$ .

A traversing path  $\mathfrak{M}_p$  can be intuitively thought as a point moving on  $\mathfrak{M}$  from a starting point at time  $t_1$  ( $\tau_{start}, \mathbf{x}_{start}$ ) to an ending point ( $\tau_{end}, \mathbf{x}_{end}$ ) at time  $t_2$ . A path is not just a subset of  $\mathfrak{M}$  which “looks like” a curve, it also includes a natural parametrization as,  $g_{\zeta \rightarrow \tau} : [0, 1] \rightarrow \mathfrak{M}$ , s.t.  $g(0) = \tau_{start}$  and  $g(1) = \tau_{end}$ . So, a new latent variable  $\zeta \in [0, 1]$  is associated with every point in this path. Since  $\mathfrak{M}$  is embedded in  $\mathbb{R}^D$  by  $f(\cdot)$ , essentially the traversing path can be described as a non-linear multivariate time series as,

$$\mathbf{x}(t) = h(\zeta_t) + \epsilon \quad (1)$$

where  $h(\cdot) = f(g(\cdot))$  is called *STM mapping function* which maps  $\zeta$  to  $\mathbb{R}^D$  ( $h^{-1}(\cdot)$  is  $\mathbb{R}^D$  to  $\zeta$ ).

**STM for Human Motion Data.** Given a length  $L_x$  human action sequence (e.g. stretching), the joint-trajectories can be represented as a matrix  $\mathbf{X}_{1:L_x} = [\mathbf{x}_1 \ \mathbf{x}_2 \ \dots \ \mathbf{x}_{L_x}] \in \mathbb{R}^{D \times L_x}$ , where  $\mathbf{x}_t$  is the joint-positions at temporal index  $t$ . In 3D (Mocap),  $\mathbf{x}_t = [\mathbf{p}_1^t, \dots, \mathbf{p}_M^t]^T \in \mathbb{R}^{3M \times 1}$  and  $\mathbf{p}_i^t = (p_{i1}^t \ p_{i2}^t \ p_{i3}^t)$  is the coordinate of the  $i$ th marker in  $\mathbb{R}^3$ . Or in 2D (tracking trajectories),  $\mathbf{x}_t = [\mathbf{u}_1^t, \dots, \mathbf{u}_K^t]^T \in \mathbb{R}^{2K \times 1}$  ( $K \leq M$ ),  $\mathbf{u}_i^t = (u_{i1}^t \ u_{i2}^t)$  is the pixel location of the  $i$ th key point. Although  $\mathbf{x}_t$  lies in a high dimensional space, the natural property of human pose suggests  $\mathbf{x}_i$  has lower intrinsic number of degrees of freedom. So,  $\mathbf{X}_{1:L_x}$  is just a sequence of sampled observations on a STM. The newly introduced variable  $\zeta$  is assigned a semantic meaning which indicates the “completion” degree of an action. For a complete action sequence, i.e., an *action unit*, we assume the starting point of the action is  $\zeta = 0$  and the ending point is  $\zeta = 1$ <sup>1</sup>. Without specific notes, an action sequence is corresponding to a complete *action unit* in this paper.

### 3.1. Structure Learning

The set of all points in a STM is empirically found to be a 1D *smooth* manifold by Tensor Voting. Given  $L$  ordered data points  $\{\mathbf{x}_t\}_{t=1}^{L_x}$  sampled from a STM, the *goal of learning* is to recover the latent “completion” variable  $\zeta_t$  (or  $h^{-1}(\cdot)$ ) from these samples. Note that our goal is different from most latent variable models, which aim to identify  $\tau$  [19] and sometimes  $f(\cdot)$  [5, 21, 22].

**Estimating  $d_{Geo}(\cdot)$ .** We use Tensor Voting to estimate the minimum traversing distance between  $\mathbf{x}_s$  and  $\mathbf{x}_{s+1}$  ( $1 \leq s \leq L_x - 1$ ). It approximates the geodesic distance  $d_{Geo}(\cdot)$ . Tensor Voting is a non-parametric framework proposed to estimate the geometric information of manifolds [8]. Let  $\mathbf{x}_s(0) = \mathbf{x}_s$ , we have

$$d_{Geo}(\mathbf{x}_s, \mathbf{x}_{s+1}; \mathfrak{M}_p) \approx \sum_{r=0}^R \|\mathbf{x}_s(r) - \mathbf{x}_{s+1}(r)\|_{L_2} \quad (2)$$

<sup>1</sup>For periodic motion, e.g., walking, it defines a motion cycle.

where  $\mathbf{x}_s(r+1)$  is updated from the current point  $\mathbf{x}_s(r)$ ,

$$\mathbf{x}_s(k+1) = \mathbf{x}_s(k) + \alpha \mathbf{J}^*(\mathbf{x}_s(r)) \mathbf{J}^*(\mathbf{x}_s(r))^T (\mathbf{x}_{s+1} - \mathbf{x}_s(r))$$

until  $\mathbf{x}_s(r+1)$  converges to  $\mathbf{x}_{s+1}$ .  $\alpha$  is a step length, and  $\mathbf{J}^*(\mathbf{x}_s(r))$  is the tangent space estimation on  $\mathbf{x}_s(r)$  by Tensor Voting [8].

