Int J Comput Vis (2010) 87: 118–139
DOI 10.1007/s11263-009-0266-5

Coupled Visual and Kinematic Manifold Models for Tracking

C.-S. Lee · A. Elgammal

Received: 13 January 2008 / Accepted: 29 June 2009 / Published online: 15 July 2009

Abstract In this paper, we consider modeling data lying on multiple continuous manifolds. In particular, we model the shape manifold of a person performing a motion observed from different viewpoints along a view circle at a fixed camera height. We introduce a model that ties together the body configuration (kinematics) manifold and visual (observations) manifold in a way that facilitates tracking the 3D configuration with continuous relative view variability. The model exploits the low-dimensionality nature of both the body configuration manifold and the view manifold, where each of them are represented separately. The resulting representation is used for tracking complex motions within a Bayesian framework, in which the model provides a low-dimensional state representation as well as a constrained dynamic model for both body configuration and view variations. Experimental results estimating the 3D body posture from a single camera are presented for the HUMANEVA dataset and other complex motion video sequences.

Keywords Visual manifold · Human motion tracking · Kinematic manifold · Manifold learning · Bayesian tracking · Pose estimation

Electronic supplementary material The online version of this article (http://dx.doi.org/10.1007/s11263-009-0266-5) contains supplementary material, which is available to authorized users.

C.-S. Lee (✉)
Department of Electronic Engineering, School of Electronic Engineering, Communication Engineering and Computer Science, Yeungnam University, Gyeongsan, South Korea
e-mail: chansu@ynu.ac.kr

A. Elgammal
Department of Computer Science, Rutgers University, Piscataway, NJ, USA
e-mail: elgammal@cs.rutgers.edu

1 Introduction

Human motion analysis is a challenging computer vision problem with wide interest emanating from various potential real-world applications, such as visual surveillance, human-machine interface, video archival and retrieval, computer graphics animation, autonomous driving, and virtual reality. Despite the high-dimensionality of the human body configuration space, many human activities lie intrinsically on low-dimensional manifolds (Elgammal and Lee 2007). Exploiting this property is essential for constraining the solution space for many problems, such as tracking, posture estimation, and activity recognition. Recently, increasing interest has been placed on learning low-dimensional representations for the manifolds of the body configuration during motions, as in Elgammal and Lee (2004a), Sminchisescu and Jepson (2004), Urtasun et al. (2005), Christoudias and Darrell (2005), Morariu and Camps (2006) for tracking and posture estimation. We can discriminate between approaches that learn joint angle configuration manifolds (e.g., Urtasun et al. 2005), with the goal of creating a better dynamic model for tracking, and approaches that focus on modeling the visual manifold (e.g. Elgammal and Lee 2004a; Christoudias and Darrell 2005) with the aim of inference of configuration from visual input.

The goal of this paper is to model the visual manifold of an articulated object observed from different viewpoints. Modeling visual manifolds is a challenging task. In particular, we focus on modeling human motion observed from different viewpoints. Traditionally, generative model-based approaches have been used for tracking and posture estimation. These approaches utilize a 3D body model and a camera model, and the problem is formulated as a search problem in high-dimensional spaces (articulated body configuration and geometric transformation). Alternatively, discriminative mappings have also been introduced. The model introduced here is generative. However, it generates observations for a certain motion observed from different viewpoints without any explicit 3D body model. This is achieved through modeling the visual manifold corresponding to different postures and views.

Modeling the visual manifolds for rigid objects under different views and illuminations has been studied in Murase and Nayar (1995) for object recognition. However, dealing with articulated objects is more challenging. Consider the simple example of observing a human performing a periodic motion, such as walking, from different viewpoints along a view circle. For a given viewpoint, it has been shown in Elgammal and Lee (2004a) that, the observed motion lies on a low-dimensional manifold (one dimensional for gait). This corresponds to the configuration manifold observed from a single viewpoint. Given a single body posture observed from different viewpoints along a viewing circle, the observations will lie on a one-dimensional manifold as well. That is the view manifold for that particular posture. In other words, each posture has its own view manifold, and each view has its own configuration manifold. If both the motion and the view are one-dimensional manifolds (e.g., gait observed from a view circle), then this product space is equivalent to a torus manifold (Lee and Elgammal 2006). In this previous work, a torus was used to model such a two-dimensional manifold (configuration × view) jointly. However, the approach in Lee and Elgammal (2006) is limited to the particular setting of a one-dimensional motion. The fundamental question we address here is: *How to learn a representation of a view manifold that is invariant to the body posture and, therefore, exhibits the one-dimensional behavior expected due to the camera setting.*

The contribution of this paper can be summarized in the following goals that we achieve:

I To model the posture, view, and shape manifolds of an observed motion with three separate low-dimensional representations: (1) a view-invariant, shape-invariant configuration manifold; (2) a configuration-invariant, shape-invariant view manifold; (3) a configuration-invariant, view-invariant shape representation.

II To model the view and posture manifolds in a general setting, in which the motion is not assumed to be one dimensional. We show results with complex motions.

III To link the configuration manifold learned from 3D motion-captured data with the visual manifold. A distinguishing feature of our work here is that we utilize both the input (visual) and output (kinematic) manifolds to constrain the problem. We model the kinematic manifold and the observation manifold, tied together with a parameterized generative mapping function.[1]

We consider tracking and inferring the view and body configuration of a human motion from a single monocular camera. In this setting, a person can change his/her pose with respect to the camera while being tracked (equivalently, the camera can be moving). In this paper, we limit the view variability to a one-view circle (we use different viewpoints at a fixed camera height as an example of a view circles). However, this is not a theoretical limitation of the approach but rather a practical choice. Our main goal is to model a person's pose with respect to the camera and not the camera's motion. The camera is typically fixed and mounted at a fixed height in many applications, and the person can change his/her orientation with respect to the camera. Our experimental results, in which no camera calibration is assumed, reveal that a one-view circle provides a good approximation of the expected viewpoint variability in such scenarios.

The paper's organization is as follows: After literature review on human motion analysis in Sect. 2, Sect. 3 summarizes the framework. Sections 4 and 5 describe the learning procedure. A Bayesian tracking framework using the proposed generative model is presented in Sect. 6. Section 7 shows experimental results for different motions with varying complexity.

2 Related Work

In the last two decades, extensive research has been performed on understanding human motion from image sequences. We refer the reader to the excellent surveys covering this topic, such as Aggarwal and Cai (1999), Gavrila (1999), Moeslund et al. (2006). The problems of tracking and recovery of body configuration have been traditionally addressed through generative model-based approaches, e.g., O'Rourke (1980), Hogg (1983), Rohr (1994), Rehg and Kanade (1995), Gavrila (1996), Kakadiaris and Metaxas (1996), Sidenbladh et al. (2000). 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). Partial recovery of body configuration can also be achieved through view-based representations (models), e.g. Darrell and Pentland (1993), Campbell and Bobick (1995), Shakhnarovich et al. (2002), Yacoob (1999). In such case, constancy of the local appearance of individual body

[1] Since we use kinematic data to learn the configuration manifold, in this paper we use the terms kinematic manifold and configuration manifold interchangeably.

parts is exploited. The main limitation with such approaches is that they deal with limited view configurations, i.e., single view or a small set of discrete views.

Alternatively, discriminative approaches have been proposed where recovering body posture can be achieved directly from the visual input by posing the problem as a supervised learning problem through searching a pre-labelled database of body posture (Mori and Malik 2002; Grauman et al. 2003; Shakhnarovich et al. 2003) or through learning regression models from input to output (Rosales et al. 2001; Grauman et al. 2003; Agarwal and Triggs 2004; Sminchisescu et al. 2005). All these approaches pose the problem as a regression 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 mapping hard to generalize.

Despite the high dimensionality of both the human joint angle space and the visual input space, many human activities lie on low dimensional manifolds. In the last few years, there have been increasing interest in exploiting such a fact by using intermediate activity-based manifold representations (Brand 1999; Elgammal and Lee 2004a; Sminchisescu and Jepson 2004; Rahimi et al. 2005; Urtasun et al. 2005; Morariu and Camps 2006; Moon and Pavlovic 2006; Urtasun et al. 2006). In our earlier work (Elgammal and Lee 2004a, 2007), 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 Elgammal and Lee (2004a) is a view-based approach; the manifold was learned for discrete views. In contrast to Elgammal and Lee (2004a), in this paper the manifold of both the configuration and view is learned in a continuous way. In Sminchisescu and Jepson (2004), manifold representations learned from the body configuration space were used to provide constraints for tracking. In both Elgammal and Lee (2004a) and Sminchisescu and Jepson (2004) 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 Urtasun et al. (2006), coupled learning of the representation and dynamics was achieved through introducing Gaussian Process Dynamic Model (Wang et al. 2005) (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. Similarly, in Moon and Pavlovic (2006), Lin et al. (2006), Li and Tian (2007), models that coupled learning dynamics with embedding were introduced.

Manifold-based representations of the motion can be learned from kinematic data, to learn the body-configuration manifold, or from visual data to learn the visual manifold. The former fits generative model-based approaches and provides better dynamic-modeling for tracking, e.g., Sminchisescu and Jepson (2004), Urtasun et al. (2005). Learning motion manifolds from visual data, as in Elgammal and Lee (2004a), Christoudias and Darrell (2005), Morariu and Camps (2006), 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 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 Elgammal and Lee (2004a) or as a way to constrain the solution space for discriminative approaches as in Tian et al. (2005). The representation we introduce in this paper can be used as a generative model for tracking.

Also related to this paper is the research on multilinear models which extends subspace analysis to decompose multiple orthogonal factors using bilinear models and multilinear tensor analysis (Tenenbaum 2000; Vasilescu and Terzopoulos 2002). Tenenbaum (2000) formulated the separation of style and content using a bilinear model framework (Magnus and Neudecker 1988). In Vasilescu and Terzopoulos (2002), 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 (Lathauwer et al. 2000) is used to fit multilinear models. Multilinear tensor analysis was also used in Vasilescu (2002) to factorize human motion styles. The applications of bilinear and multilinear models in Tenenbaum (2000), Vasilescu and Terzopoulos (2002), Vasilescu (2002) to decompose variations into orthogonal factors were performed in the original observation space. In contrast, in Elgammal and Lee (2004b), bilinear and multilinear analysis were used in the space of the mapping functions between a central representation and the observations to decompose variation 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.

3 Framework

We consider two manifolds: (1) the body configuration manifold during motion in the kinematic space, and (2) the visual input (observation) manifold of the same motion observed

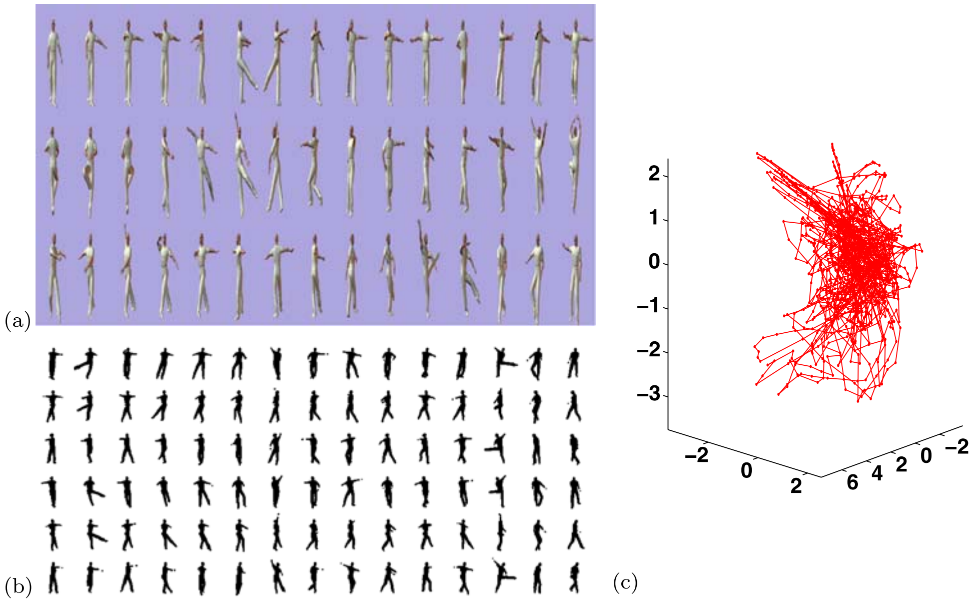


Fig. 1 Example of a complex motion from different views: (**a**) Example postures from a ballet motion. The 8th, 16th, . . . , 360th frames are shown from a sequence. (**b**) Sampled shapes from different views and postures. *Rows*: different views (30°, 90°, . . . , 330°). *Columns*: body postures at frames 25th, 50th, . . . , 375th. (**c**) Visual manifold embedding using LLE, combining both the view and body configuration variations

from different viewpoints along a view circle at a fixed camera height. It is clear that the kinematic manifold can be embedded using nonlinear dimensionality reduction techniques to achieve a low-dimensional representation of the manifold that can be used for tracking. For example, Gaussian Process Dynamic Models (GPDM) (Wang et al. 2005) achieve such embedding in addition to learning a dynamic model for such manifolds. The challenge lies in the visual manifold, since it involves variability in both the body configuration and view. Embedding such a complex manifold will not result in any useful representation that can be used for inferring the configuration and view separately. This can be noticed in Fig. 1(c), where LLE (Roweis and Saul 2000) is used to embed the visual manifold of a ballet motion from different views. Any other nonlinear dimensionality reduction technique can be applied with qualitatively similar results. Here we summarize our approach:

(1) Using joint angle data, we obtain an embedding of the kinematics, which represents the motion manifold invariant to the view. We learn a parameterization of the motion manifold in the embedding space and learn the dynamics by learning a flow field.
(2) Given view-based observation, from different viewpoints, we learn view-based nonlinear mapping functions from the embedded kinematic manifold to the observations in each of the views.
(3) Given the view-based mapping function coefficients, we factorize the view factor arranged as a tensor using higher order singular value decomposition (HOSVD) (Lathauwer et al. 2000).
(4) Given the view factors, we explicitly model the view manifold in the coefficient space, which leads to a representation of the view manifold that is invariant to body configuration.
(5) We also factorize the variability of different people's shapes within the same model.

