International Journal of Wavelets, Multiresolution
and Information Processing
Vol. 10, No. 6 (2012) 1250058 (24 pages)
© World Scientific Publishing Company
DOI: 10.1142/S0219691312500580

World Scientific
www.worldscientific.com

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.

BIASED MANIFOLD LEARNING FOR VIEW INVARIANT BODY POSE ESTIMATION[a]

DONGCHEOL HUR* and HEUNG-IL SUK†
Department of Computer Science and Engineering, Korea University
Anam-Dong, Seongbuk-Ku, Seoul 136-713, Korea
**dcheo@image.korea.ac.kr*
†*hisuk@image.korea.ac.kr*

CHRISTIAN WALLRAVEN‡ and SEONG-WHAN LEE§,b
Department of Brain and Cognitive Engineering, Korea University
Anam-Dong, Seongbuk-Ku, Seoul 136-713, Korea
‡*wallraven@korea.ac.kr*
§*swlee@image.korea.ac.kr*

Received 24 February 2012
Revised 23 June 2012
Published 11 December 2012

In human body pose estimation, manifold learning has been considered as a useful method with regard to reducing the dimension of 2D images and 3D body configuration data. Most commonly, body pose is estimated from silhouettes derived from images or image sequences. A major problem in applying manifold estimation to pose estimation is its vulnerability to silhouette variation caused by changes of factors such as viewpoint, person, and distance.

In this paper, we propose a novel approach that combines three separate manifolds for viewpoint, pose, and 3D body configuration focusing on the problem of viewpoint-induced silhouette variation. The biased manifold learning is used to learn these manifolds with appropriately weighted distances. The proposed method requires four mapping functions that are learned by a generalized regression neural network for robustness. Despite the use of only three manifolds, experimental results show that the proposed method can reliably estimate 3D body poses from 2D images with all learned viewpoints.

Keywords: 3D pose estimation; manifold learning; nonlinear dimensionality reduction.

AMS Subject Classification: 68T10, 68Q32

1. Introduction

Reconstructing 3D human body poses from 2D images is one of the most challenging issues in computer vision, because there are many factors that should be considered:

[a]A preliminary partial version of this paper was presented in [10].
[b]Corresponding author.

These include, for example, changes of view and body shape, corrupted background and foreground, and body-part occlusion.[2,18] Since most approaches in this field rely on estimating poses from silhouettes, let us consider a series of 2D silhouette images originating from a sequence depicting a human action.[3] The pixel data in these images exhibit a nonlinear, high-dimensional dynamics that result in a potentially very large search space for pose reconstruction.

In order to overcome the "curse of dimensionality", manifold learning[15] has established itself as one of the core techniques for human body-pose estimation and tracking. Manifold learning allows to explicitly represent the relationship of high-dimensional and nonlinear data (such as the dynamics of 2D images and 3D body configurations) within a *low-dimensional* space.[19] Changes in the silhouette due to other factors than body pose, however, can cause problems for these approaches. One of the major sources of error is the variation in viewpoint as illustrated in Fig. 1. It shows six manifolds that were constructed from image sequences of a walking action on a treadmill observed from six different viewpoints. These manifolds were

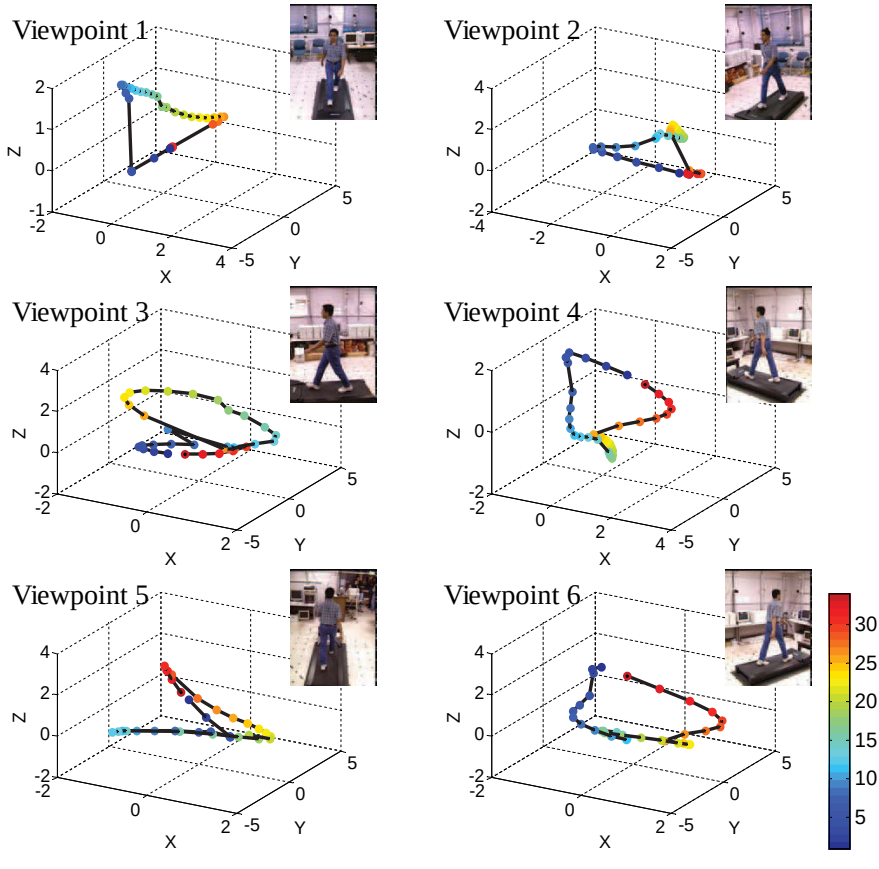


Fig. 1. Six manifolds for a walking action observed from six different viewpoints.

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.

generated from the CMU Mobo dataset[20] from which we took walking data from several individuals and six viewpoints. In order to create a manifold associated with a specific viewpoint, we use all individuals and walking sequences associated with that viewpoint. It is easy to see from Fig. 1 that the manifold distributions are highly variable among different viewpoints — despite the same underlying individual and action. The simultaneous consideration of view and pose variation, therefore, is one of the challenging problems in pose estimation.

Previous methods for solving the problem of viewpoint variation include, for example, building multiple manifolds for all possible viewpoints. Each viewpoint then has its own manifold to represent silhouette variation caused by pose variation. However, learning and indexing into a large number of manifolds are very complex and time-consuming. Another method involves separating view factors from mapping functions using a tensor decomposition method.[12,17] This approach tries to extract the influence of viewpoint changes from coefficients in mapping functions between the 3D body configuration manifold and visual input. However, it does not guarantee a unique mapping between an input image and a corresponding 3D body pose. Without this unique mapping, different postures and viewpoints can be mixed in the manifold space resulting in a critical problem for 3D pose estimation.

In this paper, we propose a novel approach to the problem of viewpoint variation using manifold learning. In order to tackle this problem, three kinds of manifolds are considered that represent view, pose, and body configuration separately: The view manifold represents view variation in 2D silhouettes, the pose manifold represents pose variation in 2D silhouettes, and the body configuration manifold represents variation of body configuration in 3D. In order to learn these manifolds, we employ biased manifold learning[4] that uses modified distances generated from labeled data. The application of the biased distance is the key to the success of our approach as they ensure clear separation of pose and viewpoint variations during the learning stage. In order for robust mapping among feature spaces (input image, view manifold, pose manifold, kinematics manifold, 3D body pose), we also employ a generalized regression neural network to efficiently learn several different mapping functions.[6,16] Finally, the viewpoints are estimated via the view manifold, and 3D body configurations are estimated via the pose and body configuration manifolds.

The remainder of the paper is organized as follows. In Sec. 2, we review related studies on human pose estimation. Section 3 describes the proposed method of modeling pose, view, and body configuration manifolds, and learning the different mapping functions. In Sec. 4, we present the experimental results and analysis on the synthesized data and the CMU Mocap dataset. We conclude the paper in Sec. 5.

2. Related Work

Reconstructing human body poses from 2D images has received a lot of interest in recent decades. A large number of human pose estimation approaches have been model-based utilizing the knowledge of the human skeletal structure. Those

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.

approaches use parameter optimization to infer hidden information of joint angles from image data. In general, such methods can be divided into two categories: top-down and bottom-up.

A top-down approach infers 3D human body poses with kinematic constraints in the human body.[9] Because the total degrees of freedom for a human skeleton is 59, this results in a high-dimensional search space making those approaches rather time-consuming and in general prone to the curse of dimensionality. The bottom-up approaches, meanwhile, try to find body parts in 2D images before inferring whole body poses, for example, by applying a belief propagation algorithm.[5] Such bottom-up approaches are especially vulnerable to occlusion and are computationally expensive in finding the body parts.

Other methods[1,7] have also been introduced, which try to directly infer 3D poses with a learned function from visual inputs and body configuration data. Such approaches have shown great potential in solving the fundamental problem of initialization for model-based approaches, as well as in recovering from tracker failures. However, those methods are exclusively discriminative, learning a transformation function by transforming visual inputs into 3D body configurations or other intermediate representations.