**Recovering  $\zeta_t$ .** A two stage approach is possible, first estimate  $\tau$  (or  $f(\cdot)$ ) on a collection of time series, and then optimize  $\{\zeta_{1:L_x}\}$ . Instead, we propose a solution which performs direct estimation for an individual sequence based on the learnt approximated geodesic distance.

$$\zeta_t^* = \frac{\sum_{s=1}^{t-1} d_{Geo}(\mathbf{x}_s, \mathbf{x}_{s+1}; \mathfrak{M}_p)}{\sum_{s=1}^{L_x-1} d_{Geo}(\mathbf{x}_s, \mathbf{x}_{s+1}; \mathfrak{M}_p)} \quad (3)$$

Since the traversing path is continuous and smooth, the global geodesic distance is approximately decomposed to the sum of the local distances, inspired by ISOMAP [19]. A visualization of  $\zeta_t$  in a motion sequence is given in Fig. 2.

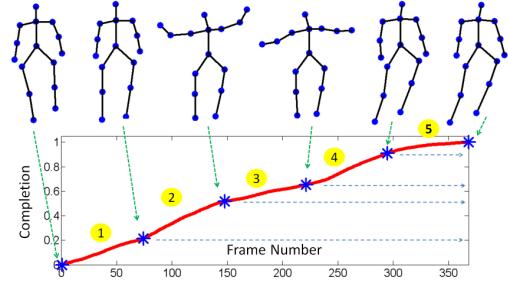


Figure 2. An illustration of the non-linearity of  $\zeta(t)$ . Top, 6 samples in an action “stretching”; bottom, estimated  $\zeta(t)$ .

## 4. Motion Sequence Matching

In order to recognize an action, for any two human action sequences  $\mathbf{X}_{1:L_x} \in \mathbb{R}^{D_x \times L_x} (\mathfrak{M}_p^x)$  and  $\mathbf{Y}_{1:L_y} \in \mathbb{R}^{D_y \times L_y} (\mathfrak{M}_p^y)$ <sup>2</sup>, we need to calculate the motion distance score  $S(\mathbf{X}_{1:L_x}, \mathbf{Y}_{1:L_y})$ , even if two sequences are of different actions.  $S(\mathbf{X}_{1:L_x}, \mathbf{Y}_{1:L_y})$  is calculated based on proper spatial and temporal alignment. The problem is inherently challenging because of the large spatial/temporal scale difference between human actions, ambiguity between human poses, as well as the inter/intra subject variability [26].

Under the STM model, we propose a novel approach Dynamic Manifold Warping (DMW), which aligns two multivariate time series with intrinsic manifold structure and provides a robust motion similarity score.

<sup>2</sup>Both can be 3D joint-trajectories, or one 3D and the other 2D joint-trajectories from a video clip.

## 4.1. Temporal Alignment

The first part of DMW is temporal alignment, which is called Dynamic Manifold Temporal Warping (DMTW).

**Formulation.** Given two time series  $\mathbf{X}_{1:L_x} \in \mathbb{R}^{D_x \times L_x}$  and  $\mathbf{Y}_{1:L_y} \in \mathbb{R}^{D_y \times L_y}$ , find the optimal alignment path  $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_L] \in \mathbb{R}^{2 \times L}$  by minimizing the following loss function ( $\|\cdot\|_F$  is the Frobenius norm operator),

$$\begin{aligned} \mathcal{L}_{DMTW}(\mathcal{F}_x(\cdot), \mathcal{F}_y(\cdot), \mathbf{W}_x, \mathbf{W}_y) \\ = \|\mathcal{F}_x(\mathbf{X}_{1:L_x})\mathbf{W}_x^T - \mathcal{F}_y(\mathbf{Y}_{1:L_y})\mathbf{W}_y^T\|_F^2 \end{aligned} \quad (4)$$

where  $L$  is the length of the aligned path,  $\mathbf{W}_x = \{w_{t,t_x}^x\} \in \{0, 1\}^{L \times L_x}$  and  $\mathbf{W}_y = \{w_{t,t_y}^y\} \in \{0, 1\}^{L \times L_y}$  are binary selection matrices encoding the temporal alignment path  $\mathbf{Q}$  [26].  $w_{t,t_x}^x = w_{t,t_y}^y = 1$  is equivalent to  $\mathbf{q}_t = [t_x \ t_y]^T$ , which means  $\mathbf{x}_{t_x}$  corresponds to  $\mathbf{y}_{t_y}$  at step  $t$  in the alignment path.  $\mathcal{F}(\cdot)$  maps  $\mathbf{X}_{1:L_x}$  and  $\mathbf{Y}_{1:L_y}$  to a shared subspace with the same dimensionality. Essentially,  $\mathcal{F}_x(\cdot)$  and  $\mathcal{F}_y(\cdot)$  are *spatial mapping* functions and  $\mathbf{W}_x$  and  $\mathbf{W}_y$  are *temporal warping* matrices.

**Algorithm.** The key factor in temporal alignment is to build the *temporal aligning matrix*  $\mathbb{A} = \{a_{t_x, t_y}\}_{(L_x \times L_y)}$ , where  $a_{t_x, t_y}$  is the metric between  $\mathbf{x}_{t_x}$  and  $\mathbf{y}_{t_y}$  in the transform domain (induced by  $\mathcal{F}(\cdot)$ ). If  $\mathcal{F}(\cdot)$  is an identity function, then  $\mathbb{A}$  is just the *Euclidean* metric matrix, which is the case of DTW [14, 18].

As a straightforward association, constructing  $\mathbb{A}$  is related to unsupervised metric learning [22]. But these works focus on *spatial* metric learning, not *temporal* alignment. In contrast, STM model incorporates both temporal information and latent spatial structure. Thus, the difficulties of temporal alignment, non-linear motion dynamic and spatial transformation, are well handled by the latent variable  $\zeta_t$  (or  $h^{-1}(\cdot)$ ) learnt from STM.