These procedures result in two low-dimensional embeddings, one for body configuration and one for the view, as well as a generative model that can generate an observation given the two manifolds' parameterizations. This fits perfectly into the Bayesian tracking framework, because it directly provides: (1) a low-dimensional state representation for each of the view and the body configuration, (2) a constrained dynamic model, since the manifolds are modeled explicitly, and (3) an observation model, which comes directly from the generative model used.

4 Learning Configuration and View Manifolds

4.1 Learning View-Invariant Configuration Manifold

As a common representation of the body configuration invariant to viewpoint, we use an embedding of the kinematic manifold. This embedding represents the body configuration in a low-dimensional space. The kinematic manifold embedding is also invariant to different people's shapes and appearances. We can obtain a low-dimensional representation of the kinematic manifold by applying nonlinear dimensionality reduction to motion-captured data using approaches such as LLE (Roweis and Saul 2000), Isomap (Tenenbaum et al. 2000), GPLVM (Lawrence 2004). The choice of the embedding technique is orthogonal to the proposed framework. In particular, without loss of generality, we used LLE in this

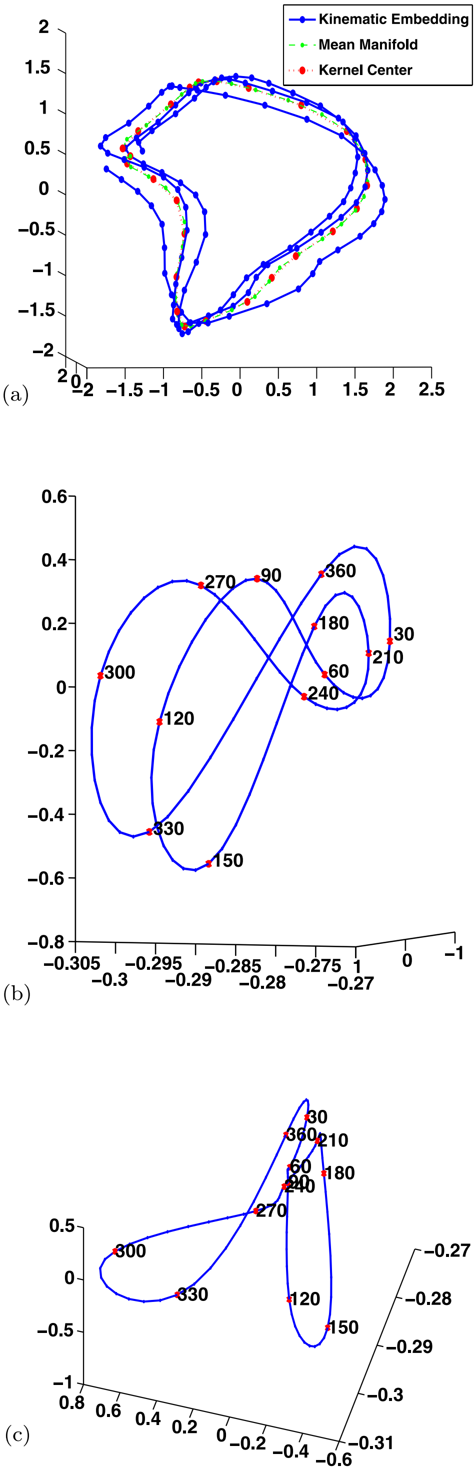


Fig. 2 Configuration and View Manifolds for Gait: (**a**) Embedded kinematic manifold. (**b**), (**c**) Configuration-invariant view manifold (the first three dimensions from different views are shown)

paper. Alternatively, embedding approaches which use temporal information (Lin et al. 2006; Li and Tian 2007) can also be used. Since we need to achieve an embedding of the kinematics, invariant to the person's transformation with respect to the world coordinate system, we represent the kinematics using the body's joint locations in a human-centered coordinate system. We aligned for a global transformation in advance so as to count only motion due to body configuration changes.

Figure 2(a) shows an embedded kinematic manifold for a gait motion (three walking cycles from one person). For a periodic motion like gait, the embedding shows the kinematic manifold as a one-dimensional twisted closed manifold as expected. Gait is fundamentally a one-dimensional manifold (factoring out all other sources of variability). Variations on the motion (or different style) would add different twists to such a manifold (Elgammal and Lee 2004b) and increase embedding dimension. Without counting style variations, the kinematic manifold for the gait motion can be embedded free of intersection in a three-dimensional Cartesian coordinate as in Fig. 2(a). For more complex motions, the manifold is not necessarily one-dimensional. However, we can always achieve an embedding of the kinematic manifold in a low-dimensional Euclidean space. Figures 3(a), (b) show an example embedding for the ballet dance routine data shown in Fig. 1.

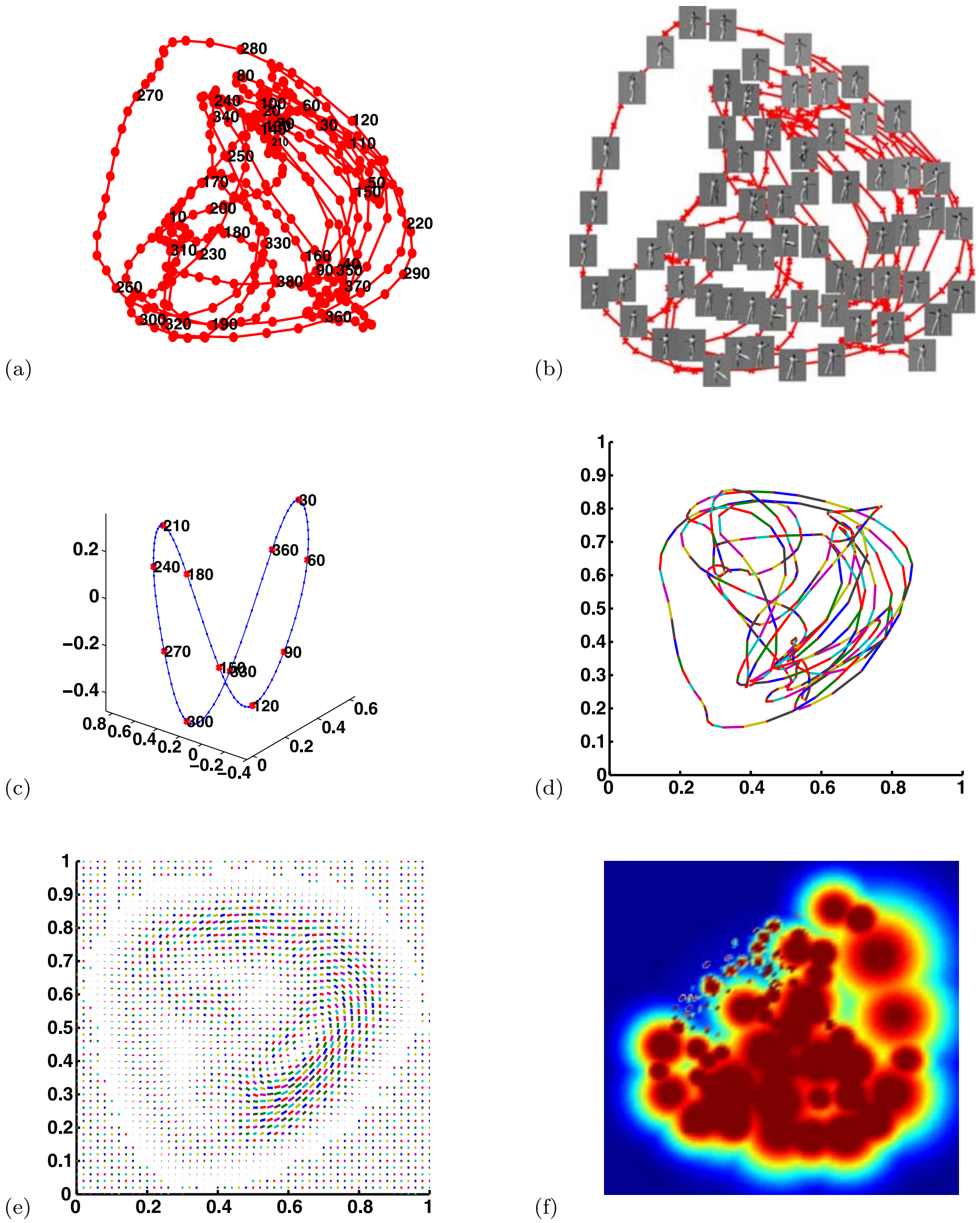
4.2 Learning Posture-Invariant View Manifold

Given an embedding of the kinematic manifold, we can achieve a representation of different views by analyzing the coefficient space of nonlinear mappings between the kinematic manifold embedding and view-dependent observation sequences. Elgammal and Lee (2004b) introduced a framework to separate "style" factors in the space of the coefficients of nonlinear functions that map from a unified "content" manifold and style-dependent observations. In our case, we consider the kinematic manifold embedding as the "content" manifold and the view is considered as a "style" factor. "Style" variations are factorized in the space of the nonlinear mapping coefficients from an embedded manifold to the view-dependent observations. Unlike (Elgammal and Lee 2004b), the view (a style factor) in our case lies on a continuous manifold. Unlike (Elgammal and Lee 2004b), in which the content manifolds were view-dependent, the use of the kinematic manifold in our case provides a view invariant content representation. Differences between the view-dependent observed data will, therefore, be preserved in the nonlinear mapping of each view-dependent input sequence.

Given a set of N body configuration embedding coordinates on the kinematic manifold, $X = \{\boldsymbol{x}_1 \cdots \boldsymbol{x}_N\}$, and their corresponding view-dependent shape observations (silhouettes) $\boldsymbol{Y}^k = \{\boldsymbol{y}_1^k \cdots \boldsymbol{y}_N^k\}$ for each view k where $k = 1, \dots, V$, we can fit view-dependent regularized nonlinear mapping functions in the form of a generalized radial basis function satisfying

$$\boldsymbol{y}_i^k = \boldsymbol{B}^k \psi(\boldsymbol{x}_i), \tag{1}$$

Fig. 3 Configuration and View Manifolds for a ballet Motion: (**a**), (**b**) Embedded kinematic manifold in 2D. (**c**) One-dimensional configuration-invariant view manifold embedding (the first three dimensions are shown). (**d**), (**e**) Velocity field and its interpolation on the configuration manifold. (**f**) Prior probabilistic distribution of body configuration on the kinematic embedding



for each view k. Here, each observation $\boldsymbol{y}$ is represented as a D dimensional vector, and we denote the embedding space dimensionality by e. $\psi(\cdot)$ is an empirical kernel map (Schlkopf and Smola 2002) $\psi_{N_c}(\boldsymbol{x}) : \mathbb{R}^e \to \mathbb{R}^{N_c}$ defined using the N_c kernel functions centered around arbitrary points $\{z_i \in \mathbb{R}^e, i = 1, \dots, N_c\}$ along the kinematic manifold embedding, i.e.,

$$\psi_{N_c}(\boldsymbol{x}) = [\phi(\boldsymbol{x}, z_1), \dots, \phi(\boldsymbol{x}, z_{N_c})]^\top, \quad (2)$$

where $\phi(\cdot,\cdot)$ is a radial basis function (we use a Gaussian function). The coefficient matrices $\boldsymbol{B}^k$ can be obtained by solving a linear system for each view, k, in the form

$$[\boldsymbol{y}_1^k \cdots \boldsymbol{y}_N^k] = \boldsymbol{B}^k[\psi(\boldsymbol{x}_1), \dots, \psi(\boldsymbol{x}_N)].$$

To avoid overfitting to the training data, regularization is needed. Regularizing the RBF mapping in (1) is a standard procedure and can be achieved by adding a regularization term to the diagonal of the matrix $[\psi(\boldsymbol{x}_1), \dots, \psi(\boldsymbol{x}_N)]$ (Poggio and Girosi 1990).