Contrary to those discriminative approaches, manifold learning is a generative method that involves learning a mapping function by transforming a learned low-dimensional manifold representation into a visual input. It is possible to synthesize a visual silhouette and also fits well within a Bayesian tracking framework. However, direct application of manifold learning to the Bayesian framework is problematic since it is "unsupervised".[14] That is, it fails to represent shape, geometric or view variations. To this end, Lee and Elgammal proposed a method of modeling view and posture manifolds to track human body poses.[12,13] Their method constructs view- and pose-invariant body configuration manifolds from body configuration data. After building the body configuration manifold, view factors are extracted from a mapping function using Higher-Order Singular Value Decomposition (HOSVD). After extracting the view factors, the view and pose manifolds are built. Again, however, the method is based on unsupervised learning, resulting in potential mapping problems in the learning stage because different pose silhouettes can be mapped onto the same body configuration.

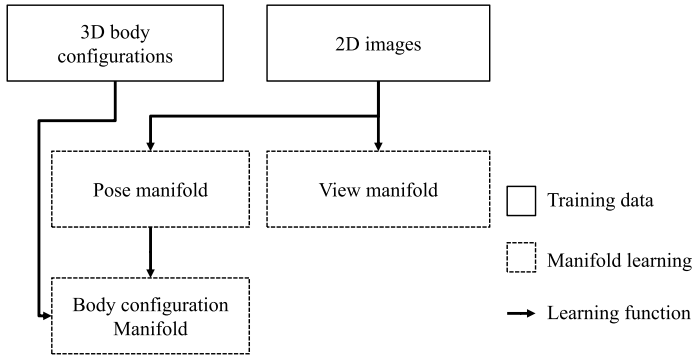
Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.

3. Proposed Method

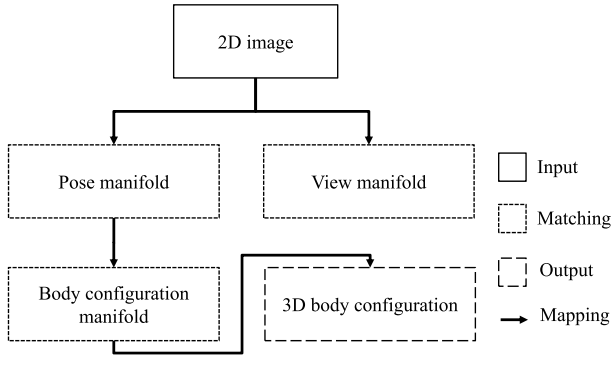
3.1. *Overview*

In this section, we provide a brief overview of the proposed method that can be divided into two phases: training and estimation. In the *training phase*, the pose, view, and body configuration manifolds are learned using biased manifold learning. In addition, the mapping functions among spaces are learned using a generalized regression neural network. In the *estimation phase*, the 3D body configuration is estimated using these learned manifolds and mapping functions.

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.



(a) Training.



(b) Estimation.

Fig. 2. Framework of the proposed method.

3.1.1. *Training*

Figure 2(a) shows the training phase in our approach. Three kinds of manifolds and four mapping functions are learned. For view-invariant body pose estimation, we need to represent view, pose, and body configuration variations. In order to represent a silhouette variation caused by changes of viewpoint in 2D images, we build a view and a pose manifold. The view manifold is constructed in a supervised fashion using 2D silhouette training data and its view label data. This is to ensure that all samples in the view manifold space are sorted with a viewpoint number. For representing the pose and body configuration variations, pose and body configuration manifolds are constructed using 2D silhouette training data, 3D body configuration data, and pose label data.

After building three manifolds, four different mapping functions are learned, namely, Silhouette-To-View manifold (STV), Silhouette-To-Pose manifold (STP), Body configuration-To-Body configuration manifold (BTB), and Pose manifold-To-Body configuration manifold (PTB). We employ a generalized regression neural

network for learning these four mapping functions. Traditional approaches for pose estimation based on manifold learning have mostly used a radial basis function method for learning mapping functions among the original and embedding spaces. A radial basis function method is not suited for our approach, because it assumes that samples are linearly distributed in a local area. In our case, however, the distances among samples are replaced with *biased* distances. Thus, the distance between two samples in the original space and the distance between two samples in the manifold space are different. As a result, the neighborhoods of a point in the original space are different from the neighborhoods of a point in an embedding space. We apply a general regression neural network learning scheme to overcome those problems.

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.

3.1.2. *Estimation*

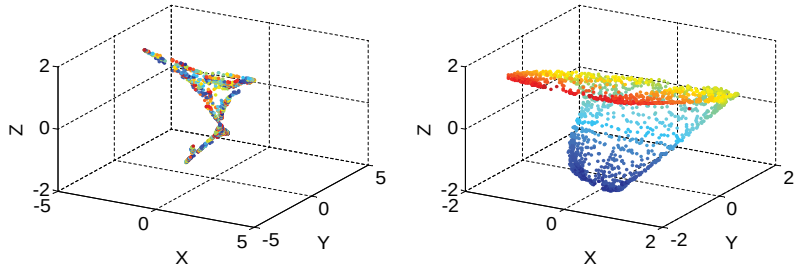
Figure 2(b) shows the estimation phase in our approach. Using the learned three manifolds and four mapping functions, we can estimate 3D body configurations from 2D images with various viewpoints in the following way. Given a 2D image, the corresponding point in the view manifold is estimated using the STV mapping function. At the same time, a corresponding point in the pose manifold is estimated using the STP mapping function. With this point and the PTB mapping function, the corresponding point in the body configuration manifold is estimated. Finally, this point is mapped to a 3D body configuration with the BTB mapping function.

3.2. *Biased manifold learning*

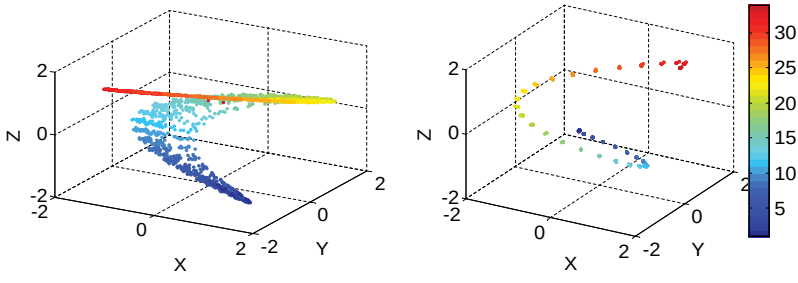
In this paper, we use three kinds of manifolds, namely, pose, view, and body configuration manifolds, where pose and view manifolds are independent to each other. One of the main problems in disentangling the effects of view and pose during learning of the manifolds is to ensure that, indeed, training samples with the same pose or view are closer to each other in the latent space than training samples of different poses or views. As Fig. 1 shows, this is not possible to achieve with standard approaches, which rely on Euclidean distance matrices during learning. In order to overcome this problem, we employ a biased manifold learning method[4] that is a supervised method based on a biased distance matrix extracted from a label matrix. This matrix is used to distinguish within-class and between-class samples to ensure that distances between the former are smaller than between the latter. We can modify the distances among two extrinsic samples by using the following equation,

$$\tilde{D}(i,j) = \lambda_F \times F(i,j) + \lambda_G \times G(i,j), \tag{3.1}$$

where $\tilde{D}(i,j)$ is the biased distance, $F(i,j)$ is a normalizing function for Euclidean distance, $G(i,j)$ is a normalizing function for label distance between samples x_i and x_j, λ_F is the weight of the function $F(\cdot)$, and λ_G is the weight of function $G(\cdot)$. Using λ_F and λ_G, we can control the tradeoff between normalized Euclidean distance and



(a) $\lambda_F = 0.9$, $\lambda_G = 0$ (b) $\lambda_F = 0.5$, $\lambda_G = 0.4$



(c) $\lambda_F = 0.4$, $\lambda_G = 0.5$ (d) $\lambda_F = 0.9$, $\lambda_G = 0.9$

Fig. 3. The distribution of samples in a manifold space.

normalized label distance. Figure 3 shows the change of the distributions according to the values of these two parameters.

The normalizing function $F(\cdot)$, used to transform the sample distribution in Euclidean space, is defined as

$$F(i,j) = \frac{\alpha \times |D(i,j)|}{D_{\max} - D(i,j)}, \tag{3.2}$$

where α is a constant, $D(i,j)$ is the original distance between samples X_i and X_j, and $D_{\max}$ is the largest Euclidean distance in D. The normalizing function $G(\cdot)$ for label distance is defined as

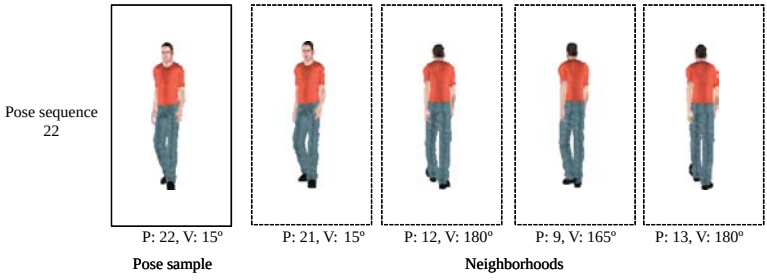
$$G(i,j) = \frac{\beta \times |L(i,j)|}{L_{\max} - L(i,j)}, \tag{3.3}$$

where β is a constant, and $L_{\max}$ is the largest label distance in L. The distance $L(i,j)$ is defined as follows,

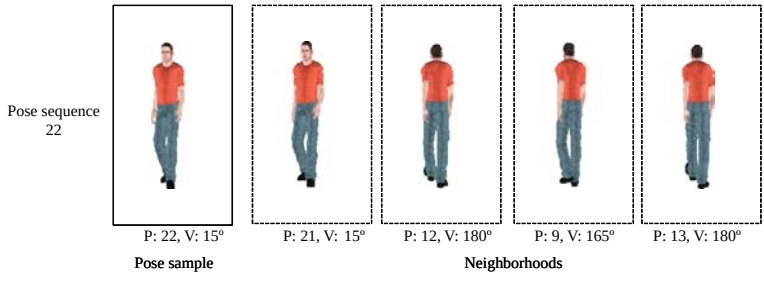
$$L(i,j) = |L_i - L_j|. \tag{3.4}$$

Figures 4(a) and 4(b) show, respectively, the nearest pose and view neighborhoods determined based on the Euclidean distance. It is obvious that it is not possible to obtain ordered samples in this case. In contrast, Figs. 4(c) and 4(d) demonstrate that biased distances ensure proper ordering of the samples in the latent space.