Step 1. Under the STM model in Sec. 3, we choose  $\mathcal{F}_x(\mathbf{X}_{1:L_x})$  to be  $\zeta_{1:L_x}^x \in \mathbb{R}^{1 \times L_x}$  and  $\mathcal{F}_y(\mathbf{X}_{1:L_y})$  to be  $\zeta_{1:L_y}^y \in \mathbb{R}^{1 \times L_y}$ .  $\zeta_t$  represents the universal structure for all STMs, making aligning two sequences with different actions possible. If the sequence is labeled data (i.e. Mocap), then eq. 2 can be used to estimate  $d_{Geo}(\cdot)$ . Otherwise, instead of performing the variable-length path estimation, we can directly estimate the  $d_{Geo}(\cdot)$  by using the fixed-length (i.e., 1 or 2) traversing path, without re-performing Tensor Voting at each step. After learning  $d_{Geo}(\cdot)$ , combining with eq. 3, we can obtain the estimated results for  $\zeta_{1:L_x}^x$  and  $\zeta_{1:L_y}^y$ , denoted as  $\widetilde{\zeta}^x \in \mathbb{R}^{1 \times L_x}$  and  $\widetilde{\zeta}^y \in \mathbb{R}^{1 \times L_y}$ . Rigorously, the derivative of  $\zeta_t$  measures the normalized geodesic distances between consecutive samples, and  $\zeta_t$  can be recovered with arbitrary small error by eq. 3 when a sufficient number of samples on the STM is available<sup>3</sup>.

<sup>3</sup>Under the local isometric embedding assumptions.

Step 2. Replace  $\mathcal{F}_x(\cdot)$  and  $\mathcal{F}_y(\cdot)$  with  $\zeta^x$  and  $\zeta^y$  in eq. 4,  $\mathcal{L}_{DMTW}$  reduces to the following formulation,

$$\mathcal{L}_{DMTW}(\mathbf{W}_x, \mathbf{W}_y) = \|\widetilde{\zeta}^x \mathbf{W}_x^T - \widetilde{\zeta}^y \mathbf{W}_y^T\|_F^2 \quad (5)$$

This is equivalent to performing DTW in the transform domain, i.e.,  $\zeta^x$  and  $\zeta^y$ . The *temporal aligning matrix*  $\mathbb{A} = \{a_{t_x, t_y}\}$  with  $a_{t_x, t_y} = (\widetilde{\zeta}_{t_x}^x - \widetilde{\zeta}_{t_y}^y)^2$ , is a compact representation of  $\widetilde{\zeta}^x$  and  $\widetilde{\zeta}^y$ . Optimizing eq. 5 results in *variable length* path  $L$  (vary from  $\max(L_x, L_y)$  to  $L_x + L_y - 1$ ), so is not proper for similarity metric. Thus, referenced DTW is proposed to fix the path length by setting one warping matrix to be identity,

$$\|\widetilde{\zeta}^x \mathbf{I}_{L_x} - \widetilde{\zeta}^y \mathbf{W}_y^T\|_F^2 \quad (6)$$

where  $\mathbf{I}_{L_x}$  is an identity matrix.  $\mathbf{X}_{1:L_x}$  is chosen as the reference sequence, and  $\mathbf{Y}_{1:L_y}$  is aligned to  $\mathbf{X}_{1:L_x}$  by the warping matrix  $\mathbf{W}_y \in \mathbb{R}^{L_x \times L_y}$ . The path  $\mathbf{Q}$  in eq. 6 has fixed length  $L_x$ . Since  $\widetilde{\zeta}^x$  and  $\widetilde{\zeta}^y$  are monotonically increasing sequences, dynamic programming provides an efficient solution ( $O(L_x L_y)$ ) and satisfies boundary conditions, that  $\mathbf{q}_1 = [1 \ 1]^T$  and  $\mathbf{q}_{L_x} = [L_x \ L_y]^T$ .

**Analysis.** While the objective function of DMTW (eq. 4) is inspired from CTW [26], key differences exist. CTW uses linear  $\mathcal{F}(\cdot)$ , and its optimization process may lead to local extreme since the objective function is non-convex. In DMTW,  $\mathcal{F}(\cdot)$  is chosen as the non-linear mapping  $h^{-1}(\cdot)$ , which can guarantee a global solution. It is notable that CTW does not need smooth manifold assumption, and thus has more general applications than DMTW, while DMTW focuses on time series with intrinsic manifold structure.

DMTW is also related with Profile Models [6]. Although the ideas of Profile and  $\zeta_t$  seem similar, they differ in many aspects. In particular, Profile Models need multiple training examples and the size of the discrete Profile space increases exponentially with the precision requirement, which is not only computationally impractical but also causes over-fitting. In contrast, DMTW does not need training stage, and  $\zeta_t$  is continuous in nature.

