Hindawi Publishing Corporation
EURASIP Journal on Image and Video Processing
Volume 2008, Article ID 326896, 18 pages
doi:10.1155/2008/326896

Research Article

Monocular 3D Tracking of Articulated Human Motion in Silhouette and Pose Manifolds

Feng Guo[1] **and Gang Qian**[1, 2]

[1] *Department of Electrical Engineering, Arizona State University, Tempe, AZ 85287-9309, USA*
[2] *Arts, Media and Engineering Program, Department of Electrical Engineering, Arizona State University, Tempe, AZ 85287-8709, USA*

Correspondence should be addressed to Gang Qian, gang.qian@asu.edu

Received 1 February 2007; Revised 24 July 2007; Accepted 29 January 2008

Recommended by Nikos Nikolaidis

This paper presents a robust computational framework for monocular 3D tracking of human movement. The main innovation of the proposed framework is to explore the underlying data structures of the body silhouette and pose spaces by constructing low-dimensional silhouettes and poses manifolds, establishing intermanifold mappings, and performing tracking in such manifolds using a particle filter. In addition, a novel vectorized silhouette descriptor is introduced to achieve low-dimensional, noise-resilient silhouette representation. The proposed articulated motion tracker is view-independent, self-initializing, and capable of maintaining multiple kinematic trajectories. By using the learned mapping from the silhouette manifold to the pose manifold, particle sampling is informed by the current image observation, resulting in improved sample efficiency. Decent tracking results have been obtained using synthetic and real videos.

Copyright © 2008 F. Guo and G. Qian. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. INTRODUCTION

Reliable recovery and tracking of articulated human motion from video are considered a very challenging problem in computer vision, due to the versatility of human movement, the variability of body types, various movement styles and signatures, and the 3D nature of human body. Vision-based tracking of articulated motion is a temporal inference problem. There exist numerous computational frameworks addressing this problem. Some of the frameworks make use of training data (e.g., [1]) to inform the tracking, while some attempt to directly infer the articulated motion without using any training data (e.g., [2]). When training data is available, the articulated motion tracking can be cast into a statistical learning and inference problem. Using a set of training examples, a learning and inference framework needs to be developed to track both seen and unseen movements performed by known or unknown subjects. In terms of the learning and inference structure, existing 3D tracking algorithms can be roughly clustered into two categories, namely, generative-based and discriminative-based approaches. Generative-based approaches, for example [2–4], usually assume the knowledge of a 3D body model of the subject and dynamical models of the related movement, from which kinematic predictions and corresponding image observations can be *generated*. The movement dynamics are learned from training examples using various dynamic system models, for example, autoregressive models [5], hidden Markov models [6], Gaussian process dynamical models [1], and piecewise linear models in the form of a mixture of factor analyzers [7]. A recursive filter is often deployed to temporally propagate the posterior distribution of the state. Especially, particle filters have been extensively used in movement tracking to handle nonlinearity in both the system observation and the dynamic equations. Discriminative-based approaches, for example [8–13], treat kinematics recovery from images as a regression problem from the image space to the body kinematics space. Using training data, the relationship between image observation and body poses is obtained using machine-learning techniques. When compared against each other, both approaches have their own pros and cons. In general, generative-based methods utilize movement dynamics and produce more accurate tracking results, although they are more time consuming, and usually the conditional distribution of the kinematics given the current image observation is not utilized directly. On the other hand, discriminative-based methods learn such conditional distributions of kinematics given image observations from training data and often result in fast image-based kinematic inference. However, movement kinematics are usually not fully explored by discriminative-based methods. Thus, the rich temporal correlation of body kinematics between adjacent frames is unused in tracking.

In this paper, we present a 3D tracking framework that integrates the strengths of both generative and discriminative approaches. The proposed framework explores the underlying low-dimensional manifolds of silhouettes and poses using nonlinear dimension reduction techniques such as Gaussian process latent variable models (GPLVM) [14] and Gaussian process dynamic models (GPDM) [15]. Both Gaussian process models have been used for people tracking [1, 16–18]. The Bayesian mixture of experts (BME) and relevance vector machine (RVM) are then used to construct bidirectional mappings between these two manifolds, in a manner similar to [10]. A particle filter defined over the pose manifold is used for tracking. Our proposed tracker is self-initializing and capable of tracking multiple kinematic trajectories due to the BME-based multimodal silhouette-to-kinematics mapping. In addition, because of the bidirectional inter-manifold mappings, the particle filter can draw kinematic samples using the current image observation, and evaluate sample weights without projecting a 3D body model. To overcome noise present in silhouette images, a low-dimensional vectorized silhouette descriptor is introduced based on Gaussian mixture models. Our proposed framework has been tested using both synthetic and real videos with different subjects and movement styles from the training. Experimental results show the efficacy of the proposed method.

1.1. Related work

Among existing methods on integrating generative-based and discriminative-based approaches for articulated motion tracking, the 2D articulated human motion tracking system proposed by Curio and Giese [19] is the most revelent to our framework. The system in [19] conducts dimension reduction in both image and pose spaces. Using training data, one-to-many support vector regression (SVR) is learned to conduct view-based pose estimation. A first-order autoregressive (AR) linear model is used to represent state dynamics. A competitive particle filter defined over the hidden state space is deployed to select plausible branches and propagate state posteriors over time. Due to SVR, this system is capable of autonomous initialization. It draws samples using both current observation and state dynamics. However, there are four major differences between the approach in [19] and our proposed framework. Essentially, [19] presents a tracking system for 2D articulated motion, while our framework is for 3D tracking. In addition, In [19] a 2D patch-model is used to obtain the predicted image observation, while in our proposed framework this is done through nonlinear regression without using any body models. Furthermore, during the initialization stage of the system in [19], only the best body configuration obtained from the view-based pose estimation and the model-based matching is used to initialize the tracking. It is obvious that using a single initial state has the risk of missing other admissible solutions due to the inherent ambiguity. Therefore, in our proposed system multiple solutions are maintained in tracking. Finally, BME is used in our proposed framework for view-based pose estimation instead of SVR as in [19]. BME has been used for kinematic recovery [10]. In summary, our proposed framework can be considered as an extension of the system in [19] to better address the integration of generative-based and discriminative-based approaches in the case of 3D tracking of human movement, with the advantages of tracking multiple possible pose trajectories over time and removing the requirement of a body model to obtain predicted image observations.

Dimension reduction of the image silhouette and pose spaces has also been investigated using kernel principle component analysis (KPCA) [12, 20] and probabilistic PCA [13, 21]. In [7, 22], a mixture of factor analyzers is used to locally approximate the pose manifold. Factor analyzers perform nonlinear dimension reduction and data clustering concurrently within a global coordinate system, which makes it possible to derive an efficient multiple hypothesis tracking algorithm based on distribution modes. Recently, nonlinear probabilistic generative models such as GPLVM [14] have been used to represent the low-dimensional full body joint data [16, 23] and upper body joints [24] in a probabilistic framework. Reference [16] introduces the scaled GPLVM to learn dynamical models of human movements. As variants of GPLVM, GPDM [15, 25], and balanced GPDM [1] have shown to be able to capture the underlying dynamics of movement, and at the same time to reduce the dimensionality of the pose space. Such GPLVM-based movement dynamical models have been successfully used as priors for tracking of various types of movement, including walking [1] and golf swing [16]. Recently, [26] presents a hierarchical GPLVM to explore the conditional independencies, while [27] extends GPDM into a multifactor analysis framework for style-content separation. In our proposed framework, we follow the balanced GPDM presented in [1] to learn movement dynamics due to its simplicity and demonstrated ability to model human movement. Furthermore, we adopt GPLVM to construct the silhouette manifold using silhouette images from different views, which has been shown to be promising in our experiments. Additional results using GPLVM for 3D tracking have been reported recently. In [18], a real-time body tracking framework is presented using GPLVM.

Since image observations and body poses of the same movement essentially describe the same physical phenomenon, it is reasonable to learn a joint image-pose manifold. In [17] GPLVM has been used to obtain a joint silhouette and pose manifold for pose estimation. Reference [28] presents a joint learning algorithm for a bidirectional generative-discriminative model for 2D people detection and 3D human motion reconstruction from static images with cluttered background by combining the top-down (generative-based) and bottom-up (discriminative-based) processings. The combination of top-down and bottom-up approaches in [28] is promising for solving simultaneous people detection and pose recovery in cluttered images. However, the

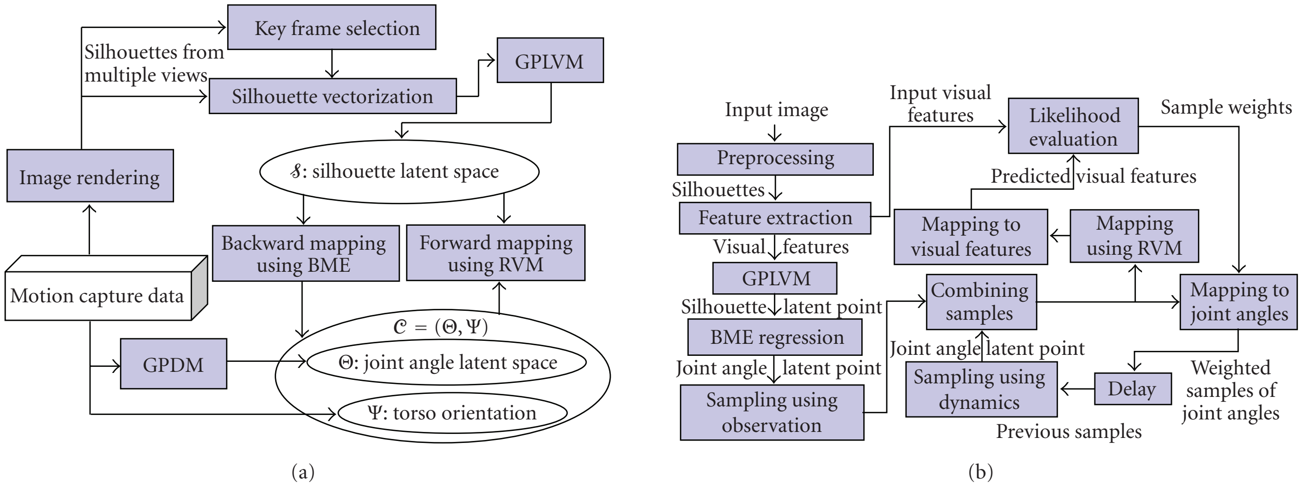


Figure 1: An overview of the proposed framework, (a): training phase; (b): tracking phase.

emphasis of [28] is on parameter learning of the bidirectional model and movement dynamics are not considered. Comparing with [17, 28], the separate kinematics and silhouette manifold learning is a limitation of our proposed framework.

View-independent tracking and handling of ambiguous solutions are critical for monocular-based tracking. To tackle this challenge, [29] represents shape deformations according to view and body configuration changes on a 2D torus manifold. A nonlinear mapping is then learned between torus manifold embedding and visual input using empirical kernel mapping. Reference [30] learned a clustered exemplar-based dynamic model for viewpoint invariant tracking of the 3D human motion from a single camera. This system can accurately track large movements of the human limbs. However, neither of the above approaches explicitly considers multiple solutions and only one kinematic trajectory is tracked, which results in an incomplete description of the posterior distribution of poses. To handle the multimodal mapping from the visual input space to the pose space, several approaches [10, 31, 32] have been proposed. The basic idea is to split the input space into a set of regions and approximate a separate mapping for each individual region. These regions have *soft* boundaries, meaning that data points may lie simultaneously in multiple regions with certain probabilities. The mapping in [31] is based on the joint probability distribution of both the input and the output data. An inverse mapping function is used to formulate an efficient inference. In [10, 32], the conditional distribution of the output given the input is learned in the framework of mixture of experts. Reference [32] also uses the joint input-output distribution and obtains the conditional distribution using the Bayes rule while [10] learns the conditional distribution directly. In our proposed framework, we adopt the extended BME model [33] and use RVM as experts [10] for multimodal regression. A related work that should be mentioned here is the extended multivariate RVM for multimodal multidimensional 3D body tracking [8]. Impressive full body tracking results of human movement have been reported in [8].