Each $D \times N_c$ matrix $\boldsymbol{B}^k$ is a view-dependent coefficient matrix that encodes the view variability. Given such view-dependent mapping coefficients, we can fit a model in the form

$$\boldsymbol{y}_i^k = \boldsymbol{\mathcal{A}} \times_1 \boldsymbol{v}^k \times_2 \psi(\boldsymbol{x}_i), \quad (3)$$

where $\boldsymbol{\mathcal{A}}$ is a third-order tensor with dimensionality $D \times V \times N_c$ and $\times_j$ is the mode-j tensor multiplication (Lathauwer et al. 2000). This equation represents a generative model for synthesizing an observation vector $\boldsymbol{y}_i^k \in \mathbb{R}^D$ of view k given a view vector $\boldsymbol{v}^k$, and body configuration represented by the embedding coordinate $\boldsymbol{x}_i \in \mathbb{R}^e$ on the kinematic manifold embedding. To fit such a model, the view-dependent coefficient matrices $\boldsymbol{B}^k, k = 1, \dots, V$ are stacked as columns in a $(DN_c) \times V$ matrix $\boldsymbol{C}$, and then the view factors are decomposed by fitting an asymmetric bilinear model (Tenenbaum 2000), i.e., $\boldsymbol{C} = \boldsymbol{A} \cdot [\boldsymbol{v}^1 \cdots \boldsymbol{v}^V]$. The third-order $(D \times V \times N_c)$ tensor $\boldsymbol{\mathcal{A}}$ in (3) is the tensor representation of the matrix $\boldsymbol{A}$, which can be obtained by unstacking its columns.

The resulting representation of the view variations is discrete and high-dimensional. The dimensionality of the view vector in (3) depends on the number of views, i.e., V dimensional. This high-dimensional representation is not desirable as a state representation for tracking. The dimensionality can be reduced when fitting the asymmetric model by decreasing the number of view bases. Figures 2(b), (c) and Fig. 3(c) show the embedded posture-invariant view manifold in the mapping coefficient space for gait and ballet motion, respectively. They clearly show a one-dimensional manifold that preserves the proximity between nearby views. Here, the first three dimensions are shown. The actual view manifold can then be explicitly represented as shown in Sect. 5.1.

4.3 Learning Observation Shape Variability

The model in (3) can be further generalized to model people of different shapes by including a variable for shape style variability between different people. The use of the kinematic manifold provides a representation invariant to observation variability, which allows us to generalize the model. Given the view-dependent shape observations for different people, we can fit view- and person-dependent mapping functions in the form of (1). This yields a set of coefficient matrices $\boldsymbol{B}^{kl}$ for each person l and view k. Given such coefficient matrices, we can fit a generalized model in the form

$$\boldsymbol{y}_i^{kl} = \boldsymbol{\mathcal{D}} \times_1 \boldsymbol{s}^l \times_2 \boldsymbol{v}^k \times_3 \psi(\boldsymbol{x}_i), \tag{4}$$

where $\boldsymbol{\mathcal{D}}$ is a forth-order tensor with dimensionality $D \times S \times V \times N_c$. This equation represents a generative model for synthesizing an observation vector $\boldsymbol{y}_i^{kl} \in \mathbb{R}^D$ of a view k, shape style l, and configuration i given a view vector $\boldsymbol{v}^k \in \mathbb{R}^V$, shape style vector $\boldsymbol{s}^l \in \mathbb{R}^S$, and body configuration represented by an embedding coordinate $\boldsymbol{x}_i \in \mathbb{R}^e$ on the kinematic manifold embedding. Fitting such model can be achieved using HOSVD (Lathauwer et al. 2000; Vasilescu and Terzopoulos 2002).

5 Parameterizations of View and Configuration Manifolds

5.1 Parameterizing the View Manifold

Given the view space defined by the decomposition in (3), different view vectors are expected to lie on a low-dimensional nonlinear manifold. Obviously, a linear combination of view vectors in (3) will not result in valid view vectors. We need to explicitly model the view manifold in the coefficient space to be able to predict and synthesize new views. Therefore, we model view variations as a one-dimensional nonlinear manifold. We employ a one-dimensional continuous variable using spline fitting with third-level parametric continuity ($C2$) constraints between the last and the first sample views, since the view manifold is presumed to be closed. As a result, we represent the view manifold with a one-dimensional view parameter θ and a spline function $g_v : \mathbb{R} \to \mathbb{R}^V$ that maps from the parameter space into the factorized view space. In this representation, a certain view $\boldsymbol{v}_t$ can be represented as $\boldsymbol{v}_t = g_v(\theta_t)$. Figures 2(b), (c) and Fig. 3(c) show a spline-parameterized one-dimensional view manifold embedded in three-dimensional space. Since the training data are sampled at equidistance viewpoints along a view circle, the data are represented using equidistance points in the spline parameter space, i.e., the parameter θ linearly relates to the physical view location. This representation of the view manifold can be directly extended to a two-dimentional parameterization for the case where the view changes along a part of (or a whole) a view sphere.

5.2 Parameterizing the Configuration Manifold

In general, we make no assumptions regarding the dimensionality of the body configuration manifold. However, we discriminate between two cases: (1) the case of a one-dimensional motion, which can be a periodic closed trajectory (e.g., walking or running) or a non-periodic open trajectory (e.g., golf swings or tennis serves), and (2) the case of a general motion where the actual configuration manifold dimensionality is unknown, as in dance or aerobics. In both cases we parameterize the body configuration with a parameter β and a function $g_b(\cdot)$ which maps from the parameter space to the kinematic manifold embedding space.

For one-dimensional motions, the kinematic manifold can be represented using a one-dimensional spline parameter $\beta_t \in \mathbb{R}$ and a spline function $g_b : \mathbb{R} \to \mathbb{R}^e$ that maps from a parameter space into the embedding space and satisfies $\boldsymbol{x}_t = g_b(\beta_t)$. Here $\boldsymbol{x}_t \in \mathbb{R}^e$ denotes the embedding space coordinate, and β_t denotes the parameter at time t. Since the motion is one-dimensional manifold motion, the parameter β_t fully describes the intrinsic body configuration. Using the spline parameter is advantageous over embedding $\boldsymbol{x}_t$ because it produces a constant-speed dynamic model. Equidistance time steps in the training data (frame rate) corresponds to equidistance steps in the spline parameter space, which transforms nonlinearly to variable steps in the embedding space. The parameter β_t will change at a constant speed between frames, whereas the embedding $\boldsymbol{x}_t$ will change in variable steps on the manifold. This can be seen in the results in Fig. 4(f) and Fig. 9(d). Now, we can represent the view and body configuration manifolds using two continuous parameters θ_t and β_t and generate new observations jointly as:

$$\boldsymbol{y}_t^v = \boldsymbol{\mathcal{A}} \times_1 g_v(\theta_t) \times_2 \psi(g_b(\beta_t)). \tag{5}$$

Any combination of view manifold parameter θ_t and body configuration manifold parameter β_t can generate a new image using (5).

For complex motions like aerobics or dance, in which the manifold dimensionality is unknown, a two-dimensional embedding space is used to represent the manifold. In this case, the body configuration parameter space is the same as the kinematic embedding space. To be consistent with the notation used in (5), we denote the body configuration parameter at time t by $\beta_t \in \mathbb{R}^2$. In this case, the function $g_b(\cdot)$, is defined as $g_b : \mathbb{R}^2 \to \mathbb{R}^2$ where $g_b(\beta_t) = \boldsymbol{x}_t$. In such cases, the kernel function centers in (2) are fit to the embedded manifold through fitting a Gaussian mixture model.

To learn the dynamics in such cases, we learn a flow field in the embedding space. Given a sequence of N body configuration embedding coordinates on the kinematic manifold, $X = \{\boldsymbol{x}_1 \cdots \boldsymbol{x}_N\}$, $\boldsymbol{x}_t \in \mathbb{R}^2$, we can directly obtain flow vectors, which represent the velocity in the embedding space, as $v(\boldsymbol{x}_t) = \boldsymbol{x}_t - \boldsymbol{x}_{t-1}$. Given this set of flow vectors, we estimate a smooth flow field over the whole embedding domain, where the flow $v(\boldsymbol{x})$ at any point $\boldsymbol{x}$ in the space can be estimated as

$$v(\boldsymbol{x}) = \sum_{i=1}^{N} b_i k(\boldsymbol{x}, \boldsymbol{x}_i)$$

using Gaussian kernels $k(\cdot,\cdot)$ and linear coefficients b_i that can be obtained by solving a linear system similar to that used to fit (1) (Elgammal and Lee 2007). The smooth flow field is used to estimate how the body configuration will change in the embedding space. This smooth flow field is used in tracking to propagate the particles. Figures 3(d), (e) shows an example of the motion flow field for a ballet dance motion.

5.3 Parameterizing the Shape Space

The shape variable $\boldsymbol{s}$ in (4) can be high-dimensional. To constrain the shapes generated by the model in (4), we represent any shape as a linear convex combination of the shape clusters in the training data. The shape style vector $\boldsymbol{s}_t$ is written as a linear combination of Q shape style vectors $\boldsymbol{s}^q$ in the shape space such that

$$\boldsymbol{s}_t = \sum_{q=1}^{Q} w_t^q \boldsymbol{s}^q, \qquad \sum_{q=1}^{Q} w_t^q = 1, \quad \forall q \; w_t^q > 0.$$

The shape state at time t is denoted by λ_t and represented by the coefficients w_t^q, i.e., $\lambda_t = [w_t^1, \ldots, w_t^Q]^\top$.

Overall, our generative model can be described as

$$\boldsymbol{y}_t = \boldsymbol{\mathcal{D}} \times_1 (\boldsymbol{S}\lambda_t) \times_2 g_v(\theta_t) \times_3 \psi(g_b(\beta_t)), \tag{6}$$

where $\boldsymbol{x}_t$ is the embedded representation of the body configuration, $\boldsymbol{\mathcal{D}}$ is forth-order tensor in (4). The matrix $\boldsymbol{S} = [\mathbf{s}^1, \ldots \mathbf{s}^Q]$ contains the shape style vectors representing the shape style space.

6 Tracking on the Manifold Using Particle Filtering

The Bayesian tracking framework enables recursive update of the posterior $P(\boldsymbol{X}_t|\boldsymbol{Y}^t)$ of the object state $\boldsymbol{X}_t$ given all observations $\boldsymbol{Y}^t = \boldsymbol{Y}_1, \boldsymbol{Y}_2, \ldots, \boldsymbol{Y}_t$ up to time t:

$$P(\boldsymbol{X}_t|\boldsymbol{Y}^t) \propto P(\boldsymbol{Y}_t|\boldsymbol{X}_t) \times \int_{\boldsymbol{X}_{t-1}} P(\boldsymbol{X}_t|\boldsymbol{X}_{t-1}) P(\boldsymbol{X}_{t-1}|\boldsymbol{Y}^{t-1}) d\boldsymbol{X}_{t-1}. \tag{7}$$

We can update the state posterior based on the observation likelihood estimation with the transition probability $P(\boldsymbol{X}_t|\boldsymbol{X}_{t-1})$ (the dynamic model), the previous time step state posterior $P(\boldsymbol{X}_{t-1}|\boldsymbol{Y}^{t-1})$, and the observation (measurement) model $P(\boldsymbol{Y}_t|\boldsymbol{X}_t)$.

The generative models in (6) fits directly to the Bayesian tracking framework to generate observation hypothesis from the state $\boldsymbol{X}_t$. The state is represented by the view parameter θ_t, configuration parameter β_t, and shape parameter λ_t, i.e., $\boldsymbol{X}_t = (\theta_t, \beta_t, \lambda_t)$. We use a particle filter to realize the tracker. Separate particle representations for the view manifold, configuration manifold, and shape space are used. We assume independence of each substate as each substate comes from the decomposition of multiple orthogonal factors by HOSVD. We represent the body configuration with N_β particles, the viewpoint with N_θ particles, and the shape style with N_λ particles.

For a body configuration particle i, view particle j, and style particle k, the observation probability can be computed

$$P(\boldsymbol{y}_t|\theta_t^{(j)}, \beta_t^{(i)}, \lambda_t^{(k)}) = N(\boldsymbol{\mathcal{D}} \times_1 (\boldsymbol{S}\lambda_t^{(k)}) \times_2 g_v(\theta_t^{(j)}) \times_3 \psi(g_b(\beta_t^{(i)})), \boldsymbol{\Sigma}), \tag{8}$$

with observation covariance $\boldsymbol{\Sigma}$ to update the particles' weights. To propagate the particles, we use a flow field to propagate the body configuration particles and a random walk to propagate both the view and shape particles. We evaluate the performance with and without dynamics in Sect. 7.4.1. For one-dimensional motions, we use a constant speed dynamic model, which is directly followed by construction from the spline fitting and leads to superior tracking results.