## 4.2. Temporally Local Spatial Alignment

**Formulation.** After DMTW, spatial alignment is performed to leverage the subjects' variability, i.e., body-skeleton scales variations, or 2D viewpoint variations. In particular, we propose Dynamic Manifold Spatial Warping (DMSW) to focus on *temporally local* manifolds as,

$$\|\mathbf{V}_x(\mathcal{U}(\mathbf{X}_{t_1:t_2})) - \mathbf{V}_y(\mathcal{U}(\widetilde{\mathbf{Y}}_{t_1:t_2}))\|_F^2 \quad (7)$$

$\mathbf{X}_{t_1:t_2} \in \mathbb{R}^{D_x \times (t_2 - t_1 + 1)}$  are consecutive frame features  $\mathbf{x}_{t_1}$  to  $\mathbf{x}_{t_2}$  in the reference sequence, and  $\widetilde{\mathbf{Y}}_{t_1:t_2} \in$

$\mathbb{R}^{D_y \times (t_2 - t_1 + 1)}$  are temporally corresponding samples in the aligned sequence  $\tilde{\mathbf{Y}}_{1:L_x}$ .  $\mathbf{V}_x(\cdot)$  is the spatial alignment function (same for  $\mathbf{V}_y(\cdot)$ ) and  $\mathcal{U}(\cdot)$  is the pre-defined feature extraction function.

**Algorithm.** Denoting the extracted features by  $\mathcal{U}(\cdot)$  as two zero mean feature sets,  $\mathbf{U}_x \in \mathbb{R}^{d_1 \times n}$  and  $\mathbf{U}_y \in \mathbb{R}^{d_2 \times n}$ , we consider an unsupervised learning approach, i.e., Canonical Correlation Analysis (CCA), in which a pair of linear alignment matrices is optimized in the sense of *maximizing* the correlation  $E(\cdot)$  in transformed features as follows,

$$\begin{aligned} E(\mathbf{V}_x, \mathbf{V}_y) &= \text{Tr}(\mathbf{V}_x^T \mathbf{U}_x (\mathbf{V}_y^T \mathbf{U}_y)^T) \\ \text{s.t.}, \mathbf{V}_x^T \mathbf{U}_x \mathbf{U}_x^T \mathbf{V}_x &= \mathbf{V}_y^T \mathbf{U}_y \mathbf{U}_y^T \mathbf{V}_y = \mathbf{I}_d \end{aligned} \quad (8)$$

where  $\mathbf{V}_x \in \mathbb{R}^{d_1 \times d}$  and  $\mathbf{V}_y \in \mathbb{R}^{d_2 \times d}$  are two *linear* spatial alignment matrices for  $\mathbf{U}_x$  and  $\mathbf{U}_y$ , and  $\mathbf{I}_d$  is the identity matrix of size  $d \times d$ .  $\text{Tr}(\cdot)$  is the trace operator. Minimizing this objective function is equivalent to solving a generalized eigenvalue problem [1]. The metric can be calculated as,

$$D_{DMSW}(\mathbf{X}_{t_1:t_2}, \tilde{\mathbf{Y}}_{t_1:t_2}) = \|\mathbf{V}_x^{*T} \mathbf{U}_x - \mathbf{V}_y^{*T} \mathbf{U}_y\|_F^2 \quad (9)$$

$\mathbf{V}_x^* \in \mathbb{R}^{d_1 \times d}$  and  $\mathbf{V}_y^* \in \mathbb{R}^{d_2 \times d}$  are the solutions of eq. 8. Eq. 9 can handle two feature sets with different dimensionalities, making the alignment between 2D and 3D possible.

Directly applying CCA to local manifolds (segments) is often impossible due to *limited number of samples* v.s. *high dimensionality*, i.e.,  $t_2 - t_1 + 1 < D_x$  (or  $D_y$ ). This problem is handled by exploring the implicit structure of the joint-position space in two proposed feature extraction functions. Instead of treating  $\mathbf{x}_t \in \mathbb{R}^{D_x \times 1}$  (or  $\mathbf{y}_t$ ) as a multi-dimensional vector, an implicit structure in the joint-position space is considered. In sec. 3,  $\mathbf{x}_t = [\mathbf{p}_1^t, \dots, \mathbf{p}_M^t]^T \in \mathbb{R}^{3M \times 1}$ , and we reformulate  $\mathbf{x}_t$  as,

$$\mathbf{x}_t = \begin{pmatrix} p_{11} & \dots & p_{M1} \\ p_{12} & \dots & p_{M2} \\ p_{13} & \dots & p_{M3} \end{pmatrix} \in \mathbb{R}^{3 \times M} \quad (10)$$

which turns to be  $M$  samples in  $\mathbb{R}^3$  (similar operation for  $\mathbf{x}_t \in \mathbb{R}^{2K}$  to  $\mathbf{x}_t \in \mathbb{R}^{2 \times K}$ ). This operation is defined as  $\mathcal{T}^{3D} : \mathbb{R}^{3M} \rightarrow \mathbb{R}^{3 \times M}$ , or  $\mathcal{T}^{2D} : \mathbb{R}^{2K} \rightarrow \mathbb{R}^{2 \times K}$ . The first feature extraction function is chosen as  $\mathcal{U}_1(\mathbf{x}_t) = \mathcal{T}(\mathbf{x}_t)$ , which is the static pose feature (joint-position in the matrix formulation). The second one is  $\mathcal{U}_2(\mathbf{x}_t, \mathbf{x}_{t+1}) = \mathcal{T}(\mathbf{x}_t) - \mathcal{T}(\mathbf{x}_{t+1})$ , which is the motion pose feature between two consecutive frames. Thus, the similarity score  $S_1(\mathbf{X}_{1:L_x}, \mathbf{Y}_{1:L_y})$  given by the static features is,

$$\sum_{t=1}^{L_x} D_{DMSW}(\mathcal{T}(\mathbf{x}_t), \mathcal{T}(\tilde{\mathbf{y}}_t)) \quad (11)$$

where  $\tilde{\mathbf{y}}_t$  is the temporally corresponding frame estimated by DMTW. The similarity score  $S_2(\mathbf{X}_{1:L_x}, \mathbf{Y}_{1:L_y})$  given

by the motion features is,

$$\sum_{t=1}^{L_x} D_{DMSW}(\mathcal{T}(\mathbf{x}_t) - \mathcal{T}(\mathbf{x}_{t+1}), \mathcal{T}(\tilde{\mathbf{y}}_t) - \mathcal{T}(\tilde{\mathbf{y}}_{t+1})) \quad (12)$$