Another highlight of our proposed system is that predicted visual observations can be obtained directly from a pose hypothesis without projecting a 3D body model. This feature allows efficient likelihood and weight evaluation in a particle filtering framework. The 3D-model-free approaches for image silhouette synthesis from movement data reported in [34, 35] are most related to our proposed approach. The main difference is that our approach achieves visual prediction using RVM-based regression, while in [34, 35] multilinear analyis [36] is used for visual synthesis.

2. SYSTEM ARCHITECTURE

An overview of the architecture of our proposed system is presented in Figure 1, consisting of a training phase and a tracking phase.

The training phase contains training data preparation and model learning. In data preparation, synthetic images are rendered using animation software from motion capture data, for example, Maya. The model-learning process has five major steps as shown in Figure 1(a). In the first step, key frames are selected from synthetic images using multidimensional scaling (MDS) [37, 38] and k-means. In the second step, silhouettes in the training data are then be vectorized according to its distances to these key frames. Then in the following step, GPLVM is used to construct the low-dimensional manifold $\mathcal{S}$ of the image silhouettes from multiple views using their vectorized descriptors. The fourth step is to reduce dimensionality of the pose data and obtain a related motion dynamical model. GPDM is used to obtain the manifold Θ of full-body pose angles. This latent space is then augmented by the torso orientation space Ψ to form the *complete* pose latent space $\mathcal{C} \equiv (\Theta, \Psi)$. Finally in the last step, the forward and backward nonlinear mappings between $\mathcal{C}$ to $\mathcal{S}$ are constructed in the learning phase. The forward mapping from $\mathcal{C}$ to $\mathcal{S}$ is established using RVM, which will be used to efficiently evaluate sample weights in the tracking phase. The multimodal (one-to-many) backward mapping from $\mathcal{S}$ to $\mathcal{C}$ is obtained using BME.

The essence of tracking in our proposed framework is the propagation of weighted movement particles in $\mathcal{C}$ based on the image observation up to the current time instant and learned movement dynamic models. In tracking, the body silhouette is first extracted from an input image and then vectorized. Using the learned GPLVM, its corresponding latent position is found in $\mathcal{S}$. Then BME is invoked to find a few plausible pose estimates in $\mathcal{C}$. Movement samples are drawn according to both the BME outputs and learned GPDM. The sample weights are evaluated according to the distance between the observed and predicted silhouettes. The empirical posterior distributions of poses are then obtained as the weighted samples. The details of the learning and tracking steps are described in the following sections.

3. PREPARATION OF TRAINING DATA

To learn various models in the proposed framework, we need to construct training data sets including complete pose data (body joint angles, torso orientation), and the corresponding images. In our experiments, we focus on the tracking of gait. Three walking sequences (07_01, 16_16, 35_03) from different subjects were taken from CMU motion capture database [39], with each sequence containing two gait cycles. These sequences were then downsampled by a factor of 4, constituting 226 motion capture frames in total. There are 56 original local joint angles in the original motion capture data. Only 42 major joint angles are used in our experiments. This set of local joint angles is denoted as Θ_T.

To synthesize multiple views of one body pose defined by a frame of motion capture data, sixteen frames complete pose data were generated by augmenting the local joint angles with 16 different torso orientation angles. To obtain silhouettes from diverse view points, these orientation angles are randomly altered from frame to frame. Given one frame of motion capture data, these 16 torso orientation angles were selected as follows. A circle centered at the body centroid in the horizontal plane of the human body can be found. To determine the 16 body orientation angles, this circle is equally divided into 16 parts, corresponding to 16 cameras views. In each camera view, an angle is uniformly drawn in an angle interval of 22.5°. Hence for each given motion capture frame, there are 16 complete pose frames with different torso orientation angles, resulting 3616 (226×16) complete pose frames in total. This training set of complete poses is denoted as $\mathbf{C}_T$.

Using $\mathbf{C}_T$, corresponding silhouettes were generated using animation software. We denote this silhouette training set $\mathbf{S}_T$. Three different 3D models (one female and two males) were used for each subject to obtain a diverse silhouette set with varying appearances.

4. IMAGE FEATURE REPRESENTATION

4.1. GMM-based silhouette descriptor

Assume that silhouettes can be extracted from images using background subtraction and refined by morphological operation. The remaining question is how to represent the silhouette robustly and efficiently. Different shape descriptors have been used to represent silhouettes. In [40], Fourier descriptor, shape context, and Hu moments were computed from silhouettes and their resistance to variations in body built, silhouette extraction errors, and viewpoints were compared. It is shown that both Fourier descriptor and shape context perform better than the Hu moment. In our approach, Gaussian mixture models (GMM) are used to represent silhouettes and it performs better than shape context descriptor. We have used GMM-based shape descriptor in our previous work on single-image-based pose inference [41].

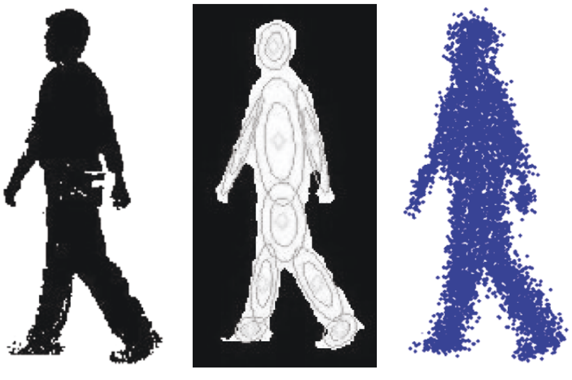


FIGURE 2: (a): the original silhouette, (b): learned Gaussian mixture components using EM, (c): point samples drawn such a GMM.

GMM assumes that the observed unlabeled data is produced by a number of Gaussian distributions. The basic idea of GMM-based silhouette descriptor is to consider a silhouette as a set of coherent regions in the 2D space such that the foreground pixel locations are generated by a GMM. Strictly speaking, foreground pixel locations of a silhouette do not exactly follow the Gaussian distribution assumption. Actually a uniform distribution confined to a closed area given by the silhouette contour would be a much better choice. However, due to its simplicity, GMM is selected in the proposed framework to represent silhouettes. From Figure 2, we can see that the GMM can model the distribution of the silhouette pixels well. It has good locality to improve the robustness compared the global descriptor such as shape moment. The reconstructed silhouette points look very similar to the original silhouette image.

Given a silhouette, the GMM parameters can be obtained using an EM algorithm. Initial data clustering can be done using the k-means algorithm. The full covariance matrices of the Gaussian are estimated. In our implementation, a GMM with 20 components is used to represent one silhouette. It takes about 600 milliseconds to extract the GMM parameters from an input silhouette (~120 pixel-high) using Matlab.

4.2. KLD-based similarity measure

It is critical to measure the similarities between silhouettes. Based on the GMM descriptor, the Kullback-Leibler divergence (KLD) is used to compute the distance between two silhouettes. Similar approaches have been taken for

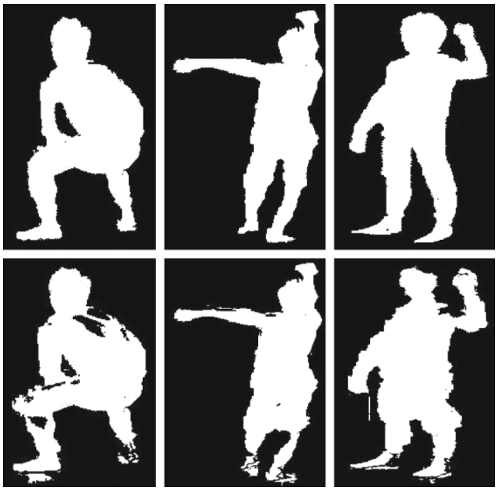


Figure 3: Clean (top row) and noisy silhouettes of some dance poses.

GMM-based image matching for content-based image retrieval [42]. Given two distributions p_1 and p_2, the KLD from p_1 to p_2 is

$$D(p_1 \| p_2) = \int p_1(x) \log \frac{p_1(x)}{p_2(x)} dx. \tag{1}$$

The symmetric version of the KLD is given by

$$d(p_1, p_2) = \frac{1}{2}[D(p_1 \| p_2) + D(p_2 \| p_1)]. \tag{2}$$

In our implementation, such symmetric KLD is used to compute the distance between two silhouettes and the KLDs are computed using a sampling-based method.

GMM representation can handle noise and small shape model differences. For example, Figure 3 has three columns of images. In each column, the bottom image is a noisy version of the top image. The KLD between the noisy and clean silhouettes in the left, middle, and right columns are 0.04, 0.03, and 0.1, respectively. They are all below 0.3, which is an empirical KLD threshold indicating similar silhouettes. This threshold was obtained according to our experiments running over a large number of image silhouettes of various movements and dance poses.

4.3. Vectorized silhouette descriptor

Although GMM and KLD can represent silhouettes and compute their similarities, sampling-based KLD computation between two silhouettes is slow, which harms the scalability of the proposed method when a large number of training data is used. To overcome this problem, in the proposed framework a vectorization of the GMM-based silhouette descriptor is introduced. The nonvectorized GMM-based shape descriptor has been used in our previous work on single-image-based pose inference [41]. Vector representation of silhouette is critical since it will simplify and expedite the GPLVM-based manifold learning and mapping from silhouette space to its latent space.

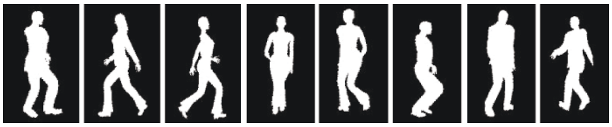


Figure 4: Some of the 46 key frames selected from the training samples.

To obtain a vector representation for our GMM descriptor, we use the relative distances of one silhouette to several key silhouettes to locate this point in the silhouette space. The distance between this silhouette and each key silhouette is one element in the vector. The challenge here is to determine how many of them will be sufficient and how to select these key frames.

In our propose framework, we first use MDS [37, 38] to estimate the underlying dimensionality of the silhouette space. Then the k-means algorithm is used to cluster training data and locate the cluster centers. Silhouettes that are the closest to these cluster centers are then selected as our key frames. Given training data, the distance matrix D of all silhouettes is readily computed using KLD. MDS is a nonlinear dimension reduction method if one can obtain a good distance measure. An excellent review of MDS can be found in [37, 38]. Following MDS, $\overline{D} = -P^e D P^e$ can be computed. When D is a distance matrix of a metric space (e.g., symmetric, nonnegative, satisfying triangle inequality), $\overline{D}$ is positive semidefinite (PSD), and the minimal embedding dimension is given by the rank of $\overline{D}$. Here $P^e = 1 - ee^T/N$ is the centering matrix, where N is the number of training data and ee^T is an $N \times N$ matrix of all ones. Due to observation noise and errors introduced in the sampling-based KLD calculation, the KLD matrix D we obtained is only an approximate distance matrix and $\overline{D}$ might not be purely PSD in practice. In our case, we just ignored the negative eigenvalues of $\overline{D}$ and only considered the positive ones. Using the 3616 training samples in $\mathbf{S}_T$ described in Section 3, 45 dimensions are kept to count over 99% of the energy in the positive eigenvalues. To remove a representation ambiguity, distances from 46 key frames are needed to locate a point in a 45-dimensional space. To select these key frames, all the training silhouettes are clustered into 46 groups using the k-means algorithm. The closest silhouette to the center of each cluster is chosen as the key silhouette. Some of these 46 key frames are shown in Figure 4. Given these key silhouettes, we obtain the GMM vector representation as $[d_1, \ldots, d_i, \ldots, d_N]$, where d_i is the KLD distance between this silhouette and the ith key silhouette.

