

Tracking People on a Torus

Ahmed Elgammal *Member, IEEE*, and Chan-Su Lee, *Student Member, IEEE*,
 Department of Computer Science, Rutgers University
 Piscataway, NJ, USA

Abstract—We present a framework for monocular 3D kinematic posture tracking and viewpoint estimation of periodic and quasi-periodic human motions from an uncalibrated camera. The approach we introduce here is based on learning both the visual observation manifold and the kinematic manifold of the motion using a joint representation. We show that the visual manifold of the observed shape of a human performing a periodic motion, observed from different viewpoints, is topologically equivalent to a torus manifold. The approach we introduce here is based on supervised learning of both the visual and kinematic manifolds. Instead of learning an embedding of the manifold, we learn the geometric deformation between an ideal manifold (conceptual equivalent topological structure) and a twisted version of the manifold (the data). Experimental results show accurate estimation of the 3D body posture and the viewpoint from a single uncalibrated camera.

Index terms Human Motion Analysis, Tracking, Manifold Learning, Supervised Manifold Learning, Periodic Motion

I. INTRODUCTION

Tracking the human body and recovering the 3D body posture from images is a challenging computer vision problem with many applications such as visual surveillance, human-machine interface, video archival and retrieval, computer animation, autonomous driving, virtual reality, etc. Traditionally, these problems have been addressed through generative model-based approaches. Such approaches typically require a body model and a calibrated camera in order to obtain observation hypotheses from configurations. The recovery of the 3D configuration is formulated as a search problem for the best model configuration that minimizes an error metric given the visual observation. Similarly, 2D view-based body models can be used. Alternatively, discriminative approaches have been suggested, which learn mapping functions from the visual observation to the 3D body configuration. Whether, the approach is generative or discriminative, the main challenge is in the high dimensionality of both the body configuration space and the input space. Another challenge stems from the large variability in human appearance in images.

Despite the high dimensionality of the body configuration space, many human activities lie intrinsically on low-dimensional manifolds. Recently, researchers (e.g. [1], [2], [3], [4], [5], [6]) have been increasingly interested in exploiting the manifold structures of different human motions. For many motions such as locomotion, kicking, golf swing, gestures, etc., the body configuration changes along a one-dimensional manifold. Such body configuration manifolds can be closed

for periodic motion, such as walking or running, or it can be an open trajectory for non periodic motions, such as a golf swing or kicking. It follows that, the observed motion, in terms of body shape or other feature configurations, lies on low-dimensional manifolds as well (we call that the visual manifold or the observation manifold). Exploiting such properties, as well as the relation between the configuration manifolds and the visual manifolds, is essential to constrain the solution space for many problems such as tracking, posture estimation, and activity recognition. However, it is not clear what is the best way to exploit such manifold constraints. Is it through learning the visual observation manifold, the body configuration manifold, or both?

When dealing with the visual manifold of a motion, we need to model the different sources of perceptual variability affecting this manifold, including the view and the body configuration. Modeling the visual manifold for human motion with both the view and body configuration factors jointly is a very challenging task and is useful for tracking, posture estimation, and view estimation. In this paper, we consider this problem for simple classes of motion. We consider learning the visual manifold (in terms of shape) of a person performing a one-dimensional manifold motion, such as walking, running, or golf swing, observed from different viewpoints. Such a setting, although limited, is very useful for many applications such as surveillance where walking and running are the most frequent motions. Similarly, this setting is useful in sport analysis where motions such as golf swings, baseball pitches, tennis serves, etc., are all one-dimensional manifold motion. No previous work has addressed the full modeling of views and configurations for tracking and inference even for such simple motions.

The approach we introduce in this paper is based on learning the visual observation manifold in a supervised manner. Traditional manifold learning approaches are unsupervised, where the goal is to find a low-dimensional embedding of the data. However, for manifolds with known topology, manifold learning can be formulated in a different way. Manifold learning is then the task of learning a mapping from/to a topological structure to/from the data where that topological structure is homeomorphic to the data. In this paper we argue that this supervised setting is suitable to model human motions that lie intrinsically on one-dimensional manifolds, whether closed and periodic (such as walking, jogging, running, etc.) or open (such as golf swing, kicking, tennis serve, etc.). We show that we can model the visual manifold of such motions (in terms of shape) as observed from different viewpoints by mapping that manifold to a toroid topological structure.

There are different ways to see the contribution of this

Chan-Su Lee is currently with the School of Electronic Eng., Communication Eng. & Computer Science, Yeungnam University, Republic of Korea.

This research is partially funded by NSF CAREER award IIS-0546372.

paper:

- A generative model for dynamic shapes: the approach presented here serves as a generative model (a generative function) for the observed shapes of human motion from different viewpoints and for different people's shapes. For example, for gait, the model can generate walking figures at different body configurations, from different viewpoints, and for different people's shapes, where each of these factors is parameterized in a low-dimensional continuous space. The generative model does not use any 3D body model to generate observations; rather, it is based on learning the manifold of the observed shapes.
- Supervised manifold learning: the approach presented in this paper is a supervised approach to learn manifolds with *known topology*. In such cases, we can think of manifold learning as learning the deformation from an idealistic structure to the data. In other words, we learn a generative model of the manifold.
- The approach presented in the paper links together both the visual manifold (input) with the kinematic manifold (output) through a shared representation. By linking these two manifolds, we can track visual observations (input) and infer body posture (output) in a direct way.
- Efficient tracking: The approach presented in the paper provides a practical solution for tracking a certain class of human motions. The resulting state space is low in dimensionality, and therefore, efficient tracking can be achieved.

The organization of the paper is as follows: Section II discusses the relation to previous work in different areas. Section III describes the general solution framework. Sections IV and V describe using a torus manifold for jointly modeling both view and configurations. Section VI describes how the approach fits in the Bayesian tracking framework. Section VII describes several experimental results and evaluations performed to validate the approach and its applicability.

II. RELATED WORK

There are different bodies of research related to this paper. Here we focus on closely related work to put this paper into context.

A. Human Motion Analysis - generative vs. discriminative

In the last two decades, extensive research has focused on understanding human motion from image sequences. We refer the reader to excellent surveys covering this topic, such as [7], [8], [9]. The problems of tracking and recovering body configuration have been traditionally addressed through generative model-based approaches, e.g., [10], [11], [12], [13], [14], [15], [16]. In such approaches, explicit 3D articulated models of the body parts, joint angles, and their kinematics (or dynamics), as well as models for camera geometry and image formation, are used. Recovering body configuration in these approaches involves searching high-dimensional spaces (body configuration and geometric transformation), which can be formulated deterministically as a nonlinear optimization problem, e.g. [17], [13], or probabilistically as a maximum

likelihood problem, e.g. [16]. Such approaches achieve significant success when the search problem is constrained as in the tracking context. However, initialization remains the most challenging problem. The dimensionality of the initialization problem increases as models for variations between individuals in physical body style, variations of action style, and/or models for clothing are incorporated. Partial recovery of body configuration can also be achieved through view-based representations (models), e.g. [18], [19], [20], [21], [22], [23], [24]. In such cases, constancy of the local appearance of individual body parts is exploited. The main limitation with such approaches is that they deal with limited view configurations, i.e., a single view or a small set of discrete views.

Alternatively, discriminative approaches have been proposed. In such approaches, recovering the body posture can be achieved directly from the visual input by posing the problem as a supervised learning problem through searching a pre-labeled database of body postures [25], [26], [27] or through learning regression models from input to output [28], [26], [29], [30]. All these approaches pose the problem as a machine learning problem, where the objective is to learn an input-output mapping from input-output pairs of training data. Such approaches have great potential for solving the initialization problem for model-based vision. However, these approaches are challenged by the existence of a wide range of variability in the input domain. Another challenge is the high dimensionality of the input and output spaces of the mapping, which makes such a mapping hard to generalize.

The approach we introduce in this paper is generative. However, unlike traditional generative approaches, there is no explicit 3D or 2D body model. The observations are generated through a regression model based on an embedded representation of the motion manifold. Similar to discriminative approaches, an input-output mapping is learned. However, unlike discriminative approaches, the mapping in our case is from a shared manifold representation to both the input and the output. Therefore, instead of being "model-based", the representation we use is "manifold-based". Instead of an explicit 3D model, a continuous view-based representation is used.

B. Manifold-based Human Motion Analysis

Despite the high dimensionality of both the human body configuration space and the visual input space, many human activities lie on low-dimensional manifolds. In the last few years, there has been increasing interest in exploiting this fact by using intermediate activity-based manifold representations [31], [32], [1], [33], [34], [3], [4], [6], [35]. In our earlier work [1], the visual manifolds of human silhouette deformations, due to motion, have been learned explicitly and used for recovering 3D body configuration from silhouettes in a closed-form. In that work, knowing the motion provided a strong prior to constrain the mapping from the shape space to the 3D body configuration space. However, the approach proposed in [1] is a view-based approach; a manifold was learned for each of the discrete views. In contrast, in this paper the manifold of both the configuration and view is learned in a

continuous way. In [33], manifold representations learned from the body configuration space were used to provide constraints for tracking. In both [1] and [33] learning an embedded manifold representation was decoupled from learning the dynamics and from learning a regression function between the embedding space and the input space. In [35], coupled learning of the representation and dynamics was achieved through introducing Gaussian Process Dynamic Model [36] (GPDM), in which a nonlinear embedded representation and a nonlinear observation model were fitted through an optimization process. GPDM is a very flexible model since both the state dynamics and the observation model are nonlinear. The problem of simultaneously estimating a latent state representation coupled with a nonlinear dynamic model was earlier addressed in [37]. Similarly, in [6], models that coupled learning dynamics with embedding were introduced.

Manifold-based representations of the motion can be learned from kinematic data, or from visual data, e.g., silhouettes. The former is suitable for generative model-based approaches and provides better dynamic-modeling for tracking, e.g., [33], [3]. Learning motion manifolds from visual data, as in [1], [38], [4], provides useful representations for recovery and tracking of body configurations from visual input without the need for explicit body models. The approach we introduce in this paper learns a shared representation for both the visual manifold and the kinematic manifold. Learning a representation of the visual motion manifold can be used in a generative manner as in [1] or as a way to constrain the solution space for discriminative approaches as in [2]. The representation we introduce in this paper can be used as both: a generative model for tracking or as a discriminative model for recovering the body configuration from a visual input.

C. Modeling Visual Manifolds

The approach we introduce in this paper focuses on learning the visual manifold of deformable shapes where the deformation is due to both the underlying motion and the view variability. Learning the view and illumination manifolds of rigid objects has been studied in [39]. In that work, PCA was used to achieve a linear embedding for the joint view and illumination manifold for object recognition. Learning motion manifolds can be achieved through a linear subspace approximation, as in [40], [41]. Alternatively, exemplar-based approaches, such as [42], [43], implicitly model nonlinear manifolds of observations through points (exemplars) along the manifold. Such exemplars were represented in the visual input space. Hidden Markov Models (HMM) were used to provide a probabilistic piecewise linear approximation, which can be used to learn nonlinear manifolds as in [31]. Recently, unsupervised nonlinear manifold learning techniques, such as LLE [44], Isomap [45], etc., have been popular for learning low-dimensional representations of visual manifolds. Unlike most previous work on learning the visual manifolds that focused on unsupervised embedding of the visual manifolds, the approach we introduce here uses the knowledge about the topology of the manifold to achieve an embedded representation of the visual data.

Also, related to this paper, is research on multilinear models, which extends subspace analysis to decompose multiple orthogonal factors using bilinear models and multilinear tensor analysis [46], [47]. Tenenbaum and Freeman [46] formulated the separation of style and content using a bilinear model framework [48]. In [47], multilinear tensor analysis was used to decompose face images into orthogonal factors controlling the appearance of the face including geometry (people), expressions, head pose, and illumination. N-mode SVD [49] is used to fit multilinear models. Multilinear tensor analysis was also used in [50] to factorize human motion styles. The applications of bilinear and multilinear models in [46], [47], [50] to decompose variations into orthogonal factors were performed in the original observation space. In contrast, in our earlier work [51], bilinear and multilinear analysis were used in the space of the mapping functions between an average manifold representation and the observations to decompose the variability factors in such functions. In this paper, we used a similar approach to decompose shape “style” variabilities in the space of the mapping functions between the embedded manifold representation and visual observations.

III. FRAMEWORK

A. Graphical Model

This section aims to discuss the characteristics of useful representations for tracking the visual observation of articulated motion. In particular, we consider visual observations deforming through a one-dimensional manifold motion, such as gait, golf swings, etc. We start by defining the basic manifold terms, which we are going to use throughout the paper, and then we will explain their relationships.

Let us consider an articulated object motion observed from a camera (stationary or moving). Such a motion can be represented as a kinematic sequence $Z_{1:T} = z_1, \dots, z_T$ and observed as an observation sequence $Y_{1:T} = y_1, \dots, y_T$. In this paper, without loss of generality, by observations we mainly mean shape contours. A similar argument applies to any vector representation of the observations. With an accurate 3D body model, camera calibration, and geometric transformation information, we can explain $Y_{1:T}$ as a projection of an articulated model. However, in this paper, we are interested in a different interpretation of the relation between the observations and the kinematics that does not involve any body model.