The final score  $S_{DMW}(\mathbf{X}_{1:L_x}, \mathbf{Y}_{1:L_y})$  is,

$$\lambda S_1(\mathbf{X}_{1:L_x}, \mathbf{Y}_{1:L_y}) + (1 - \lambda) S_2(\mathbf{X}_{1:L_x}, \mathbf{Y}_{1:L_y}) \quad (13)$$

where  $\lambda \in [0, 1]$  can be either optimized by cross-validation in the supervised setting (recognition), or chosen manually in the unsupervised setting (clustering). This score is not symmetric, so we set the testing sequence as the reference.

**Analysis.** Both DMSW and CTW algorithms use CCA, but key differences exist. Spatial alignment in DMSW is restricted to *temporally local* manifolds, since *global linear* matching on entire sequences (CTW) is often not accurate due to non-linear variations. But this global matching is not necessarily a disadvantage: CTW can provide dimension reduction results, which is useful in some applications.

Combining all features of DMW (DMTW and DMSW) together, we can align 2KD video tracks with 3MD Mocap sequences, which is not addressed by previous works.

## 5. Action Recognition from Videos

**Tracking.** To apply our approach to videos, we have a pre-processing step to extract joint 2D trajectories from image observations. The problem itself is challenging and tightly connected to human pose estimation [17], an important subarea in computer vision. Currently, we use the IVT Tracker [15], an online updated appearance model is used to model the objects dynamic variation.

**Alignment of  $\mathbf{X}_{mocap}$  and  $\mathbf{Y}_{video}$ .** Given  $K$  tracks in 2D, we have a 2KD spatio-temporal manifold (STM)  $\mathbf{Y}_{1:L_y} \in \mathbb{R}^{2K \times L_y}$ , where the  $t$ th column  $\mathbf{y}_t = [\mathbf{u}_1, \dots, \mathbf{u}_K]^T \in \mathbb{R}^{2K}$  is the joint-coordinates vector of  $K$  tracked points ( $\mathbf{u}_{j(1 \leq j \leq K)}$ ) at frame  $t$ . Assume the underlying 3D joint-trajectories for the people in the video is  $\mathbf{Z}_{1:L_y} \in \mathbb{R}^{3M \times L_y}$  ( $\mathbf{z}_{t(1 \leq t \leq L_y)} = [\mathbf{p}_1, \dots, \mathbf{p}_M]^T, \mathbf{p}_{j(1 \leq j \leq M)} \in \mathbb{R}^3$ ), it can be shown that the STM in 2KD space is the projection of the STM in 3MD space. In particular, under the linear projection model  $\mathbf{P} \in \mathbb{R}^{2 \times 3}$  (from 3D position  $\mathbf{p}_j$  to 2D image coordinate  $\mathbf{u}_j$ ), we can have  $\mathbf{y}_t$  equal to,

$$\begin{pmatrix} \mathbf{P} & \mathbf{0} & \dots \\ \dots & \mathbf{P} & \dots \\ \dots & \mathbf{0} & \mathbf{P} \end{pmatrix} \begin{pmatrix} \mathbf{W}_{11} & \mathbf{W}_{12} & \dots \\ \dots & \mathbf{W}_{ij} & \dots \\ \dots & \dots & \mathbf{W}_{KM} \end{pmatrix} \begin{pmatrix} \mathbf{p}_1^T \\ \dots \\ \mathbf{p}_M^T \end{pmatrix}$$

$\mathbf{W}_{ij}$  is a binary selection matrix, which equals to  $\mathbf{I}_{3 \times 3}$  if the  $j$ th key point is available in the tracking results, otherwise equals to  $\mathbf{0}_{3 \times 3}$ . This projection relationship can be compactly represented in matrix notations as,

$$\mathbf{y}_{(2K \times 1)} = \tilde{\mathbf{P}}_{(2K \times 3K)} \tilde{\mathbf{W}}_{(3K \times 3M)} \mathbf{z}_{(3M \times 1)} \quad (14)$$

i.e., the  $2KD$  manifold is just the linear projection of the  $3MD$  manifold.  $\tilde{\mathbf{P}} \in \mathbb{R}^{2K \times 3K}$  is the compact projection matrix and  $\tilde{\mathbf{W}} \in \mathbb{R}^{3K \times 3M}$  is the compact selection matrix. For perspective projection, the derivation is similar, and the manifold should be represented in homogeneous space.

The alignment between  $\mathbf{Y}_{1:L_y} \in \mathbb{R}^{2K \times L_y}$  (video) and  $\mathbf{X}_{1:L_x} \in \mathbb{R}^{3M \times L_x}$  (Mocap) is performed as:

- (1) Structure Learning. Use the algorithms in Sec. 3.1.
- (2) Temporal Alignment. Use the DMTW (eq. 4) algorithm to get temporal aligned  $\tilde{\mathbf{X}}_{1:L_y} \in \mathbb{R}^{3M \times L_y}$ .