4.4. Comparison with other common shape descriptors

To validate the proposed vectorized silhouette representation based on GMM, extensive experiments have been conducted to compare GMM descriptor, vectorized GMM descriptor, shape context, and the Fourier descriptor. To produce shape context descriptors, a code book of the 90-dimensional shape context vectors is generated using the 3616 walking

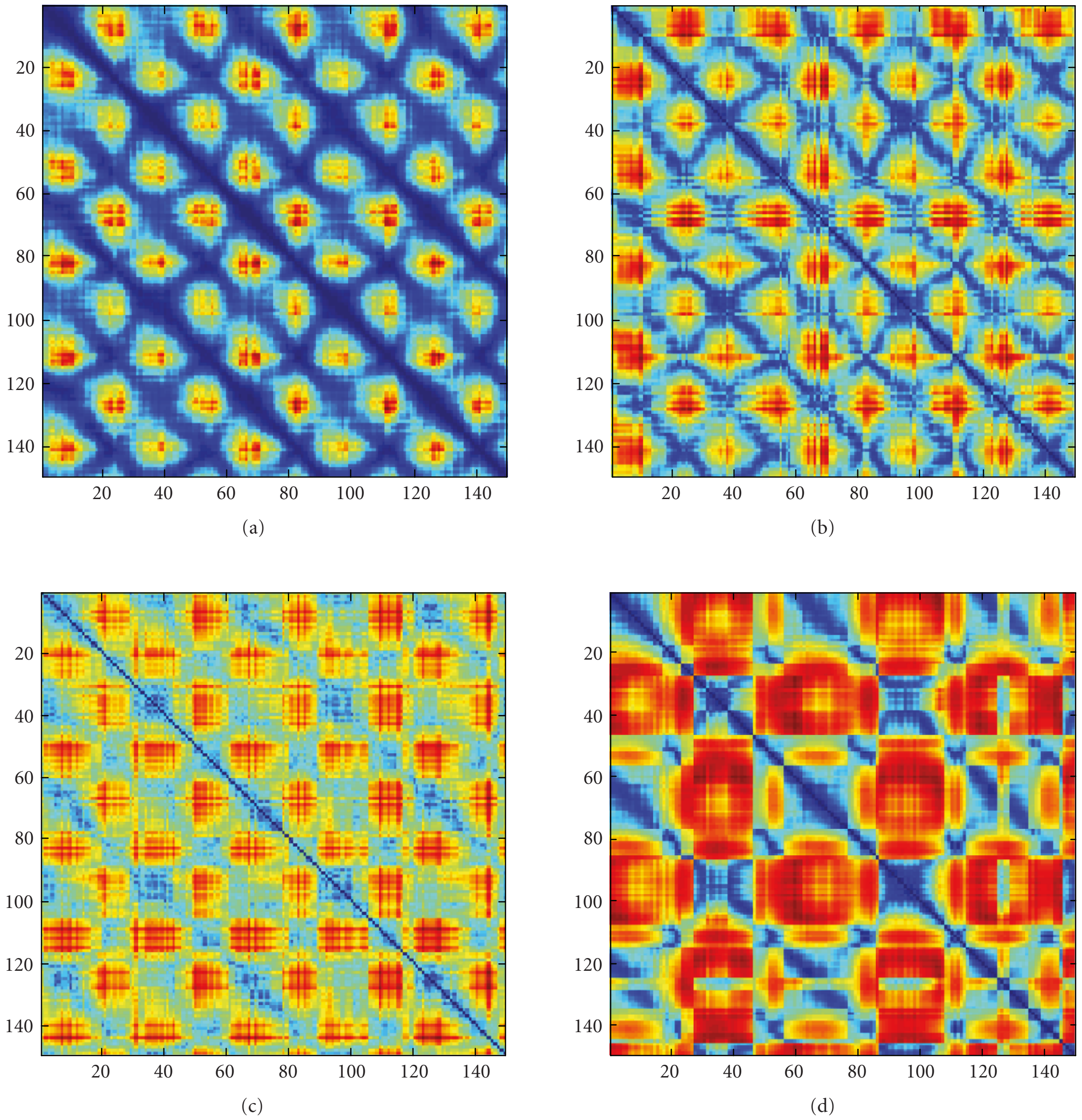


Figure 5: Distance matrices of a 149-frame sequence of side-view walking silhouettes computed using (a) GMM, (b) vectorized GMM using 46 key frames, (c) shape context, and (d) Fourier descriptor.

silhouettes from different views in $\mathbf{S}_T$ described in Section 3. Two hundred points are uniformly sampled on the contour. Each point has a shape context (5 radial, 12 angular bins, size range 1/8 to 3 on log scale). The code book center is clustered from shape context of all sampling points. To compare these four types of shape descriptor, distance matrices between silhouettes of a walking sequence are computed based on these descriptors. This sequence has 149 side views of a person walking parallel to a fixed camera over about two and half gait cycles (five steps). The four distance matrices are shown in Figure 5. All distance matrices are normalized with respect to the corresponding maxima. Dark blue pixels indicate small distances. Since the input is a side-view walking sequence, significant inter-frame similarity is presented, which results in a periodic pattern in the distance matrices. This is caused by both repeated movement in different gait cycles and the half cycle ambiguity in a side-view walking sequence in the same or different gait cycles (e.g., it is hard to tell the left arm from the right arm from a side-view walking silhouette even for humans). Figure 6 presents the distance values from the 10th frame to the remaining frames according to the four different shape descriptors. It can be seen from Figure 5 that the distance matrix computed using KLD based on GMM (Figure 5(a)) has the clearest pattern as a result of smooth similarity measure as shown by Figure 6(a). The continuity of the vectorized GMM is slightly deteriorated comparing to the original GMM. However, it is still much better than that of the shape context as shown by Figures 5(b), 5(c), 6(b), and 6(c). The Fourier descriptor is the least robust among the four shape descriptors. It is

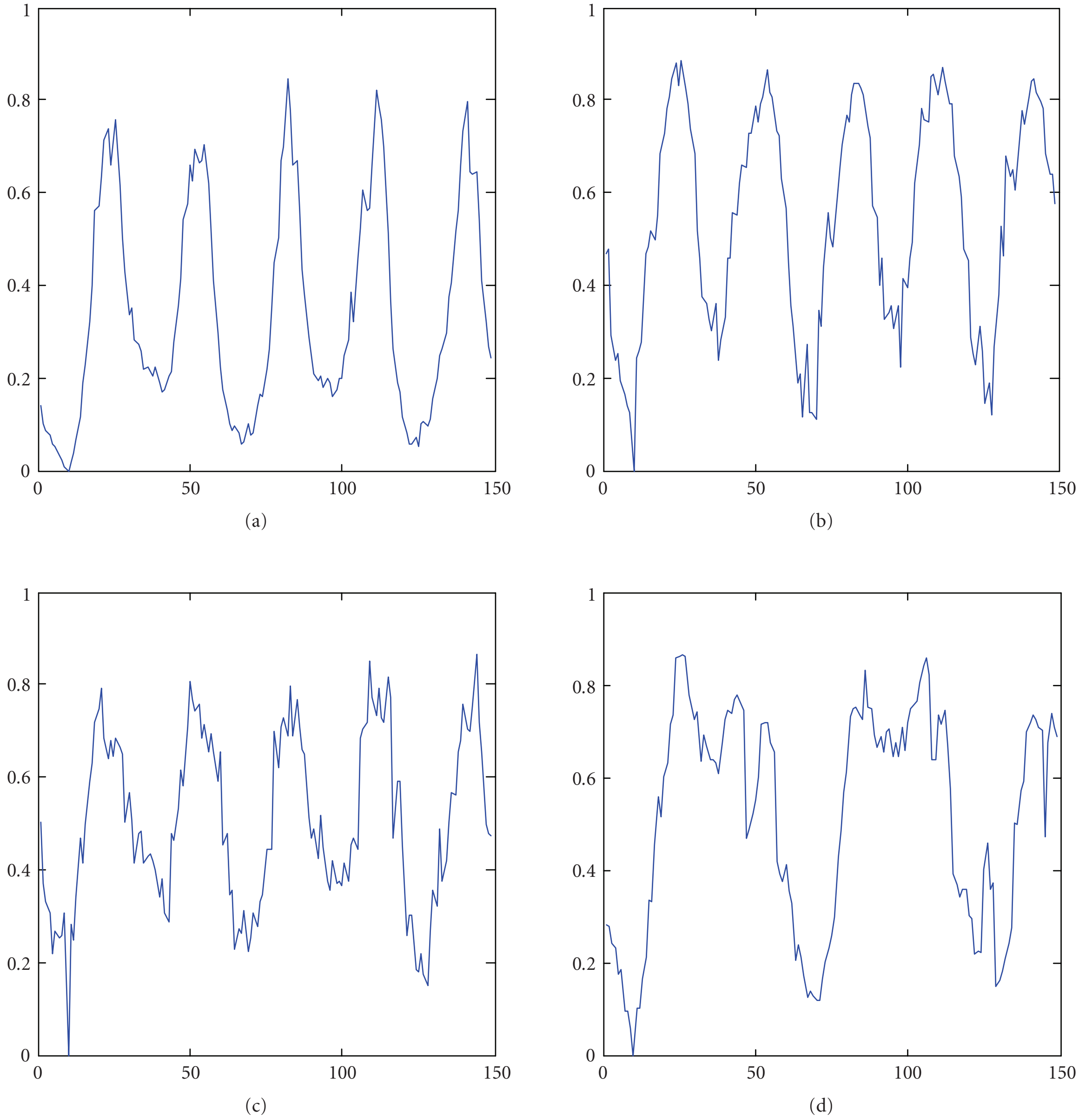


Figure 6: Distances between the 10th frame of the side-view walking sequence and all the other frames computed using (a) GMM, (b) vectorized GMM using 46 key frames, (c) shape context, and (d) Fourier descriptor.

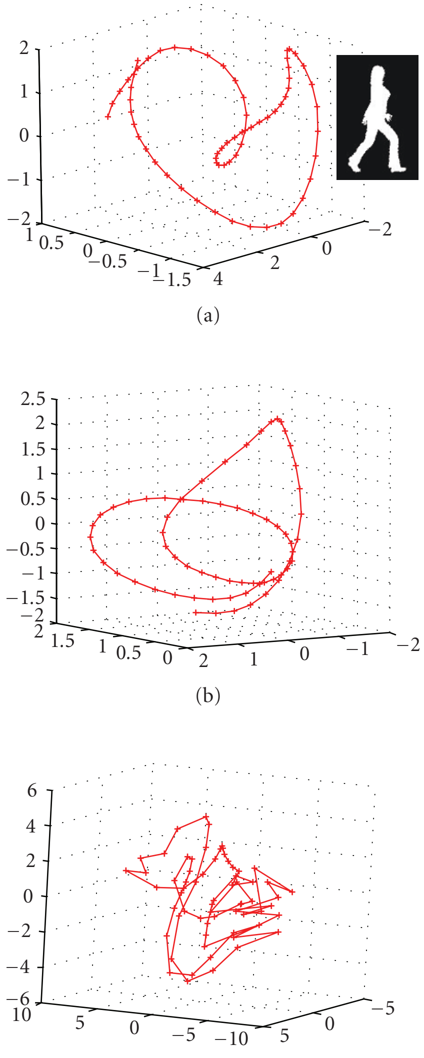
difficult to locate similar poses (i.e., find the valleys in Figure 6). This is because the outer contour of a silhouette can change suddenly between successive frames. Thus, the Fourier descriptor is discontinuous over time. Other than these four descriptors, the columnized vector of the raw silhouette is actually also a reasonable shape descriptor. However, the huge dimensionality (∼1000) of the raw silhouette makes the dimension reduction using GPLVM very time consuming and thus computationally prohibitive.

To take a close look at the smoothness of the three shape descriptors, original GMM, vectorized GMM, and shape context, we examine the resulting manifolds after dimension reduction and dynamic learning using GPDM. A smooth trajectory of latent point in the manifold indicates smoothness of the shape descriptor. Figure 7 shows three trajectories corresponding to these three shape descriptors. It can be seen that the vectorized GMM has a smoother trajectory than that of the shape context, which is consistent to our findings based on distance matrices.

5. DIMENSION REDUCTION AND DYNAMIC LEARNING

5.1. Dimension reduction of silhouettes using GPLVM

GPLVM [43] provides a probabilistic approach to nonlinear dimension reduction. In our proposed framework, GPLVM is used to reduce the dimensionality of the silhouettes and to recover the structure of silhouettes from different views.



(a)

(b)

(c)

Figure 7: Movement trajectories of 73 frames of side-view walking silhouette in the manifold learned using GPDM from three shape descriptors, including (a) GMM, (b) vectorized GMM using 46 key frames, and (c) shape context.