In actual computation, the nonlinear mapping coefficient matrix between a sample state and a corresponding sample observation depends on the view vector $\boldsymbol{v}_t^{(j)} = g_v(\theta_t^{(j)})$ and the style vector $\boldsymbol{s}_t^{(k)} = \boldsymbol{S}\lambda_t^{(k)}$. For a given view vector and style vector, a mapping coefficient matrix can be computed by partial evaluation of the product $\boldsymbol{\mathcal{D}} \times_1 \boldsymbol{s}_t^{(k)} \times_2 \boldsymbol{v}_t^{(j)}$. Using such a mapping coefficient matrix we can obtain N_β shape hypotheses corresponding to the N_β body configuration particles. Each substate's posterior is evaluated in a sequential manner. For updating the configuration substate posterior, the MAP estimate of the style and view distributions in the previous frame are used, assuming that the style and view change smoothly. Similarly, given a view and body configuration MAP estimates, we can estimate style substate posterior. Similarly, the view substate posterior can be estimated. This procedure reduces the required particle number from $N_\beta \times N_\theta \times N_\lambda$ to $N_\beta + N_\theta + N_\lambda$.

7 Experimental Results

We evaluated the performance of the proposed approach with different types of motions using both synthetic and real data. First we will describe the shape and body configuration representations we used in our experiments. We divide the description of the experiments according to the dimensionality of the motion manifold. Even though the same framework can be applicable to motion manifolds of different dimensionality, the dynamic models and prior probabilistic distribution are more complicated with high-dimensional motion manifolds. We evaluate the performance of the approach in tracking with different settings in Sect. 7.4. In all the experiments, the testing was done using a single uncalibrated camera.

7.1 Shape and Body Configuration Representation

7.1.1 Shape Representation: Implicit Signed-Distance Function

For training the model, we use normalized shapes to represent shapes in a manner invariant to both the distance from the camera (silhouette image size) and the body translation in an observed image space. The extracted foreground shapes are normalized by scaling its vertical axis to a fixed silhouette height and re-centering its horizontal axis. We represent each shape instance as a level-set represented using 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. In particular we used a signed distance function. Such a representation imposes smoothness on the distance between shapes. Given such a representation, the input shapes are points $y \in R^D$ where D is the dimensionality of the input space. We use a 6000-dimensional vector ($D = 6000$) for shape representation from an implicit function of size 100 × 60 representing the normalized silhouettes. The model in (6) generates shapes in an implicit function form with the same dimensionality. The generated shapes can be used to evaluate observations in different formats: (1) the observation can be in the form of background subtracted silhouettes, (2) the observation can be edge fragments extracted using any edge or boundary detectors where a suitable metric can be used to evaluate the observation (we used a probabilistic form of edge-oriented chamfer distance), (3) the model can also be used within a level-set segmentation and tracking framework since the generated silhouettes are in the form of level-sets.

7.1.2 Body Configuration Representation: Body-Centered Coordinate

We represent the body configuration as a set of joints' locations in a body-centered coordinate system. Therefore, the body configuration is invariant to body rotations and body translations. We used a body model containing 23 joints, i.e., the kinematic space is 69 dimensional. Motion-captured data usually fits a 3D model to global marker locations that vary with body transformation. If the motion-captured data has a *root* node representing global transformations (e.g. BVH format), it is easy to achieve a body-centered coordinate representation by simply removing the global transformation (i.e., assigning zero or constant values for global translation and rotation parameters). When we have a global transformation and location for each joint angle without a root node (as in the HUMANEVA dataset), we can achieve a similar representation that is invariant to the global transformation by applying the inverse of the global transformation. We perform this transformation on a node (such as *pelvis*) that can be considered a root node for all other nodes of the given frame.[2]

To evaluate the 3D configuration estimation, the embedded body configuration is mapped to a 3D joint location space by learning an RBF mapping from the embedding space to the joint location space. 3D reconstruction error for a given body configuration is computed from the average

[2] Since we use a body-centered coordinate system, we cannot directly use the evaluation routine supported in the HUMANEVA database.

Table 1 Training and evaluation dataset type

Experiment	Training dataset	Evaluation dataset	Observed shapes
Golf swing	synthetic	synthetic	silhouette
HUMANEVA	synthetic	real	silhouette
Ball passing	synthetic	real	silhouette
Circular tracking (RU)	synthetic	real	edges
Shape adaptive tracking (RU)	real	real	silhouette

Table 2 Average error of normalized 3D body posture estimation from single camera

Subject	Start	End	Duration	Cycle	Mean error
S1	76	534	459	6	26.16 mm
S2	21	436	416	5	37.11 mm
S3	91	438	348	5	40.47 mm
Average			407.6	5.3	32.91 mm

absolute distance between individual markers and the recovered 3D joint location, similar to the Brown HUMANEVA dataset (Sigal 2006).

7.2 Evaluation for One-Dimensional Motion Manifolds

We evaluated the approach using different one-dimensional manifold motions, such as locomotion, golf swings, ball passing, etc. We used both synthesized data and real-data for evaluating the proposed approach in 3D body posture reconstruction from a single camera. The synthesized data facilitate quantitative analysis of the configuration and view estimation. The HUMANEVA dataset provides ground truth data for quantitative evaluation. To fit our model, we need not only ground truth data for 3D joint locations but also visual data from multiple views. Since the HUMANEVA database did not provide visual data sampled along a view circle, we used motion-captured data similar to the evaluation data in the HUMANEVA-I dataset to acquire dynamic shape images from multiple views. We used graphics software (*Poser*®) to render the synthetic input sequences required to train the model. In addition, we captured multiple view locomotion sequences from different people to model the style variations of the visual manifold. Table 1 summarizes the types of data used for training and evaluation.

7.2.1 Brown HUMANEVA Dataset

We tested the 3D body posture estimation accuracy using the Brown HUMANEVA dataset (Sigal 2006), which provides ground truth data for 3D joint locations for different types of motions. We used three circular-trajectory walking sequences, which have continuous view variations with respect to the camera. We normalized the original joint locations in the HUMANEVA dataset into a body-centered coordinate system. We trained the model using synthetic data with 12 discrete views rendered based on our own motion-captured walking sequences. Although 12 discrete views are used for training, the estimation of the view parameter is continuous along the learned one-dimensional view manifold. The evaluation is done using a single camera. We did not fit the model from any of the subjects in the HUMANEVA dataset. For the estimation of the 3D body posture, we selected one cycle from the training sequence to learn the mapping from the embedded kinematic manifold to the 3D kinematic space. Figures 4(e), (f), (d) shows the estimated view, body configuration, and 3D body posture. As can be noticed, the estimated configuration and view parameters fit very well to a constant speed linear dynamic system. Figure 4(g) shows the average error for all joints per frame. The large error around frame 350 corresponds to a frontal view of the subject which is ambiguous for recovering the body posture. The average error of all joints for the three subjects is 32.91 mm. Table 2 shows the average error, frame duration, and number of cycles for each subject. Figure 5 compares the ground truth joint location (blue, *) and estimated joint location (red, O) of the *lower left leg distal*. The tracking is achieved using only 30 particles for estimation of the configuration parameter and 30 particles for estimation of the one-dimensional view parameter.

7.2.2 Golf Swing: One Dimensional Open Manifold

A golf swing is a one-dimensional non-periodic motion. Figure 6(i) shows the embedding of a golf swing kinematic manifold. We collected 12 discrete views of a golf swing sequence (108 frames each) and learned a one-dimensional parameterization of the view manifold (Fig. 6(j)). We tested the performance with a synthetic sequence, which has a continuous constant speed camera motion during the golf swing motion. The estimated view in Fig. 6(e) correctly reflects the

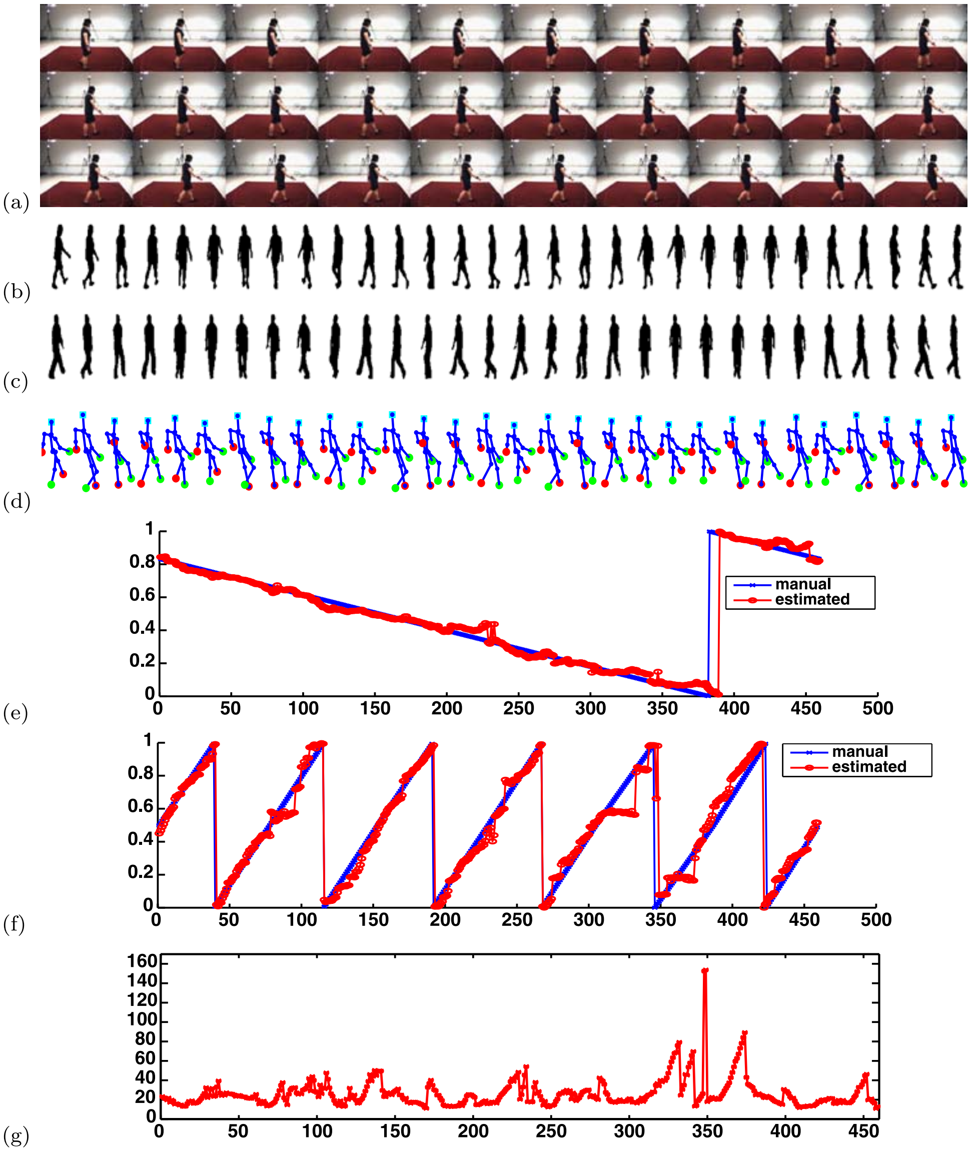


Fig. 4 A walking sequence from HUMANEVA: (**a**) Input raw images. (**b**) Input silhouettes. (**c**) Synthesized silhouettes after view and body configuration estimation. (**d**) Reconstructed 3D postures. (**e**) Estimated view parameter. (**f**) Estimated body configuration parameter. (**g**) Joint location error in each frame (in mm)

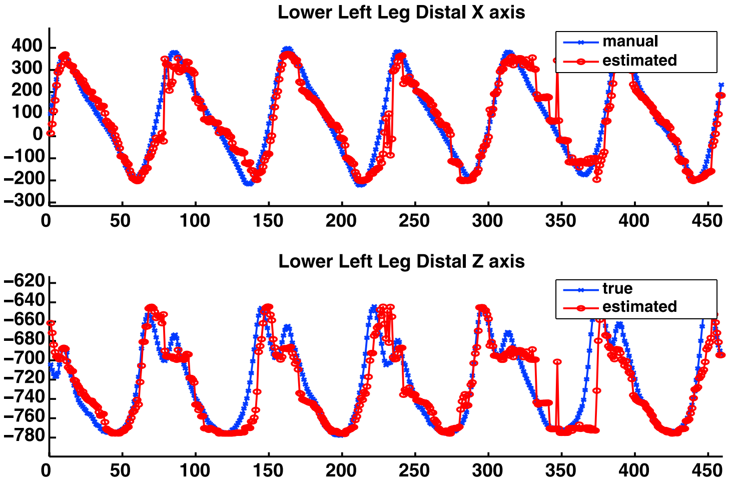


Fig. 5 Evaluation of joint location estimation (HUMANEVA): Estimated joint locations and ground truth for each frame: x and z values for *Lower left leg distal*