The dynamic sequence $Z_{1:T}$ lies on a manifold. Let us denote this by *the kinematic manifold*. The kinematic manifold is the manifold of body configuration changes in the kinematic space. In addition, the observations lie on a manifold, *the visual manifold*. Obviously, the observation is a function of not only the body configuration but also the viewpoint, person’s appearance, etc. What is the relation between the kinematic manifold and the visual manifold? We can think of a graphical model connecting the two manifolds through two latent variables: a body configuration variable, b_t , and a viewpoint variable, v_t . The body configuration variable is shared between both the kinematic manifold and the visual manifold. The viewpoint variable represents the camera location relative to

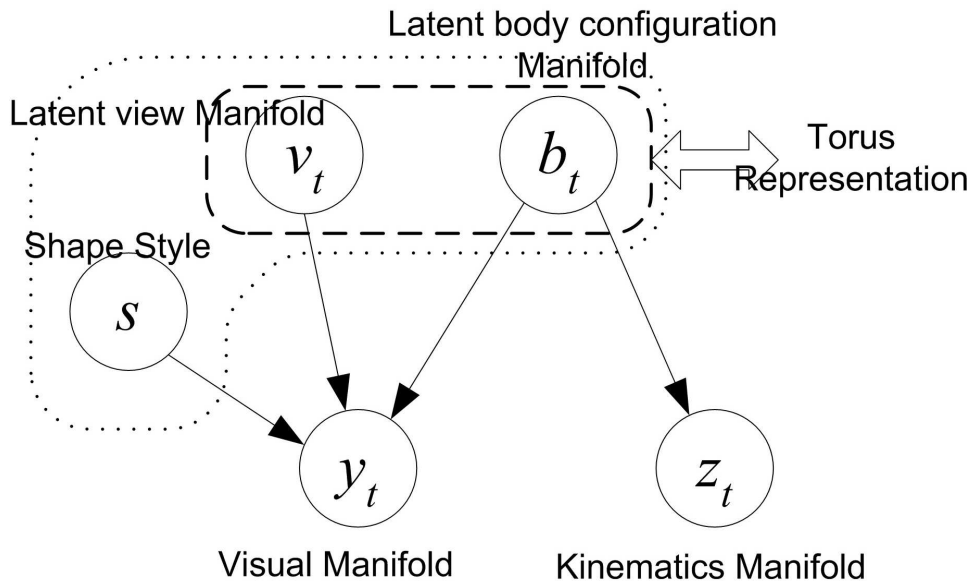


Fig. 1. A graphical model relating the different manifold structures

a human-centered coordinate system. The body configuration variable moves on a manifold in the latent space; we denote it by *the latent body configuration manifold*. Similarly, the viewpoint variable lies on a manifold, denoted by *the latent view manifold*.

Another variable affecting the observation is the shape variability among different subjects, i.e., the human shape space, or the shape style as was denoted in [51]. We denote this variable by s , which is a time invariant variable. Given this graphical model, the tracking problem is an inference problem where, given the observation y_t at time t , we want to recover the state, which consists of three components: body configuration b_t , viewpoint v_t , and shape style s . The Bayesian tracking framework can be used to recursively update the state posterior where the latent space manifold representation should provide a constraint on the state's dynamics.

Given the definition of the various manifolds, we can summarize the goals of our representation as follows:

1) We want to relate the kinematic manifold with the visual manifold in order to be able to infer the configuration of the joint angles from input.

2) We want to model the visual manifold with all its variabilities, due to the motion, the viewpoint, and shape style. In particular, we want to be able to deal with both body configuration and viewpoints as continuous variables. This facilitates tracking subjects from varying viewpoints due to camera motion or changing subject view w.r.t. the camera.

3) We want the tracking state space to be low in dimensionality and continuous. The low dimensionality of the tracking state space is preferred for efficient particle filter tracking. Moreover, despite the nonlinearity in dynamics in both the kinematics and the observations, it is preferred that the latent body configuration variable would exhibit simple dynamics, i.e., linear dynamics or even constant speed dynamics.

Given these goals stated, there are certain questions we need to answer: What manifolds do we need to learn, the

visual manifold, the kinematic manifold, or both? What are the tools we need to learn such manifolds? Here we argue in favor of a representation that is conceptual, central, and generative. Conceptual means that *the representation is not learned from the data, but the data is fitted to a manifold model in a supervised way*. Central means *the representation jointly models both the kinematics and observations as well as jointly models different people's manifolds*. Generative means that *the same representation is used to generate both the kinematics and observations*.

B. Supervised Manifold Learning

Unsupervised nonlinear dimensionality reduction (NLDR) techniques, such as LLE [44], Isomap [45], GPLVM [52], etc., have been popular recently in learning low-dimensional representations of both visual and kinematic data, as detailed in Section II. Such techniques aim to achieve low-dimensional embedded representations of high-dimensional data for visualization and other data analysis purposes. The main assumptions in such approaches are: 1) the data lie on a manifold; 2) enough dense data are available in order to be able to recover the manifold structure from the local structure of the data.

Perceptually, we might think the visual data lie on a low-dimensional manifold with certain topology. However, in reality, visual data might not exhibit such manifold structure. In other words, the manifold structure might not be recoverable from the data using unsupervised nonlinear embedding approaches. There are many reasons for this problem that we will discuss shortly, but we first show an example to demonstrate the point. Let us consider the visual manifold of a simple motion, such as gait, observed from a view circle around the person. Such data can be seen in Fig. 2-a. Here, for illustration, we have no other variations in the data, e.g., no translation, scaling, noise, or shape styles. The observed

shapes represent a product space of two one-dimensional manifolds representing body configuration and view. Therefore, the data is expected to lie on a two-dimensional manifold representing the body configuration and view variations. As will be shown in Section IV, this manifold is expected to resemble a distorted torus.

Figs. 2-d and 2-e show the resulting embedding when LLE and Isomap are used to embed such visual data¹. We argue that the resulting embedding is not necessarily useful, nor desirable as a latent embedded representation for tracking. The resulting embedding using any NLDR approach, although it preserves the local manifold structure, depends on many other aspects including: 1) the way the visual data is represented (shape representation in this case); 2) the distance metric used between visual instances; 3) the choice of the objective function; 4) the existence noise and other sources of variations. As a result, the embedding obtained is stretched at some parts and squashed at others to fit the actual data local manifold structure. These stretches reflect the factors mentioned above and do not reflect the conceptual distance between instances. This results in a representation that is not continuous. Continuity is very important in any latent state space, since sampling on the surface of the manifold is needed for tracking. Notice that the embeddings shown in Figs. 2-d and 2-e are 3D embeddings and sampling on the surface of the manifold will be needed while tracking.

For manifolds with known topology, instead of unsupervised embedding, a conceptual supervised topology-preserving representation would be advantageous for tracking. Before we further justify the use of such a conceptual representation, we will first explain the framework. If the manifold topology is known, manifold learning can be formulated in a different way. Manifold learning is then the task of learning the nonlinear deformation of the data manifold from an *ideal manifold* model. For example, for a periodic motion, such as gait, the data lie on a closed trajectory in the space, and hence, the data manifold is homeomorphic to a unit circle. We can think of the data as a deformed circle and the task of learning would be to learn the nonlinear function that deforms a unit circle to the data. In other words, how can the data be generated, knowing an equivalent “idealistic” topological structure? In fact, this view can be even extended to the case where the data manifold does not share the exact topology with the ideal manifold. For example, the gait manifold can intersect itself in the visual space but, still, we can learn the deformation from a unit circle to the data. We will show in the next section that a torus manifold is a good manifold model for visual data similar to the one we consider in this section (periodic motion observed from a view circle), as in Fig. 2-a. Enforcing a topological structure can also be achieved by adding a constraint to the objective function in unsupervised NLDR as was done in [53].

There are several reasons why such a supervised approach is advantageous and more suitable when the topology is known.

1- The ideal manifold model would be advantageous as a

¹To obtain these results we represent shapes as implicit level set functions to impose smoothness between shape instances in order to recover the manifold. The shape representation is described in section VI. Other shape representation might be used with more or less qualitatively similar results.

state representation over a data-driven representation (such as the one obtained through unsupervised learning as shown in Figs. 2-d, 2-e) because of its low dimensionality, continuity, and unambiguity. The ideal manifold would also be more uniform and would not exhibit stretches or squashes, yet it would preserve the topological structure of the data manifold since they are topologically equivalent. The nonlinear deformation function will take care of stretching and/or squashing the ideal manifold to fit the data.

2- Using a conceptual representation provides a common representation for different people’s manifolds. The visual manifolds for different people are expected to twist differently depending on the shape of the person performing the motion. Therefore, a representation learned from a certain person will not be useful to generalize to other people. On the other hand, learning such embedded representations from multiple people is not easily achievable since nonlinear dimensionality reduction techniques, as mentioned above, deal with one manifold and would have trouble dealing with inter-person and intra-person manifolds. Many researchers are actively pursuing style-content separation approaches that can learn embedded representations of the posture from multiple data sets invariant to the identity of the person [54], [51]. Specifying the topology of the posture manifold, as we do here, achieves a conceptual content representation, invariant to the identity. Modeling the style variability is achieved by factorizing the deformation functions as will be shown in Section V-C.

3- The ideal manifold is a shared latent (common) representation between both the visual (input) manifold and the kinematic (output) manifold since it is topologically equivalent to both of them. Therefore, the ideal manifold would serve both in tracking the visual observation and in recovering the configuration of the body joints. Learning a shared manifold representation between an input and an output manifold can also be achieved in unsupervised way as in [55], [56]. Here, the shared representation is specified by enforcing the known topology.

4- Instead of learning nonlinear dynamics from the actual data, we can impose simpler dynamic models on the ideal manifold. The nonlinear deformation function would transform such simple dynamics to the nonlinear dynamics of the data. For example, for a periodic motion, moving along a unit circle with equal steps would correspond to nonlinear steps in the data space. This yields a simple constant-speed linear dynamic system in the latent space and, therefore, simplifies the tracking.

5- This model is generative, and therefore, can be directly used for synthesizing data. This fits directly into the Bayesian tracking framework, since in such a setting we need to generate observation hypotheses given a state representation to evaluate the state posterior.

IV. VIEW AND CONFIGURATION JOINT REPRESENTATION

A. Why a Torus Manifold?

In this section, we show that a torus manifold is topologically equivalent to the visual manifold resulting from a certain class of motion observed from a view circle. We will start with

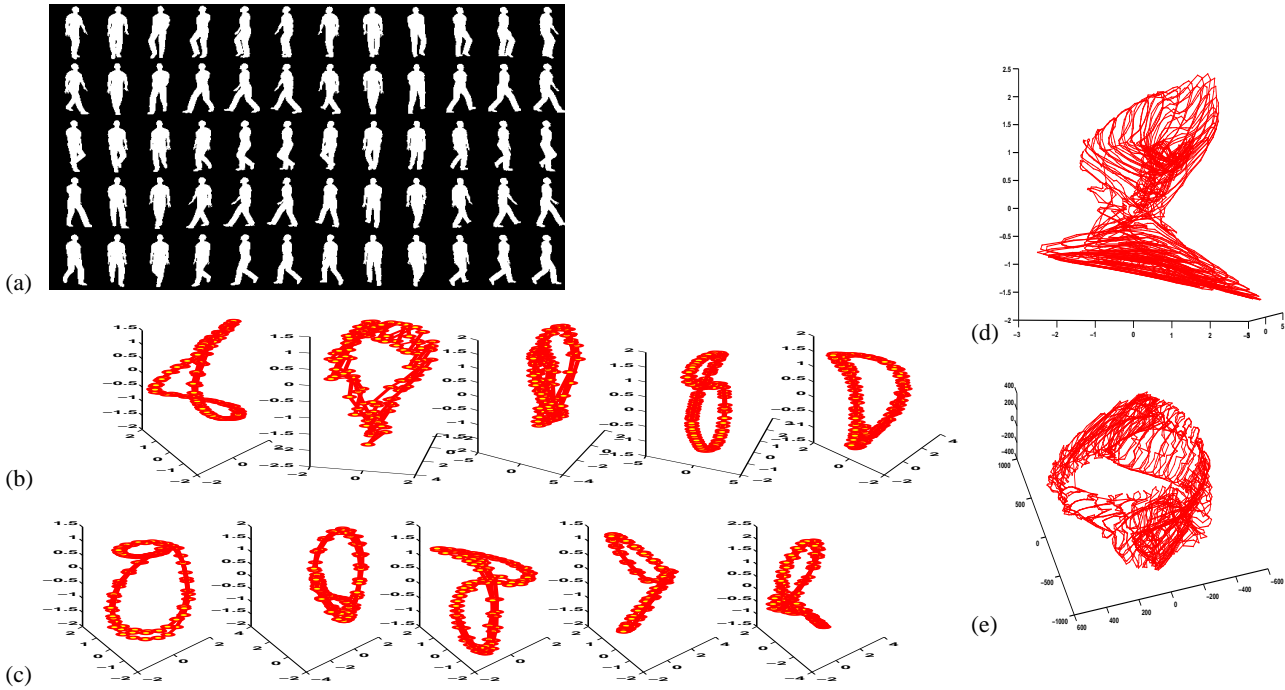


Fig. 2. Data-driven view and body configuration manifolds:(a) Examples of sample data with different views and configurations. Rows: body posture at $0, \frac{1}{5}T, \frac{2}{5}T, \frac{3}{5}T, \frac{4}{5}T$. Cols:view $0, 30, 60, \dots, 330$. (b) The intrinsic configuration manifolds when the view angle is $0, 60, 120, 180$, and 240 . (c) The view manifolds for five different fixed body postures. (d) (e) The combined view and body configuration manifold embedding by LLE and Isomap.

a simple periodic motion, such as a simple aerobic exercise or gait, observed from a view circle around the person. Later, we show how to deal with other motions and extend the approach to the whole view sphere. For illustration, we will use the gait motion. The gait motion, observed from the same viewpoint, is a one-dimensional closed manifold, which is topologically equivalent to a unit circle. This was earlier shown in [1] and can be noticed from embedding view-based observations as shown in the examples in Fig. 2-b. We can think of the gait manifold as a twisted or deformed circle in the visual input space. Similarly, the view manifold of a certain body posture is topologically equivalent to a unit circle, as can be seen in Fig. 2-c. For a joint configuration and view manifold, where the view varies along a view circle, this is a product space and the resulting manifold is the product of two circles, i.e., either a torus or a sphere. Since the configuration manifold of a given view and the view manifold of a give posture intersect at exactly one point, this is equivalent to a torus manifold. Therefore, the data in Fig. 2-a lie on a deformed torus in the visual input space. This is also observable from the embeddings shown in Figs. 2-d, 2e. The supervised learning task in this case reduces to learning the deformation from a torus to the data.

On the other hand, the kinematic manifold, which is invariant to the viewpoint, is also a deformed circle in the kinematic space. Starting from a torus, the kinematic manifold can be achieved through collapsing the torus along one of its axes to form a circle and then deforming that circle. Therefore, a torus manifold acts as an “ideal” manifold to represent both the latent body configuration and view variables, b_t, v_t . On one hand, the torus can deform to form the visual manifold, where the observations lie. On the other hand, the torus can

deform to form the kinematic manifold, where the kinematic data lie.