A detailed tutorial on GPLVM can be found in [14]. Here we briefly describe the basic idea of the GPLVM for the sake of completeness.

Let $\mathbf{Y} = [y_1, \dots, y_i, \dots, y_N]^T$ be a set of D-dimensional data points and $\mathbf{X} = [x_1, \dots, x_i, \dots, x_N]^T$ be the d-dimensional latent points associated with $\mathbf{Y}$. Assume that $\mathbf{Y}$ is already centered and $d < D$. $\mathbf{Y}$ and $\mathbf{X}$ are related by the following regression function,

$$y_i = W\varphi(x_i) + \eta_i, \tag{3}$$

where $\eta_i \sim \mathcal{N}(0, \beta^{-1})$ and the weight vector $W \sim \mathcal{N}(0, \alpha_W^{-1})$. $\varphi(x_i)$'s are a set of basis functions. Given $\mathbf{X}$, each dimension of $\mathbf{Y}$ is a Gaussian process. By assuming independence among different dimensions of $\mathbf{Y}$, the marginalized distribution of $\mathbf{Y}$ over W given $\mathbf{X}$ is

$$P(\mathbf{Y} \mid \mathbf{X}) \propto \exp\left(-\frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^T\right)\right), \tag{4}$$

where $\mathbf{K}$ is the gram matrix of the $\varphi(x_i)$'s. The goal in GPLVM is to find $\mathbf{X}$ and the parameters that maximize the marginal distribution of $\mathbf{Y}$. The resulting $\mathbf{X}$ is thus considered as a low-dimensional embedding of $\mathbf{Y}$. By using the kernel trick, instead of defining what $\varphi(x)$ is, one can simply define a kernel function over $\mathbf{X}$ and compute $\mathbf{K}$ so that $\mathbf{K}(i, j) = k(x_i, x_j)$. By using a nonlinear kernel function, one introduces a nonlinear dimension reduction. In our approach, the following radial basis fundtion (RBF) kernel is used:

$$k(x_i, x_j) = \alpha \exp\left(-\frac{\gamma}{2}\|x_i - x_j\|^2\right) + \beta^{-1}\delta_{x_i, x_j}, \tag{5}$$

where α is the overall scale of the output, γ is the inverse width of the RBFs. The variance of the noise is given by β^{-1}. $\Lambda = (\alpha, \beta, \gamma)$ are the unknown model parameters. We need to maximize (4) over Λ and $\mathbf{X}$, which is equivalent to minimizing the negative log of the objective function:

$$L = \frac{D}{2}\ln|\mathbf{K}| + \frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^T\right) + \frac{1}{2}\sum_i \|x_i\|^2 \tag{6}$$

with respect to the Λ and $\mathbf{X}$. The last term in (6) is added to take care of the ambiguity between the scaling of $\mathbf{X}$ and γ by enforcing a low energy regurlization prior over $\mathbf{X}$. Once the model is learned, given a new input data y_n its corresponding latent point x_n can be obtained by solving the likelihood objective function:

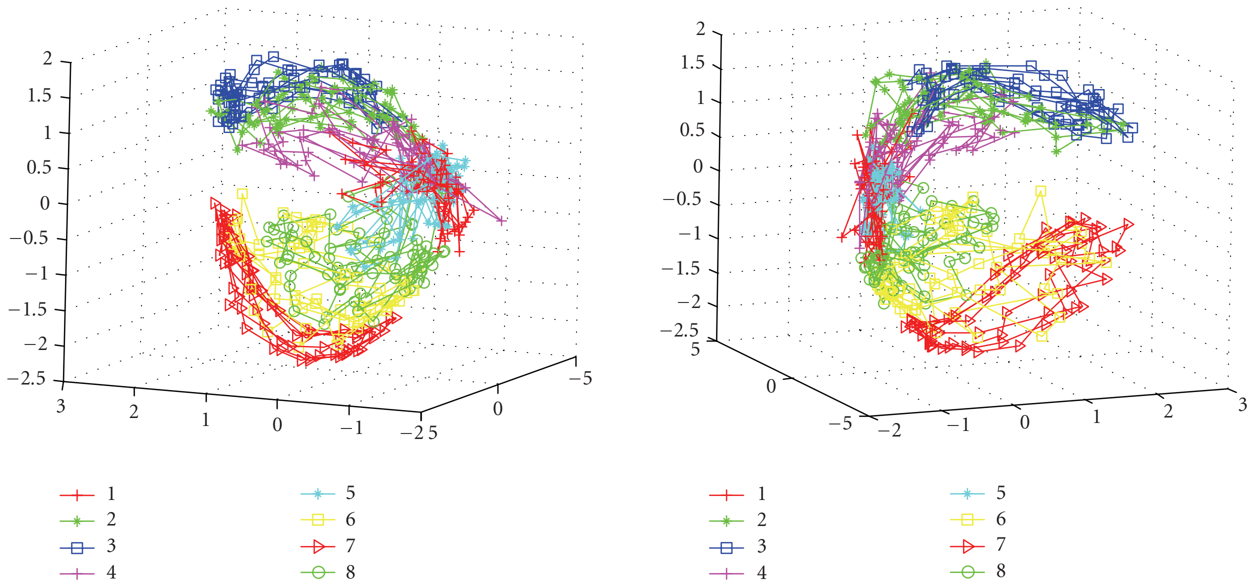
$$L_m(x_n, y_n) = \frac{\|y_n - \mu(x_n)\|^2}{2\sigma^2(x_n)} + \frac{D}{2}\ln\sigma^2(x_n) + \frac{1}{2}\|x_n\|^2, \tag{7}$$

where

$$\mu(x_n) = \mu + \mathbf{Y}^T\mathbf{K}^{-1}\mathbf{k}(x_n), \tag{8}$$

$$\sigma^2(x_n) = k(x_n, x_n) - \mathbf{k}(x_n)^T\mathbf{K}^{-1}\mathbf{k}(x_n), \tag{9}$$

$\mu(x_n)$ is the mean pose reconstructed from the latent point x_n, and $\sigma^2(x_n)$ is the reconstruction variance. μ is the mean of the training data $\mathbf{Y}$. $\mathbf{k}(x_n)$ is the kernel function of x_n evaluated over all the training data. Given input y_n, the initial latent position is obtained as $x_n = \arg\min_{x_n} L_m(x_n, y_n)$. Given x_n, the mean data reconstructed in high dimension can be obtained using (8). In our implementation, we make use of the FGPLVM Matlab toolbox (http://www.cs.man.ac.uk/neill/gpsoftware.html) and the fully independent training conditional (FITC) approximation [44] software provided by Dr. Neil Lawrence for GPLVM learning and bidirectional mapping between $\mathbf{X}$ and $\mathbf{Y}$. Although the FITC approximation was used to expedite the silhouette learning process, it took about five hours to process all the 3616 training silhouettes. As a result, it will be difficult to extend our approach to handle multiple motions simultaneously.



(a)

(b)

Figure 8: The first three dimensions of the silhouette latent points of 640 walking frames.

When applying GPLVM to silhouettes modeling, the image feature points are embedded in a 5D latent space $\mathcal{S}$. This is based on the consideration that three dimensions are the minimum representation of walking silhouettes [34]. One more dimension is enough to describe view changes along a body-centroid-centered circle in the horizontal plane of the subject. We then add the fifth dimension to allow the model to capture extra variations, for example, introduced by body shapes of different 3D body models used in synthetic data generation. By using the FGPLVM toolbox, we obtained the corresponding manifold of the training silhouette data set $\mathbf{S}_T$ described in Section 3. In Figure 8, the first three dimensions of 640 silhouette latent points from $\mathbf{S}_T$ are shown. They represent 80 poses of one gait cycle (two steps) with 8 views for each pose. It can be seen in Figure 8 that silhouettes in different ranges of view angles are generally in different part of the latent space with certain levels of overlapping. Hence, the GPLVM can partly capture the structure of the silhouettes introduced by view changes.

5.2. Movement dynamic learning using GPDM

GPDM simultaneously provides a low-dimensional embedding of human motion data and dynamics. Based on GPLVM, [15] proposed GPDM to add a dynamic model in the latent space. It can be used for the modeling of a single type of motion. Reference [1] extended the GPDM to balanced-GPDM to handle multiple subjects' stylistic variation by raising the dynamic density function.

GPDM defines a Gaussian process to relate latent points x_t to x_{t-1} at time t. The model is defined as:

$$\begin{aligned} x_t &= A\varphi_d(x_{t-1}) + n_x \\ y_t &= B\varphi(x_t) + n_y, \end{aligned} \tag{10}$$

where A and B are regression weights, and n_x and n_y are Gaussian noise. The marginal distribution of $\mathbf{X}$ is given by

$$p(\mathbf{X} \mid \Lambda_d) \propto \exp\left(-\frac{1}{2}\operatorname{tr}\left(\mathbf{K}_x^{-1}(\hat{\mathbf{X}} - \tilde{\mathbf{X}})(\hat{\mathbf{X}} - \tilde{\mathbf{X}})^T\right)\right), \tag{11}$$

where $\hat{\mathbf{X}} = [x_2, \dots, x_t]^T$, $\tilde{\mathbf{X}} = [x_1, \dots, x_{t-1}]^T$, and Λ_d consists of the kernel parameters which will be introduced later. $\mathbf{K}_x$ is the kernel associated with the dynamics Gaussian process and is constructed on $\tilde{\mathbf{X}}$. We use an RBF kernel with a white noise term for the dynamics as in [14]

$$k_x(x_t, x_{t-1}) = \alpha_d \exp\left(-\frac{\gamma_d}{2}\left\| x_t - x_{t-1} \right\|^2 \right) + \beta_d^{-1}\delta_{t,t-1}, \tag{12}$$

where $\Lambda_d = (\alpha_d, \gamma_d, \beta_d)$ are parameters of the kernel function for the dynamics. GPDM learning is similar to GPLVM learning. The objective function is given by two marginal log-likelihoods:

$$\begin{aligned} L_d &= \frac{d}{2}\ln\left|\mathbf{K}_X\right| + \frac{1}{2}\operatorname{tr}\left(\mathbf{K}_x^{-1}(\hat{\mathbf{X}} - \tilde{\mathbf{X}})(\hat{\mathbf{X}} - \tilde{\mathbf{X}})^T\right) \\ &\quad + \frac{D}{2}\ln|\mathbf{K}| + \frac{1}{2}\operatorname{tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^T\right), \end{aligned} \tag{13}$$

$(\mathbf{X}, \Lambda, \Lambda_d)$ are found by maximizing L_d. Based on Λ_d, one is ready to sample from the movement dynamics, which is important in particle filter-based tracking. Given x_{t-1}, x_t can be inferred from the learned dynamics $p(x_t \mid x_{t-1})$ as follows:

$$\begin{aligned} \mu_x(x_t) &= \hat{\mathbf{X}}^T\mathbf{K}_X^{-1}\mathbf{k}_x(x_{t-1}), \\ \sigma_x^2(x_t) &= k_x(x_{t-1}, x_{t-1}) - \mathbf{k}_x(x_{t-1})^T\mathbf{K}_X^{-1}\mathbf{k}_x(x_{t-1}), \end{aligned} \tag{14}$$

where $\mu_x(x_t)$ and $\sigma_x^2(x_t)$ are the mean and variance for prediction. $\mathbf{k}_x(x_{t-1})$ is the kernel function of x_{t-1} evaluated