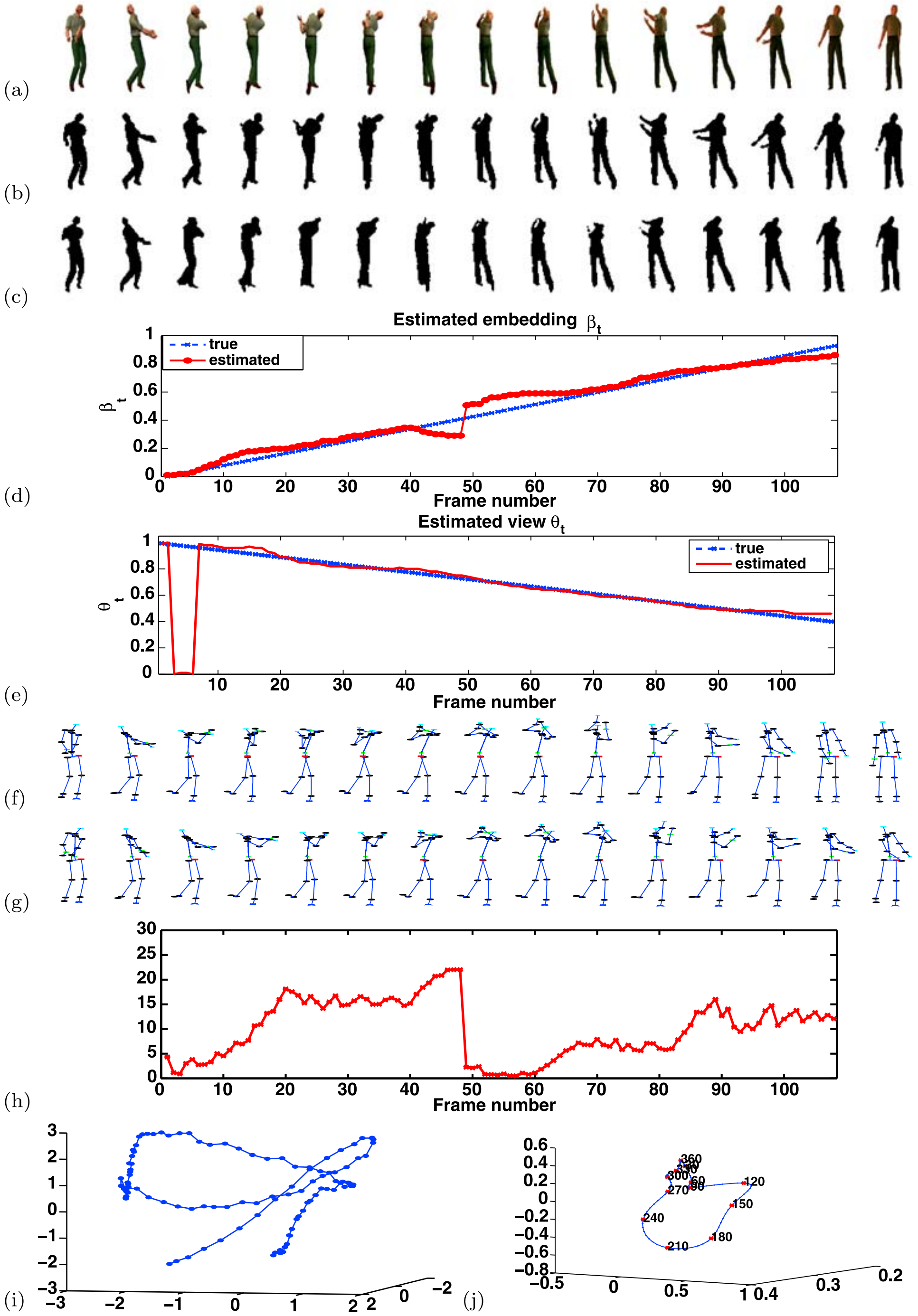


Fig. 6 Golf swing: (**a**) Rendered input for a view variant sequence. (**b**) Input silhouettes. (**c**) Output silhouettes generated based on the estimated view and configuration parameters. (**d**) Estimated body configuration parameter. (**e**) Estimated view parameter. (**f**) True 3D body posture from motion-captured data. (**g**) Reconstructed 3D body posture. (**h**) Errors in estimated 3D body posture (in mm). (**i**) Embedding of the kinematic manifold. (**j**) Configuration invariant view manifold of the motion

constant change of the camera view. Note that only 12 views are learned; all intermediate views were correctly estimated. The offset in the estimated view comes from the additional body rotation in the original motion-captured data, because our body-centered model removed the body rotation. We used 30 particles for tracking the body configuration parameter β and 30 particles for tracking the view parameter θ. In Fig. 6(e), the large dip in the viewpoint estimation in the first few frames is a visualization effect. The view is parameterized from 0 to 1, corresponding to 0 to 2π on a circle, i.e., 0 is the same as 1. The apparent dip is just because we visualize the scale as from 0–1 on a line instead of a circle. Figure 6(h) shows the reconstruction of the 3D body posture from the estimated body configuration parameter. The average error in the estimated 3D body posture is 94.36 mm. We also tested the performance of the 3D body posture estimation during a tilt camera motion, as will be described in Sect. 7.4.

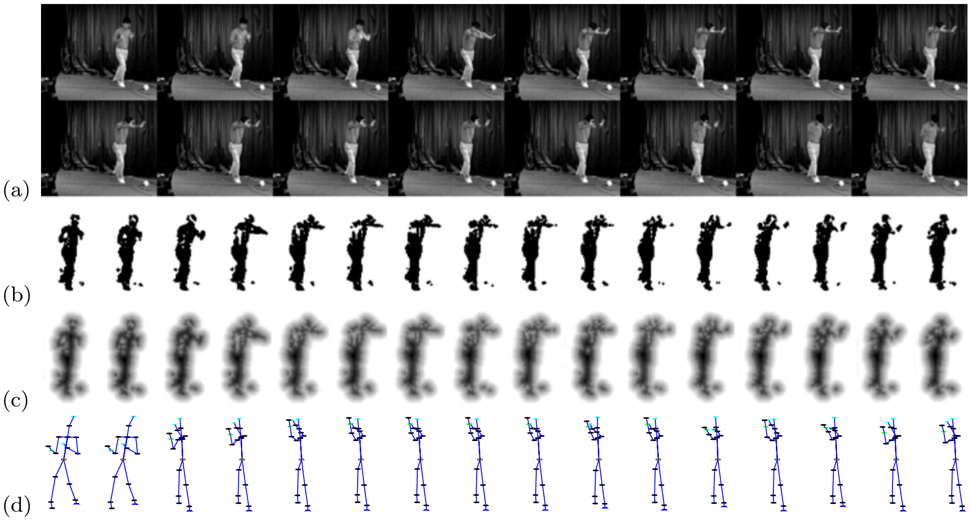


Fig. 7 Basketball pass: (**a**) Captured images (frame number: 6, 12, ..., 96). (**b**), (**c**) Extracted silhouettes and corresponding implicit shape representations. (**d**) Reconstructed 3D body posture based on estimated configuration parameters

7.2.3 *Basketball Pass Motion*

A basketball pass motion, similar to many other simple sport activity primitives, is a one-dimensional manifold motion (when we consider a single cycle). Because there are many camera motions and human body rotations in the arbitrary views of sport video sequences, modeling actions in arbitrary views is crucial for general sport activity tracking and recognition. In our experiment, we fit the model using synthetic data generated from 12 views using motion-captured data. For evaluation, we used real video data from an arbitrary. Sample results are shown in Fig. 7. The proposed approach reliably estimated the change of the body configuration in spite of noisy silhouette inputs as shown in Fig. 7(d). Notice that one of the arms is inside the body silhouette, which makes this example challenging.

7.2.4 *Edge-Based Tracking*

As mentioned earlier, the proposed model generates dynamic shapes in an implicit function representation and can track motion from video data without the need for background subtraction. Instead, edge detector can be used to represent the visual observation. Figure 8 shows an example of tracking using detected edges. Figure 8(b) shows the edges detected using a Canny edge detector. Typically, the detected edges are fragmented, and the outdoor environment generates many additional edges from the cluttered background.

Our model, however, can estimate view and body configuration parameters as shown in Fig. 8(d) with visually acceptable accuracy. In the case of edge-based tracking, since no foreground is segmented, we need to estimate the global transformation. A total of 650 particles were used to estimate global transformation (scale and translation). We estimate the global transformation parameter by Chamfer matching to the normalized shape given the estimated view and body configuration. After selecting the best matching, we further estimated the view and style parameters. Figure 8(e) shows the reconstructed 3D posture from the estimated body configuration parameters.

7.2.5 *Style-Adaptive Tracking*

We captured eight different views of four subjects walking on a treadmill to fit our model with shape style factorization. After extracting foreground silhouettes from one cycle for each subject, we fit the model in (4). We tested the model on outdoor sequences, in which people were walking in S-shaped trajectories. In the shown sequence, the subject walks for nine gait cycles, which were successfully tracked, as shown in Fig. 9. Figure 9(e) shows the estimated view, which exhibits a directional change due to the S-shape walking trajectory. The estimated view parameter decreased from 1 to 0.5 (180 degree counter-clockwise variations), and then it changed back from 0.5 to 1 (180 degree clockwise variations). This variation simulates the actual S-shape walking pattern. The estimated style starts from an average shape style and gradually fits the observed model by a combination of the styles used in training as shown Fig. 9(f).

7.3 Estimation from General Motion Manifolds

Many interesting activities like dancing, aerobics, and sport activities are high-dimensional in their kinematic manifolds. Even simple sport motions like catching and throwing cannot be parameterized by a one-dimensional manifold due to the variability in the body configuration during repeated cycles of the motion. When we catch and throw a ball repeatedly in the air, for example, the catch action changes according to the falling ball location. In this section, we describe

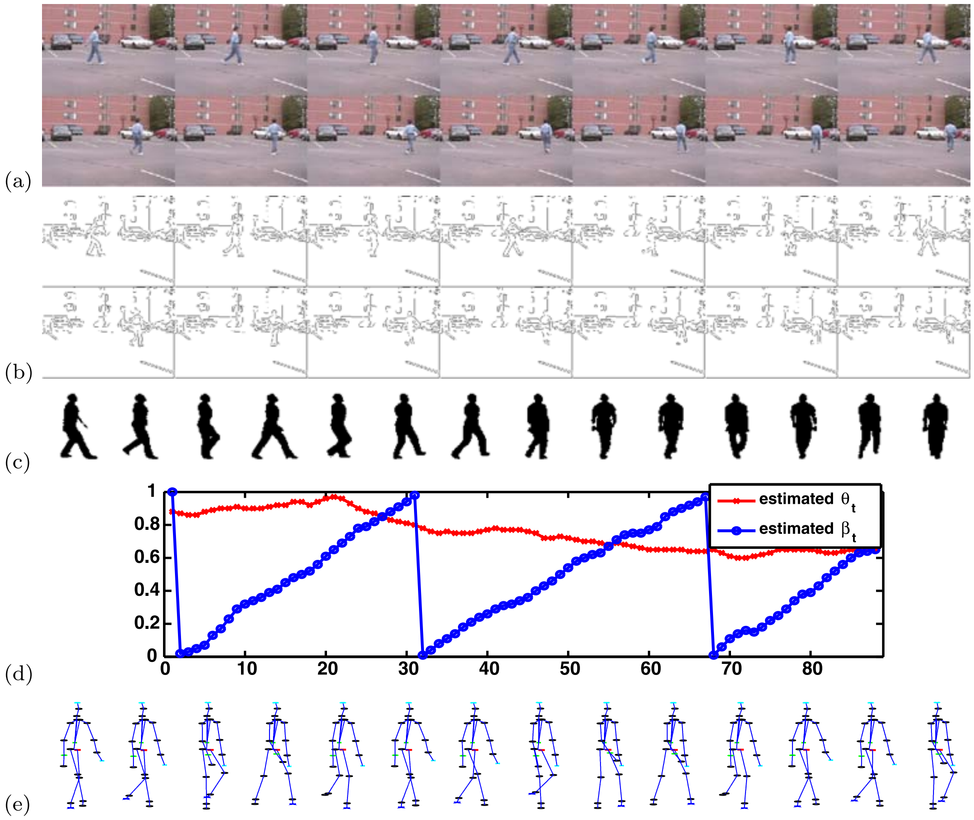


Fig. 8 (Color online) Edge-based tracking: (**a**) Raw input images. (**b**) Input edges. (**c**) Synthesized silhouettes after view and body configuration estimation. (**d**) Estimated body configuration (*blue*, *) and view (*red*, O) parameters. (**e**) Reconstructed 3D postures

experiments on estimating the 3D body posture and view parameters for catch/throw, ballet, and dancing sequences. In each case the model is fitted from synthetic data generated from *Poser*®, and it is tested using other synthetic sequences under different conditions.