B. Torus Manifold Geometry

A torus manifold, is a two-dimensional manifold embedded in a three-dimensional space with a single hole. The torus manifold can be constructed from a rectangle with two orthogonal coordinates with the range $[0, 1] \times [0, 1]$ by gluing both pairs of opposite edges together with no twists [57]. Therefore, the torus surface can be parameterized by two variables $\mu, \nu \in [0, 1]$. Let the radius from the center of the hole to the center of the torus tube be R_c , and the radius of the tube be R_a , then an azimuthally symmetric torus about the z axis can be described

$$\left(R_c - \sqrt{x^2 + y^2}\right)^2 + z^2 = R_a^2,$$

and the parametric equations are

$$\begin{aligned} x &= (R_c + R_a \cos 2\pi\nu) \cos 2\pi\mu, \\ y &= (R_c + R_a \cos 2\pi\nu) \sin 2\pi\mu, \\ z &= R_a \sin 2\pi\nu. \end{aligned} \quad (1)$$

Fig. 3-a shows a torus manifold when $R_c = 2, R_a = 1$.

As justified above, the torus is an ideal representation for a periodic motion observed from a view circle and, therefore, it can be used as a conceptual representation for both the viewpoint (along one viewing circle) and the body configuration jointly. The view and body configuration manifolds can be parameterized in the rectangle coordinate system with the two orthogonal axes of the torus. Any point on the torus has two

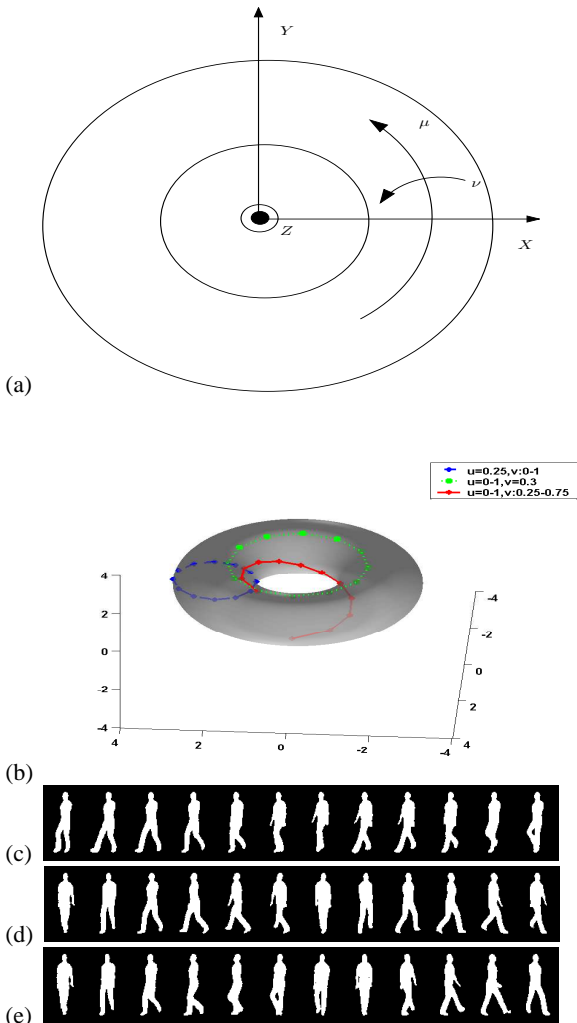


Fig. 3. Torus representation for continuous view-variant dynamic human motion:(a) The coordinate system on the torus: 3D Cartesian coordinates and the μ and ν axes. The Z axis is perpendicular to the paper plane (b) Three different trajectories on the torus manifold according to the view and configuration changes shown in parts c,d, and e. (c) Syntheses of body posture variations with a fixed view ($\mu = 0.25, \nu : 0 \rightarrow 1$). (d) Syntheses of view variations with a fixed body configuration. $\mu : 0 \rightarrow 1, \nu = 0.3$. (e) Syntheses of both view and body configuration variation: $\mu : 0 \rightarrow 1, \nu : 0.25 \rightarrow 0.75$

orthogonal circles²: one is in the plane of the torus, which we use to model the view variable and is parameterized with μ . The other circle is perpendicular to the torus plane, which we use to represent the body configuration and is parameterized by ν .

C. Embedding points on the torus

Given a sequence of kinematic data, $Z = \{z_1, \dots, z_N\}$, representing a motion, we use graphics software to render body silhouettes from different viewpoints along a given viewing circle. Let us denote the silhouette sequence from a given viewpoint, $v \in [0, 2\pi)$, by $Y^v = \{y_1^v, \dots, y_N^v\}$. Here, we are dealing with the data as a discrete set of points rather than a temporal sequence. By embedding points on the torus, we mean to find the corresponding torus coordinates

$(\mu_{(i,v)}, \nu_{(i,v)})$ for an input instance (z_i, y_i^v) . The torus coordinates μ and ν are used to indicate the view and the body configuration respectively. We propose two ways to achieve such an embedding.

1) *Constant Speed Dynamics*: For tracking, we not only know the topology of the manifold but also may know the desired dynamics in the state space. It is desired to embed the data on a torus in a conceptual way that does not necessarily reflect their Euclidean distance neither in the kinematic space nor in the visual input space. Instead, the objective is to embed them on the torus in a way to simplify the tracking. For example, for periodic motion, such as walking and running, despite the nonlinearity of the dynamics in both the kinematic and the visual input manifolds, we need the latent state variable to reflect a constant speed on the latent manifold. The nonlinear mapping, which will be described in Section V, will transform this linear dynamics to nonlinear dynamics. This can be achieved by embedding the points on equidistant points along the configuration axis of the torus. Therefore, given a sequence $(z_i, y_i^v), i = 1, \dots, N, v \in [0, 2\pi)$ for one cycle of the motion (starting and ending with the same posture), the corresponding torus coordinates are set to

$$\begin{aligned}\mu_{(i,v)} &= v/2\pi, \\ \nu_{(i,v)} &= i/N.\end{aligned}$$

2) *Geodesics-based Embedding*: The torus representation is not only suitable for periodic motion but also suitable for quasi-periodic and non-periodic motions. By a quasi-periodic motion we mean a motion that starts and ends with a similar body posture (e.g. a jump motion, or a facial expression starting and ending with a neutral expression) and, therefore, the motion lies on a closed (or almost closed) trajectories. Obviously, for such motions, similar to the periodic case, the torus is a good representation to model the visual manifold under viewpoint and configuration variability. On the other hand, the torus representation is also useful for non-periodic motions where the body configuration manifold is a one-dimensional open trajectory (e.g., a golf swing), and the motion starts and ends at different body posture. In such cases, any linear embedding is also homeomorphic to the configuration manifold. For example, a cylinder can be used to represent the visual manifold of such a motion. A torus can also be used in this case, where a part of the torus configuration axis can be used to represent the open trajectory. In this section, we will describe how to use the torus for representing both quasi-periodic and non-periodic open-trajectory motions.

The approach described here is suitable in general for motions where the data might exhibit different acceleration along the course of the motion, and therefore, constant speed dynamics might not be achievable in the latent space. It is desired to embed the data on the torus in a way that preserves their kinematic manifold structure. This can be achieved through embedding the points such that the geodesic distances on the torus are proportional to the geodesic distances on the kinematic manifold.

To achieve this, we first embed the kinematic manifold using an unsupervised nonlinear dimensionality reduction approach

²Actually, two additional circles called Villareau circles can be drawn at the any given point, but we focus on the two simple perpendicular ones.

(such as LLE or Isomap). This embedding is used for: 1) measuring the gap between the beginning and end postures of the motion in order to map the manifold to a proportional part of the torus; 2) measuring the geodesic distances along the kinematic manifold. The points are embedded on the torus in such a way that only a part of the torus ν axis is used proportional to the embedded manifold length. Let $x_i, i = 0, \dots, N$ be the embedding coordinate of the kinematic sequence $z_i, i = 0, \dots, N$. The coordinate of point z_i on the torus ν -axis is denoted by ν_{z_i} and is set to be $\nu_{z_i} = S_i/S$, where S_i is the geodesic distance of point x_i to the starting point, x_o , i.e., $S_i = \sum_{j=1}^i \|x_j - x_{j-1}\|$ and S is defined to be $S = S_N + \|x_N - x_o\|$. The gap between the first and last embedding points on the torus will be $\frac{\|x_N - x_o\|}{S}$.

Fig. 4 and Fig. 5 show two examples for the geodesic-based embedding for two non-periodic motions. In the first example, a jump motion from 36 views is embedded on a torus surface. Example images from that motion are shown in Fig. 13 and Fig. 14. In Fig. 4-a, Isomap is used to embed the kinematics of the motion. Notice the dense embedding at the beginning and the end of the motion. This non-uniformity in the embedding of the kinematics is reflected in the torus coordinates of the points (notice the dense embedding at the beginning and end of the motion on the torus). This jump motion is an example of a quasi-periodic motion where the start and the end postures are close. As a result, the gap on the torus is very small. In contrast, a golf swing motion, as shown in Fig. 5, is an open trajectory motion and, therefore, the gap on the torus is larger. Fig. 5-b shows the embedding of the motion kinematics and Fig. 5-c shows the area of the torus surface that is used for the embedding. Fig. 5-c also shows three trajectories on the torus corresponding to three different views. The corresponding rendered silhouettes are shown in Fig. 5-a.

D. Generalization to the Full View Sphere

Generalization to the full view sphere around the person is straightforward. The visual manifold of a periodic motion observed from the full view sphere around the person is a deformed order-3 torus or $S^1 \times S^1 \times S^1$ structure. This can be justified by discretizing the view sphere into view circles at different heights. Each view circle will yield a deformed torus manifold. The combination of all these tori is an order-3 torus or a family of tori. Each point in this $S^1 \times S^1 \times S^1$ manifold represents a certain body configuration observed from a certain azimuth angle and a certain elevation angle.

In this case, the joint configuration and view manifold can be mapped to a family of tori, which is a subset of the product space of three circles $S^1 \times S^1 \times S^1$, one for the body configuration, one for the horizontal view circle and one for the vertical view circle. In practice, only a small range of any vertical view circle is considered. Therefore, this can be modeled as a set of rectangles each representing a torus manifold for a given view circle, i.e., this can be parameterized by three parameters μ, ν, ξ for the body configuration, the azimuth view angle, and the elevation view angle respectively.

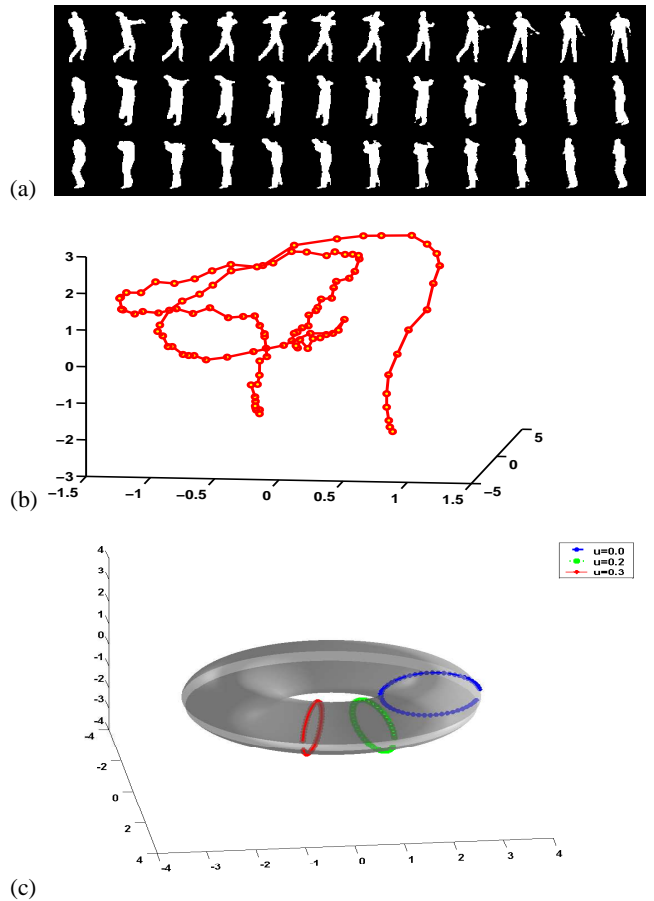


Fig. 5. Torus manifold with a gap: (a) Example sequence of a golf swing from three different views $\mu = 0, 0.2, 0.3$. (b) Embedding of golf swing motion capture data. (c) Visualization of a torus manifold with the trajectories of the three different views shown in (a).