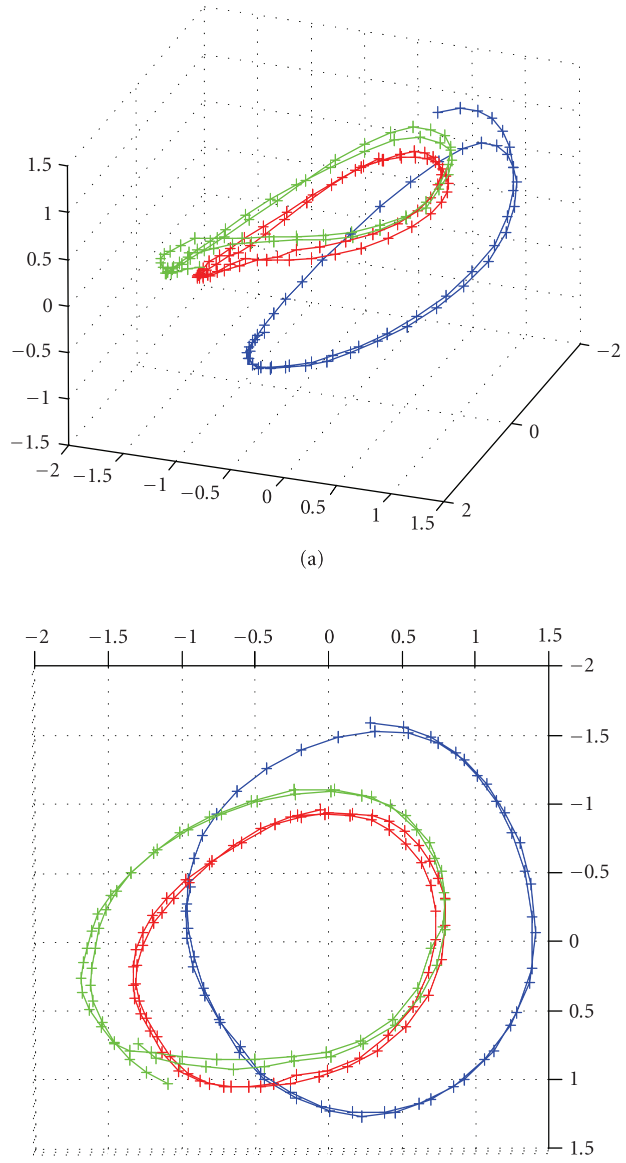


Figure 9: Two views of a 3D GPDM learned using gait data set Θ_T (see Section 3), including six walking cycles' frames from three subjects.

over $\hat{X}$. In our implementation, the balanced GPDM [1] is adopted to balance the effect of the dynamics and the reconstruction. As a data preprocessing step, we first center the motion capture data and then rescale the data to unit variance [45]. This preprocessing reduces the uncertainty in high-dimensional pose space. In addition, we follow the learning procedure in [14] so that the kernel parameters in Λ_d are prechosen instead of being learned for the sake of simplicity. This is also due to the fact that these parameters carry clear physical meanings so that they can be reasonably selected by hand [14]. In our experiment, $\Lambda_d = (0.01, 10^6, 0.2)$. The local joint angles from motion capture are projected to joint angle manifold Θ. By augmenting Θ with the torso orientation space Ψ, we obtain the complete pose latent space $\mathcal{C}$. A 3D movement latent space learned using GPDM from the joint angle data set Θ_T described in Section 3 (six walking cycles from three subjects) are shown in Figure 9.

6. BME-BASED POSE INFERENCE

The backward mapping from the silhouette manifold $\mathcal{S}$ to the joint space of the pose manifold and the torso orientation $\mathcal{C}$ is needed to conduct both autonomous tracking initialization and sampling from the most recent observation. Different poses can generate the same silhouette, which means this backward mapping is one-to-many from a single-view silhouette.

6.1. The basic setup of BME

The BME-based pose learning and inference method we use here mainly follows our previous work in [41]. Let $s \in \mathcal{S}$ be the latent point of an input silhouette and $c \in \mathcal{C}$ the corresponding complete pose latent point. In our BME setup, the conditional probability distribution $p(c \mid s)$ is represented as a mixture of K predictions from separate experts:

$$p(c \mid s, \Xi) = \sum_{k=1}^{K} g(z_k = 1 \mid s, V)\, p(c \mid s, z_k = 1, U_k), \qquad (15)$$

where $\Xi = \{\mathbf{V}, \mathbf{U}\}$ denotes the model parameters. z_k is a latent variable such that $z_k = 1$ indicates that s is generated by the kth expert, otherwise $z_k = 0$. $g(z_k = 1 \mid s, \mathbf{V})$ is the *gate* variable, which is the probability of selecting the kth expert given s. For the kth expert, we assume that c follows a Gaussian distribution:

$$p(c \mid s, z_k = 1, U_k) = \mathcal{N}(c; f(s, \mathbf{W}_k), \Omega_k), \qquad (16)$$

where $f(s, \mathbf{W}_k)$ and Ω_k are the mean and covariance matrix of the output of the kth expert. $U_k \equiv \{W_k, \Omega_k\}$ and $\mathbf{U} \equiv \{U_k\}_{k=1}^{K}$. Following [33], in our framework we consider the joint distribution $p(c, s \mid \Xi)$ and assume the marginal distribution of s is also a mixture of Gaussian. Hence, the gate variables are given by the posterior probability

$$g(z_k = 1 \mid s, \mathbf{V}) = \frac{\lambda_k \mathcal{N}(s; \mu_k, \Sigma_k)}{\sum_{l=1}^{K} \lambda_l \mathcal{N}(s; \mu_l, \Sigma_l)}, \qquad (17)$$

where $\mathbf{V} = \{V_k\}_{k=1}^{K}$. $V_k = (\lambda_k, \mu_k, \Sigma_k)$ and λ_k, μ_k, Σ_k are the mixture coefficient, the mean and covariance matrix of the marginal distribution of s for the kth expert, respectively. λ_k's sum to one.

Given a set of training samples $\{(s^{(i)}, c^{(i)})\}_{i=1}^{N}$, the BME model parameter vector Ξ needs to be learned. Similar to [10], in our framework the expectation-maximization (EM) algorithm is used to learn Ξ. In the E-step of the nth iteration, we first compute the *posterior gate* $h_k^{(i)} = p(z_k = 1 \mid s^{(i)}, c^{(i)}, \Xi^{(n-1)})$ using the current parameter estimate $\Xi^{(n-1)}$. $h_k^{(i)}$ is basically the posterior probability that $(s^{(i)}, c^{(i)})$ is generated by the kth expert. Then in the M-step, the estimate of

Ξ is refined by maximizing the expectation of the log likelihood of the complete data including the latent variables. It can be easily shown [33] that the object function can be decomposed into two subfunctions: one related to gate parameters $\mathbf{V}$ and the other one to the expert parameters $\mathbf{U}$. Details about the update of $\mathbf{V}$ can be found in [33], which are essentially the basic equations in the M-step for Gaussian mixture modeling of $\{s^{(i)}\}_{i=1}^{N}$ using EM.

6.2. *Experts learning using weighted RVM*

In this section, we present our method for the learning of the expert parameters $\mathbf{U} = \{U_k\}_{k=1}^{K}$. There are K data pair clusters in BME. For each cluster, we need to construct an expert for the mapping from silhouette latent point s to the complete pose latent point c. The learning process of the parameters for all of the K experts is identical. We now consider the learning of $U_k = (\mathbf{W}_k, \Omega_k)$. The input to the learning algorithm is $\{(s^{(i)}, c^{(i)}), h_k^{(i)}\}_{i=1}^{N}$, including the original training data pairs and their associated posterior gate values with respect to the kth expert. $h_k^{(i)}$'s are the outputs of the E-step of the BME learning mentioned in the previous section. Following [33], the objective function for the optimization of the expert parameters is given by

$$L_e = \sum_{i=1}^{N} h_k^{(i)} p\left(c^{(i)} \mid \left(s^{(i)}, U_k\right)\right). \tag{18}$$

In our proposed framework, we deployed RVM [46] to solve this maximization problem. In our current implementation, individual dimensions of c are considered separately assuming independence between dimensions. To be concise in notation, in the remaining of this section we assume that c is a scalar. When c is a vector, the expert learning processes in all dimensions are identical. Denote $\mathbf{S} = \{s^{(i)}\}_{i=1}^{N}$, $\mathbf{C} = [c^{(1)}, \ldots, c^{(i)}, \ldots, c^{(N)}]^T$, and $\mathbf{H} = \operatorname{diag}(h_i)$, $i = 1, \ldots, N$. The RVM regression from s to c takes the following form:

$$c \sim \mathcal{N}\left(c; \phi(s)^T \mathbf{W}_k, \Omega_k\right), \tag{19}$$

where $\phi(s) = [1, k(s, s^{(1)}), \ldots, k(s, s^{(N)})]^T$ is a column vector of known kernel functions. Hence, the likelihood of $\mathbf{C}$ is

$$p\left(\mathbf{C} \mid \mathbf{S}, \mathbf{W}_k, \Omega_k\right) \propto \exp\left\{-\frac{\left(\mathbf{C} - \Phi\mathbf{W_k}\right)^T \mathbf{H}\left(\mathbf{C} - \Phi\mathbf{W_k}\right)}{2\Omega_k}\right\}, \tag{20}$$

where $\Phi = [\phi(s^{(1)}), \ldots, \phi(s^{(N)})]^T$ is the kernel matrix. To overfitting, a diagonal hyper-parameter matrix $\mathbf{A}$ is introduced to model the prior of $\mathbf{W_k}$: $p(\mathbf{W_k} \mid \mathbf{A}) \sim \mathcal{N}(\mathbf{W_k}; \mathbf{0}, \mathbf{A}^{-1})$. Following the derivation in [46], it can be easily shown that in the case of weighted RVM, the conditional probability distribution of $\mathbf{W}$ is given by

$$p\left(\mathbf{W_k} \mid \mathbf{C}, \mathbf{S}, \mathbf{H}, \mathbf{A}, \Omega_k\right) = \mathcal{N}\left(\mathbf{W_k}; \widehat{\mathbf{W}}_k, \mathbf{\Sigma}\right), \tag{21}$$

$\widehat{\mathbf{W}}_k, \mathbf{\Sigma}$ are computed through the following iterative procedure:

$$\begin{aligned}
\mathbf{\Sigma} &= \left(\mathbf{\Omega}_{\boldsymbol{k}}^{-1} \mathbf{\Phi}^T \mathbf{H} \mathbf{\Phi} + \mathbf{A}\right)^{-1}, \\
\widehat{\mathbf{W}}_k &= \Omega_k^{-1} \mathbf{\Sigma} \Phi^T \mathbf{H} \mathbf{C} \\
\alpha_i^{\text{new}} &= \frac{1 - \Sigma_{ii}}{\widehat{w}_i^2}, \\
\left(\Omega_k\right)^{\text{new}} &= \frac{\left(\mathbf{C} - \Phi\widehat{\mathbf{W}}_k\right)^T \mathbf{H}\left(\mathbf{C} - \Phi\widehat{\mathbf{W}}_k\right)}{N - \sum_{i=1}^{N}\left(1 - \alpha_i \Sigma_{ii}\right)},
\end{aligned} \tag{22}$$

where $\widehat{w}_i$ is the ith element of $\widehat{\mathbf{W}}_k$. α_i and Σ_{ii} are the ith diagonal terms of $\mathbf{A}$ and $\mathbf{\Sigma}$, respectively. Once the parameters have been estimated, given a new input s_*, the conditional probability distribution of the output is given by

$$c_* \sim \mathcal{N}\left(c_*; \phi\left(s_*\right)^T \widehat{\mathbf{W}}_k, \widehat{\Omega}_k\right) \tag{23}$$

with $\widehat{\Omega}_k = (\mathbf{C}_j - \Phi\widehat{\mathbf{W}}_k)^T \mathbf{H}(\mathbf{C} - \Phi\widehat{\mathbf{W}}_k)$.

6.3. *Experiments results for 3D pose inference*

To demonstrate the validity of the above BME-based pose inference framework, some experimental results are included in this section. The resulting BME constitutes a mapping from $\mathcal{S}$ to $\mathcal{C}$. The training data used includes the projection of silhouette training set $\mathbf{S}_T$ onto $\mathcal{S}$ using GPLVM and the projection of the pose data $\mathbf{C}_T$ on $\mathcal{C}$ using GPDM. The number of experts in BME is the number of mappings from $\mathcal{S}$ to $\mathcal{C}$. When the local body kinematics is fixed, usually five mappings are sufficient to cover the variations introduced by different torso orientations. When the torso orientation is fixed, the number of mappings needed to handle changes due to different body kinematics depends on the complexity of the actual movement. In the case of gait, three mappings are sufficient. Therefore, in our experiment when both torso orientation and body kinematics are allowed to vary, fifteen experts were learned in BME for pose inference of gait.