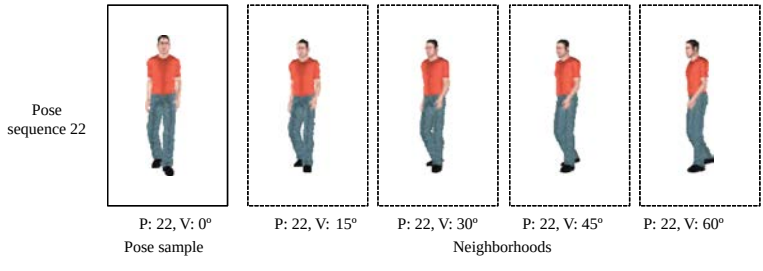
Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.



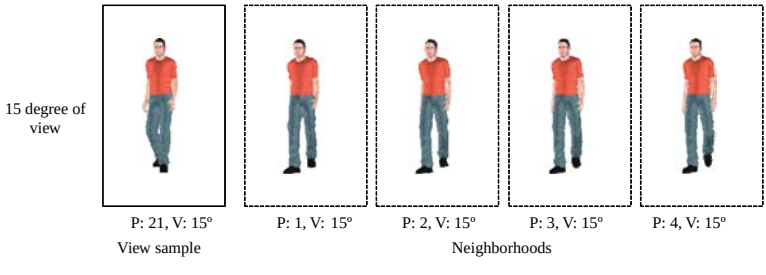
(a) Pose neighbors with Euclidean distance.



(b) View neighbors with Euclidean distance.



(c) Pose neighbors with biased distance.



(d) View neighbors with biased distance.

Fig. 4. Examples of the pose neighbors determined by K-NN with the Euclidean distance and the biased distance.

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.

The view manifold should be invariant to pose variation as image silhouettes are changed easily by view and pose variations. In order to ensure that it is invariant to pose variations and other factors, we model the view manifold using biased manifold learning. Before modeling the manifold, we align samples along with the viewpoint angle. For learning the biased manifold, we again need labeling data to modify the distances appropriately.

In this paper, we model a view manifold from synthesized 2D images and body configurations using Poser 7. We take silhouettes from 24 viewpoints. Silhouettes are grouped with their viewpoint. Figure 5 shows pose samples from one viewpoint.

Similar to the view manifold, we model a pose manifold using a biased manifold learning approach that is invariant to view variations and other factors. Before modeling the manifold, we align samples based on the pose sequence number. We use 30 poses for a walking action. We label the pose data with a sequence number and produce a biased distance matrix. Figure 6 shows viewpoint samples with one pose.

Figure 7 compares Euclidean distance-based manifold learning and biased distance-based manifold learning. We can see that all samples are sorted along viewpoints for the biased view manifold, whereas they are mixed in the Euclidean-based view manifold. Figure 8 compares Euclidean distance-based manifold learning and biased distance-based manifold learning in terms of modeling poses. Similarly to Fig. 7, in Fig. 8(a), all samples in the Euclidean distance-based manifold are mixed, but in Fig. 8(b), all samples in the biased distance-based manifold are sorted along pose numbers.

3.3. *Learning mapping functions*

To estimate the 3D configuration of a human body from a 2D image, we learn four mapping functions: STP Ψ_{ip}, STV Ψ_{iv}, PTB Ψ_{pk}, and BTB Ψ_{bk}. In order to

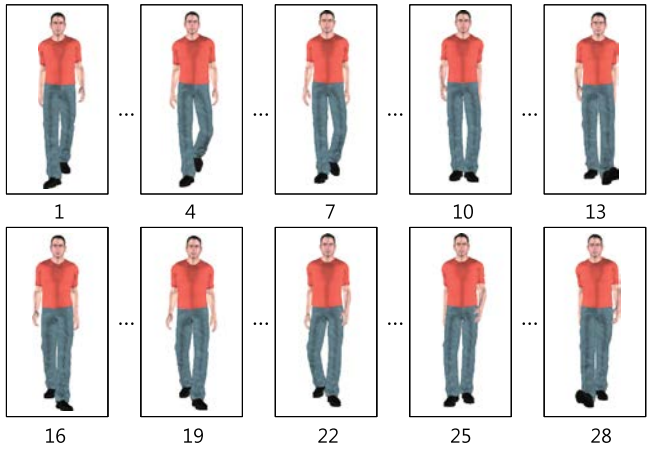


Fig. 5. Pose samples in a viewpoint.

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.

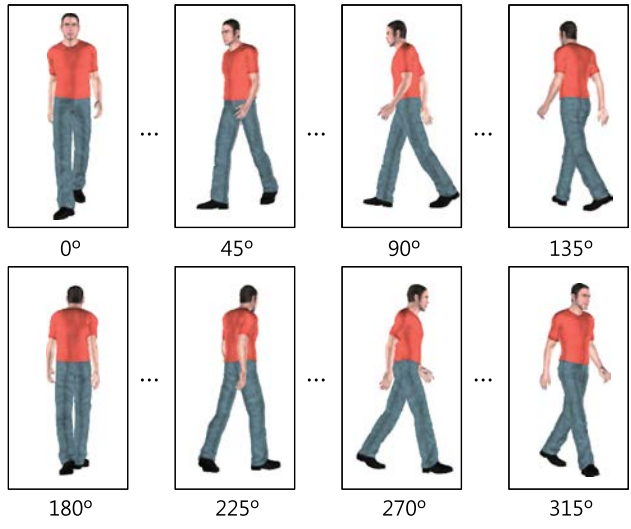
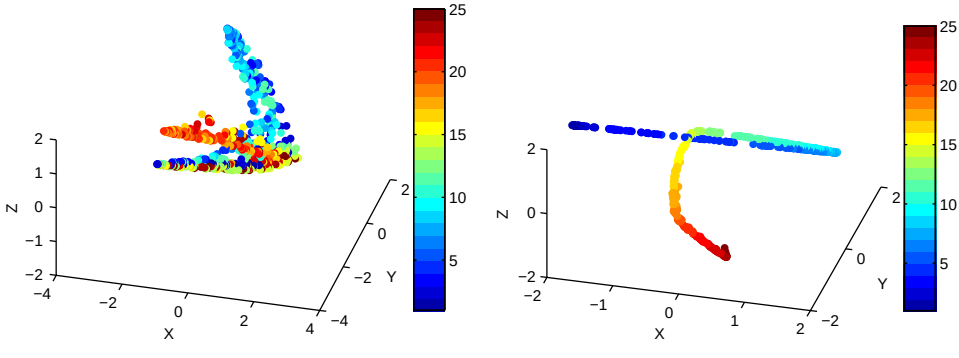


Fig. 6. Viewpoint samples from one pose.



(a) View manifold. (b) Biased view manifold.

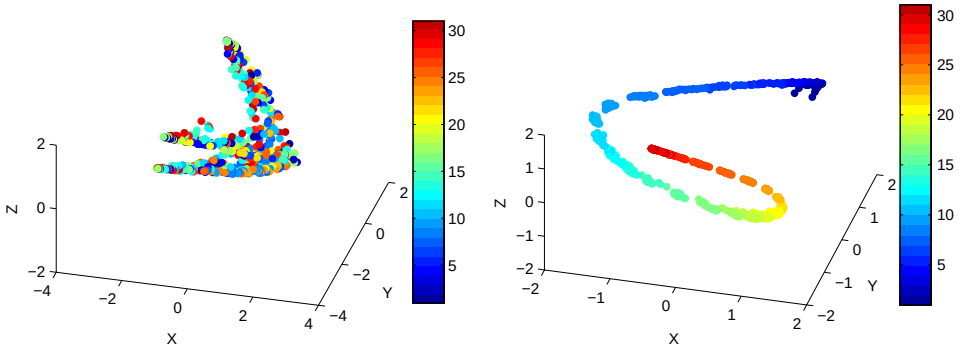
Fig. 7. The comparison of Euclidean distance-based manifold learning and biased distance-based manifold learning in terms of modeling viewpoints.

build these mapping functions, we use a Generalized Regression Neural Network (GRNN).[6,16] The GRNN function is defined by the following equations,

$$E(Y \mid X) = \hat{Y}(X) = \frac{\sum_{i=1}^{n} Y_i \exp\left(-\frac{D_i^2}{2\sigma^2}\right)}{\sum_{i=1}^{n} \exp\left(-\frac{D_i^2}{2\sigma^2}\right)}, \tag{3.5}$$

where $D_i^2 = (X - Xi)^T (X - X_i)$, Y_i is the ith actual output in a training dataset, and σ is a smoothing parameter. The GRNN is trained by the following

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.