V. LEARNING MANIFOLD DEFORMATION

A. Learning Manifold Deformation

Learning a mapping from a topological structure to the data, where that topological structure is homeomorphic to the data, can be achieved through learning a regularized nonlinear warping function. Let \mathcal{T} denote a topological structure and let \mathcal{M} denote the data manifold, where \mathcal{T} and \mathcal{M} share the same topology. Given a set of point $x_i \in \mathbb{R}^d, i = 1, \dots, K$ on \mathcal{T} and their corresponding points $y_i \in \mathbb{R}^D, i = 1, \dots, K$ on a manifold \mathcal{M} , we can learn a nonlinear mapping function $g : \mathbb{R}^d \rightarrow \mathbb{R}^D$ from \mathcal{T} to \mathcal{M} . According to the representer theorem [58], such a function admits a representation in the form of

$$y = \sum_{j=1}^N b_j k(x, z_j),$$

where $\{z_j\}, j = 1, \dots, N$, is a finite set of points on \mathcal{T} , not necessarily corresponding to the data points and $k(\cdot, \cdot)$ is a kernel function that combines the coefficient vectors, $b_j \in \mathbb{R}^D$. If radial basis kernels are used, then this can be interpreted as a form of radial basis function interpolation. Such a mapping can be written in the form

$$y = \mathbf{B}\psi(x), \quad (2)$$

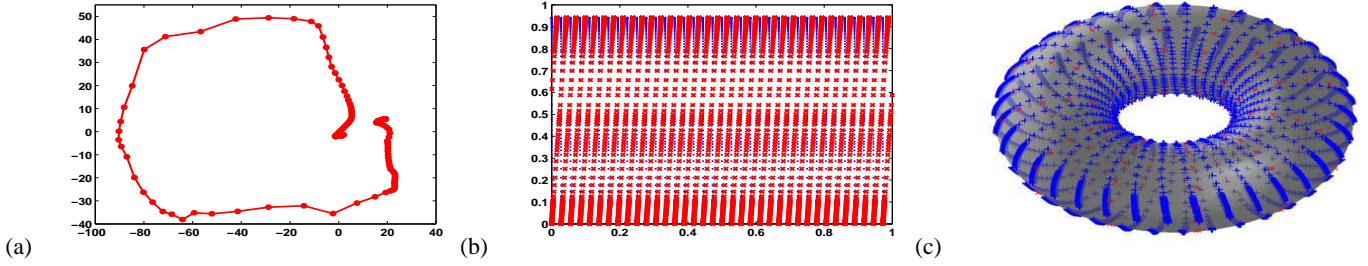


Fig. 4. Geodesics-based embedding on a torus for a jump motion. (a) Isomap embedding for a jump motion’s kinematics from motion capture data. (b) Training data from 36 views embedded on the ν (view - horizontal), μ (configuration - vertical) parameter space. (c) Embedded points on the torus for the training data set and their empirical kernel bases (red x’s).

where $\mathbf{B} = [b_1, \dots, b_N]$ is a $D \times N$ coefficient matrix and

$$\psi(x) = [k(x, z_1), \dots, k(x, z_N)]^T \quad (3)$$

is a vector of the kernel functions given a point x . $\psi(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}^N$ represents an empirical kernel map [59] to a feature space defined by the $\{z_j\}$ points on \mathcal{T} . The coefficient matrix \mathbf{B} can be obtained by solving a linear system in the form

$$[y_1 \dots y_K] = \mathbf{B}[\psi(x_1), \dots, \psi(x_K)].$$

To avoid overfitting to the training data, regularization is needed. Regularizing the RBF mapping in Eq. 2 is a standard procedure and can be achieved by adding a regularization term to the diagonal of the matrix $[\psi(x_1), \dots, \psi(x_K)]$ [60].

B. Deforming the Torus:

The input is a kinematic sequence and its corresponding visual data from multiple views. Given the data embedded on the torus, as described earlier, we can learn the deformation between the torus and both the visual manifold and the kinematic manifold. This can be achieved through learning two regularized nonlinear mapping functions in the form of Eq. 2, as will be described next.

Let us denote the visual observations by $\{y^{vb}\}$ and their corresponding kinematic data by $\{z^b\}$, where v and b are the view and posture indices. After embedding this data on the torus, we obtain the embedding coordinates $\{\mu_v, \nu_b\}$ on the torus coordinate system, where μ_v is the view representation on the μ axis, and ν_b is the body configuration representation on the ν axis.

1) *Torus to Visual Manifold:* Deforming the torus to the visual manifold can be achieved through learning a nonlinear mapping function in the form of Eq. 2. Such a function maps from the Euclidean space where the torus is embedded to the visual data. Notice that we need to map from the Euclidean space where the torus lives, and not from the torus coordinate system (μ, ν) since this coordinate system is not continuous at the boundary. The torus representation in the Euclidean space provides continuity that is needed in the representation. Eq. 1 maps between the torus coordinate system and the Euclidean space where the torus is embedded in (\mathbb{R}^3) . Any point on the torus surface can be represented by a function $g : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ of the two variables μ and ν such that $\mathbf{x} = [x \ y \ z]^T = g(\mu, \nu)$.

Given the embedding coordinates on the torus surface, $\mathbf{x}^{vb} = g(\mu_v, \nu_b)$ and their corresponding visual data (silhouettes), $y^{vb} \in \mathbb{R}^D$, for discrete posture b and view v , we can

fit the mapping function $h : \mathbb{R}^3 \rightarrow \mathbb{R}^D$, which maps from the torus to the shape space in the form

$$y = h(\mathbf{x}) = \mathbf{D} \cdot \psi(g(\mu, \nu)), \quad (4)$$

satisfying

$$y^{vb} = h(\mathbf{x}^{vb}).$$

In this case, we need a set of N basis functions covering the torus surface, which are set uniformly across the surface. Using this model, for any view v and body configuration b , we can generate the corresponding observation y^{vb} . Figs. 3-c, 3-d, and 3-e show examples of the generation of new sequences using the torus manifold embedding and the nonlinear mapping function. Corresponding manifold trajectories are shown in Fig. 3-b.

2) *Torus to Kinematic Manifold:* Similarly, deforming the torus to the kinematic manifold can be achieved through learning a nonlinear mapping from the torus configuration-axis to the kinematic manifold. Given the embedded points on the torus, (μ_v, ν_b) and their corresponding kinematic points $z^b \in \mathbb{R}^{D_k}$, where D_k is the dimensionality of the kinematic space (number of joint angles \times 3), we can fit a mapping function $f : \mathbb{R} \rightarrow \mathbb{R}^{D_k}$ in the form

$$z = f(\nu) = \mathbf{B} \cdot \psi(\nu), \quad (5)$$

stratifying

$$z^b = f(\nu_b).$$

Given this mapping, any point (μ, ν) on the torus can be directly mapped to a 3D kinematic configuration (positions of the joints).

C. Learning Different People Manifolds from Partial Views

Different people have variations in their observed shapes due to different body types and different clothing styles. Any tracker needs to be able to adapt to the specific tracked person contour shape. In this section, we show how to learn the torus model from different people’s training data. The torus model, as presented so far, needs data from all viewpoints (dense sampling of the view manifold) in order to model the visual manifolds. Such data cannot be easily obtained in practice. We can get around this by using graphics software to synthesize such data given the kinematics. However, such synthetic data will not model the variability in actual contours, i.e., the shape space of different people. Therefore, there are two goals: 1) to

learn different people posture-view manifolds from only sparse partial views' data to achieve successful tracking, 2) to adapt the model to new people as we track them. In this section we propose solutions for these two problems.

In our earlier work [51], we showed that shape "style" variations can be decomposed in the space of the coefficients of the nonlinear mapping functions from an embedded manifold representation to the observation space. In this paper we use a similar approach to learn style-dependent mappings, in the form of Eq. 4, from the torus to each person's data. The torus is a unified manifold representation invariant to people's appearance. This results in a general generative model for contours for different postures, different views, and for different shape styles in a tensor product form

$$y_{vb}^s = \mathcal{A} \times_1 a^s \times_2 \psi(\mu_v, \nu_b). \quad (6)$$

Such a model combines a body configuration parameter ν_b , a view parameter μ_v , and a shape style vector a^s to generate an observation y_{vb}^s of style s and view v and posture b . The tensor \mathcal{A} is a third order tensor with dimensions $D \times S \times N$, which controls the correlation between the configuration, view and style, where S is the dimensionality of the shape space. The product notation \times_n is the tensor multiplication as defined in [49].

We now show how to fit the model in Eq. 6 from sparse view-based data for different people performing the same motion. Given different persons' sequences from different sparse viewpoints, each sequence can be embedded on the torus as described in Sec. IV. Let $Y_{v_k}^s$ be sequences of visual data for person s , where $s = 1, \dots, K$, from viewpoints v_k . Such sequences can be embedded on the torus, which leads to a set of torus coordinates (μ_{v_k}, ν_b) . The viewpoints do not need to be the same across subjects and the sequences do not need to be of the same length; only the beginning and end of the motion is needed to be aligned on the torus ν -axis. Given the embedding points and their corresponding contours for person s , a person-specific mapping function in the form of Eq. 4 can be fitted, which leads to a $D \times N$ coefficient matrix D^s . Notice that the kernel space, defined by $\psi(\cdot)$ in Eq. 4 is the same across all subjects since the same RBF centers are used on the torus.

Given the learned nonlinear mapping coefficients D^1, D^2, \dots, D^K , for training people $1, \dots, K$, the shape style parameters are decomposed by fitting an asymmetric bilinear model [46] to the coefficient space such that

$$[d^1 \dots d^K] = \mathbf{A}\mathbf{S}, \quad (7)$$

where each d^s is the DN -dimensional vector representation of the matrix D^s using column stacking. The matrix \mathbf{A} is a $DN \times K$ matrix containing the style basis for the coefficient space. The style matrix, \mathbf{S} , is an orthonormal matrix containing the style vectors for the training data. This decomposition can be obtained using the Singular Value Decomposition (SVD). The correlation tensor \mathcal{A} in Eq. 6 can be directly obtained by restacking the matrix \mathbf{A} to a $D \times S \times N$ tensor, where the first S style bases are retained to represent the shape subspace.

The model in Eq. 6 generalizes the model we previously proposed in [51], which was a view-based model. The model proposed here provides a continuous representation of the viewpoint and the body configuration in one latent representation space.

VI. BAYESIAN TRACKING ON THE TORUS

Tracking is the recursive estimation of the state on the torus, i.e., the estimation of the configuration, the viewpoint, and the shape style parameters given new observations. In our case, the observations are the body contours; either extracted using background subtraction or just fragmented edges detected in the image. Recovering the state on the torus directly yields the body kinematics in 3D through the mapping function in Eq. 5. The Bayesian tracking framework recursively updates the posterior density $P(\mathbf{X}_t | \mathbf{Y}^t)$ of the object state \mathbf{X}_t at time t given all observations $\mathbf{Y}^t = \mathbf{Y}_1, \mathbf{Y}_2, \dots, \mathbf{Y}_t$ up to time t

$$P(\mathbf{X}_t | \mathbf{Y}^t) \propto P(\mathbf{Y}_t | \mathbf{X}_t) \int_{\mathbf{X}_{t-1}} P(\mathbf{X}_t | \mathbf{X}_{t-1}) P(\mathbf{X}_{t-1} | \mathbf{Y}^{t-1}) d\mathbf{X}_{t-1}.$$

The state can be updated based on observation likelihood estimation $P(\mathbf{Y}_t | \mathbf{X}_t)$ given the transition probability $P(\mathbf{X}_t | \mathbf{X}_{t-1})$ and the previous frame state posterior $P(\mathbf{X}_{t-1} | \mathbf{Y}^{t-1})$.

On a torus manifold, the view and body configuration are represented together. The manifold provides a natural continuous, low-dimensional representation of the joint (view-configuration) distribution for a particle filter tracker. The state has three components: viewpoint, body configuration on the manifold and shape parameter. We denote the state at time t by $\mathbf{X}_t = [\lambda_t, \mu_t, \nu_t]$, where μ_t and ν_t are the torus coordinates corresponding to view and body configurations respectively. λ_t is the shape state, where the shape at time t is assumed to be a convex combination of the shape classes in the training set. That is, the shape style vector a_t at time t is written as a linear combination of K shape style vectors s^k in the style space

$$a_t = \sum_{k=1}^K w_t^k s^k, \quad \sum_{k=1}^K w_t^k = 1.$$

The shape state is represented by the coefficients w_t^k , i.e., $\lambda_t = [w_t^1, \dots, w_t^K]^T$.

A. Dynamic Model

Since the posture, view, and shape style parts of the state are independent given the observations, the dynamic model is a product of three dynamic models, i.e. $P(\mathbf{X}_t | \mathbf{X}_{t-1}) = P(\lambda_t | \lambda_{t-1}) P(\mu_t | \mu_{t-1}) P(\nu_t | \nu_{t-1})$. Therefore, the state model we use has the general form

$$\begin{aligned} \mu_t &= a\mu_{t-1} + n_\mu, \\ \nu_t &= b\nu_{t-1} + n_\nu, \\ \lambda_t &= \mathbf{C}\lambda_{t-1} + n_\lambda, \\ y_t &= h(\mu_t, \nu_t, \lambda_t; \mathcal{A}) + n_y, \end{aligned} \quad (8)$$

where n_μ, n_ν, n_λ , and n_y are zero mean white Gaussian noise. The scalars a and b and the matrix \mathbf{C} are the linear state

dynamics. The nonlinear mapping function $h(\mu_t, \nu_t, \lambda_t; \mathcal{A})$ maps the state to the observation and has the form

$$h(\mu_t, \nu_t, \lambda_t; \mathcal{A}) = \mathcal{A} \times_1 (\mathbf{S} \lambda_t) \times_2 \psi(\mu_t, \nu_t),$$

where the matrix $\mathbf{S} = [\mathbf{s}^1, \dots, \mathbf{s}^k]$ contains the shape style vectors representing the shape style space.

The shape state is supposed to be time-invariant. However, in tracking, since the tracked person's shape style is not known, the shape style needs to change with frames until it adapts to the correct shape style. Therefore, the propagation of particles in the shape space is controlled by a variance variable that decays with time. As introduced in Sec. IV, the configuration can be embedded on the torus in a way to achieve constant speed dynamics in the state space, which is very suitable in the case of periodic motion. Therefore, the propagation of the particle in the body configuration domain uses a constant speed model, which adapts to each person's dynamics through tracking.

B. Observation Model

For learning the model, we represent each shape instance as an implicit function $y(x)$ at each pixel x such that $y(x) = 0$ on the contour, $y(x) > 0$ inside the contour, and $y(x) < 0$ outside the contour. We use a signed-distance function such that

$$y(x) = \begin{cases} d_c(x) & x \text{ inside } c \\ 0 & x \text{ on } c \\ -d_c(x) & x \text{ outside } c \end{cases}$$

where the $d_c(x)$ is the distance to the closest point on the contour c with a positive sign inside the contour and a negative sign outside the contour. Such a representation imposes smoothness on the distance between shapes. Given such a representation, the input shapes are points in \mathcal{R}^d , where d is the dimensionality of the input space. This implicit function representation is typically used in level-set methods.

The model in Eq. 8 fits directly to the Bayesian framework to generate observation hypotheses from the particles. The tracker is realized using a traditional particle filter. The view, configuration, and shape style joint state is represented with N_ξ particles $\{\xi_t^{(k)}, \xi \pi_t^{(k)}\}_{k=1}^{N_\xi}$ with weights π . We can generate samples for a given particle $\xi_{t+1}^{(k)} = [\lambda_{t+1}^{(k)}, \mu_{t+1}^{(k)}, \nu_{t+1}^{(k)}]$ by

$$y_{t+1}^{(k)} = \mathcal{A} \times \mathbf{S} \lambda_{t+1}^{(k)} \times \psi(\mu_{t+1}^{(k)}, \nu_{t+1}^{(k)}),$$

i.e., each particle will directly generate a hypothesis contour in a given configuration, in a given view, in the form of a level-set function.