Synthetic testing data were generated using different 3D human models and motion sequences from different subjects. Some reconstructed poses for the first two most probable outputs, that is, the outputs with the first two largest gate values computed using (17), are shown in Figure 10. It is clear that BME can handle ambiguous poses.

A real video (40 frames, two steps' side-view walking) was also used to evaluate this approach. Due to observation noise, the silhouettes extracted from this video were not as clean as the synthesized ones. However, BME can still produce perceptually sound results. Some recovered poses are shown in Figure 11.

7. TRACKING USING PARTICLE FILTER

A particle filter defined over $\mathcal{C}$ is used for 3D tracking of articulated motion. The state parameter at time t is $c_t = (\theta_t, \psi_t)$, where θ_t is the latent point of the body joint angles, and ψ_t is the torso orientation. Given a sequence of latent silhouette points $s_{1:t}$ obtained from input images using

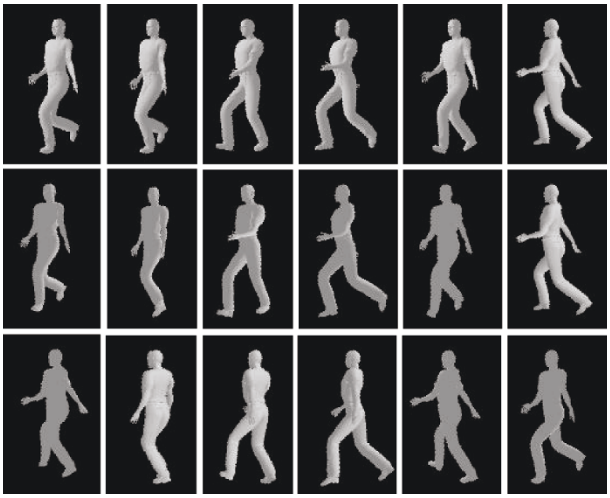


Figure 10: BME-based pose inference results of a synthetic walking sequence. Top row: input images; middle row: the most probable poses; bottom row: the second most probable poses.

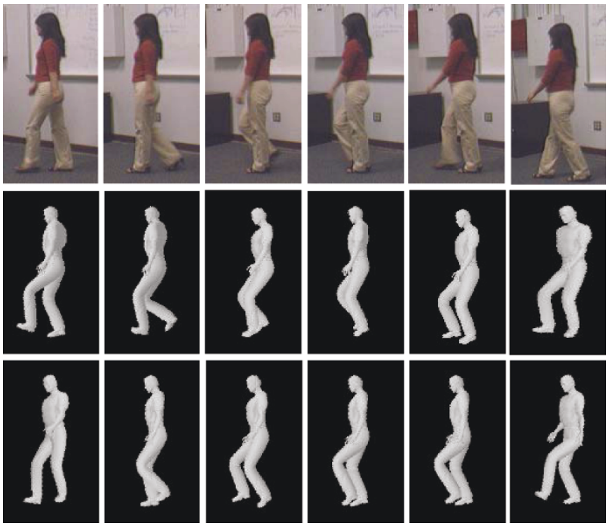


Figure 11: BME-based pose inference results of a real walking video. Top row: input video images; middle row: the most probable poses; The third row: The second most probable poses.

GPLVM, the posterior distribution of the state is approximated by a set of weighted samples $\{w_t^{(i)}, c_t^{(i)}\}_{i=1}^{M}$. The importance weights of the particles are propagated over time as follows:

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{p(s_t \mid c_t^{(i)})\, p(c_t^{(i)} \mid c_{t-1}^{(i)})}{q(c_t^{(i)} \mid c_{t-1}^{(i)}, s_t)}. \quad (24)$$

Pose estimation results from BME are used to initialize the tracking. BME cannot disambiguate, however, it provides multiple possible solutions. In our experiments, the first three most probable solutions from BME are selected as tracking seeds according to their gate values. Then samples are drawn around these seeds. Generally, a wrong initialized branch will merge with the correct ones after several frames estimation. But in some situations, due to inherent ambiguity, an ambiguous solution might also stay. For example, multiple tracking trajectories were obtained in some of our experiments as discussed in Section 8.

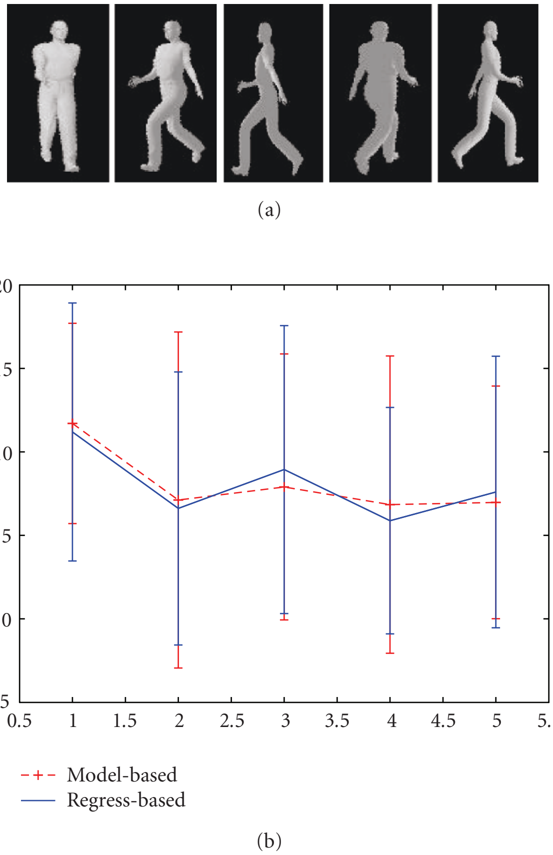


Figure 12: (a) Sample silhouettes from view number 1through view number 5, indexed starting from the leftmost figure. (b) RMS error results using rendering and regression approaches. The average error is close for both approaches.

7.1. Sampling

Particles are propagated over time from a proposal distribution q. To take into account both the movement dynamics and the most recent observation s_t, in our approach we select q to be the mixture of two distributions as follows:

$$q(c_t \mid c_{t-1}, s_t) = \pi q_b(c_t \mid s_t) + (1 - \pi) p(c_t \mid c_{t-1}), \quad (25)$$

where $q_b(c_t \mid s_t)$ is chosen as the BME output $p(c \mid s, \Xi)$ given by (15) and

$$p(c \mid s, \Xi) = \sum_{k=1}^{K} g(z_k = 1 \mid s, V)\, \mathcal{N}(c; \phi(s)^T \widehat{\mathbf{W}}_k, \widehat{\Omega}_k). \quad (26)$$

In our experiment, we only use the first three most probable components of the 15 BME outputs and draw samples

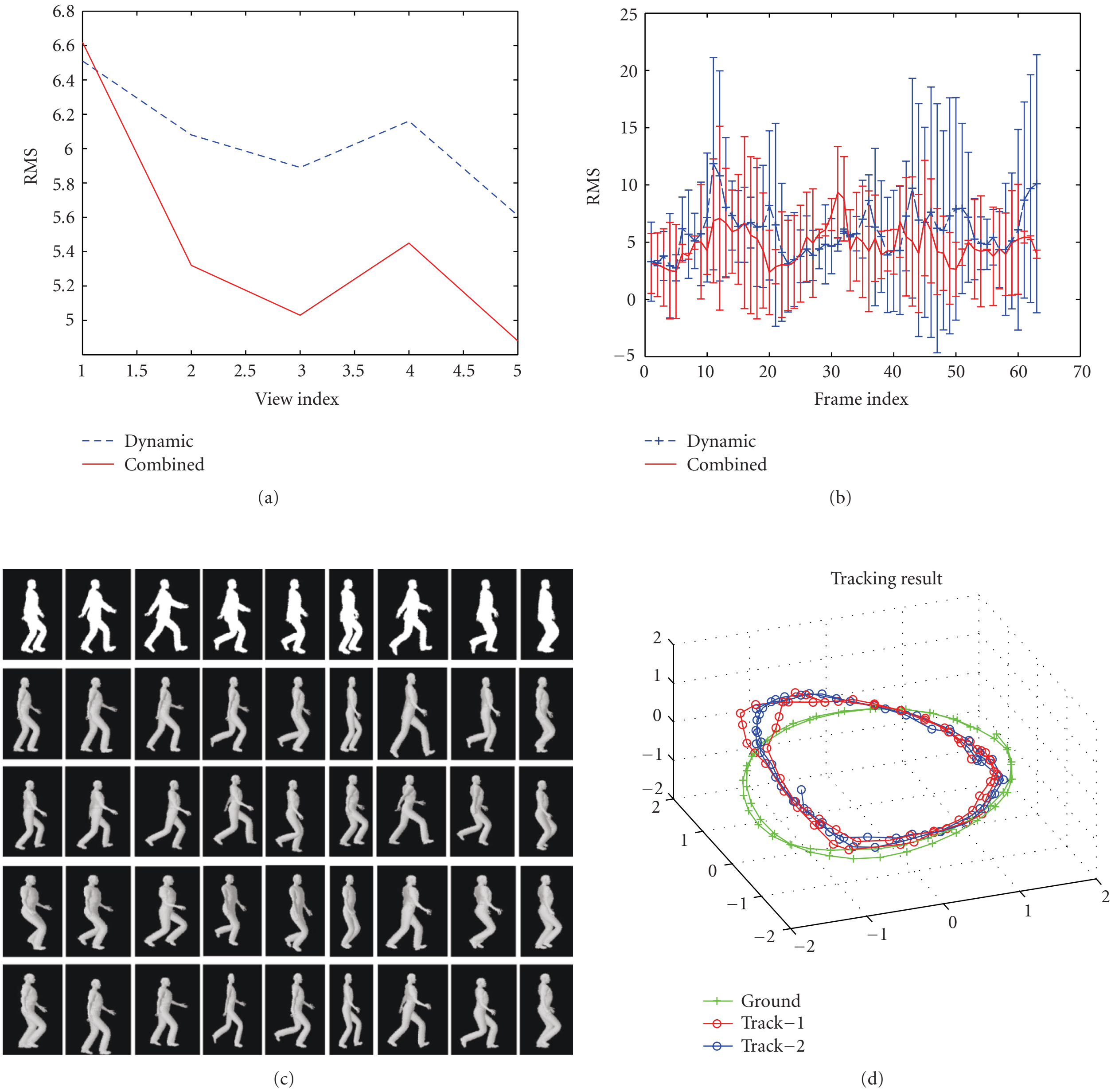


Figure 13: Experimental results obtained using synthetic data. (a) average RMS errors obtained using synthetic testing sequences from different views; (b) frame-wise RMS from the side view. (c) exemplar input silhouettes of view 5 and tracking results. Top row: some input silhouettes; the second and third rows: two plausible solutions obtained using our framework; the fourth row: the recovered poses directly from the observed image using BME; bottom row: the recovered poses obtained using only dynamic prediction; (d) the tracked movement trajectories in the joint angle manifold Θ.

according to the regression covariance. The second term in (25) is from movement dynamics learned using GPDM and a first-order AR model for the torso orientation

$$p(c_t \mid c_{t-1}) = p(\theta_t \mid \theta_{t-1})p(\psi_t \mid \psi_{t-1}). \qquad (27)$$

In (25), π is the mixture coefficient of the BME-based prediction and the dynamics-based prediction components. In our experiments, $\pi = 0.5$. Because of $\mathcal{C}$ is a 5D space, only 100 particles were used in tracking, which makes the tracking computationally efficient.