7.3.1 Catch/Throw Motion

We used catch and throw sequences with variations of the motion in each catch and throw cycle. These are represented as different trajectories in the body configuration embedding space. We used 90 and 60 particles for configuration and view tracking with a particle filter. Figure 10 shows the results with details in the caption. Figure 10(f) shows the estimated view for the test sequence shown in Fig. 10(b), which exhibits camera motion with a constant speed.

7.3.2 Ballet Motion

Ballet motion exhibits frequent body rotations, and the motion is very complicated since both the arms and legs are moving independently. However, the motion is still constrained by the physical dynamics of the motion. Figure 3 shows the two-dimensional body configuration embedding, flow field, and prior models for a ballet motion. Figures 11(e), (f) show the reconstruction of the 3D body posture based on the estimated body configuration and average errors in each frame. Figure 11(g) shows the estimated view variations and true body rotations in the motion-captured data. Since our model use body centered coordinates that are computed by removing both translation and rotation from the body center (root in the motion-captured data), the body rotation is measured as the variation of view in the opposite direction. The average error in view estimation was 23.1°. This accuracy level reflects good performance considering the fast body rotation and given the ambiguity from a single camera view. Figure 11(h) shows the differences between estimated view variations and actual body rotations in each frame.

7.3.3 Aerobic Dancing Sequence

Many complex motions can be represented by a combination of simple, primitive motions. In particular, contemporary dance sequences can be divided into simple dance steps. Here, we look at a dance sequence that combines two primitive dance steps: left-leg-up and right-leg-up. Two primitive motions are clustered separately in the embedding space, as shown in Figs. 12(a), (b). Left-leg-up is represented by the bottom horizontal cluster and right-leg-up is represented by the diagonal cluster. We used locally linear embedding (LLE) (Roweis and Saul 2000) to learn a two-dimensional embedding for the dancing sequence. Then, we fit view-dependent dynamic shape contour models from 12 synthetic views.

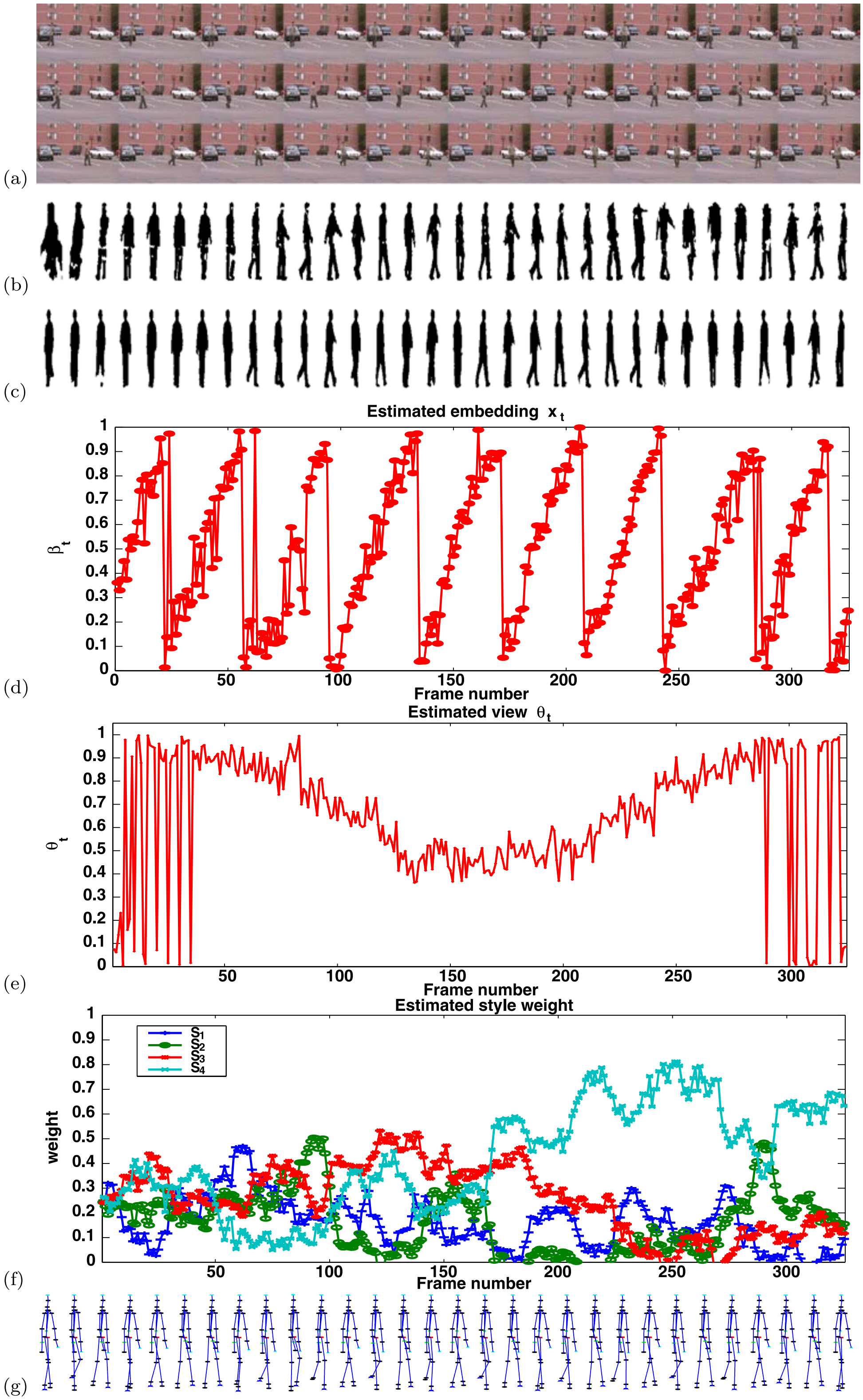


Fig. 9 Tracking with shape variability in addition to view variations: (**a**) Input images. (**b**) Input silhouettes. (**c**) Estimated output silhouettes. (**d**) Estimated body configuration parameter. (**e**) Estimated view parameter. (**f**) Estimated style weights. (**g**) Reconstructed 3D body posture

We tested the performance of the view and body configuration estimations using two types of synthetically rendered data, one with fixed camera and the other with rotating camera. Figures 12(g), (h) show the view and body configuration estimation results for a fixed view. The estimated embedded body configuration switches between the two clusters in the

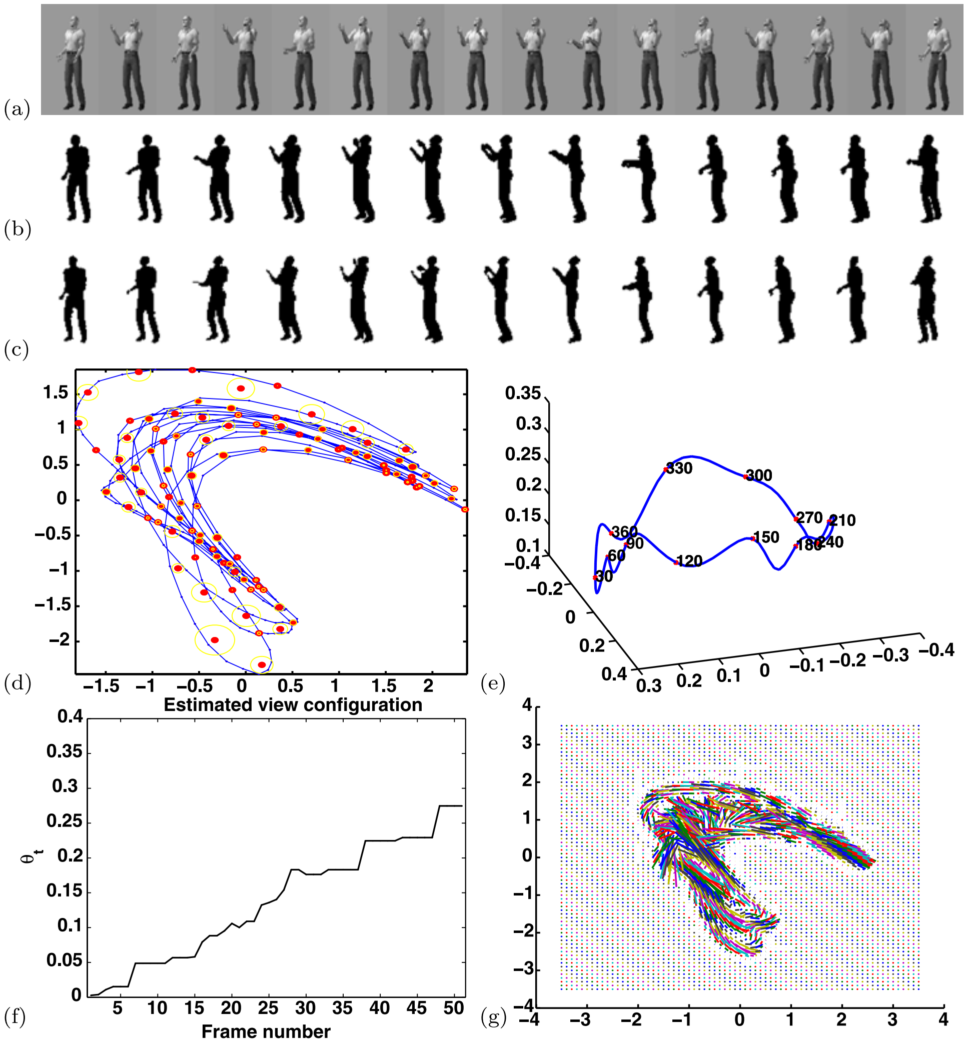


Fig. 10 Catch/throw motion (Evaluation): (**a**) Rendered image sequence (frames 3, 25, 47, 69, ..., 333). (**b**) A test sequence with a moving camera. (**c**) Estimated shapes after view and configuration estimation. (**d**) Two-dimensional configuration manifold embedding and selected basis points. (**e**) Configuration-invariant view manifold in a 3D space. (**f**) Estimated view. (**g**) Motion flow field on the embedding space

embedding space according to the primitive motion type: left-leg-up or right-leg-up. Figure 13 shows the evaluation with view variations from 0° to 90°.

7.4 Comparative Evaluation

This section describes several experiments that show the effect of different choices and parameter settings on the performance. In Sect. 6, we presented the use of the proposed approach for tracking and inferring body configuration within a Bayesian framework as in (7). However, the actual performance will be different for different settings. We tested performance with and without dynamic models; and with and without prior models. In addition, we evaluated the effect of camera tilt motion in addition to camera rotation in Sect. 7.4.2. Finally, we compared the performance of our approach to that of other approaches in Sect. 7.4.3.

7.4.1 Effects of Dynamics and Prior Models

In this experiment, we evaluated the difference of tracking performance with and without prior models and with and without dynamics. By a prior model we mean to use the density of the kinematic embedding as a prior probability distribution in the sampling process in the particle filter. By dynamics, we mean the flow field in the embedding space as described earlier. We selected 64 frames with fast body rotations from a ballet sequence. We used 150 and 60 particles for body configuration and view, respectively. Table 3 shows performance under different conditions. In our experiments, the prior model does not improve performance for body configuration estimation. This is because the prior model is based on a small number of training samples and may over-constrain the searching space for the body configuration. In general, using dynamics improves the results. According to this experiment, the prior distribution did not improve the estimation. The best performance arises when we use a dynamic model without a prior distribution constraints, and it has an average error of 85.88 mm for the joint locations in each frame. Figure 14 shows an example plot of joint locations comparing the ground truth and estimated body location.