The observation itself can be in the form of extracted silhouettes (using background subtraction) or just edges extracted directly from the images. For the case of tracking background-subtracted silhouettes, each input silhouette is represented as a level set and the observation likelihood is computed using the Euclidean distance between the two implicit function representations. In the case of edge features, observation likelihood is estimated using probabilistic Chamfer matching [61].

VII. EXPERIMENTAL RESULTS

We have designed and performed several experiments to evaluate the proposed approach for tracking, view estimation, and 3D posture estimation, including: 1) quantitative evaluation of the 3D reconstruction of the body joints' locations with ground truth data; 2) comparison to other representations to show the advantage of the proposed supervised learning over unsupervised approaches; 3) evaluation of the tracking and the 3D reconstruction with style adaptation. The evaluation involved two different kinds of observations: background subtracted silhouettes and edge-based tracking. The experiments focused on tracking gait because of the availability of ground-truth data for gait. However, we also show several experiments using other types of motions such as golf swings and jump motions where the subject changes his viewpoint with respect to the camera during the motion.

A. Quantitative Evaluation:

The goal of this experiment is to evaluate the accuracy of the tracking approach in terms of recovering the 3D positions of the joints with ground truth data. For this purpose, we used the Brown HUMANEVA-I dataset [62], which provides ground truth data for the 3D locations of the joints for different types of motions. The data set provides sequences for training and evaluation for the same subjects. In particular, we used three walking sequences where the subject walks along an elliptical trajectory. Therefore, the subject's viewpoint with respect to the camera is continuously changing throughout the sequences. We did not use any camera calibration information for this data since we want to show that a learned model from a fixed view circle would generalize to track subjects from similar sittings without knowledge of the camera parameters. We also did not use any visual data from the HUMANEVA-I to train the model.

Training: We generated synthetic training data of walking silhouettes from our own motion capture data using the animation software Poser[®]. Twelve different views ($10^\circ, 30^\circ, \dots, 360^\circ$) were rendered for one walking cycle. The motion capture data used for synthesis is not part of the HUMANEVA-I dataset. Given this synthetic visual data, the data was embedded on a torus, as described in Section IV, and the manifold deformation from the torus was learned as was described in Section V. The model learned with such synthetic data would be enough for tracking and recovering the posture and the view parameters on the torus. However, for the reconstruction of the 3D joints' locations we also need to learn the deformation from the torus to the kinematic space. For this purpose, we used one training cycle of the joints' locations for each of the subjects from the HUMANEVA-I dataset.

Evaluation: Given the validation sequences from the HUMANEVA-I dataset, we extracted silhouettes using background subtraction after learning a background model using a nonparametric approach [63]. On the other hand, locations of the joints in the validation set are normalized to represent *normalized posture* in a body centered coordinate system which is invariant to subject's rotation and translation. This

was performed for the three subjects' validation sequences (S1, S2, S3) to estimate the performance of inferring 3D body posture.

For tracking, we used 900 particles to represent the view and body configuration on the torus manifold. We estimated the 3D body posture from the maximum-a-posterior (MAP) estimate of the body configuration from the particles in the particle filter. Figs. 6-e and 6-f show the estimated body configuration and view parameters on the torus as well as the reconstructed 3D posture (Fig. 6-d). As can be seen, the estimated body configuration parameter on the torus clearly exhibits constant-speed linear dynamics in tracking nonlinear deformation in the observations throughout the sequence. The walking cycles can be seen clearly in the sawtooth-like shape of the estimated body configuration. The estimated view parameter (Fig. 6-e) clearly exhibits constant change in viewpoint due to the ellipse-like trajectory of the subject's motion.

We measured the errors in the estimated body posture using the average absolute distance between individual markers, as in [62]. Fig. 7-a shows the average reconstruction error for all the joints' locations in each frame for subject 'S1'. The true and the estimated limb endpoints are compared in Fig. 7-b and Fig. 7-c. Table I shows the average errors in each of the subject's test sequences³. We tested the tracking performance with and without considering the dynamics of the body configuration and view. First, we used a random walk dynamic model, i.e., particles just drift randomly in each frame. Then we applied constant-speed dynamics for the body configuration state and the view state, assuming a constant velocity for the view and the body configuration variables on the torus. As can be seen from Table I, the experiments showed better body configuration estimation when a constant-speed dynamic model is used in the particle propagation. Overall, the average error in each joint's location from three subjects is 31.36 millimeters (*mm*).

B. Comparison of Different Representations:

The goal of this experiment is to compare different embedded representations of the visual manifold for tracking. Given a set of training data, representing the visual manifold of a walking motion viewed from different viewpoints (similar to the data in Fig. 2-a), we want to compare different embedded representations of the manifold, which can be used as a low-dimensional state space for Bayesian tracking as well as for frame-based posture estimation (without tracking).

Training Data: To learn the visual manifold, we used motion capture data of a walking cycle to render silhouettes from different viewpoints along one view circle at a fixed height. 39 frames from each of 12 views were rendered representing a walking cycle, i.e., a total of 468 frames.

Evaluation Data: For evaluation, we used two different sets: 1- Synthesized data: we used another rendered walking silhouette sequence containing three walking cycles with continuous view variability along a view circle (0-360 degrees). The

sequence contained 139 frames with 3 degrees of view angle change between frames.

2- We used the HumanEva-I subject 'I' data set, as described earlier.

For this purpose we compared:

- The torus representation: the proposed approach, which takes advantage of the knowledge about the topological structure of the manifold. We evaluated the torus for both tracking using a particle filter and, for comparison, we also evaluated frame-based posture estimation based on a torus using inverse mapping as was introduced earlier in [64].
- Unsupervised learning of the manifold using nonlinear dimensionality reduction techniques: in particular we used LLE [44], Isomap [45], and GPLVM [52] as popular techniques to achieve a two-dimensional embedding of the visual manifold. We used a two-dimensional embedding space to be comparable to the torus, i.e., the state space for tracking both the body configuration and the view was two-dimensional. Figs. 8-a, 8-b, and 8-c show the two-dimensional embeddings for the training shapes using each of the approaches. We also compared a three-dimensional embedding for the case of GPLVM.
- We compared a nearest neighbor search to see the best result we can get from the collected data itself. The nearest neighbor approach can be seen as an implicit way to represent the manifold of the data by keeping all the samples on the manifold as exemplars. Nearest neighbor search is used for frame-based posture estimation by searching for the nearest silhouette from the training data and using its corresponding 3D joints' location as the 3D reconstruction.

For each of the cases of the torus and the unsupervised embeddings, we fit nonlinear mappings, in the form of Eq. 2, from the embedding space to the shape space. Such mappings will serve as the observation model to generate observation hypotheses given the particles in the embedding space. The same number of RBF centers is used in all cases (we used 148 centers), except GPLVM. For LLE and Isomap, the RBF centers were set on the cluster centers obtained after fitting a Gaussian mixture to the embedded points using the EM algorithm. For the torus case, the RBF centers are set at equidistant points on the torus surface. For the case of GPLVM, the RBF centers are chosen as a part of the optimization process (we used 200 centers for GPLVM). For tracking, we used the same number of particles in the embedding space for each of the approaches.

Table II shows the average error in the recovered joints' locations for the different approaches. The torus embedding shows much better performance than the unsupervised manifold representation. Fig. 9 shows examples of the recovered postures at the sample frames from the HumanEva-I sequence for all the approaches. Fig. 10 shows the resulting trajectory of the estimated configuration and view on the torus surface. It should be noted that the goal of this experiment is to compare different representations for embedding the visual manifold. The same approaches compared here can be used in different ways for tracking. For example, GPLVM was earlier used[3],

³We discard the *HeadProximal* location of subject *S3* during the error estimation since the validation data has inconsistency.

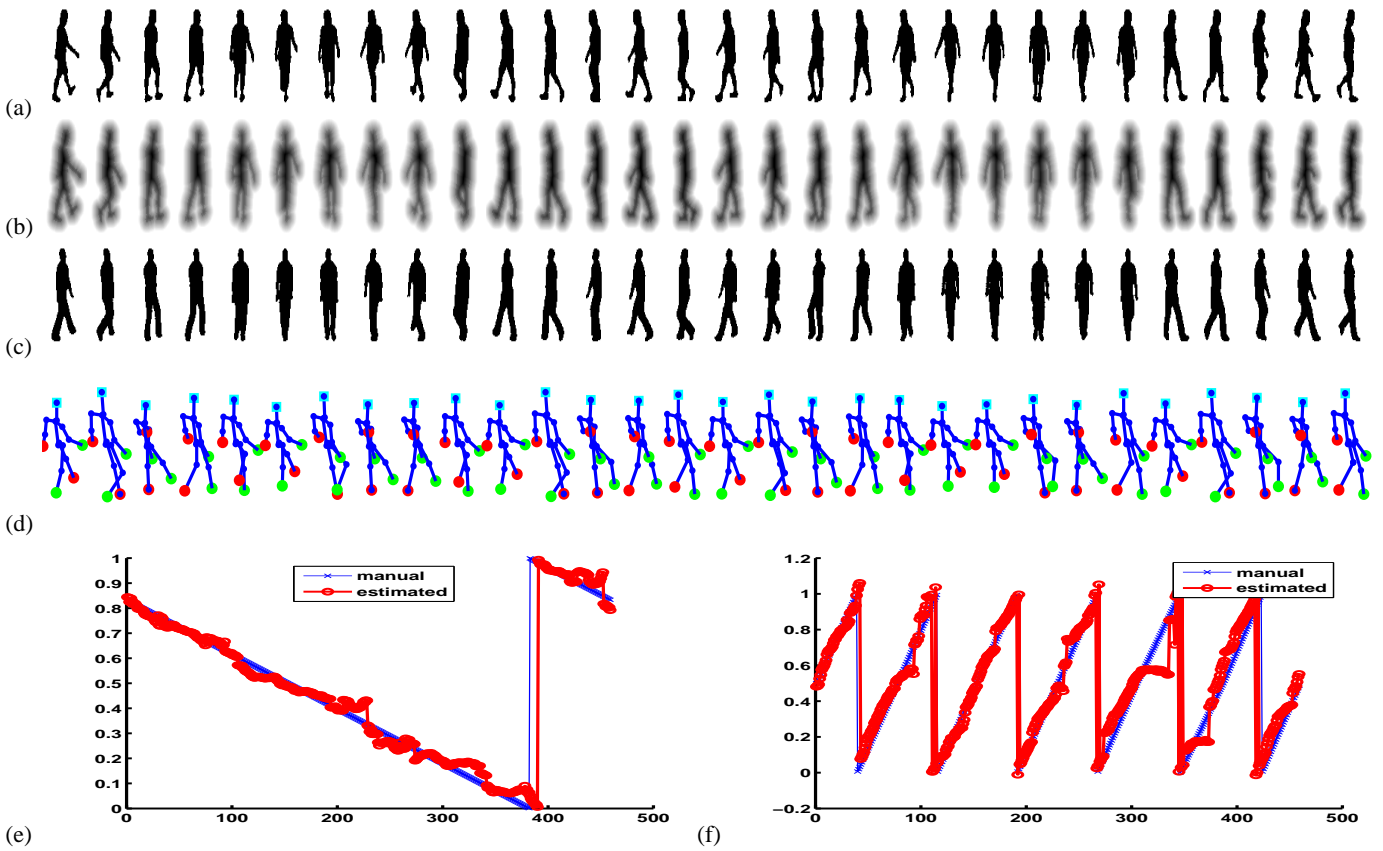


Fig. 6. Reconstruction of 3D body posture: (a) Input silhouettes. (b) Input as implicit functions used in the estimation. (c) Reconstructed silhouettes from estimated MAP parameters on the torus. (d) Reconstructed 3D posture shown from a fixed viewpoint. (e) Estimated values for the view parameter (μ). (f) Estimated values for the body configuration parameter (ν).

TABLE I
AVERAGE ERRORS IN 3D BODY POSTURE ESTIMATION FROM A SINGLE CAMERA

Subject	Start	End	Duration	Cycle	Mean Error(No Dyn.)	Mean Error(With Dyn.)
S1	76	534	459	6	26.16 mm	24.71 mm
S2	21	436	416	5	37.11 mm	31.16 mm
S3	91	438	348	5	40.47 mm	38.21 mm
S1,S2,S3					34.58 mm	31.36 mm

[35] to embed the kinematic manifold within a model-based approach and was shown to achieve good results. The point we can conclude from this experiment is that enforcing the known manifold topology leads to better results than unsupervised learning of an embedded representation.

C. Shape Style Adaptation:

The goal of this experiment is to show the ability of the tracker to adapt to different people’s shape styles while tracking the body configuration and the viewpoint, as described in Section V-C. We fit the model in Eq. 6 with real data from multiple people. To fit the model, we used sequences from four different people walking on a treadmill obtained from seven different views using synchronized cameras. For evaluation, we used the HUMANEVA-I subject ‘S1’ dataset, i.e., the person’s shape style is not in the training data. The subject in the evaluation sequence is walking on an elliptical trajectory and captured from a stationary camera. The tracker

adapts to the observed shapes by estimating the style factor a^s in Eq. 6. The tracker starts from a mean style and adapts to the correct person’s shape style. Fig. 11 shows the tracking results. As can be seen, after large errors at the beginning, the 3D reconstruction errors decreased as the model adapted to the correct person’s style. Fig. 12 shows tracking for a subject walking on an ellipse-like trajectory in an outdoor scene.