7.2. *Likelihood evaluation*

In our framework, we take RVM as the regression function to construct a forward mapping from $\mathcal{C}$ to $\mathcal{S}$. The hypothesized pose latent point c_* is first projected to s_*, and then to the image feature space using the inverse mapping in GPLVM. In the RVM learning, we used the same training set as that in the BME learning described in Section 6.3. The final number of the relevance vectors accounts about 10%–20% of the total data. To evaluate the effectiveness of the RVM-based mapping, it was compared against a model-based approach, in which the hypothesized torso orientation and body kinematics were obtained from c_*, and then Maya was used to render the corresponding silhouettes of the 3D body model. The silhouette distance is measured in the vectorized GMM feature space. Comparison results using five walking images are included in this section. For each input silhouette, fifteen poses were inferred using BME learned according to the method presented in Section 6.3. Given a pose, two vectorized GMM descriptors were obtained using both the RVM- and model-based approaches. The root mean square errors (RMSEs) between the predicted and true image features were then computed. Figure 12(a) shows exemplar input silhouettes from view number 1 through view number 5, indexed starting from the leftmost figure. For each view and each method, given an input silhouette, we found the smallest RMSE among all of the 15 candidate poses provided by BME. We then compute the average of the smallest RMSE over all the input silhouettes. The average RMSEs of all the five views from both methods are shown in Figure 12(b). It can be seen that the average RMSEs are close for these two approaches, which indicates that the likelihoods of a good pose candidate computed using both methods are similar. Hence, we can use the example-based approach for computation efficiency. In addition, the example-based approach does not need a 3D body model of the subject, which also simplifies the problem.

8. EXPERIMENTAL RESULTS

The proposed framework has been tested using both synthetic and real image sets. The system was trained using training data described in Section 3.

During tracking, the preprocessing of the input image takes about 800 milliseconds per frame, including silhouette extraction, GMM, and vectorization. Out of these three operations, GMM is the most time consuming, taking about 600 milliseconds. The mapping from vectorized GMM to the silhouette manifold $\mathcal{S}$ is the most time consuming operation in our current implementation, which takes about 3 seconds per frame. BME inference, sampling, and sample weight evaluation is fairly fast, taking about 200 milliseconds per frame. The total time to process one frame of input image is about 4 seconds.

We first used synthetic data to evaluate the accuracy of our tracking system. The test sequence was created using motion capture data (sequence 08_02 in the CMU database) of a new subject not included in training sets and a new 3D body model. Some of the camera views are also new. This test sequence has 63 frames of two walking cycles. The five camera views used to create the testing data are the same as those shown in Figure 12(a). The RMSEs between the ground truth and the estimated joint angles are given to show the tracking accuracy. The tracking results based only on sampling from the GPDM movement dynamics are also included for comparison purposes. One hundred particles were used in both cases. The average RMS errors from different views are shown in Figure 13(a). The tracking from view number 1 (frontal view) is rather ambiguous. The frame-wise RMSE of view number 5 (side view walking from left to right) is given in Figure 13(b). Figure 13(c) presents some input silhouettes from view 5 (top row) and their estimated poses (the second and third rows). To show the effectiveness of the proposed framework, results from the static image estimation using only BME and results from sampling from dynamics are also shown in the fourth and fifth rows, respectively. It can be seen that our proposed framework provided the most accurate tracking results among all three methods.

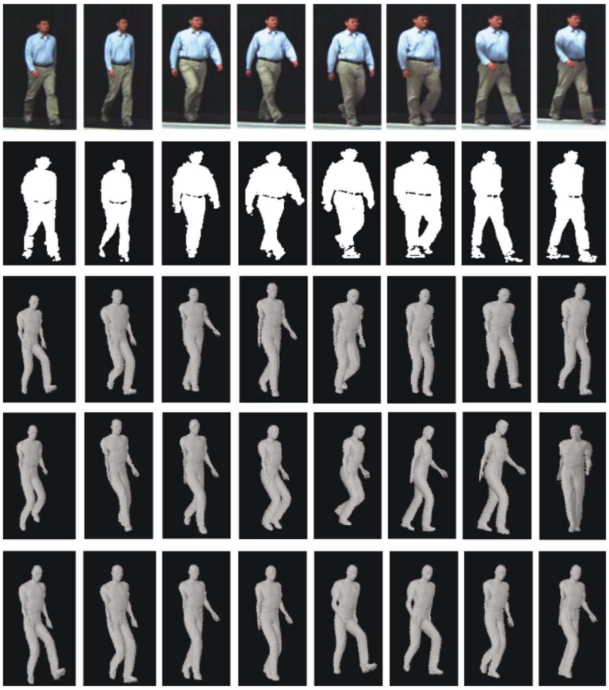
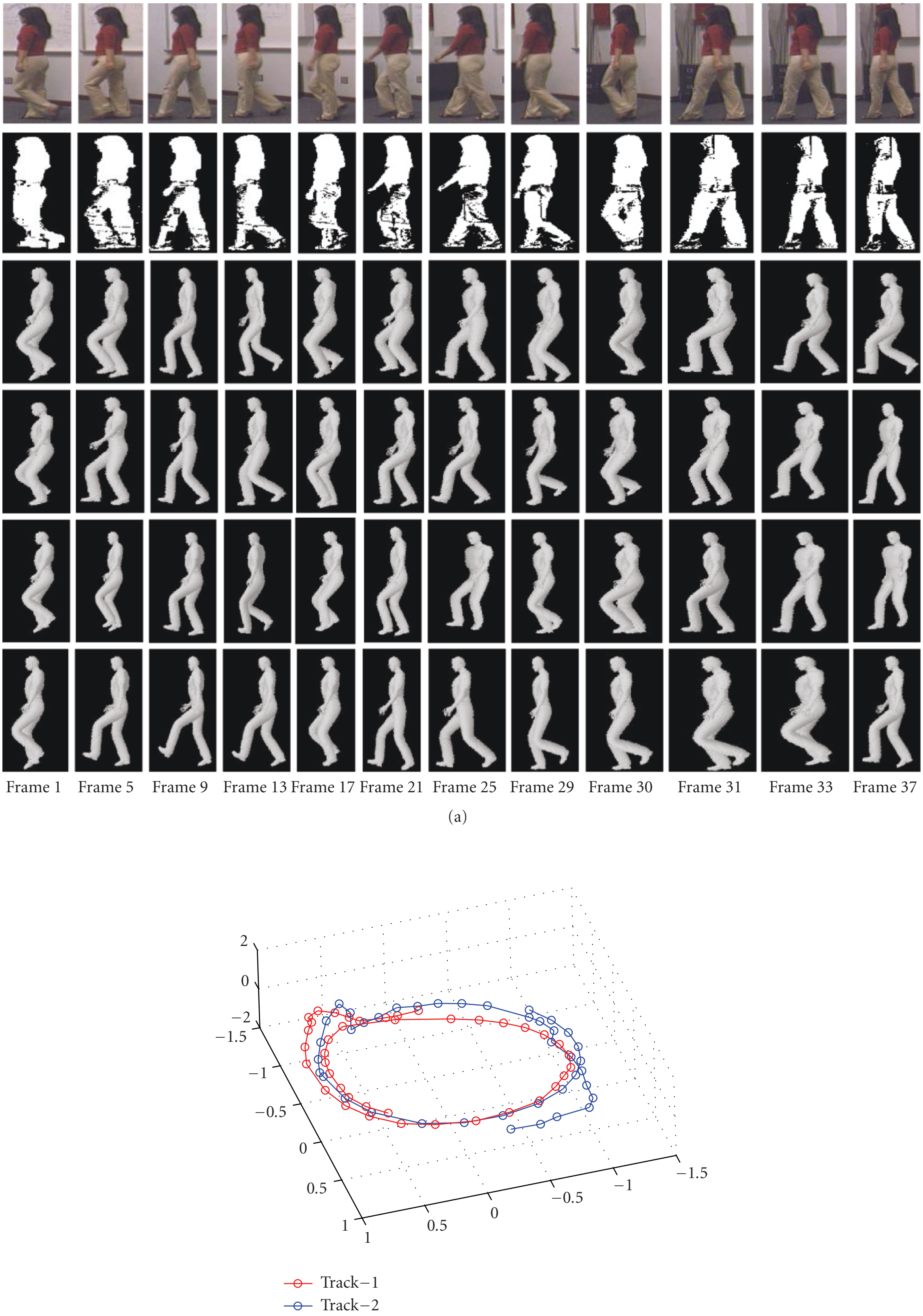


Figure 14: Reconstructed poses for a real image sequence of 42 frames. Top row: sample input images; the second row: extracted silhouettes; the third row: recovered poses using the proposed framework; the fourth row: recovered poses directly from the observed image using only BME; bottom row: recovered poses obtained using only dynamic prediction.

The result obtained using the proposed method successfully describes the inherent left-and-right ambiguity present in the input silhouette. It can been seen that the initial and the continuous silhouettes are difficult to be distinguished from the left and the right. The proposed framework returned both admissible results, although we cannot tell which one corresponds to the true movement. Both movement trajectories tracked in Θ are shown in Figure 13(d).

A real video sequence of 42 frames of two steps walking along diagonal direction to the camera was used to evaluate the proposed system. The subject was not seen in the system training. One hundred particles were used in the tracking. Due to observation noise in the video, the extracted silhouettes were not as clean as the synthesized ones. However, the proposed approach can still produce plausible results. Some recovered poses are shown in Figure 14.

Another real video sequence (40 frames, two steps side view walking) was used to evaluate the proposed system. This video is slightly more challenging than the previous



(a)

(b)

Figure 15: Pose tracking results obtained using a real image sequence of 40 frames, where there is a movement jump between frame 29 to frame 30. (a) Top row: sample input images; the second row: extracted silhouettes; the third and fourth rows: two plausible solutions tracked using our framework; the fifth row: recovered poses directly from the corresponding images using BME; bottom row: recovered poses obtained using only dynamic prediction. (b) The tracked movement trajectories in the joint angle manifold Θ.

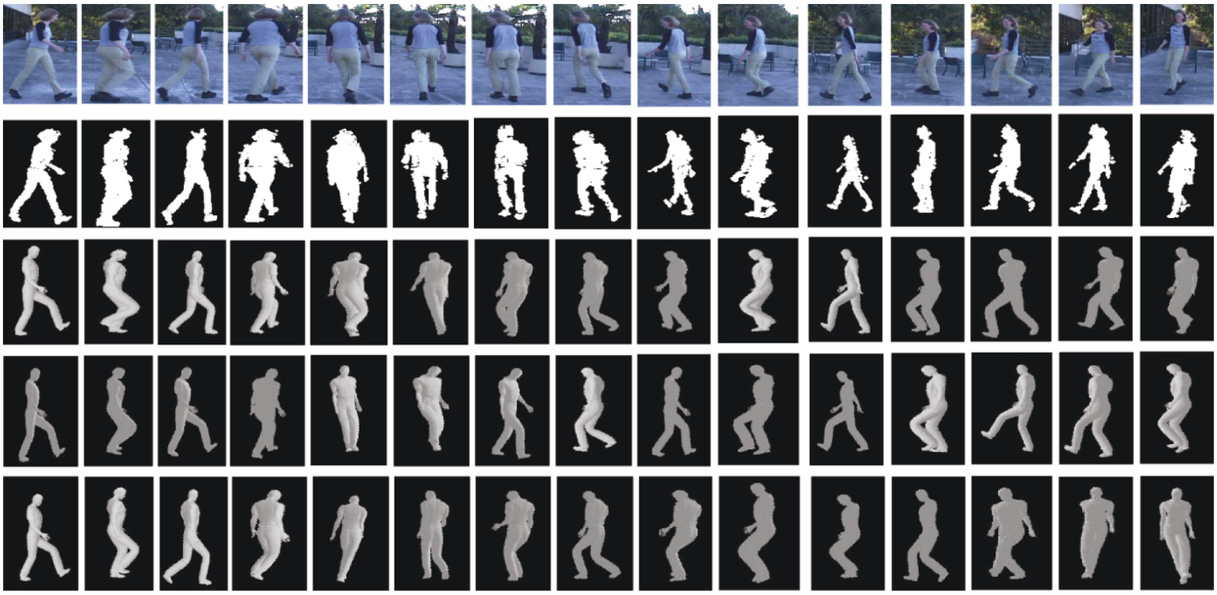


Figure 16: Pose tracking results using a real image sequence of circular walking. Top row: sample input images; the second row: extracted silhouettes; the third row: recovered poses using the proposed framework; the fourth row: recovered poses directly from the corresponding images using only BME; bottom row: recovered poses obtained using only dynamic prediction.

one because there is a jump between frame 29 to frame 30 due to missing frames caused by a misoperation during the video recording. One hundred particles were used in the tracking. The proposed framework still recovered two reasonable movement trajectories. Some of the results are shown in Figure 15(a). Both admissible tracking trajectories in the joint angle manifold Θ are shown in Figure 15(b). The last set of experimental results included here shows the generalization capability of our proposed tracking framework. A video of a circular walking from [3] was used. Two hundred particles were used in the tracking. The number of samples used in this experiment was more than the other experiments because of the increased movement complexity present in a circular walking. The corresponding results are shown in Figure 16. It can be seen that our proposed framework can track this challenging video fairly well. Our results are much better than those obtained either using only BME or direct sampling from movement dynamics.