(3) Spatial Alignment. Select  $K$  markers from  $\tilde{\mathbf{X}}_{1:L_y}$ , resulting in  $\tilde{\mathbf{X}}_{1:L_y}^K \in \mathbb{R}^{3K \times L_y}$ .  $\tilde{\mathbf{X}}_{1:L_y}^K$  is spatial aligned to  $\mathbf{Y}_{1:L_y}$  by using DMSW (eq. 9), since  $M - K$  markers' information is missed from video tracking results. Then,  $S(\mathbf{Y}_{1:L_y}, \tilde{\mathbf{X}}_{1:L_y}^K)$  is calculated by eq. 13. It is notable that  $\mathcal{T}^{3D}$  is applied to  $\tilde{\mathbf{X}}_{1:L_y}^K$  and  $\mathcal{T}^{2D}$  is applied to  $\mathbf{Y}_{1:L_y}$ .

Assume there are  $N$  labeled Mocap sequences  $\{\mathbf{X}_{mocap}^i\}_{i=1}^N$  associated with action label  $I^i \in \mathcal{I}$ , where  $\mathcal{I} = 1, 2, \dots, C$  indicates  $C$  action classes. Given a joint trajectories  $\mathbf{Y}_{video}$  from a video clip by the tracker, the estimated action label  $I^y$  is given by

$$I^y = \arg \min_{i \in \{1, 2, \dots, N\}} S_{DMW}(\mathbf{Y}_{video}, \{\mathbf{X}_{mocap}^i, I^i\}) \quad (15)$$

Furthermore, the matching process can infer complete  $3MD$  skeleton  $\mathbf{Z}_{1:L_y}$  from  $\mathbf{Y}_{1:L_y}$ . The inference is done by selecting the most similar aligned mocap manifold.

## 6. Experiments

We evaluate the performance of our system from three aspects; (1) temporal alignment, (2) action recognition on Mocap, (3) action recognition for realistic videos.  $M = 15$  key points are used to represent the human skeleton, resulting in joint 3D trajectories in 45D space. Assume motion sequences are temporally segmented, which can be done manually (training) or using algorithms like [27].

### 6.1. Temporal Alignment Evaluation

In this section, qualitative comparison of temporal alignment methods is provided. Quantitative results for how different alignment methods affect action recognition rates are provided in Sec. 6.2 and 6.3. DMTW (the temporal alignment part of DMW) is compared with other state-of-the-art methods, i.e., DTW [14] and CTW [26]. Profile methods like [6] are not considered since they need a training stage.

To make the comparison clear, sequences may include more than one *action units*. Fig. 3 shows the visual comparison of two motion sequences, one is boxing (twice) and the other is side jumping (twice). DTW does not consider the spatial transformation, making it difficult to align two motion sequences by two people. CTW significantly outperforms DTW. Our DMTW gets the most visually appealing

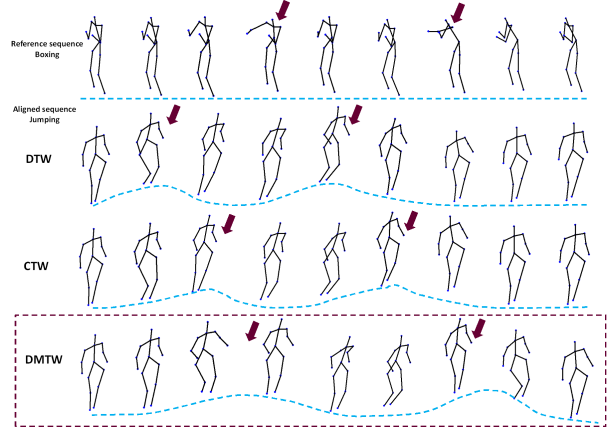


Figure 3. **Temporal Alignment Results.** The reference sequence is shown in the first row, followed by the aligned results. 2 red arrows indicate 2 key states in the reference sequence, i.e., the peaks of the first and the second boxing. All aligned sequences also have 2 red arrows, indicating the peaks of the first and the second jump. DMTW is able to align the two peak states in the jumping sequence to the peak states in the boxing sequence very well.

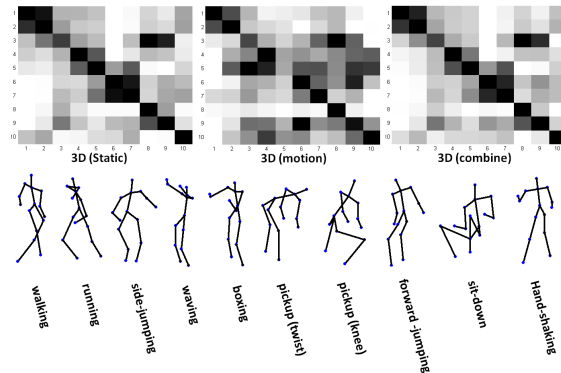


Figure 4. **Action recognition results on Mocap.** Top: confusion matrix in 3D; bottom, 1 Mocap example for each action.

results among three methods. It is notable that our temporal alignment step *does not involve* spatial matching (unlike CTW). More visual comparison results on Mocap and HumanEva videos are not provided due to lack of space, and DMTW gets similar performance in all experiments.