(a) Pose manifold. (b) Biased pose manifold.

Fig. 8. Comparison of Euclidean distance-based manifold learning and biased distance-based manifold learning in terms of modeling poses.

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.

error function,

$$\varepsilon = \frac{1}{N}\sum_{i=1}^{N}(\hat{Y}(X_i) - Y(X_i))^2, \tag{3.6}$$

where $Y(X_i)$ is the actual output of ith input vector, $\hat{Y}(X_i)$ is an estimated vector of ith input vector, X_i is ith input vector.

In this paper, we define the GRNN function as follows,

$$E(X) = \Psi(T, X), \tag{3.7}$$

where $E(X)$ is an estimated vector of the GRNN, $\Psi(\cdot,\cdot)$ is the GRNN function, T is a target vector, and X is an input vector.

Given the biased pose embedding points M_p and the visual inputs X_i in the training data, we build a regression function Ψ_{ip} for mapping visual silhouettes to pose embedding points

$$\Psi_{ip} = \Psi(M_p, X_i). \tag{3.8}$$

Given the biased view embedding points M_v, we create a regression function Ψ_{iv} for mapping visual silhouettes X_i to view embedding points as follows:

$$\Psi_{iv} = \Psi(M_v, X_i). \tag{3.9}$$

Given the learned biased body configuration manifold M_k, we build a regression function Ψ_{bk} for mapping body configurations X_b to embedded body configuration points

$$\Psi_{bk} = \Psi(M_k, X_b). \tag{3.10}$$

For inferring 3D human body poses, we use three manifold embedding spaces. We model mapping functions between original and embedding spaces for view, pose,

and body configurations. To this end, we find correspondences between pose embedding points and body configuration embedding points by the following function:

$$\Psi_{pk} = \Psi(M_k, M_p), \tag{3.11}$$

where M_k is the point in the body configuration embedding space, and M_p is the point in the pose embedding space.

3.4. *Estimating* 3D *human body configurations*

After learning manifolds and mapping functions, we can infer 3D human body poses using learned manifolds and mapping functions. In order to find a coordinate in the pose embedding space for an input silhouette, we use an L2 norm,

$$X_{b^*} = \underset{X_b}{\operatorname{argmin}} \|\Psi_{pk}(\Psi_{ip}(X_{i^*})) - \Psi_{bk}(X_b)\|^2, \tag{3.12}$$

where X_{i^*} is an input silhouette, X_b is a body configuration, and X_{b^*} is the estimated 3D body configuration.

4. Experimental Results and Analysis

4.1. *Datasets description*

For our experiments, we use synthesized data derived from Poser 7,[21] as well as real-world video data from the CMU MoBo dataset[8] augmented by 3D body configuration data from the CMU Motion Capture dataset.[20] The synthesized data contains 24 viewpoints from one walking action for a default animated character. The interval between viewpoints is 15° and the length of each sequence is 30 frames. The CMU MoBo dataset contains background images, background subtracted images, and color images from 25 individuals, four action sequences captured from six different viewpoints. The size of an image is 640×480 pixels. From this dataset, we used only 33 frames of one walking action from seven individuals for our experiments.

In order to create more variation in viewpoint, we added two more (virtual) viewpoints using horizontally inverted images from two other viewpoints. Because the CMU Mobo dataset does not contain ground-truth on 3D body configurations, we used a walking sequence from the CMU Motion Capture dataset as ground-truth. The CMU Motion Capture dataset contains 2,605 sequences in six categories and 23 individuals of which we used 3D walking data from one individual as ground truth for our walking sequence.

4.2. *Experimental analysis*

Our experiment can be divided into three steps: (i) Construction of the various manifolds, (ii) embedding visual inputs into low-dimensional spaces, (iii) reconstruction of 3D body poses from 2D visual input.

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.

4.2.1. *Manifolds construction*

In order to train manifolds, we used data from seven individuals. The remaining two individuals' data were used for test. In our experiments, we set the parameters in Eq. (3.1), Eq. (3.2), and Eq. (3.3) as $\lambda_F = 0$, $\lambda_G = 1$, $\alpha = 0.9$, $\beta = 0.9$. Figure 9 shows the results for the construction of the manifolds. Similarly to Figs. 7 and 8 in the previous section, Figs. 9(a) and 9(b) clearly demonstrate that the Euclidean distance-based approach fails on the data, whereas as shown in Figs. 9(c) and 9(d), we were able to get neighborhood-preserving results with the proposed method. Figures 9(e) and 9(f) show the deviations of embedding samples for two manifold learning approaches. Figure 9(e) compares Euclidean pose manifold with biased pose manifold learning. For all pose sequences, biased manifold learning yielded better performance. Figure 9(f) compares the two learning approaches in construction of the view manifold. Overall, the proposed approach resulted in significantly lower errors for all viewpoints. However, for the 135° view, both methods resulted in relatively high errors.

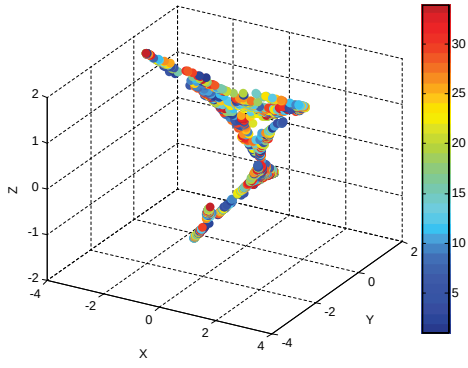
Figures 10(a) and 10(b) show the results of embedding using a Radial Basis Function (RBF) method. Figures 10(c) and 10(d) show the results of embedding using a Generalized Regression Neural Network (GRNN) method. In the figure, the red colored "*" marks are centers of the trained samples and the "+" marks are mapping results. In contrast to the RBF, we could obtain a better mapping result using the GRNN by the tighter clustering of the mapped points around the trained samples. This highlights the fact that GRNN is able to deal better with the non-homogeneous neighborhoods in the biased distance mapping procedure.

4.2.2. *Embedding visual inputs*

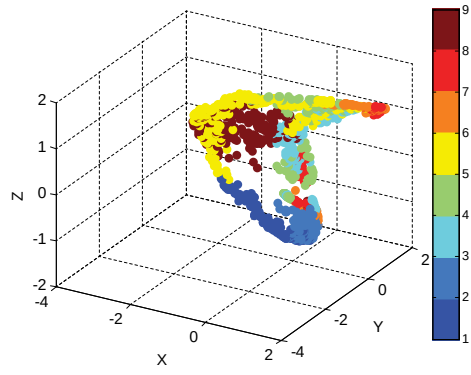
Figure 11 shows the parameter estimation results, and Fig. 12 shows the joint errors of the proposed method on the synthesized images, in Fig. 11, the green lines show the ground truth and the red lines show the estimated parameters in the view manifold. Figure 11(a) shows the view parameter estimation results from 24 viewpoints. Figure 11(b) shows the pose parameter estimation results from 30 poses. We can see that in both cases, our method correctly estimated both the view and the pose of the animated character.

Figure 12 plots the joint error of the estimation where the x-axis denotes the joint number. Here, it seems that some joints show more errors than others: these are the joints located at the back (3, 4), at the neck (5, 6, 7), at the left hand (20), at the right foot (25), and at the left leg (27, 30). Errors at these joints result from the fact that the starting posture (with the right foot located in front and the left hand in the back) and the end posture in the training data are almost the same, which adds ambiguity to the estimate. Nevertheless, Fig. 12 clearly shows that the proposed method is invariant to both view and pose variations.

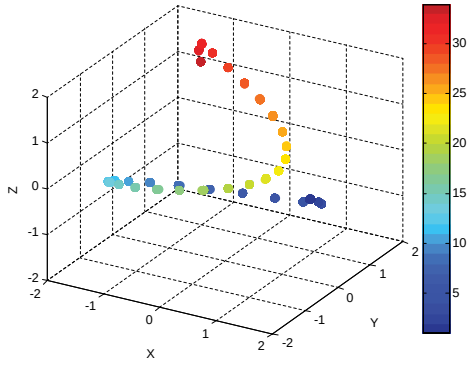
Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.



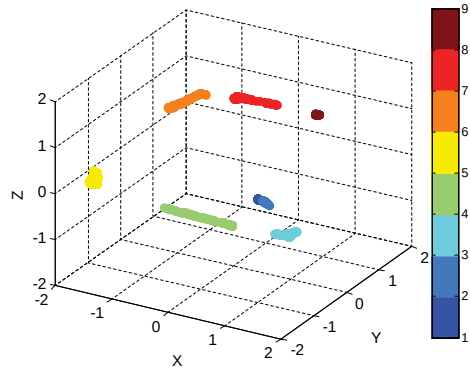
(a) Euclidean pose manifold.