D. Tracking a Non-Periodic Motion

We evaluated the approach with non-periodic open trajectory motions such as jump motions and golf swing motions. We used motion capture data to learn the model for “jump in the air” motion using geodesics-based embedding on the torus. We rendered silhouettes from 12 viewpoints to model the visual manifold. For evaluation, we used jumping sequences where the subject jumps in the air and may rotate while jumping. The tracker’s job is to recover the body configuration and viewpoint, which is changing due to the rotation during the

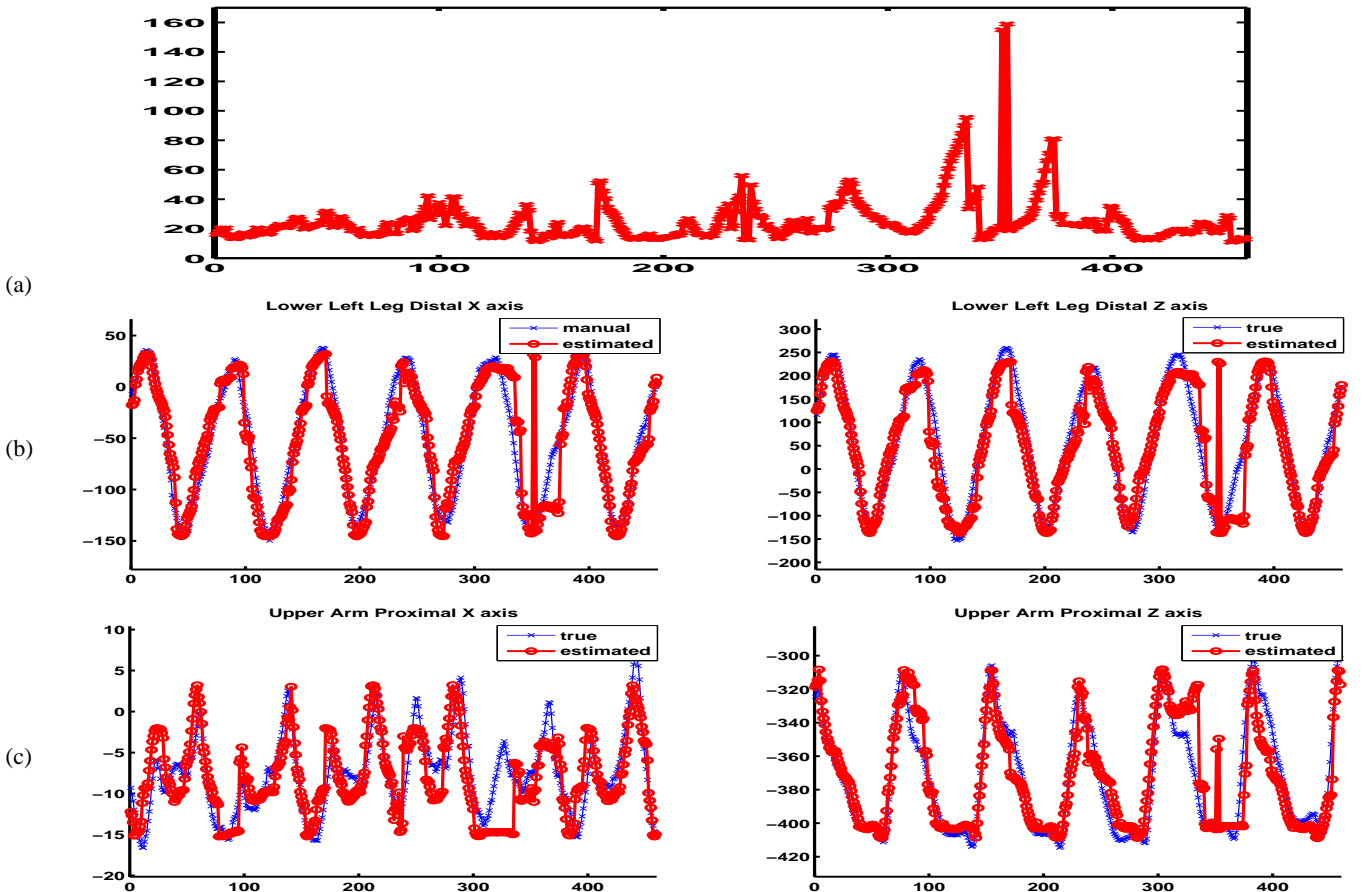


Fig. 7. A test sequence reconstruction (HUMANEVA-I S1-walking) : horizontal-axis: frame number, vertical-axis: estimated joints' location (unit:mm). (a) Average errors for all the joints in each frame. (b) True and estimated x and z values for the *Lower left leg distal* joint (c) True and estimated x and z values for the *the upper right arm proximal* joint.

TABLE II
AVERAGE ERRORS IN 3D BODY POSTURE ESTIMATION USING DIFFERENT EMBEDDINGS

Representation	Tracking					Frame-based posture recovery	
	LLE	Isomap	GPLVM-2D	GPLVM-3D	Torus	NN	Torus - inverse map [64]
Synthetic data							
Average Error in mm	76.96	77.22	43.25	46.12	15.49	25.38	52.24
HumanEva I-S1							
Average Error in mm	66.24	61.12	81.79	-	38.21	48.49	76.56

motion. Here we show two example sequences, one without rotation, and one with rotation. In both cases, background subtraction was used to extract the input silhouettes, which are used as the observations. Fig. 13 shows the estimated view and body configuration parameters for an outdoor jump sequence without rotation. Despite inaccurate silhouette extraction (Fig. 13-a), the model estimates the body configuration accurately (Fig. 13-e). The view estimation (Fig. 13-c) shows a constant viewpoint, which confirms with the non-rotational jump motion.

In contrast, Fig. 14-a shows a jump motion with a body rotation in the air. The estimated view parameter (Fig. 14-e) shows the view parameter change due to body rotation during

the motion. Simultaneously, the estimated body configuration parameter enables reconstruction of the 3D body posture (Fig. 14-d).

E. Edge-based Contour Tracking

This section shows example results for tracking based on edges, without background subtraction. The approach introduced in this paper is a generative model for the observed contours of a motion. Therefore, it can be used to generate contours at different body configurations, viewpoints, and shape styles during the motion. Such generated contour hypotheses can be evaluated given the detected edges in the image sequence using Chamfer matching directly. We tested

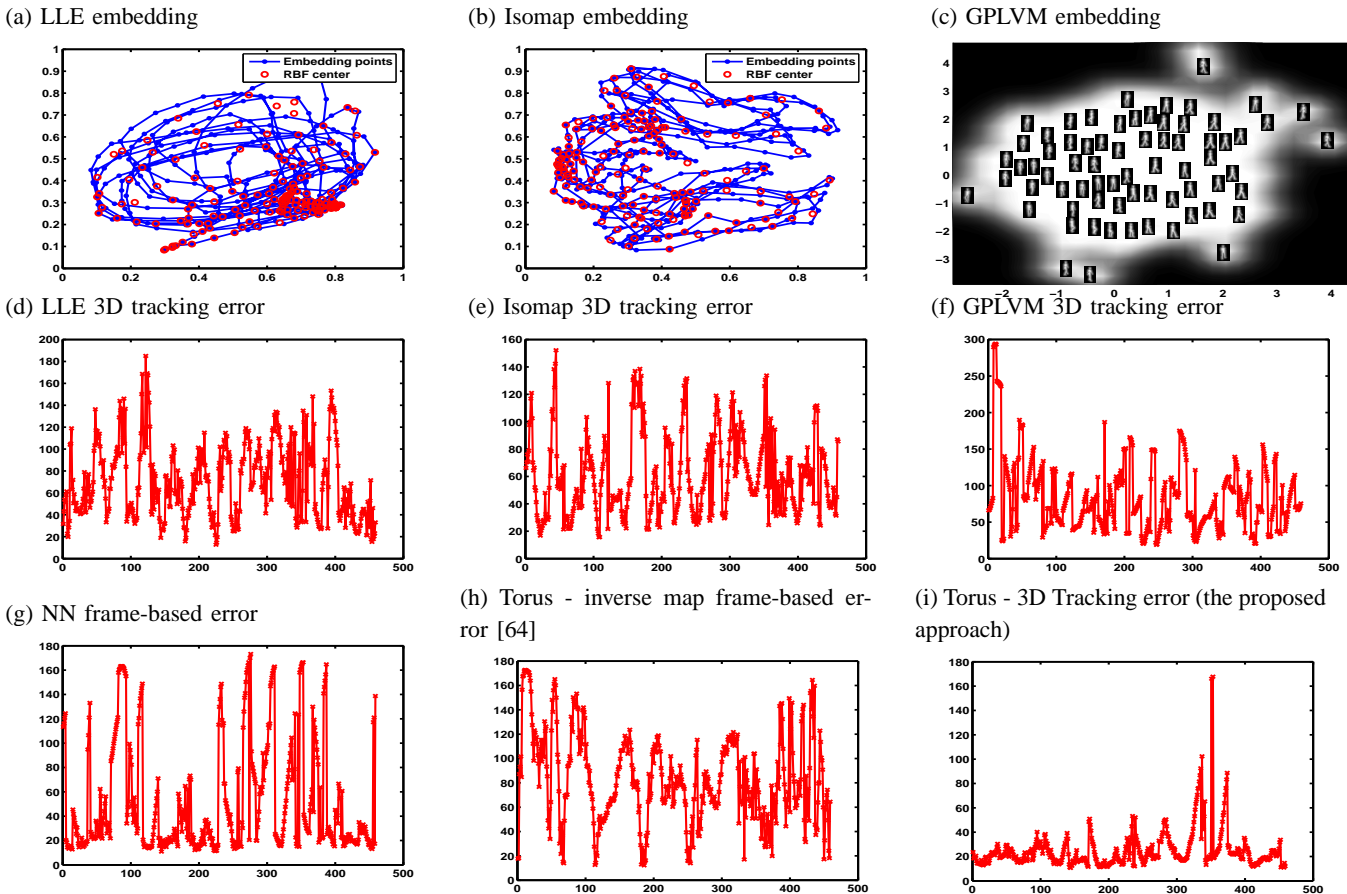


Fig. 8. Manifold embeddings and reconstruction errors for different approaches. (a-c) LLE, isomap, and GPLVM manifold embedding for the training sequence (d-h) Average errors in joints' locations in each frame for different models. (i) Average error in joints' locations estimation using the proposed approach.

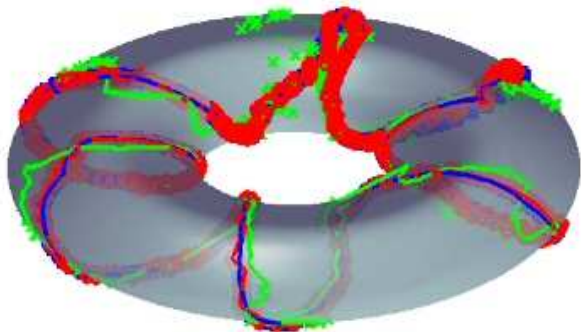


Fig. 10. The trajectory of the estimated configuration and view parameters on the torus from the particle filter: MAP estimation (green), Expected values (blue) and mode values (red).

the approach for tracking from the detected edges for different motions. Here we show examples for gait and golf swings motions.

Figs. 15-a and 15-b show tracking results for an outdoor walking sequence where the subject is walking on an elliptical trajectory. As a result, we can see a spiral motion on the torus manifold due to the simultaneous change of the view and the

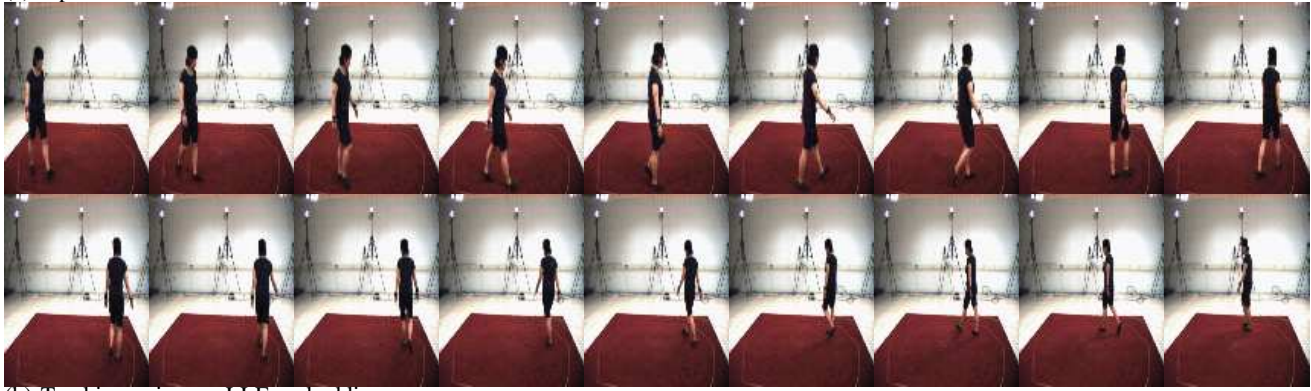
body configuration parameters. Reliable tracking results can be achieved using weak edge cues.

We also tested the approach for tracking a golf swing motion from unknown view and uncalibrated camera. Figs. 15-c and 15-d show example tracking results. Since the view is unknown, the tracker started from a uniform distribution, i.e., the particles are spread along the view circle on the torus (the same μ) at the beginning and it converged to one area as the motion is tracked.