### 6.2. Action Recognitions on Mocap

We collected 3978 frames from CMU Mocap capturing 15 people, performing 10 natural actions (details in Fig. 4). The motion distance scores between any two sequences is calculated, resulting in a  $10 \times 10$  average motion distance matrix  $\mathbf{S}$  for these 10 actions (Fig. 4). It is clear that the diagonal area has the smallest variations, which shows the effectiveness of our similarity function (eq. 13). For action recognition, we use the leave-one-out procedure. Since

| Methods  | DTW+DMSW (3MD) | CTW+DMSW (3MD) | DMW (3MD)  | DMW (2KD)  |
|----------|----------------|----------------|------------|------------|
| Rate (S) | 60%            | 85%            | <b>95%</b> | <b>90%</b> |
| Rate (F) | 62%            | 91%            | <b>99%</b> | <b>87%</b> |

Table 1. **Action Recognition Rates on Mocap.** Rate is measured by # of sequences (S) or # of frames (F).

each person only performs a specific action once, the recognition can not benefit from the fact that the same person repeating the same action results in quite large similarity.  $\lambda$  in eq. 13 is set to be 0.5 and results (Table 1) show that our approach only misclassifies 5% sequences, or 1.2% by weighing with the number of frames. Furthermore, to demonstrate the ability to recognize actions from arbitrary 2D view, Mocap sequences are projected to joint 2D trajectories using a synthetic camera ( $K = 15$ ). We achieve 90% accuracy in this 2D view recognition. [7] also reports recognition rate for Mocap data, but [7] requires a large number of training sequences and 2D view recognition is not considered.

To investigate how temporal alignment affects recognition, results of using DTW and CTW are also provided (3D). To make a fair comparison, only the temporal alignment step is changed. Results show that this change reduces accuracy significantly, which supports the effectiveness of DMW not only in temporal alignment, but also in action recognition (quantitatively).

### 6.3. Action Recognitions on HumanEva

To validate our approach on video based action recognition, we chose a number of video sequences from Brown HumanEva dataset (1 and 2), which is a benchmark proposed for human motion analysis [17]. The reason that standard action benchmark datasets, such as KTH [16] or Weizmann [2] are not used, is that the low resolution of videos makes key points tracking results unreliable. For IVT [15], we manually label key points in the first frame and do not tune any parameter in the tracking process. Only  $K = 7$  or 8 key points are estimated from the side view, resulting in joint 2D trajectories in 14D or 16D space. Tracked trajectories are associated with the labeled Mocap sequences from 10 action categories (Fig. 6). Although tracking results are often noisy (Fig. 5) (up to 30 pixels on selected frames with manually labeled ground-truth), we can correctly estimate actions from these noisy and occluded 2D input. Furthermore, the complete 3MD skeleton is inferred. [11] also reports recognition performance on HumanEva-1, but direct comparison is difficult. In particular, [11] focuses on supervised learning on HumanEva while we focus on transfer learning on HumanEva.

## 7. Conclusion

In this paper, we proposed a spatio-temporal model STM to analyze sequential data with latent spatial structure. Fur-

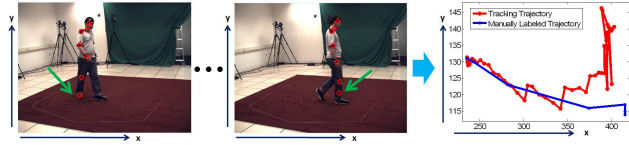


Figure 5. **Noisy Tracking Results on HumanEva videos.** Left, image sequences; right, trajectories of the right feet provide by the tracker and manual labeling. Our approach can recognize actions from this noisy and occluded input.

thermore, a robust and efficient alignment algorithm DMW is designed to calculate the similarity between two multivariate time series. Based on STM and DMW, we achieved view-invariant action recognition on videos by associating a few Mocap examples. In the future, we will evaluate our approach on more data sets, and apply it to 3D motion recovery and temporal motion segmentation.

## Acknowledgements

This work was supported in part by NIH Grant EY016093. The authors would like to thank Yan Liu, Vivek Kumar Singh and Sikai Zhu for their helpful discussions.

## References

- [1] F. R. Bach and M. I. Jordan. Kernel independent component analysis. *JMLR*, 3:1–48, 2003. **2, 5**
- [2] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *Proc. ICCV*, volume 2, pages 1395–1402, 2005. **1, 7**
- [3] T. K. Kim and R. Cipolla. Canonical correlation analysis of video volume tensors for action categorization and detection. *IEEE PAMI*, 31:1415–1428, 2009. **2**
- [4] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *Proc. CVPR*, 2008. **1**
- [5] N. Lawrence. Probabilistic non-linear principal component analysis with gaussian process latent variable models. *JMLR*, 6:1783–1816, November 2005. **2, 3**
- [6] J. Listgarten, R. M. Neal, S. T. Roweis, and A. Emili. Multiple alignment of continuous time series. In *NIPS*, volume 17, 2005. **4, 6**
- [7] F. Lv and R. Nevatia. Recognition and segmentation of 3-d human action using hmm and multi-class adaboost. In *Proc. ECCV*, volume 3954, pages 359–372, 2006. **2, 7**
- [8] P. Mordohai and G. Medioni. Dimensionality estimation, manifold learning and function approximation using tensor voting. *JMLR*, 11:411–450, 2010. **3**