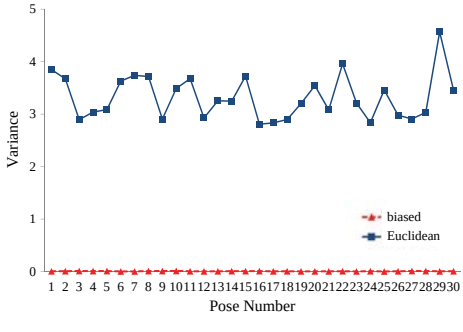
(b) Euclidean view manifold.



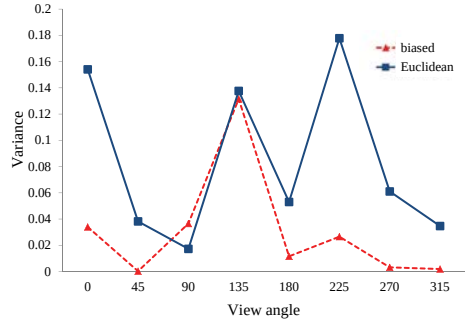
(c) Biased pose manifold.



(d) Biased view manifold.



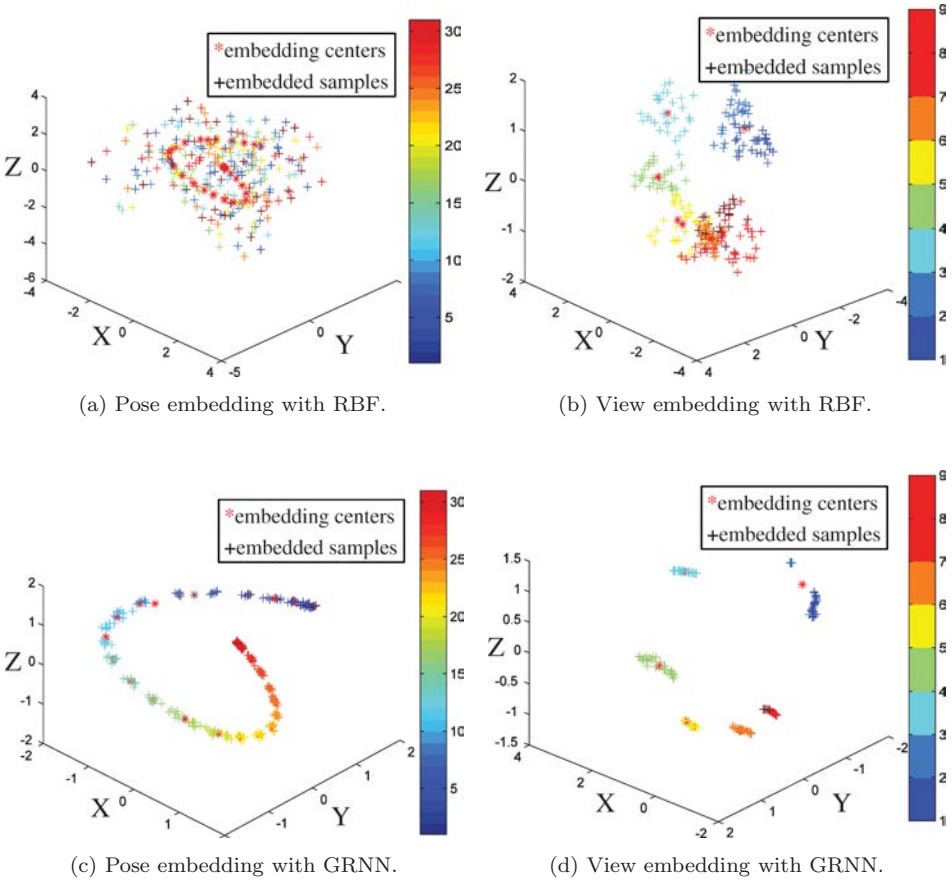
(e) Variance of Euclidean pose manifold and biased pose manifold.



(f) Variance of Euclidean view manifold and biased view manifold.

Fig. 9. Results of manifold construction.

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.



(a) Pose embedding with RBF.

(b) View embedding with RBF.

(c) Pose embedding with GRNN.

(d) View embedding with GRNN.

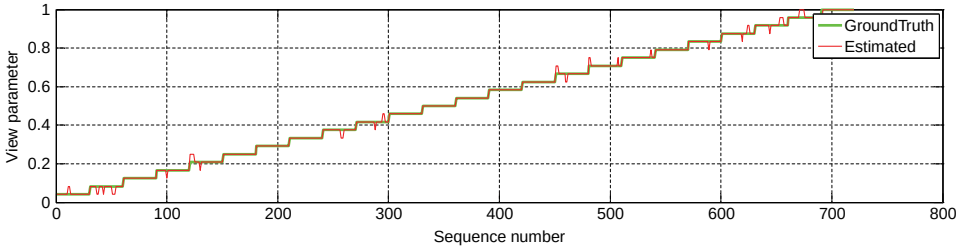
Fig. 10. Results of embedding samples.

4.2.3. *Reconstruction of* 3D *body poses*

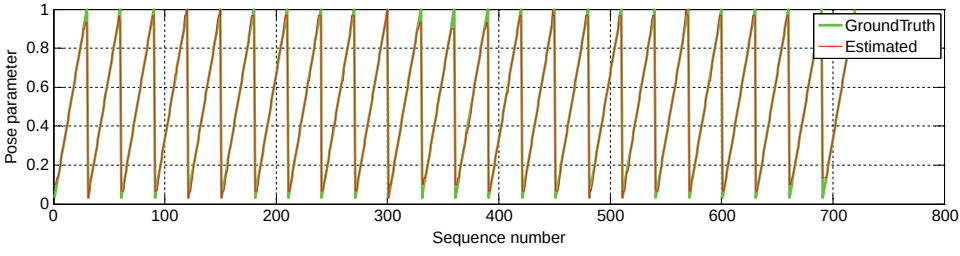
Figures 13, 14 and 15 illustrate the reconstructed poses from input images. The leftmost column shows the synthesized images. The rightmost column shows the estimated 3D body configurations. The second column shows the view embedding results. The third column shows the pose embedding results.

In Fig. 13, we can see the estimation results that include six walking postures. In this case, the viewpoint is fixed. In Fig. 13(b), the corresponding points in the view manifold are located at the same coordinate and the corresponding points in the pose manifold changes with the pose variations.

In Fig. 14, estimation results with six viewpoints are shown. In this case, the pose is fixed. In Fig. 14(c), the corresponding points in the pose manifold are located at the same coordinate and the corresponding points in the view manifold changes with the viewpoint variations.



(a) View estimation in view manifold.



(b) Pose estimation in pose manifold.

Fig. 11. Parameter estimation.

In Fig. 15, results for concurrent view and pose variations are given. In this case, the pose and viewpoint are not fixed. Figures 14(b) and 14(c) show the corresponding points in the pose and view manifolds changing with the pose and viewpoint variations.

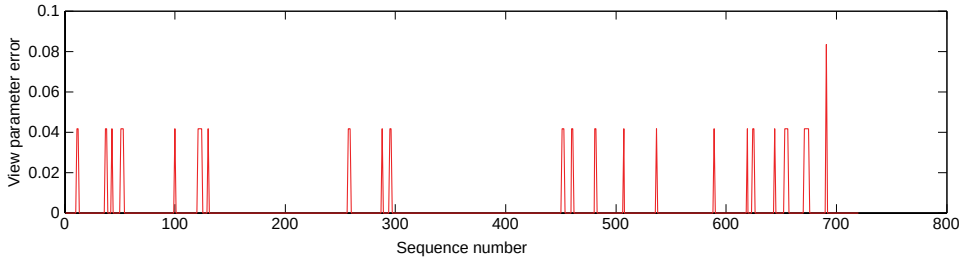
Figures 16 and 17 show the results of the CMU MoBo dataset and the CMU MoCap dataset. The first column contains the input 2D images, the second column shows the corresponding points in the view manifold, the third column shows the corresponding points in the pose manifold, and the fourth column shows the reconstructed 3D body configurations.

In spite of noise in the 2D silhouettes and viewpoint changes, we obtained very robust estimation results. Figure 16 shows the results of the CMU MoBo dataset with a one-cycle pose sequence. Figure 17 shows the results of the CMU MoBo dataset with various viewpoints. In both cases, the independence of the manifolds is well visible with the unvaried factor being reduced to a single point on the manifold.