9. CONCLUSION AND FUTURE WORK

In this paper, a 3D articulated human motion tracking framework using a single camera is proposed based on manifold learning, nonlinear regression, and particle filter-based tracking. Experimental results show that once properly trained, the proposed framework is able to track patterned motion, for example, walking.

A number of improvements can be made as part of our future work. In the proposed framework, there are two separate low-dimensional manifolds for silhouettes and poses, which requires a number of forward-backward mappings. In our future work, we will try to construct a joint silhouette-pose manifold which will greatly simplify the mapping procedure from the input silhouette to the corresponding latent pose point, in a way similar to [17]. In the proposed framework, we assume that all the entries of the vectorized GMM are independent given the latent variables. This might not be true in reality. We will investigate possible errors carried by this assumption. In our current implementation, we only learn the parameters for a first-order Markov process. To explore higher-order Markov processes, there will be another interesting research problem to work on as well. In our current BME learning, experts for different dimensions of $\mathfrak{C}$ are learned separately using univariate RVM. In our future work, we would like to adopt the multivariate RVM framework proposed in [8] for BME learning and pose inference. We will also compare the final tracking results obtained using univariate RVM and multivariate RVM. Finally, we are working on extending our proposed framework in this paper into a multiple-view setting. Research challenges include optimization of a fusion scheme of input from multiple cameras.

ACKNOWLEDGMENTS

The authors would like thank the anonymous reviewers for their insightful comments and constructive suggestions. They also want to thank Dr. Neil Lawrence for making the GPLVM and related toolboxes freely available online to the community. This paper is based upon work partly supported by U.S. National Science Foundation on CISE-RI no. 0403428 and IGERT no. 0504647. 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 U.S. National Science Foundation (NSF).

REFERENCES

[1] R. Urtasun, D. J. Fleet, and P. Fua, "3D people tracking with Gaussian process dynamical models," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 1, pp. 238–245, New York, NY, USA, June 2006.

[2] A. Blake, J. Deutscher, and I. Reid, "Articulated body motion capture by annealed particle filtering," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '00)*, vol. 2, pp. 126–133, Hilton Head Island, SC, USA, June 2000.

[3] H. Sidenbladh, M. J. Black, and D. J. Fleet, "Stochastic tracking of 3D human figures using 2D image motion," in *Proceedings of the 6th European Conference On Computer Vision (ECCV '00)*, pp. 702–718, Dublin, Ireland, June-July 2000.

[4] L. Kakadiaris and D. Metaxas, "Model-based estimation of 3D human motion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1453–1459, 2000.

[5] A. Agarwal and B. Triggs, "Tracking articulated motion using a mixture of autoregressive models," in *Proceedings of the 8th European Conference on Computer Vision (ECCV '04)*, pp. 54–65, Prague, Czech Republic, May 2004.

[6] V. Pavlovic, J. M. Rehg, and J. MacCormick, "Learning switching linear models of human motion," in *Proceedings of the Annual Conference on Neural Information Processing Systems Conference (NIPS '00)*, Denver, Colo, USA, December 2000.

[7] R. Li, T.-P. Tian, and S. Sclaroff, "Simultaneous learning of nonlinear manifold and dynamical models for high-dimensional time series," in *Proceedings of the 11th IEEE International Conference on Computer Vision (ICCV '07)*, pp. 1–8, Rio de Janeiro, Brazil, October 2007.

[8] A. Thayananthan, R. Navaratnam, B. Stenger, P. H. S. Torr, and R. Cipolla, "Multivariate relevance vector machines for tracking," in *Proceedings of the 9th European Conference on Computer Vision (ECCV '06)*, pp. 124–138, Graz, Austria, May 2006.

[9] G. Mori and J. Malik, "Estimating human body configurations using shape context matching," in *Proceedings of the 7th European Conference on Computer Vision (ECCV '02)*, pp. 150–180, Copenhagen, Denmark, May 2002.

[10] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas, "Discriminative density propagation for 3D human motion estimation," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 1, pp. 390–397, San Diego, Calif, USA, June 2005.

[11] A. Agarwal and B. Triggs, "Recovering 3D human pose from monocular images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 1, pp. 44–58, 2006.

[12] C. Sminchisescu, A. Kanujia, Z. Li, and D. Metaxas, "Conditional visual tracking in kernel space," in *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS '05)*, Vancouver, BC, Canada, December 2005.

[13] K. Grauman, G. Shakhnarovich, and T. Darrell, "Inferring 3D structure with a statistical image-based shape model," in *Proceedings of the 9th IEEE International Conference on Computer Vision (ICCV '03)*, vol. 1, pp. 641–648, Nice, France, October 2003.

[14] N. D. Lawrence, "Probabilistic non-linear principal component analysis with Gaussian process latent variable models," *Journal of Machine Learning Research*, vol. 6, pp. 1783–1816, 2005.

[15] J. M. Wang, D. J. Fleet, and A. Hertzmann, "Gaussian process dynamical models," in *Proceedings of the 20th Annual Conference on Neural Information Processing Systems (NIPS '06)*, Vancouver, BC, Canada, December 2006.

[16] R. Urtasun, D. J. Fleet, A. Hertzmann, and P. Fua, "Priors for people tracking from small training sets," in *Proceedings of the 10th IEEE International Conference on Computer Vision (ICCV '05)*, vol. 1, pp. 403–410, Beijing, China, October 2005.

[17] C. H. Ek, N. D. Laurence, and P. H. S. Torr, "Gaussian process latent variable models for human pose estimation," in *Proceedings of the 4th Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms (MLMI '07)*, Brno, Czech Republic, June 2007.

[18] S. Hou, A. Galata, F. Caillette, N. Thacker, and P. Bromiley, "Real-time body tracking using a gaussian process latent variable model," in *Proceedings of the 11th IEEE International Conference on Computer Vision (ICCV '07)*, Rio de Janeiro, Brazil, October 2007.

[19] C. Curio and M. A. Giese, "Combining view-based and model-based tracking of articulated human movements," in *Proceedings of IEEE Workshop on Motion and Video Computing (MOTION '05)*, vol. 2, pp. 261–268, Breckenridge, Colo, USA, January 2005.

[20] B. Scholkopf and A. Smola, *Learning with Kernels*, MIT Press, Cambridge, Mass, USA, 2002.

[21] M. E. Tipping and C. M. Bishop, "Mixtures of probabilistic principal component analysers," *Neural Computation*, vol. 11, no. 2, pp. 443–482, 1999.

[22] R. Li, M.-H. Yang, S. Sclaroff, and T.-P. Tian, "Monocular tracking of 3D human motion with a coordinated mixture of factor analyzers," in *Proceedings of the 9th European Conference on Computer Vision (ECCV '06)*, pp. 137–150, Graz, Austria, May 2006.

[23] K. Grochow, S. L. Martin, A. Hertzmann, and Z. Popović, "Style-based inverse kinematics," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 522–531, 2004.

[24] T.-P. Tian, R. Li, and S. Sclaroff, "Articulated pose estimation in a learned smooth space of feasible solutions," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 3, p. 50, San Diego, Calif, USA, June 2005.

[25] K. Moon and V. Pavlović, "Impact of dynamics on subspace embedding and tracking of sequences," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 1, pp. 198–205, New York, NY, USA, June 2006.

[26] N. D. Lawrence and A. J. Moore, "Hierarchical Gaussian process latent variable models," in *Proceedings of the 24th International Conference on Machine Learning (ICML '07)*, pp. 481–488, Covallis, Ore, USA, June 2007.

[27] J. M. Wang, D. J. Fleet, and A. Hertzmann, "Multifactor Gaussian process models for style-content separation," in *Proceedings of the 24th International Conference on Machine Learning (ICML '07)*, pp. 975–982, Covallis, Ore, USA, June 2007.

[28] C. Sminchisescu, A. Kanaujia, and D. Metaxas, "Learning joint top-down and bottom-up processes for 3D visual inference," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 2, pp. 1743–1750, New York, NY, USA, June 2006.

[29] C.-S. Lee and A. Elgammal, "Simultaneous inference of view and body pose using torus manifolds," in *Proceedings of the 18th International Conference on Pattern Recognition (ICPR '06)*, vol. 3, pp. 489–494, Hong Kong, August 2006.

[30] E.-J. Ong, A. S. Micilotta, R. Bowden, and A. Hilton, "Viewpoint invariant exemplar-based 3D human tracking,"

Computer Vision and Image Understanding, vol. 104, no. 2-3, pp. 178–189, 2006.

[31] R. Rosales and S. Sclaroff, "Learning body pose via specialized maps," in *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS '01)*, vol. 14, pp. 1263–1270, Vancouver, BC, Canada, December 2001.

[32] A. Agarwal and B. Triggs, "Monocular human motion capture with a mixture of regressors," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 54–65, San Diego, Calif, USA, June 2005.

[33] L. Xu, M. I. Jordan, and G. E. Hinton, "An alternative model for mixtures of experts," in *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS '94)*, pp. 633–640, Denver, Colo, USA, December 1994.

[34] A. Elgammal and C.-S. Lee, "Inferring 3D body pose from silhouettes using activity manifold learning," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, vol. 2, pp. 681–688, Washington, DC, USA, June 2004.

[35] C.-S. Lee and A. Elgammal, "Modeling view and posture manifolds for tracking," in *Proceedings of the 11th IEEE International Conference on Computer Vision (ICCV '07)*, pp. 1–8, Rio de Janeiro, Brazil, October 2007.

[36] M. A. O. Vasilescu and D. Terzopoulos, "Multilinear subspace analysis of image ensembles," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '03)*, vol. 2, pp. 93–99, Madison, Wis, USA, June 2003.

[37] I. Borg and P. Groenen, *Modern Multidimensional Scaling. Theory and Applications*, Springer, New York, NY, USA, 1997.

[38] C. J. C. Burges, "Geometric methods for feature extraction and dimensional reduction," in *Data Mining and Knowledge Discovery Handbook*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2005.

[39] CMU Human Motion Capture DataBase, http://mocap.cs.cmu.edu/.

[40] R. Poppe and M. Poel, "Comparison of silhouette shape descriptors for example-based human pose recovery," in *Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition (FGR '06)*, pp. 541–546, Southampton, UK, April 2006.

[41] F. Guo and G. Qian, "Learning and inference of 3D human poses from Gaussian mixture modeled silhouettes," in *Proceedings of the 18th International Conference on Pattern Recognition (ICPR '06)*, vol. 2, pp. 43–47, Hong Kong, August 2006.

[42] J. Goldberger, S. Gordon, and H. Greenspan, "From image gaussian mixture models to categories," in *Proceedings of the 7th European Conference on Computer Vision (ECCV '02)*, Copenhagen, Denmark, May-June 2002.

[43] N. D. Lawrence, "Gaussian process latent variable models for visualisation of high dimensional data," in *Proceedings of the 15th Annual Conference on Neural Information Processing Systems (NIPS '03)*, Vancouver, BC, Canada, December, 2003.

[44] N. D. Lawrence, "Learning for larger datasets with the gaussian process latent variable model," in *Proceedings of the 11th International Workshop on Artificial Intelligence and Statistics*, San Juan, Puerto Rico, USA, March 2007.

[45] G. Taylor, G. Hinton, and S. Roweis, "Modeling human motion using binary latent variables," in *Proceedings of the 20th Annual Conference on Neural Information Processing Systems (NIPS '06)*, Vancouver, BC, Canada, December 2006.

[46] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, no. 3, pp. 211–244, 2001.