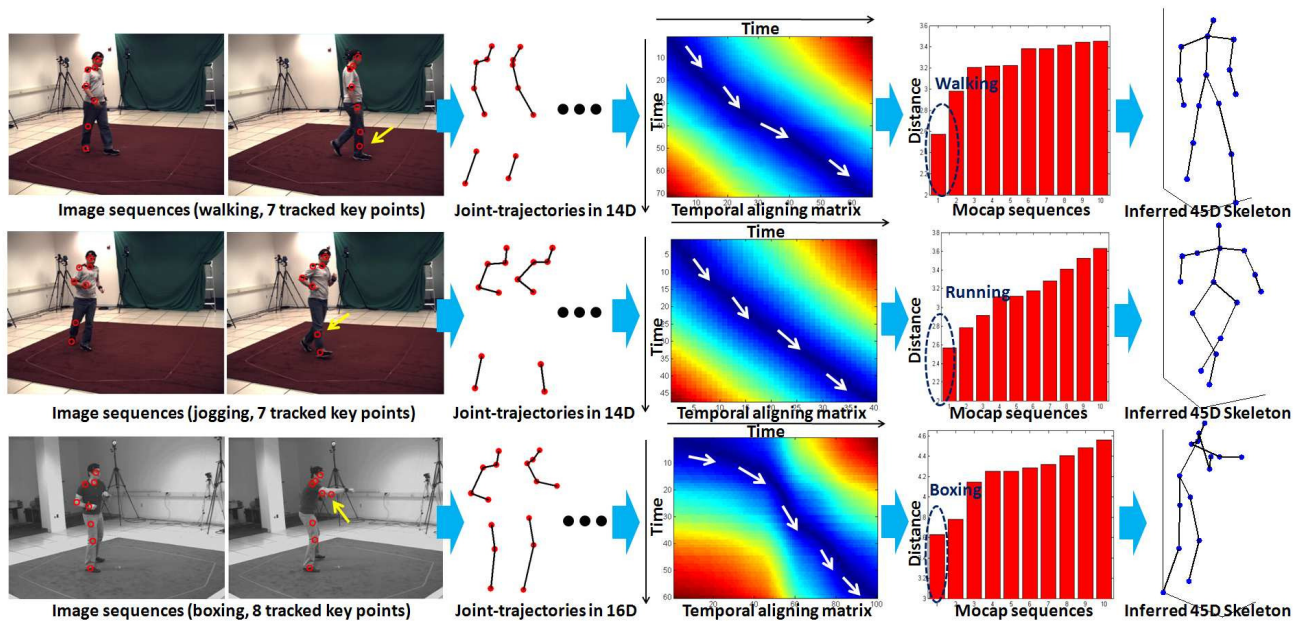


Figure 6. **Examples of action recognition results on HumanEva videos.** Top to bottom, walking, jogging and boxing. Left to right, image sequences, 2KD joint-trajectories, temporal aligning matrix with the most similar Mocap sequence, distance to 10 most similar Mocap sequences, and the inferred complete 3MD motion data. Yellow arrows in images indicate examples of noisy tracking results.

- [9] P. Natarajan and R. Nevatia. View and scale invariant action recognition using multiview shape-flow models. In *Proc. CVPR*, 2008. 2
- [10] J. Nielbes, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. *IJCV*, 79:299–318, 2008. 1
- [11] H. Ning, W. Xu, Y. Gong, and T. Huang. Latent pose estimator for continuous action recognition. In *Proc. ECCV*, volume 5303, pages 419–433. 2008. 1, 7
- [12] F. Padua, F. Carceroni, R. Santos, and G. Kutulakos. Linear sequence-to-sequence alignment. *IEEE PAMI*, 32:304–320, 2010. 2
- [13] R. Poppe. A survey on vision-based human action recognition. *Image and Vision Computing*, 28:976–990, 2010. 2
- [14] C. Rao, A. Gritaiand, M. Shah, and T. Syeda-Mahmood. View-invariant alignment and matching of video sequences. In *Proc. ICCV*, pages 939–945, 2003. 2, 4, 6
- [15] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang. Incremental learning for robust visual tracking. *IJCV*, 77:125–141, 2008. 5, 7
- [16] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. In *Proc. ICPR*, volume 3, pages 32–36, 2004. 7
- [17] L. Sigal, A. O. Balan, and M. J. Black. Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *IJCV*, 87:4–27, 2010. 5, 7
- [18] M. Singh, I. Cheng, M. Mandal, and A. Basu. Optimization of symmetric transfer error for sub-frame video synchronization. In *Proc. ECCV*, volume 5303, pages 554–567. 2008. 2, 4
- [19] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323, December 2000. 2, 3
- [20] R. Urtasun, D. Fleet, and P. Fua. 3d people tracking with gaussian process dynamical models. In *Proc. CVPR*, volume 1, pages 238–245, 2006. 2
- [21] R. Urtasun, D. J. Fleet, A. Geiger, J. Popovic, T. Darrell, and N. D. Lawrence. Topologically-constrained latent variable models. In *Proc. ICML*, pages 1080–1087, 2008. 2, 3
- [22] L. van der Maaten. Learning a parametric embedding by preserving local structure. In *Proc. AISTATS, JMLR WCP*, volume 5, pages 384–3912, 2009. 2, 3, 4
- [23] D. Weinland, E. Boyer, and R. Ronfard. Action recognition from arbitrary views using 3d exemplars. In *Proc. ICCV*, 2007. 2
- [24] D. Weinland, M. Ozuysal, and P. Fua. Making action recognition robust to occlusions and viewpoint changes. In *ECCV*, volume 6313, pages 635–648. 2010. 1, 2
- [25] P. Yan, S. M. Khan, and M. Shah. Learning 4d action feature models for arbitrary view action recognition. In *Proc. CVPR*, 2008. 1, 2
- [26] F. Zhou and F. D. la Torre. Canonical time warping for alignment of human behavior. In *NIPS*, volume 22, pages 2286–2294. 2009. 2, 3, 4, 6
- [27] F. Zhou, F. D. la Torre, and J. F. Cohn. Unsupervised discovery of facial events. In *Proc. CVPR*, pages 2574–2581, 2010. 6