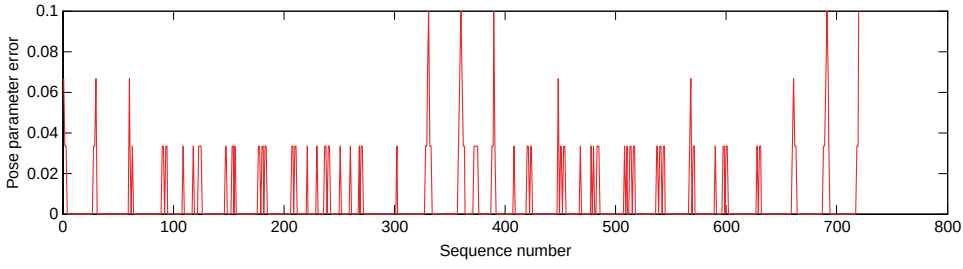
4.3. *Performance evaluation*

We compare the performance of the proposed method with K-Nearest Neighbors (K-NNs) search and with embedding representations using the Gaussian Process Latent Variable Model (GPLVM)[11] on the CMU MoBo dataset. In the case of K-NNs, we can directly obtain the 3D pose from the nearest training instance, whereas with GPLVM, using data embedding, we can directly find the embedding for each

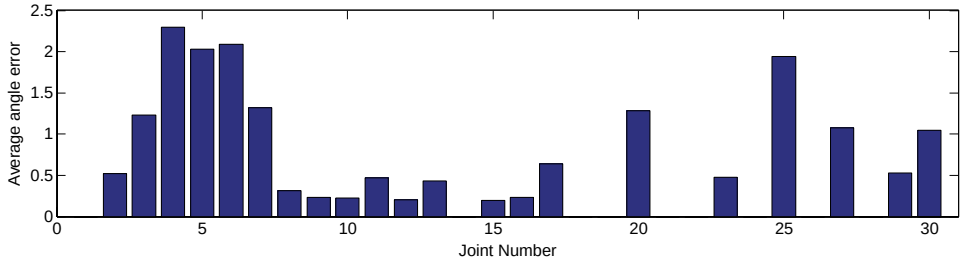
Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.



(a) View parameter error.



(b) Pose parameter error.



(c) Joint error.

Fig. 12. Parameter errors.

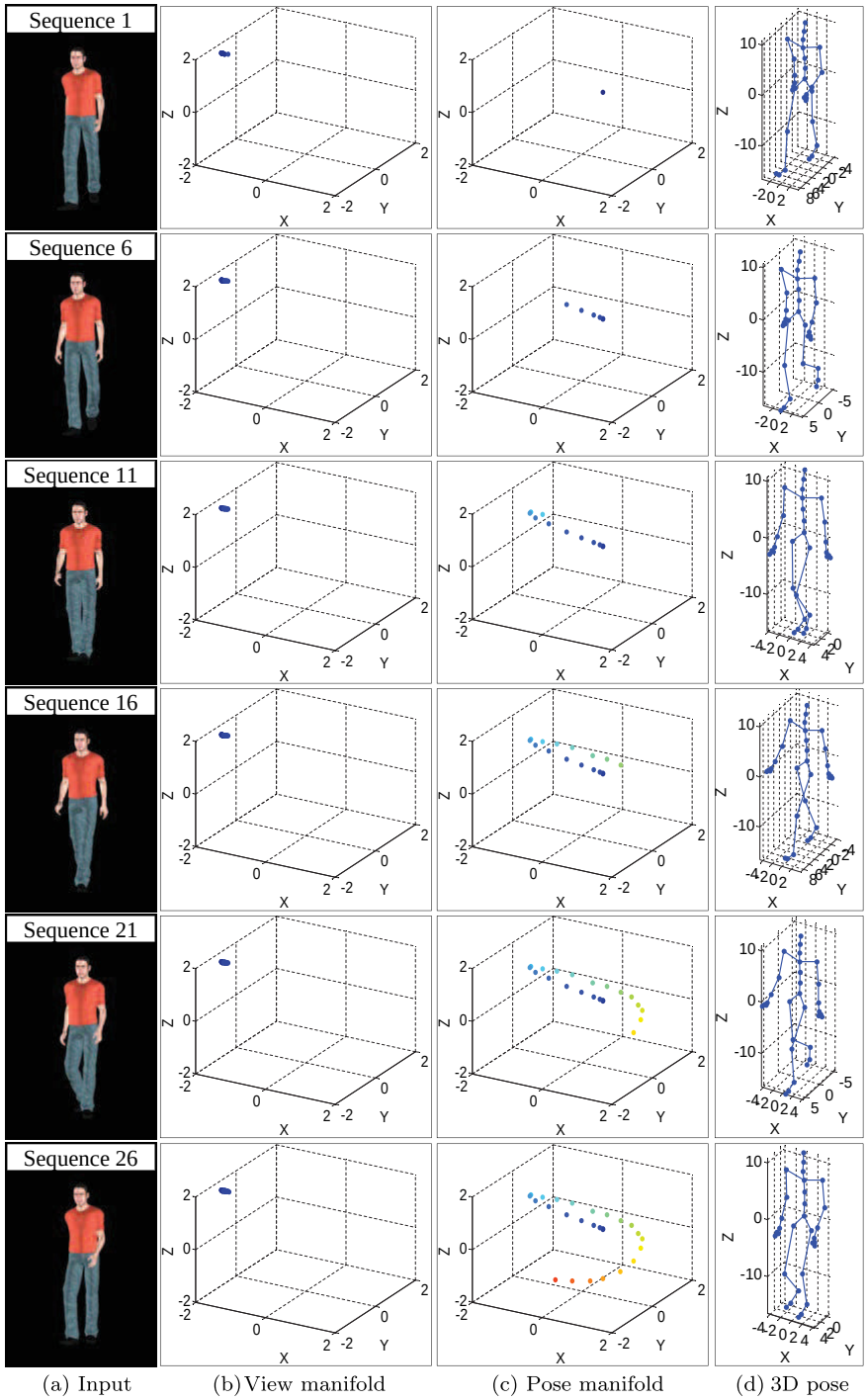
Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.

input image silhouette. To compare the performance, we calculated the average error of all joint coordinates. The average error is calculated by the following equation,

$$\hat{e} = \frac{1}{N \times P} \sum_{i=1}^{N} \sum_{j=1}^{P} \text{norm}(|X_{ij} - Y_{ij}|)$$

where N is the length of a pose sequence, P is the number of joints in the body configuration, X_{ij} is an angle of a joint in the body configuration, and Y_{ij} is the ground truth joint coordinate. The propose method and the nearest neighborhood method presented similar performance as shown in Table 1, since they use the same distance metric. There are some differences, however: the K-NNs algorithm decides based on votes from several posture, whereas in our approach, a nearest embedding point from a projected point is taken as the result. Hence, if some projected point

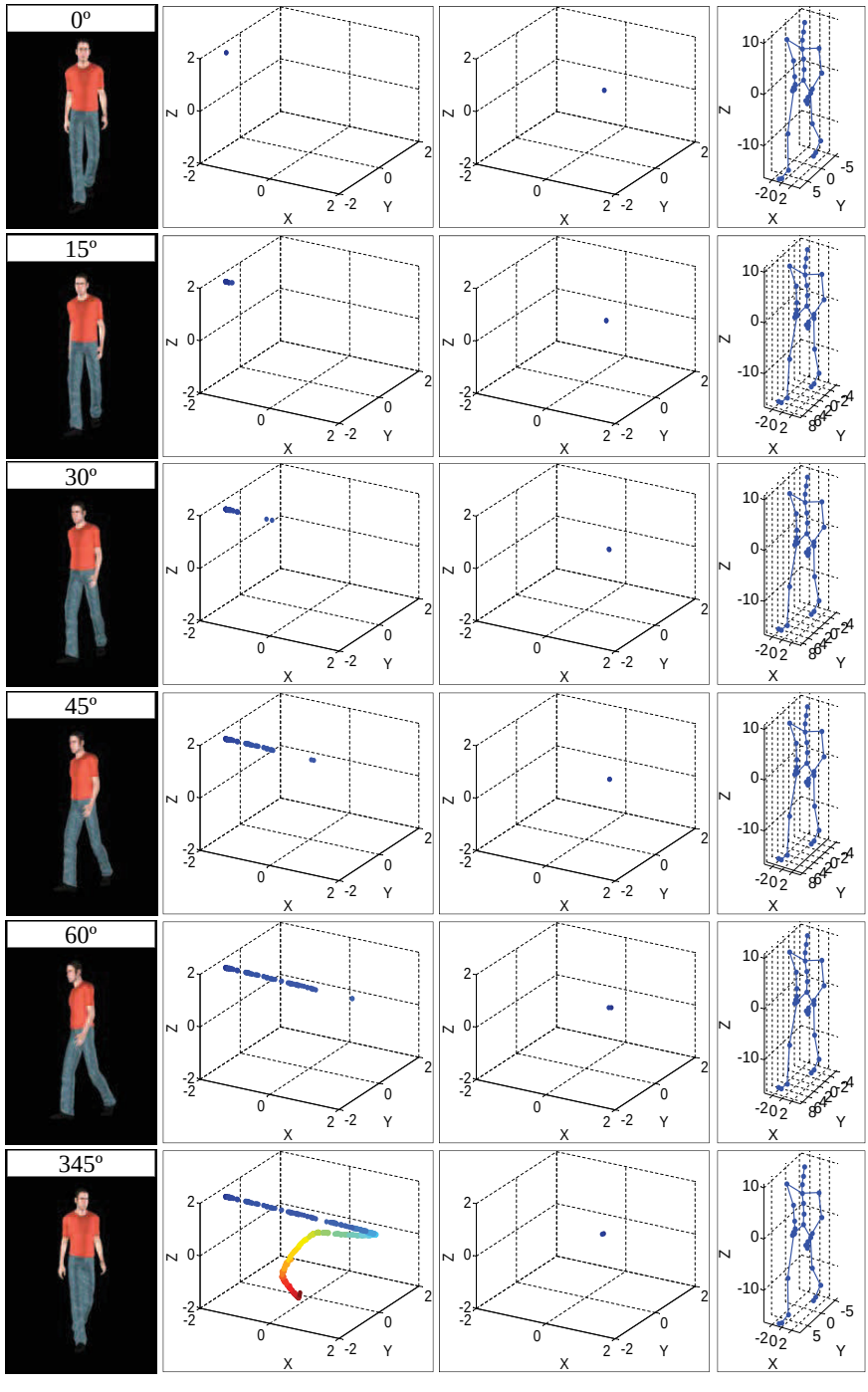
Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.