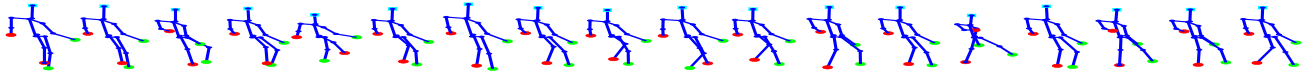
VIII. CONCLUSIONS

We formulate the problem of view variant human motion tracking as tracking on a torus surface. We use the torus as a state space for tracking both the body configuration and the viewpoint. We learn how a torus deforms to the actual visual manifold and to the kinematic manifold through two nonlinear mapping functions. The torus model is suitable for any one-dimensional manifold motion, whether periodic, (such as walking, running, etc.), quasi-periodic, or non periodic (such as golf swings, jumping, etc.). The experimental results showed that such a model is superior to other representations for the task of tracking and posture/view recovery since it provides a low-dimensional, continuous, uniformly spaced state representation. We also show how the model can be

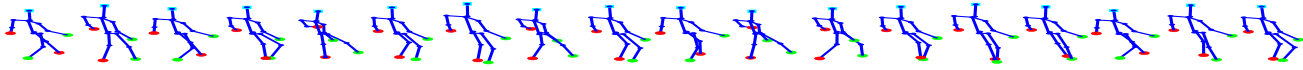
(a) Input frames



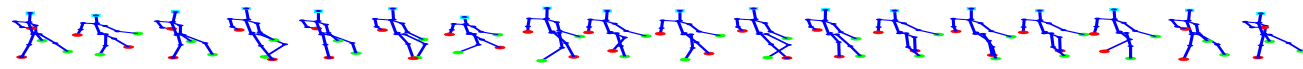
(b) Tracking using an LLE embedding:



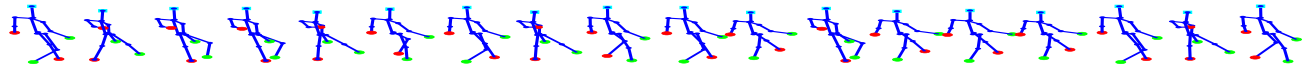
(c) Tracking using an Isomap embedding:



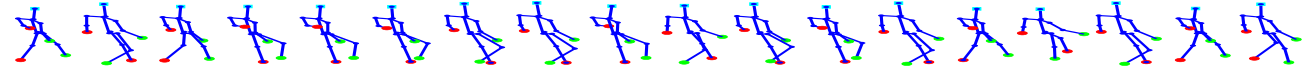
(d) Tracking using a GPLVM embedding:



(e) Frame-based posture recovery using a nearest neighbor search:



(f) Frame-based posture recovery using Torus - inverse mapping:



(g) Tracking using Torus embedding (the proposed approach):

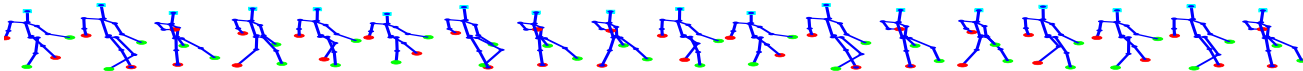


Fig. 9. Example results for posture recovery using different embedded representations.

generalized to the full view sphere and how to adapt to different persons' shapes.

The approach introduced in this paper is a parameterized generative function for the visual manifold of an observed motion. Therefore, it can be used to generate contours at different body configurations, viewpoints, and shape styles during the motion, without a 3D body model. From a perceptual point of view, the models we developed provide a computational approach that is in line with the view-based theory of object representation.

An interesting research direction is to determine how to deal with complex human motions. We believe that complex motions can be segmented into motion primitives where such primitives are one-dimensional manifold motions, whether open or closed trajectories. Segmenting complex motion into motion primitives is an active research direction that we are pursuing. Given this view, the torus representation presented in this paper can be useful for modeling motions that are more complex. In addition, complex human motions can be

dealt with through hierarchical models where different latent representations for different body joints can be achieved.

Acknowledgment This research is partially funded by NSF CAREER award IIS-0546372.

REFERENCES

- [1] A. Elgammal and C.-S. Lee, "Inferring 3D body pose from silhouettes using activity manifold learning," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2004, pp. 681–688.
- [2] T.-P. Tian, R. Li, and S. Sclaroff, "Articulated pose estimation in a learned smooth space of feasible solutions," in *Workshop on Learning in Computer Vision and Pattern Recognition*, 2005.
- [3] R. Urtasun, D. J. Fleet, A. Hertzmann, and P. Fua, "Priors for people tracking from small training sets," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2005, pp. 403–410.
- [4] V. I. Morariu and O. I. Camps, "Modeling correspondences for multi-camera tracking using nonlinear manifold learning and target dynamics," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006, pp. 545–552.

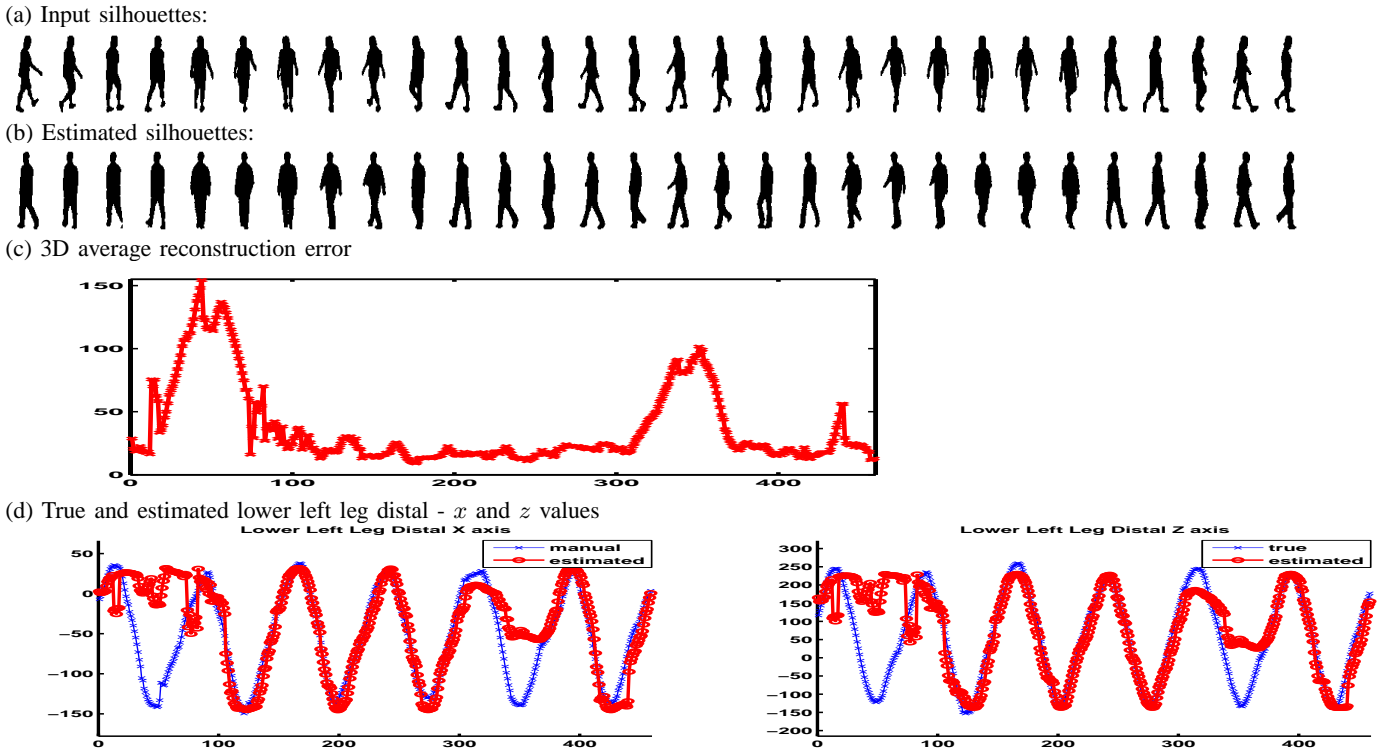


Fig. 11. Style adaptive tracking: (a) Original test silhouettes of a subject walking on an elliptical trajectory from the HumanEva-I dataset. (b) The estimated silhouettes with style adaptation. (c) Average 3D reconstruction errors in joints' locations in each frame. (d) True and estimated x and z values for the *Lower left leg distal* joint.

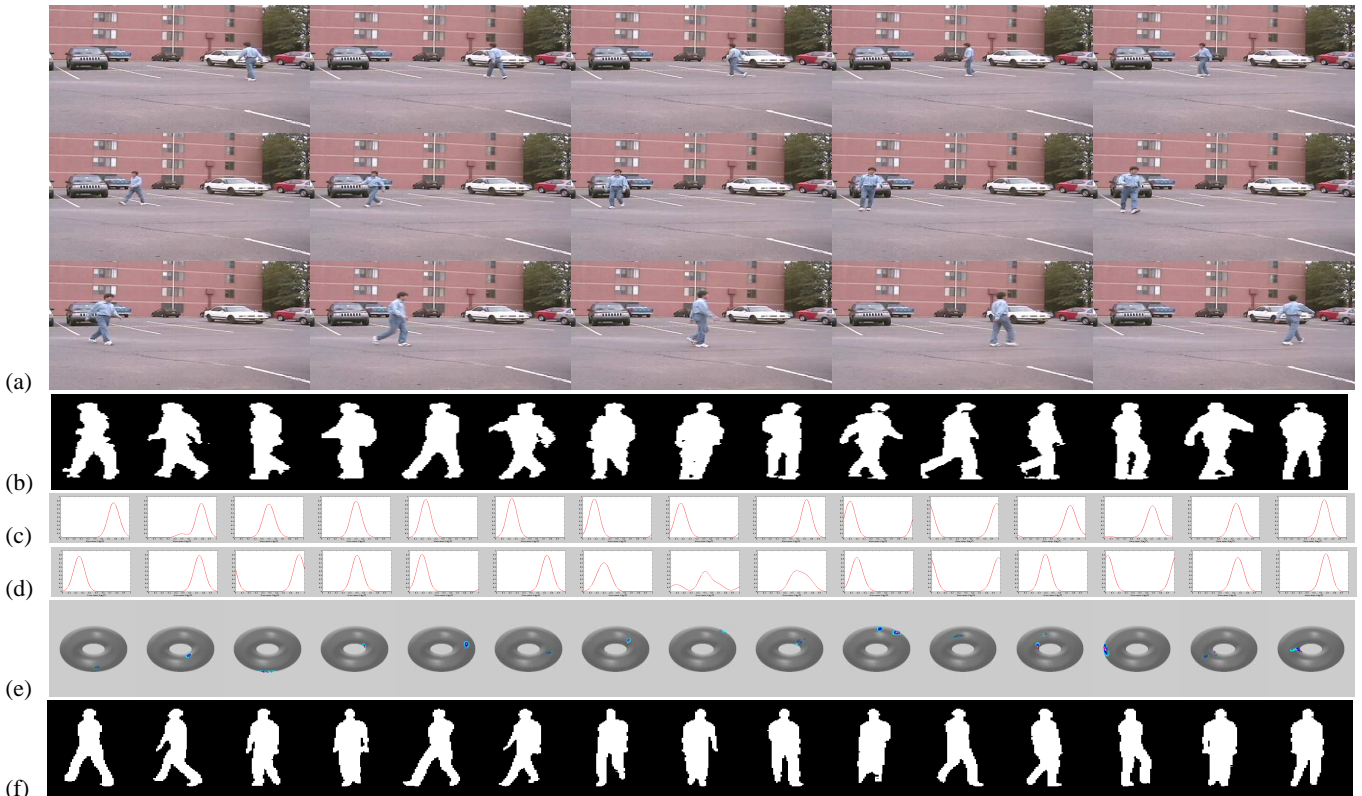


Fig. 12. Evaluation of view-variant gait tracking from real data: (a) Sample input frames (b) Input silhouettes. (c) The estimated body configuration parameter values (d) The estimated view parameter values. (e) The distributions of the particles on the torus. (f) The recovered shape from the estimated configuration and view

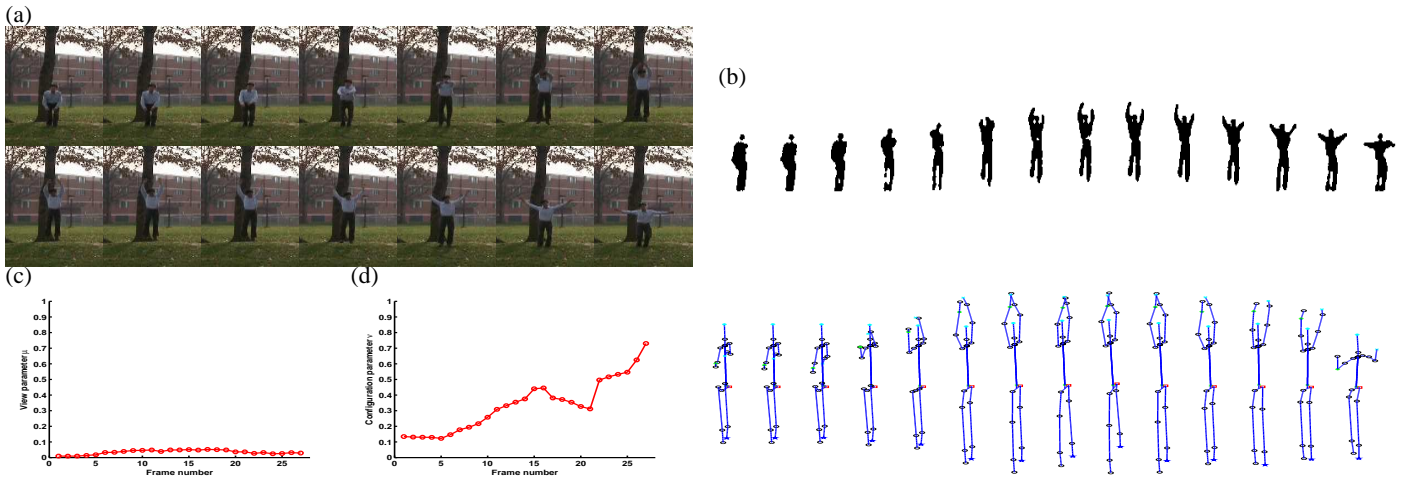


Fig. 13. An outdoor fixed-view jump motion. (a) Sample input images. (b) Sample input silhouette. (c) The estimated view parameter values for a constant-view. (d) The estimated body configuration parameter values. (e) 3D body posture reconstruction.