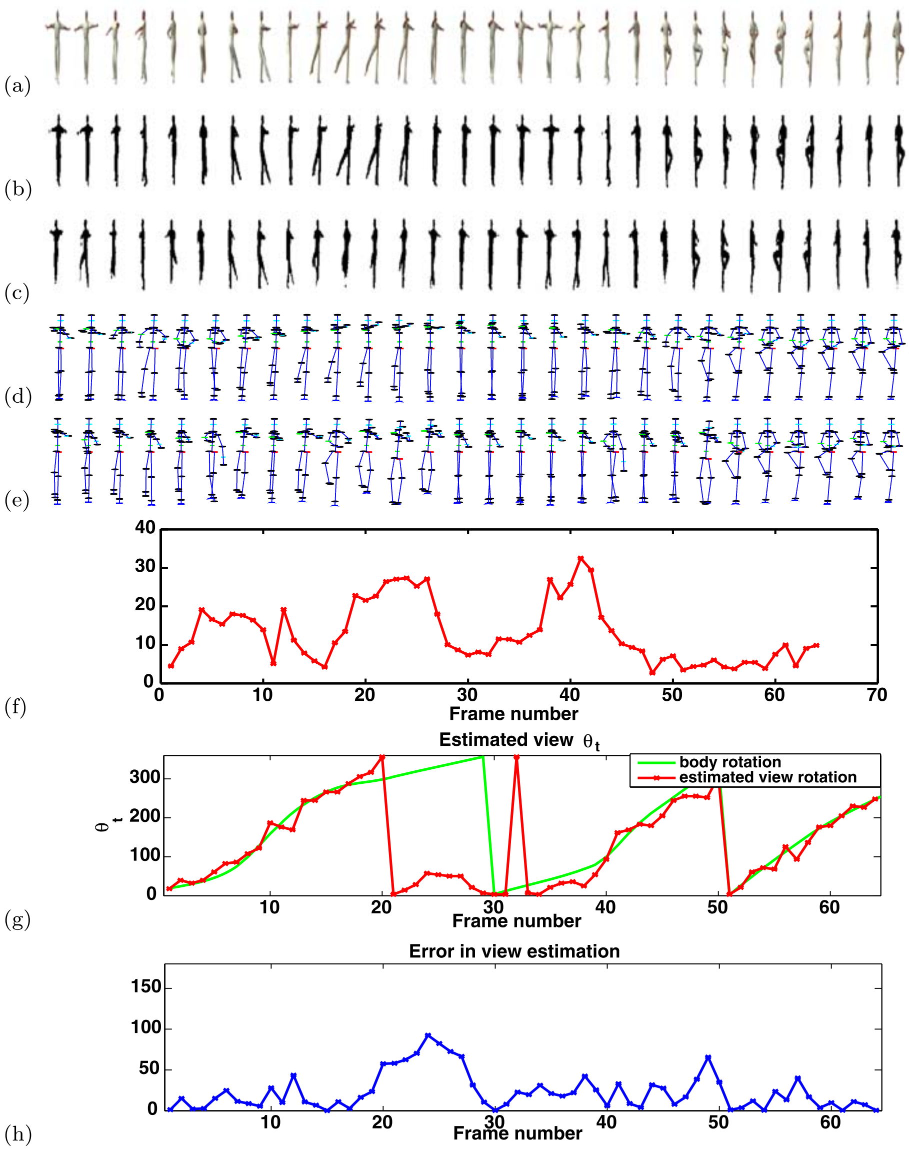


Fig. 11 A ballet motion: (**a**) A test input sequence (rendered). (**b**) A test image sequence (silhouette). (**c**) Estimated silhouette (generated from MAP estimation). (**d**) Ground truth 3D body posture (in body centered coordinates). (**e**) Estimated 3D body posture (generated from the estimated body configuration). (**f**) Average error in the joint location estimation for each frame. (**g**) Ground truth body rotation (from rotation of root in the motion-captured data), estimated view coordinates (Body rotation is measured by view rotation in the opposite direction.), and absolute error between the true and estimated rotation

Table 3 Average errors in posture estimation in the ballet sequence

	Without dynamics	With dynamics
Without prior	87.49	85.88
With prior	161.49	100.77

7.4.2 *Vertical Camera Motion: Robustness to Camera Height*

Our model assumes a one dimensional view manifold and uses sample sequence data along a view circle with a fixed height. In this experiment, we evaluated the robustness of body configuration estimation during change of the view height. We controlled the camera tilt in addition to the camera rotation along the view circle used for training. For this experiment, we choose the golf swing sequence used in Sect. 7.2.2. The original experiment has a 180 degree view rotation during the golf swing. We added tilt camera motion in addition to the continuous camera rotation along the view circle. Table 4 shows the estimated error of the 3D body configuration according to the camera tilt parameter. The average errors increase rapidly after 30 degree camera tilt motion and lost tracking after 40 degree as shown in Table 4 and Fig. 15(d).

7.4.3 *Evaluation of 3D Posture Estimation with Different Approaches*

We evaluated the performance of the proposed approach relative to other approaches for inferring 3D body posture. A nearest-neighbor (NN) search, Bayesian tracking using a torus manifold embedding (Elgammal and Lee 2009),

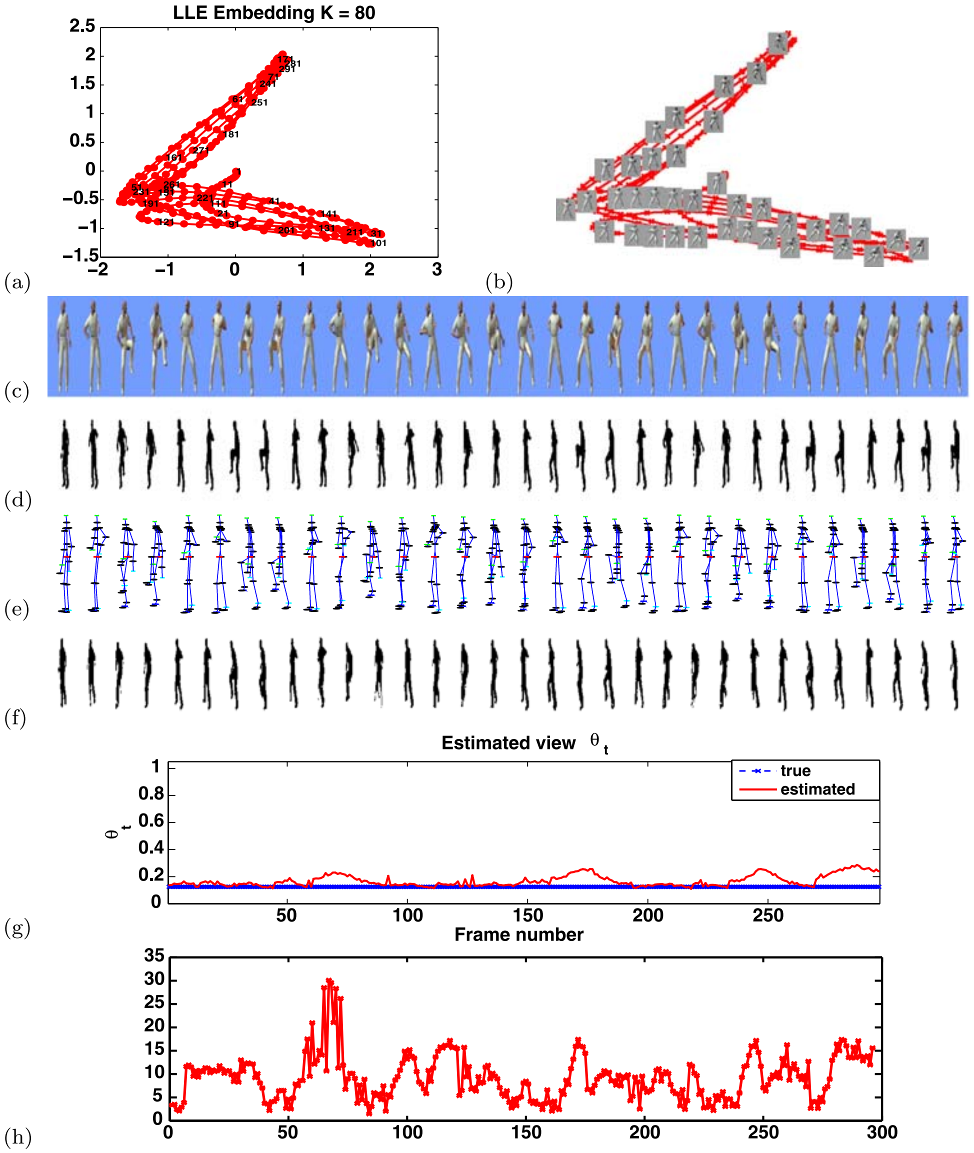


Fig. 12 Dancing sequence evaluation with a fixed view camera: (**a**) Manifold embedding of a dancing sequence. (**b**) Sample body postures on the embedded manifold. (**c**) Input image frames (rendered). (**d**) Input silhouettes for testing from a fixed view. (**e**) Ground truth 3D body posture. (**f**) Reconstructed silhouettes. (**g**) Estimated view parameters. (**h**) Average location error for all joints

direct inferring body posture using a torus manifold embedding (Lee and Elgammal 2006), and embedded representation using a Gaussian process latent variable model (GPLVM) (Lawrence 2004) was used.

Synthetic and real locomotion data were used for the evaluation. For the synthetic data, we generated 12 discrete views of a walking sequence along a view circle. We collected one cycle with 40 frames for each view. For testing, we used synthetic three cycle walking sequences with continuous view variations. For the real data, we used a subject walking sequence from the HUMANEVA database used in Sect. 7.2.1. Subject $S1$ was used in this experiment.

For the case of NN, the 3D posture is directly obtained from the nearest training instance. For the case of GPLVM, we obtained an embedding of the visual manifold from the training data. GPLVM gave an embedding space of the data and a mapping function, which can directly be used to identify the embedding coordinate for any input image silhouette. We used the provided optimization routine to find the embedding points for a given input. For all the cases (except NN), to compare the performance of the 3D posture estimation, we learned a nonlinear mapping from embedding points to the corresponding 3D body posture using an RBF mapping similar to that used in Elgammal and Lee (2004b).

The average error is shown in Table 5. The proposed approach produces better performance than other approaches in real data. NN shows relatively good results in this experiment, because the test data does not have any noise and the training posture and view have dense samples. GPLVM exhibited some problems in this experiment due to ambiguity of the body posture in different views. 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

Fig. 13 Dancing sequence evaluation with a camera rotation: (**a**) Silhouettes for a rotating view. (**b**) Reconstructed silhouettes. (**c**) Estimated view parameters. (**d**) Average location error for all joints

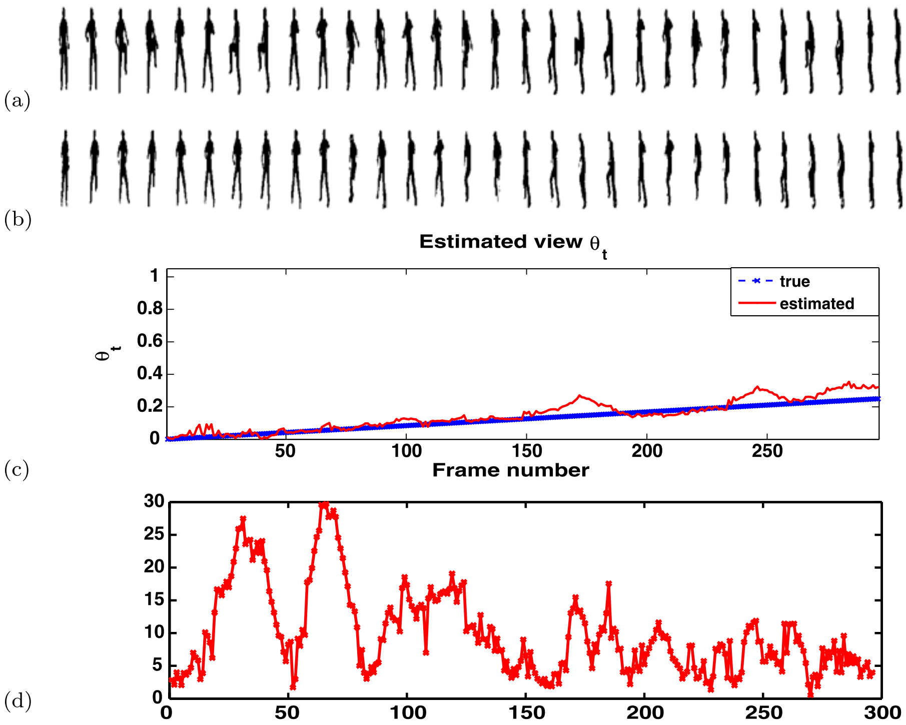


Fig. 14 Evaluation of joint location estimation (Ballet): estimated joint locations and ground truth for each frame: *x* and *z* values for *Left Wrist*