(a) Input (b) View manifold (c) Pose manifold (d) 3D pose

Fig. 13. Poser walking sequences: pose estimation results with various postures.

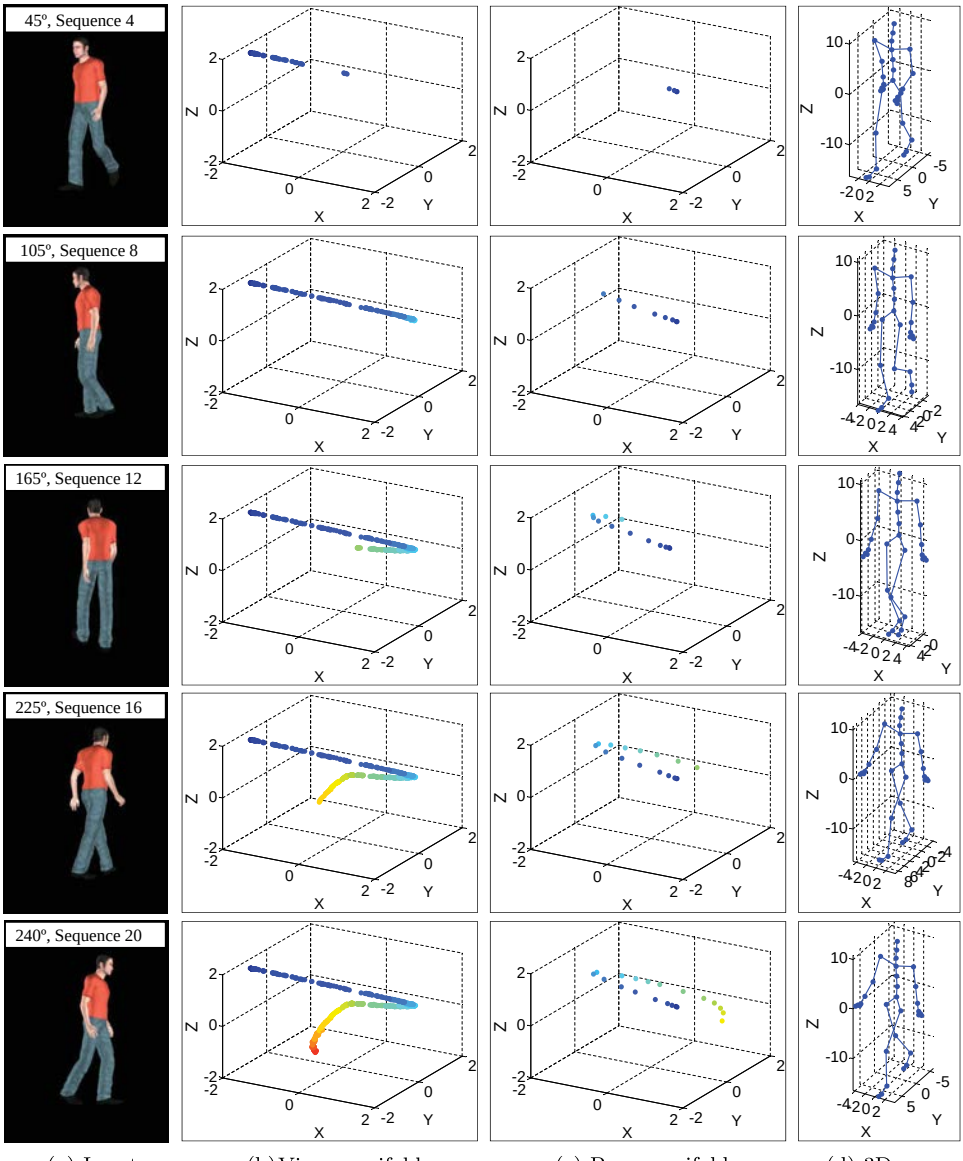
Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.



(a) Input (b) View manifold (c) Pose manifold (d) 3D pose

Fig. 14. Poser walking sequences: pose estimation results with various viewpoints.

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.



(a) Input (b) View manifold (c) Pose manifold (d) 3D pose

Fig. 15. Poser walking sequences: pose estimation results under both pose and view variations.

is located at the boundary of two distributions of two postures, K-NNs may take another neighborhood posture as a result.

In manifold learning, the size of the neighborhood and the number of samples are the main factors influencing execution time. Table 2 compares the execution time between biased manifold learning and Euclidean manifold learning on 1,848

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.

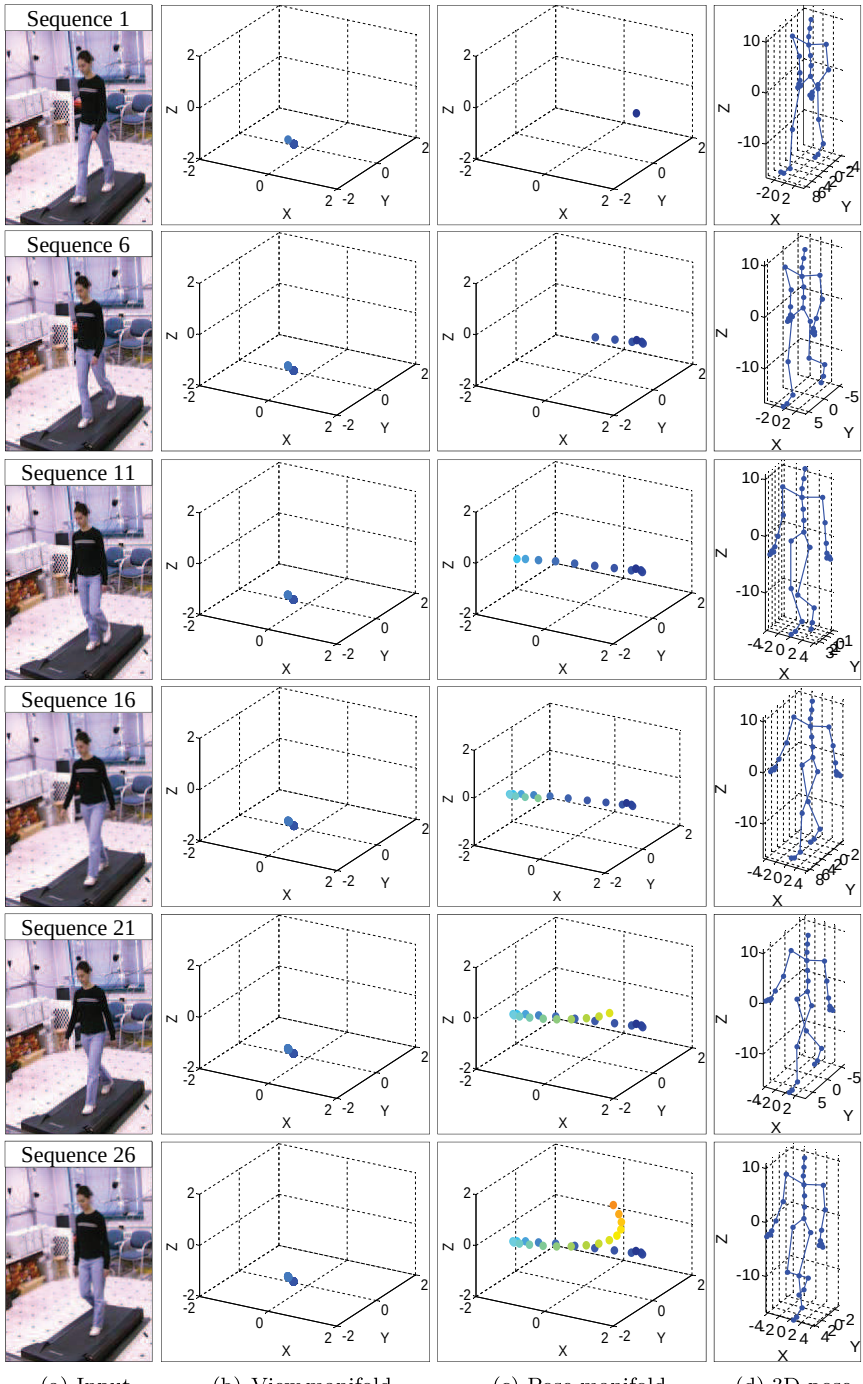
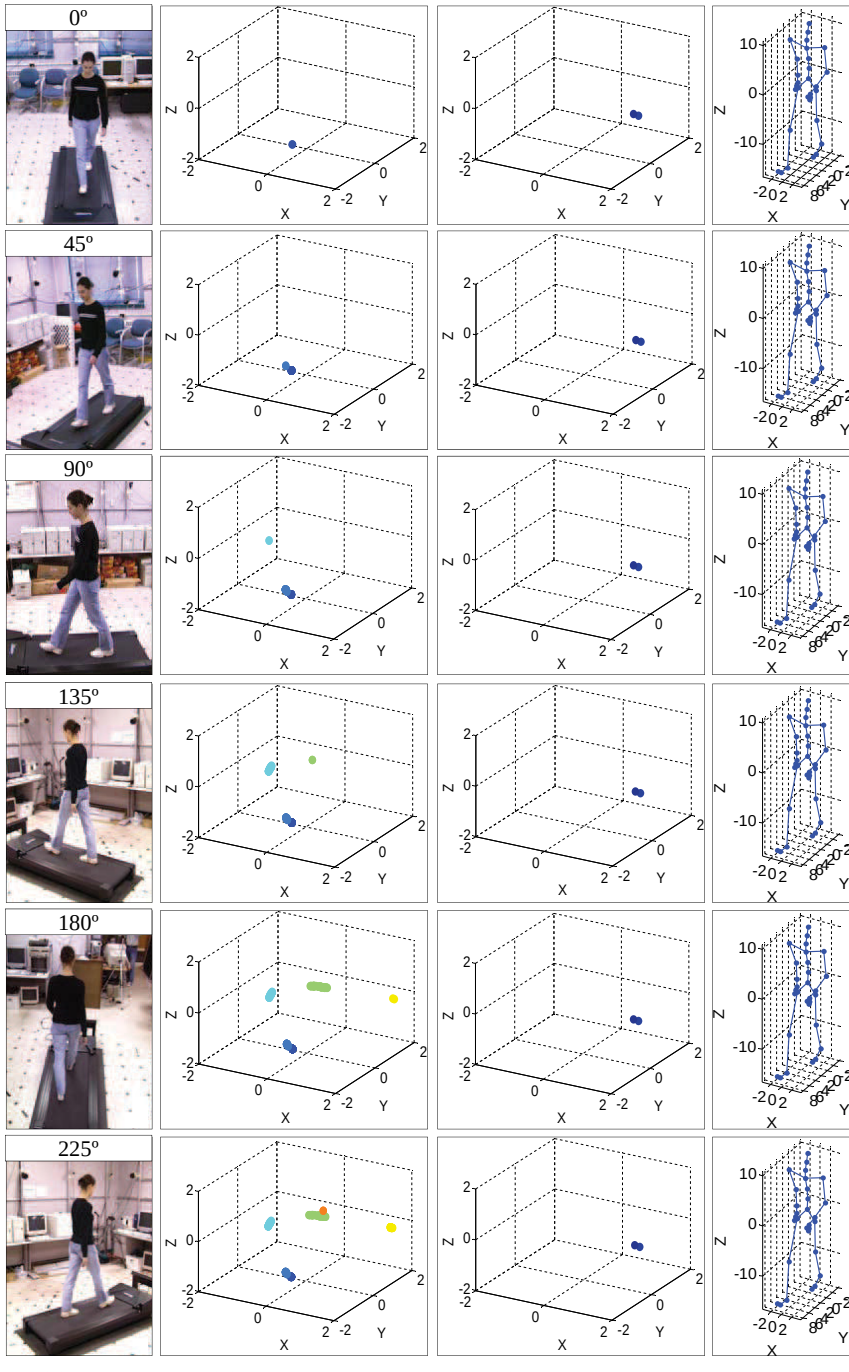