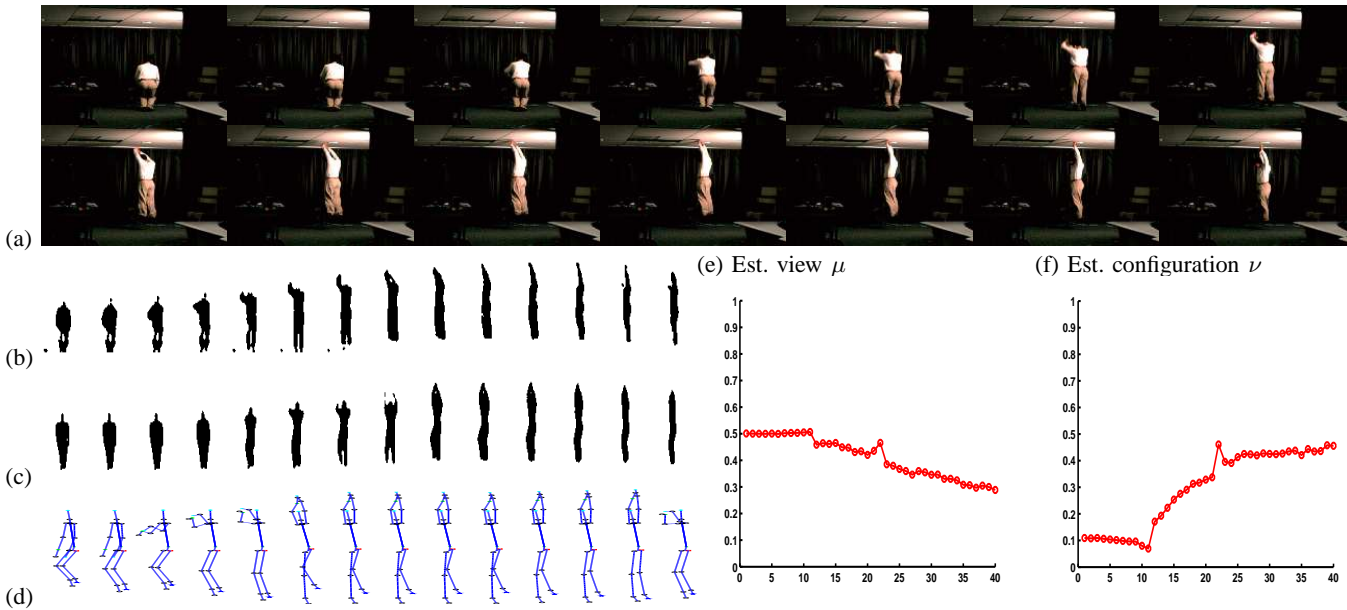


Fig. 14. Indoor jump motion with rotation. (a) Input image. (b) Input silhouettes. (c) Reconstructed silhouettes. (d) 3D body posture reconstruction based on the estimated body configuration parameter. (e) Estimated view. (f) Estimated body configuration.

- Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006, pp. 751–757.
- [6] K. Moon and V. Pavlovic, “Impact of dynamics on subspace embedding and tracking of sequences,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006, pp. 198–205.
- [7] J. K. Aggarwal and Q. Cai, “Human motion analysis: a review,” *Computer Vision and Image Understanding (CVIU)*, vol. 73, no. 3, pp. 428–440, 1999.
- [8] D. M. Gavrila, “The visual analysis of human movement: a survey,” *Computer Vision and Image Understanding (CVIU)*, vol. 73, no. 1, pp. 82–98, 1999.
- [9] T. B. Moeslund, A. Hilton, and V. Krüger, “A survey of advances in vision-based human motion capture and analysis,” *Computer Vision and Image Understanding (CVIU)*, vol. 104, no. 2, pp. 90–126, 2006.
- [10] J. O’Rourke and Badler, “Model-based image analysis of human motion using constraint propagation,” *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 2, no. 6, 1980.
- [11] D. Hogg, “Model-based vision: a program to see a walking person,” *Image and Vision Computing*, vol. 1, no. 1, pp. 5–20, 1983.
- [12] K. Rohr, “Towards model-based recognition of human movements in image sequence,” *Computer Vision, Graphics, and Image Processing*, vol. 59, no. 1, pp. 94–115, 1994.
- [13] J. M. Rehg and T. Kanade, “Model-based tracking of self-occluding articulated objects,” in *Proceedings of the International Conference on Computer Vision (ICCV)*, 1995, pp. 612–617.
- [14] D. Gavrila and L. Davis, “3-D model-based tracking of humans in action: a multi-view approach,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 1996, pp. 73–80.
- [15] I. A. Kakadiaris and D. Metaxas, “Model-based estimation of 3D human motion with occlusion based on active multi-viewpoint selection,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 1996, pp. 81–87.
- [16] H. Sidenbladh, M. J. Black, and D. J. Fleet, “Stochastic tracking of 3D human figures using 2d image motion,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2000, pp. 702–718.
- [17] J. M. Rehg and T. Kanade, “Visual tracking of high DOF articulated structures: an application to human hand tracking,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 1994, pp. 35–46.
- [18] T. Darrell and A. Pentland, “Space-time gesture,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 1993, pp. 335–340.
- [19] L. W. Campbell and A. F. Bobick, “Recognition of human body motion using phase space constraints,” in *Proceedings of the International Conference on Computer Vision (ICCV)*, 1995, p. 624.

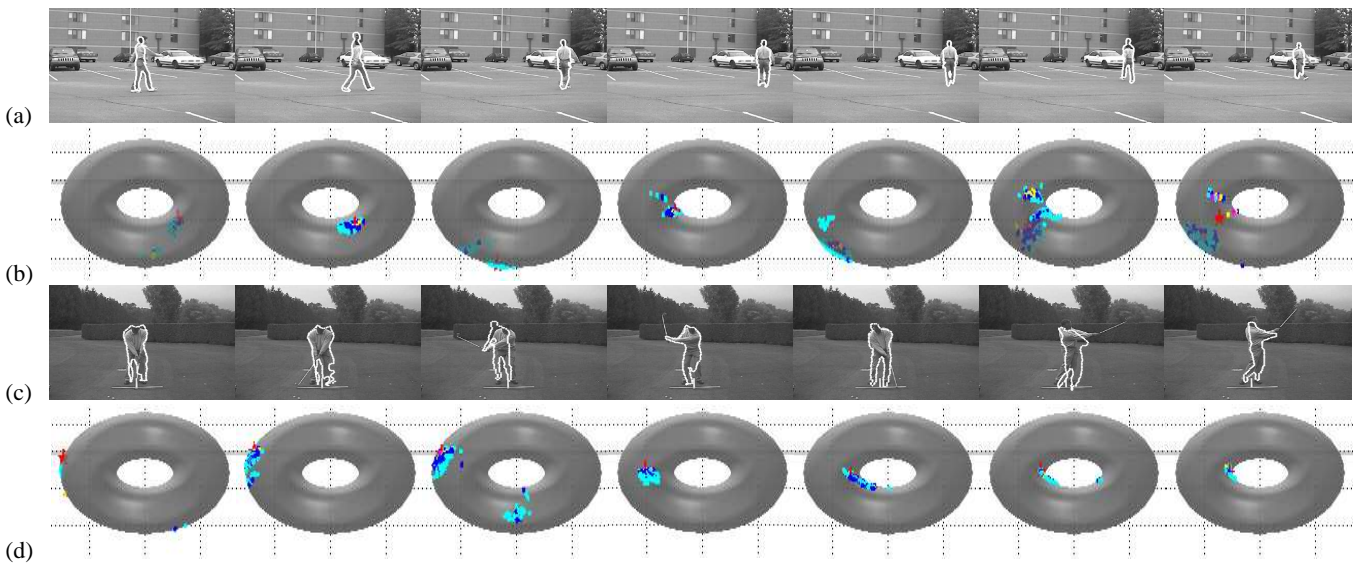


Fig. 15. Edge-based tracking: (a,b) A gait sequence tracking : (a) Estimated shape contours. (b) View and configuration particle distributions on the torus. (c,d) Golf swing tracking: (c) Estimated shape contours (d) View and configuration particle distributions on the torus.

- [20] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 19, no. 7, pp. 780–785, 1997.
- [21] S. X. Ju, M. J. Black, and Y. Yacoob, "Cardboard people: A parameterized model of articulated motion," in *Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition (FGR)*, Killington, Vermont, 1996, pp. 38–44.
- [22] G. Shakhnarovich, J. W. Fisher, and T. Darrell, "Face recognition from long-term observations," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2002, pp. 851–865.
- [23] S. X. Ju, M. J. Black, and Y. Yacoob, "Cardboard people: A parameterized model of articulated image motion," in *Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition (FGR)*, 1996, pp. 38–44.
- [24] Y. Yacoob and M. J. Black, "Parameterized modeling and recognition of activities," *Computer Vision and Image Understanding (CVIU)*, vol. 73, no. 2, pp. 232–247, 1999.
- [25] G. Mori and J. Malik, "Estimating human body configurations using shape context matching," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2002, pp. 666–680.
- [26] K. Grauman, G. Shakhnarovich, and T. Darrell, "Inferring 3D structure with a statistical image-based shape model," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2003, p. 641.
- [27] G. Shakhnarovich, P. Viola, and T. Darrell, "Fast pose estimation with parameter-sensitive hashing," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2003, pp. 750–759.
- [28] R. Rosales, V. Athitsos, and S. Sclaroff, "3D hand pose reconstruction using specialized mappings," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2001, pp. 378–387.
- [29] A. Agarwal and B. Triggs, "3D human pose from silhouettes by relevance vector regression," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2004, pp. 882–888.
- [30] C. Sminchisescu, A. Kanaujia, Z. Li, and D. N. Metaxas, "Discriminative density propagation for 3D human motion estimation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 390–397.
- [31] M. Brand, "Shadow puppetry," in *Proceedings of the International Conference on Computer Vision (ICCV)*, vol. 2, 1999, pp. 1237–1244.
- [32] D. Ormonoit, H. Sidenbladh, M. J. Black, and T. Hastie, "Learning and tracking cyclic human motion," in *Proceedings of Advances in Neural Information Processing (NIPS)*, 2000, pp. 894–900.
- [33] C. Sminchisescu and A. Jepson, "Generative modeling of continuous non-linearly embedded visual inference," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2004, pp. 140–147.
- [34] A. Rahimi, B. Recht, and T. Darrell, "Learning appearance manifolds from video," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 2005, pp. 868–875.
- [35] R. Urtasun, D. J. Fleet, and P. Fua, "3D people tracking with gaussian process dynamical models," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006, pp. 238–245.
- [36] J. Wang, D. J. Fleet, and A. Hertzmann, "Gaussian process dynamical models," in *Proceedings of Advances in Neural Information Processing (NIPS)*, 2005.
- [37] S. Roweis and Z. Ghahramani, "An EM algorithm for identification of nonlinear dynamical systems," in *Kalman Filtering and Neural Networks*, S. Haykin, Ed.
- [38] C. M. Christoulias and T. Darrell, "On modelling nonlinear shape-and-texture appearance manifolds," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 1067–1074.
- [39] H. Murase and S. Nayar, "Visual learning and recognition of 3D objects from appearance," *International Journal of Computer Vision (IJCV)*, vol. 14, no. 1, pp. 5–24, 1995.
- [40] R. Fablet and M. J. Black, "Automatic detection and tracking of human motion with a view-based representation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2002, pp. 476–491.
- [41] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky, "Dynamism of a dog on a leash" or behavior classification by eigen-decomposition of periodic motions," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Copenhagen: Springer-Verlag, LNCS 2350, May 2002, pp. 461–475.
- [42] K. Toyama and A. Blake, "Probabilistic tracking in a metric space," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2001, pp. 50–59.
- [43] B. J. Frey and N. Jojic, "Learning graphical models of images, videos and their spatial transformation," in *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence - San Francisco, CA*, 2000, pp. 184 – 191.
- [44] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
- [45] J. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319 – 2323, 2000.
- [46] J. B. Tenenbaum and W. T. Freeman, "Separating style and content with bilinear models," *Neural Computation*, vol. 12, pp. 1247–1283, 2000.
- [47] M. A. O. Vasilescu and D. Terzopoulos, "Multilinear analysis of image ensembles: Tensorfaces," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2002, pp. 447–460.
- [48] J. R. Magnus and H. Neudecker, *Matrix Differential Calculus with Applications in Statistics and Econometrics*. John Wiley & Sons, 1988.
- [49] L. D. Lathauwer, B. de Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM Journal On Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, 2000.
- [50] M. A. O. Vasilescu, "Human motion signatures: Analysis, synthesis,

- recognition,” in *Proceedings of the International Conference on Pattern Recognition (ICPR)*, vol. 3, 2002, pp. 456–460.
- [51] A. Elgammal and C.-S. Lee, “Separating style and content on a nonlinear manifold,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 2004, pp. 478–485.
- [52] N. D. Lawrence, “Gaussian process models for visualisation of high dimensional data,” in *Proceedings of Advances in Neural Information Processing (NIPS)*, 2004.
- [53] J. G. Silva, J. S. Marques, and J. M. Lemos, “Non-linear dimension reduction with tangent bundle approximation,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2005, pp. iv/85 – iv/88.
- [54] M. Brand and A. Hertzmann, “Style machines,” in *SIGGRAPH*, 2000, pp. 183–192.
- [55] J. H. Ham, D. D. Lee, and L. K. Saul, “Learning high dimensional correspondences from low dimensional manifolds,” in *ICML 2003 Workshop on The Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*, pp. 34–41.
- [56] A. P. Shon, K. Grochow, A. Hertzmann, and R. Rao, “Learning shared latent structure for image synthesis and robotic imitation,” in *Proceedings of Advances in Neural Information Processing (NIPS)*, 2006, pp. 1233–1240.
- [57] A. Gray, *Modern Differential Geometry of Curves and Surfaces with Mathematica*, 2nd ed. CRC Press, 1997.
- [58] G. S. Kimeldorf and G. Wahba, “A correspondence between bayesian estimation on stochastic processes and smoothing by splines.” *The Annals of Mathematical Statistics*, vol. 41, pp. 495–502, 1970.
- [59] B. Schölkopf and A. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press, 2002.
- [60] T. Poggio and F. Girosi, “Networks for approximation and learning,” *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1481–1497, 1990.
- [61] A. Elgammal, V. Shet, Y. Yacoob, and L. S. Davis, “Gesture recognition using a probabilistic framework for pose matching,” in *The Seventh International Conference on Control, Automation, Robotics and Vision, ICARCV 2002*, 2002.
- [62] L. Sigal and M. J. Black, “Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion,” Brown University, Tech. Rep. CS-06-08, 2006.
- [63] A. Elgammal, D. Harwood, and L. S. Davis, “Non-parametric model for background subtraction,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2000, pp. 751–767.
- [64] C.-S. Lee and A. Elgammal, “Simultaneous inference of view and body pose using torus manifolds,” in *Proceedings of the International Conference on Pattern Recognition (ICPR)*, 2006, pp. 489–494.