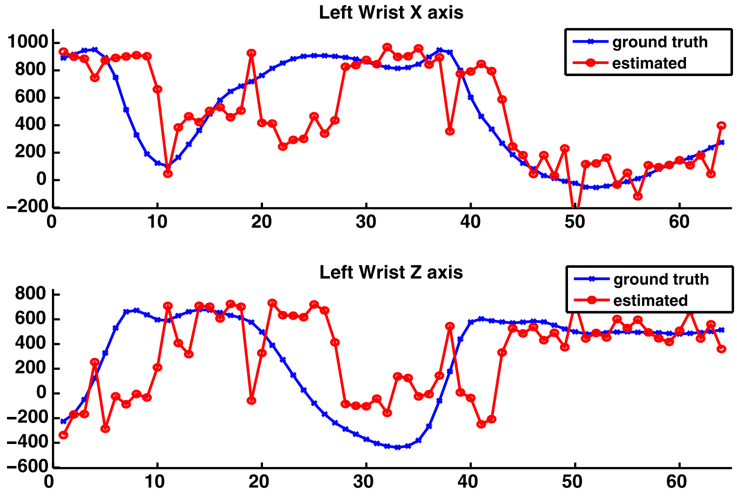


Fig. 15 3D reconstruction with camera tilt motion: (**a**) 3D reconstruction in 10°, (**b**) 30°, (**c**) 50°. (**d**) Average reconstruction errors for all joints

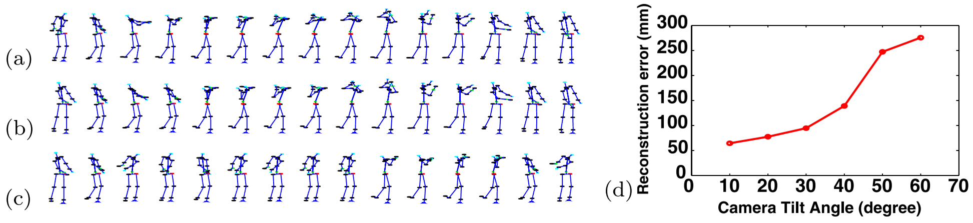


Table 4 Average errors in posture estimation from a tilted camera for a golf swing

Tilt angle (degrees)	10	20	30	40	50	60
3D posture reconstruction error (mm)	64.51	77.56	94.65	139.16	247.32	275.74

Table 5 Average error (in mm) in normalized 3D body posture estimation

Approaches	Proposed	NN	Torus tracking (Elgammal and Lee 2009)	Torus inverse map (Lee and Elgammal 2006)	GPLVM
Synthetic data	20.06	21.84	16.18	62.48	47.24
HUMANEVA data	26.16	48.49	38.21	76.56	81.79

tracking. For example, GPLVM was earlier used (Urtasun et al. 2005, 2006) to embed the kinematic manifold within a model-based approach and was shown to achieve good results.

8 Conclusions

In this paper, we introduced an approach for explicit modeling of body configuration and viewpoint with two separate low-dimensional embedded representations. The body configuration is embedded from kinematic data that is viewpoint invariant. The viewpoint manifold is represented in a posture-invariant manner. As a result, we have created a generative model that parameterizes the motion, view, and shape style. The model is appropriate for tracking and posture estimation of complex motion from uncalibrated stationary or moving cameras. We have provided several sets of experimental results and quantitative evaluations for a wide variety of motions, including simple (gait and golf swings) and complex (aerobics and ballet dancing) motions. Our model can initialize, track, and recover the parameters, which are useful for human motion analysis and recognition, for view and 3D configuration even with a moving camera. The results show a good tracking of both configuration and view when only 30 particles are used for each.

The approach presented here provides a parameterized generative function that generates dynamic shapes of a certain motion from different views and different shape styles where the motion and the viewpoint are parameterized separately. An important feature of the model is that it does not use a 3D model. In training, we used a 3D model to render data, however, in principle this is not required. From perceptual point of view, the model we introduce here, as well as in our previous work (Elgammal and Lee 2004a, 2004b, Lee and Elgammal 2005, 2006), provides a computational theory that is inline with view-based object representation. We can track, recognize the body posture and viewpoint of an articulated object in a continuous manner using a model that is learned from discrete views without the need for a 3D model representation. The limitation of the proposed approach is the requirement of training data from different views, which might be hard to get. In addition, as we did not use 3D model, the accuracy of 3D reconstruction is limited and comes from the interpolation of 3D configuration from trained data in the body-centered coordinate. So, if any body posture is very different from our trained data, we cannot estimate the body pose accurately.

We showed results using simple to complex motions. However, dealing with complex motions is still a challenge. It is hard to obtain an unambiguous embedded representation of complex motion in general. However, we believe that any complex motion can be decomposed into motion primitives which are intrinsically low in dimensionality. Segmenting complex motion into motion primitives is an active research direction that we are pursuing. Given this view, the representation presented in this paper can be useful for modeling more complex motions. Also, complex human motion can be dealt with through hierarchal models where different latent representations for different body joints can be achieved.

One of the ultimate goals of posture estimation research is to be able to build vision systems that can replace the current marker-based motion-captured systems. This paper focuses on investigating the use of manifold structures for both the posture and view estimation from a single uncalibrated camera. The use of an embedded configuration manifold as a constraint on the motion helps achieve an efficient solution. However, such a constraint would limit the accuracy of the posture recovery. Our vision is that the proposed approach (similarly, other manifold-based approaches) can be used as an initialization step to efficiently recover an initial body posture, which then can be used as an initial solution for a more sophisticated model-based nonlinear optimization technique.

The experiments shown in this paper use a single camera to recover the posture, extensions to use multiple cameras is straightforward since the approach separates the configuration space from the view space. For multiple cameras, a shared body configuration and view tracker can be used. The multiple camera geometry would provide an additional constraint on the viewpoint estimation, since the different viewpoints should to be consistent. From the experiment results, it is clear that most of the errors happen because of the inherent ambiguity of the body posture recovery from a single view. Extending the approach to multiple cameras is expected to enhance the accuracy of results by resolving such ambiguities.

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

References

Aggarwal, J. K., & Cai, Q. (1999). Human motion analysis: a review. *Computer Vision and Image Understanding*, *73*(3), 428–440. http://dx.doi.org/10.1006/cviu.1998.0744.

Agarwal, A., & Triggs, B. (2004). 3D human pose from silhuettes by relevance vector regression. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition (CVPR)* (Vol. 2, pp. 882–888).

Brand, M. (1999). Shadow puppetry. In *Proceedings of the international conference on computer vision (ICCV)* (Vol. 2, pp. 1237–1244).

Campbell, L. W., & Bobick, A. F. (1995). Recognition of human body motion using phase space constraints. In *Proceedings of the international conference on computer vision (ICCV)* (p. 624).

Christoudias, C. M., & Darrell, T. (2005). On modelling nonlinear shape-and-texture appearance manifolds. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition (CVPR)* (pp. 1067–1074).

Darrell, T., & Pentland, A. (1993). Space-time gesture. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition (CVPR)* (pp. 335–340).

Elgammal, A., & Lee, C. S. (2004a). 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, pp. 681–688).

Elgammal, A., & Lee, C. S. (2004b). 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, pp. 478–485).

Elgammal, A., & Lee, C. S. (2007). Nonlinear manifold learning for dynamic shape and dynamic appearance. *Computer Vision and Image Understanding*, *106*(1), 31–46.

Elgammal, A., & Lee, C. S. (2009). Tracking people on a torus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(3), 520–538.

Gavrila, D. M. (1999). The visual analysis of human movement: a survey. *Computer Vision and Image Understanding*, *73*(1), 82–98. http://dx.doi.org/10.1006/cviu.1998.0716.

Gavrila, D., & Davis, L. (1996). 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)* (pp. 73–80).

Grauman, K., Shakhnarovich, G., & Darrell, T. (2003). Inferring 3D structure with a statistical image-based shape model. In *Proceedings of the international conference on computer vision (ICCV)* (p. 641).

Hogg, D. (1983). Model-based vision: a program to see a walking person. *Image and Vision Computing*, *1*(1), 5–20.

Kakadiaris, I. A., & Metaxas, D. (1996). 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)* (pp. 81–87).

Lathauwer, L. D., de Moor, B., & Vandewalle, J. (2000). A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, *21*(4), 1253–1278.

Lawrence, N. D. (2004). Gaussian process models for visualisation of high dimensional data. In *Proceedings of advances in neural information processing (NIPS)*.

Lee, C. S., & Elgammal, A. (2005). Homeomorphic manifold analysis: Learning decomposable generative models for human motion analysis. In *Workshop on dynamical vision*.

Lee, C. S., & Elgammal, A. (2006). Simultaneous inference of view and body pose using torus manifolds. In *Proceedings of the international conference on pattern recognition (ICPR)* (pp. 489–494).

Li, R., Tian, T. P., & Sclaroff, S. (2007). Simultaneous learning of nonlinear manifold and dynamic models for high-dimensional time series. In *ICCV 2007* (pp. 1–8).

Lin, R. S., Liu, C. B., Yang, M. H., Ahuja, N., & Levinson, S. (2006). Learning nonlinear manifolds from time series. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 245–256).

Magnus, J. R., & Neudecker, H. (1988). *Matrix differential calculus with applications in statistics and econometrics*. New York: Wiley.

Moeslund, T. B., Hilton, A., & Krüger, V. (2006). A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, *104*(2), 90–126.

Moon, K., & Pavlovic, V. (2006). 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)* (pp. 198–205).

Morariu, V. I., & Camps, O. I. (2006). 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)* (pp. 545–552).

Mori, G., & Malik, J. (2002). Estimating human body configurations using shape context matching. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 666–680).

Murase, H., & Nayar, S. (1995). Visual learning and recognition of 3D objects from appearance. *International Journal of Computer Vision*, *14*(1), 5–24.

O'Rourke, J. (1980). Badler: model-based image analysis of human motion using constraint propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *2*(6), 522–536.

Poggio, T., & Girosi, F. (1990). Networks for approximation and learning. *Proceedings of the IEEE*, *78*(9), 1481–1497.

Rahimi, A., Recht, B., & Darrell, T. (2005). Learning appearance manifolds from video. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition (CVPR)* (Vol. 1, pp. 868–875).

Rehg, J. M., & Kanade, T. (1995). Model-based tracking of self-occluding articulated objects. In *Proceedings of the international conference on computer vision (ICCV)* (pp. 612–617).

Rohr, K. (1994). Towards model-based recognition of human movements in image sequence. *Computer Vision, Graphics, and Image Processing*, *59*(1), 94–115.

Rosales, R., Athitsos, V., & Sclaroff, S. (2001). 3D hand pose reconstruction using specialized mappings. In *Proceedings of the international conference on computer vision (ICCV)* (pp. 378–387).

Roweis, S., & Saul, L. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, *290*(5500), 2323–2326.

Schlkopf, B., & Smola, A. (2002). *Learning with Kernels: support vector machines, regularization, optimization and beyond*. Cambridge: MIT Press.

Shakhnarovich, G., Fisher, J. W., & Darrell, T. (2002). Face recognition from long-term observations. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 851–865).

Shakhnarovich, G., Viola, P., & Darrell, T. (2003). Fast pose estimation with parameter-sensitive hashing. In *Proceedings of the international conference on computer vision (ICCV)* (pp. 750–759).

Sidenbladh, H., Black, M. J., & Fleet, D. J. (2000). Stochastic tracking of 3D human figures using 2d image motion. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 702–718).

Sigal, L., & Black, M. J. (2006). *Humaneva: synchronized video and motion capture dataset for evaluation of articulated human motion* (Technical Report CS-06-08). Brown University.

Sminchisescu, C., & Jepson, A. (2004). Generative modeling of continuous non-linearly embedded visual inference. In *Proceedings of the international conference on machine learning (ICML)* (pp. 140–147).

Sminchisescu, C., Kanaujia, A., Li, Z., & Metaxas, D. N. (2005). Discriminative density propagation for 3D human motion estimation. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition (CVPR)* (pp. 390–397).

Tenenbaum, J. B., & Freeman, W. T. (2000). Separating style and content with bilinear models. *Neural Computation*, *12*, 1247–1283.

Tenenbaum, J., de Silva, V., & Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, *290*(5500), 2319–2323.

Tian, T. P., Li, R., & Sclaroff, S. (2005). Articulated pose estimation in a learned smooth space of feasible solutions. In *Workshop on learning in computer vision and pattern recognition*.

Urtasun, R., Fleet, D. J., Hertzmann, A., & Fua, P. (2005). Priors for people tracking from small training sets. In *Proceedings of the international conference on computer vision (ICCV)* (pp. 403–410).

Urtasun, R., Fleet, D. J., & Fua, P. (2006). 3D people tracking with Gaussian process dynamical models. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition (CVPR)* (pp. 238–245).

Vasilescu, M. A. O. (2002). Human motion signatures: analysis, synthesis, recognition. In *Proceedings of the international conference on pattern recognition (ICPR)* (Vol. 3, pp. 456–460).

Vasilescu, M. A. O., & Terzopoulos, D. (2002). Multilinear analysis of image ensembles: tensorfaces. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 447–460).

Wang, J., Fleet, D. J., & Hertzmann, A. (2005). Gaussian process dynamical models. In *Proceedings of advances in neural information processing (NIPS)*.

Yacoob, Y., & Black, M. J. (1999). Parameterized modeling and recognition of activities. *Computer Vision and Image Understanding*, *73*(2), 232–247.