Fig. 16. CMU MoBo walking sequences: pose estimation results with various postures.

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.



(a) Input (b) View manifold (c) Pose manifold (d) 3D pose

Fig. 17. CMU MoBo walking sequences: pose estimation results with various viewpoints.

Table 1. An average joint angle error for 30 joints.

Approach	Proposed method	K-Nearest Neighbor search	GPLVM
Average error	0.65	0.86	4.88

Table 2. Comparison of the execution time.

	Biased manifold learning	Manifold learning
Pose (K = 58)	41.075 s	62.78 s
View (K = 300)	527.742 s	1398.646 s
Kinematics (K = 62)	0.455 s	0.474 s

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.

samples for the CMU MoBo dataset. According to Table 2, the proposed method is faster than the competing method, which is due to the more efficient distance mapping.

5. Conclusion

Conventional pose estimation methods based on manifold learning suffer from many problems caused by unexpected silhouette variation. The simultaneous consideration of pose and view variations, however, is a challenging problem due to the limited representation.

In order to tackle these problems, we proposed a view-invariant body pose estimation method constructing three kinds of manifolds separating pose, view, and body configuration. Two types of label data, i.e. view and pose, are used to learn three biased manifolds. Utilizing the view and pose label data, we could sort samples in each manifold along the corresponding sequence. In our experiments, deviations of the biased samples for each group in biased embedding spaces were lower than those of the original samples, showing the efficacy of the biased distance learning.

Our experimental results showed that the mapping function based on the General Regression Neural Network outperformed the mapping functions based on the radial basis function method. The biased embedding together with the more efficient learning method made it possible to reliably and robustly estimate 3D body pose from a 2D image under various viewpoints.

While the proposed method presented good performance in our experiments, it is still limited to the estimation of the pre-trained actions with one manifold for each action. For more general applications, it is needed to develop a novel method that can represent various actions in a manifold.

Acknowledgments

This work was supported by WCU (World Class University) program through the National Research Foundation of Korea funded by the Ministry of Education, Science and Technology, under Grant R31-10008.

Int. J. Wavelets Multiresolut Inf. Process. 2012.10. Downloaded from www.worldscientific.com
by KOREA UNIVERSITY SEJONG CAMPUS LIBRARY on 03/25/13. For personal use only.

References

1. A. Agarwal and B. Triggs, Recovering 3D human pose from monocular images, *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(1) (2006) 44–58.
2. M. Ahmad and S.-W. Lee, Human action recognition using shape and CLG-motion flow from multi-view image sequences, *Pattern Recogn.* **41**(7) (2008) 2237–2252.
3. M. Ahmad and S.-W. Lee, Variable silhouette energy image representations for recognizing human actions, *Image Vision Comput.* **28**(5) (2010) 814–824.
4. V. Balasubramanian and J. Ye, Biased manifold learning: A framework for person-independent head pose estimation, in *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (Minneapolis, USA, 2007), pp. 1–7.
5. O. Bernier, P. Cheung-Mon-Chan and A. Bouguet, Fast nonparametric belief propagation for real-time stereo articulated body tracking, *Computer Vision and Image Understanding* **113**(1) (2009) 29–47.
6. Y. Chtioui, S. Panigrahi and L. Francl, A generalized regression neural network and its application for leaf wetness prediction to forecast plant disease, *Chemometrics Intelligent Laboratory Syst.* **48** (1999) 47–58.
7. A. Fathi and G. Mori, Human pose estimation using motion exemplars, in *Proc. of IEEE International Conference on Computer Vision* (Rio De Janeiro, Brazil, 2007), pp. 1–8.
8. R. Gross and J. Shi, The CMU motion of body (MoBo) database, Technical Report, (Robotics Institute, Pittsburgh, USA, 2001), No. CMU-RI-TR-01-18.
9. A. Gupta, A. Mittal and L. Davis, Integration for efficient multiview pose estimation with self-occlusions, *IEEE Trans. Pattern Anal. Mach. Intell.* **30**(3) (2008) 493–506.
10. D. Hur, C. Wallraven and S.-W. Lee, View invariant body pose estimation based on biased manifold learning, in *Proc. 20th IAPR/IEEE International Conference on Pattern Recognition* (Istanbul, Turkey, 2010), pp. 3866–3869.
11. N. Lawrence, Gaussian process latent variable models for visualization of high dimensional data, in *Proc. of Advances in Neural Information Processing Systems*, Vol. 16 (Vancouver, Canada, 2004), pp. 329–336.
12. C.-S. Lee and A. Elgammal, Modeling view and posture manifolds for tracking, in *Proc. of IEEE International Conference on Computer Vision* (Rio De Janeiro, Brazil, 2007), pp. 1–8.
13. C.-S. Lee and A. Elgammal, Coupled visual and kinematic manifold models for tracking, *Int. J. Computer Vision* **87**(1–2) (2010) 118–139.
14. E. Murphy-Chutorian and M. Trivedi, Head pose estimation in computer vision: A survey, *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(4) (2009) 607–626.
15. S. Roweis and L. Saul, Nonlinear dimensionality reduction by locally linear embedding, *Science* **290**(5500) (2000) 2323–2326.
16. D. Specht, A general regression neural network, *IEEE Trans. Neural Networks* **2**(6) (1991) 568–576.
17. X. Wu, J. Lai and X. Chen, Rank-1 tensor projection via regularized regression for action classification, *Int. J. Wavelets, Multiresolut. Inf. Process.* **9**(6) (2011) 1025–1041.
18. H.-D. Yang and S.-W. Lee, Reconstruction of 3D human body pose from stereo image sequences based on top-down learning, *Pattern Recogn.* **40**(11) (2007) 3120–3131.
19. Q. Zou, X.-L. Huang and S.-W. Luo, Multiresolution image perceptual grouping using topological structure embedding in manifold, *Int. J. Wavelets, Multiresolut. Inf. Process.* **5**(1) (2007) 39–49.
20. http://mocap.cs.cmu.edu/.
21. http://my.smithmicro.com/dr/poser.html.