International Scholarly Research Network
ISRN Machine Vision
Volume 2012, Article ID 235396, 11 pages
doi:10.5402/2012/235396

Research Article

3D Human Motion Tracking and Reconstruction Using DCT Matrix Descriptor

Alireza Behrad and Nadia Roodsarabi

Electrical Engineering Department, Faculty of Engineering, Shahed University, Tehran 33191-18651, Iran

Correspondence should be addressed to Alireza Behrad, behrad@shahed.ac.ir

Received 30 January 2012; Accepted 19 February 2012

Academic Editors: C.-C. Han and J. Heikkilä

One of the most important issues in human motion analysis is the tracking and 3D reconstruction of human motion, which utilizes the anatomic points' positions. These points can uniquely define the position and orientation of all anatomical segments. In this work, a new method is proposed for tracking and 3D reconstruction of human motion from the image sequence of a monocular static camera. In this method, 2D tracking is used for 3D reconstruction, which a database of selected frames is used for the correction of tracking process. The method utilizes a new image descriptor based on discrete cosine transform (DCT), which is employed in different stages of the algorithm. The advantage of using this descriptor is the capabilities of selecting proper frequency regions in various tasks, which results in an efficient tracking and pose matching algorithms. The tracking and matching algorithms are based on reference descriptor matrixes (RDMs), which are updated after each stage based on the frequency regions in DCT blocks. Finally, 3D reconstruction is performed using Taylor's method. Experimental results show the promise of the algorithm.

1. Introduction

One of the challenging issues in machine vision and computer graphic applications is the modeling and animation of human characters. Especially body modeling using video sequences is a difficult task that has been investigated a lot in the last decade. Nowadays, 3D human models are employed in various applications like movies, video games, ergonomic, e-commerce, virtual environments, and medicine.

3D scanners [1, 2] and video cameras are two sample tools that have been presented for 3D human model reconstruction. 3D scanners have limited flexibility and freedom constraints. In addition, the higher cost of these devices put them out of reach for general use.

Video cameras are nonintrusive and flexible devices for extraction of human motion. However, due to the high number of degrees of freedom for the human body, human motion tracking is a difficult task. In addition, self-occlusion of human segments and their unknown kinematics make the human tracking algorithm more challenging.

Existing vision-based approaches for human motion analysis may be divided in two groups, including model-based and model-free methods [3]. In model-based methods [4–8], *a priori* known human model is employed to represent human joints and segments as well as their kinematics. Model-free approaches do not employ a predefined human model for motion analysis; instead, the motion information is derived directly from video sequences. Model-free approaches mostly use a database of exemplars [9] or a learning machine [10, 11] for motion reconstruction. They are mostly restricted to known environments or images taken from a known viewpoint. Model-based approaches are more general and typically support the viewpoint independent processing or multiple viewpoints. However, they need initialization.

Various algorithms may also be divided into different categories based on the acquisition system. Some approaches are based on monocular cameras [4–14], while others employ multicamera video streams [15–20]. Also, some approaches benefit from calibrated views or cameras [15–20], while others utilize uncalibrated images [5–14].

Nowadays, monocular uncalibrated video sequences such as sports video footage are the most common source

of human motions. Generally, 3D pose estimation is not possible using a monocular camera. Therefore, it is necessary to employ special assumptions for 3D pose estimation. Furthermore, 3D reconstruction of human motion poses more additional difficulties like self-occlusion, high-dimensional representation, lack of calibration, and articulated human motion to name a few.

To compensate the lack of enough information for 3D reconstruction of human motion using uncalibrated monocular video sequences, different approaches considered some restrictive conditions. Some algorithms assumed the manual specification of key features such as joints positions or segments length [5, 21]. Furthermore, some algorithms employed a database of different motions from various human subjects to facilitate motion reconstruction [9, 13, 14].

Different algorithms for motion reconstruction using monocular videos are roughly divided into three categories, including, (i) discriminative methods [9, 13, 14], (ii) estimating and tracking methods [6–8], and (iii) method based on learning [4, 11]. In discriminative methods, 3D joint coordinates are found by using database, motion libraries and so on. In estimating and tracking methods, 3D information is extracted using a sequence of images and tracking algorithm. In methods based on learning, a machine or model is trained with some *a priori* features and used for motion reconstruction.

Various algorithms for human motion reconstruction may utilize different image descriptors for tracking, matching, or model extraction. In [9, 13], shape context descriptor was used for matching key points. A shape context is a representation of shape by a discrete set of points sampled from the internal or external contours on the shape. The contour can be obtained as the locations of edge pixels as found by an edge detector. Some image-matching algorithms employed scale invariant feature transform (SIFT) [22], to detect and describe local features in images. SIFT features are scale and rotation invariant, but computationally expensive. In [7, 23], silhouette and contour of the human body were extracted for human model reconstruction. Silhouette and contour can be easily extracted in static cameras. However, in mobile camera and cluttered background, it is difficult to extract silhouette robustly. Edge or edge lines [24] and point features [12] were also used in some algorithms as image descriptors.

In this article, we introduce a new method for 3D reconstruction of human motion in uncalibrated monocular video streams, which is based on our previous work [25]. The method utilizes a combination of discriminative and tracking algorithm. In this algorithm, the information of database is utilized to increase tracking accuracy. The method utilizes a new descriptor based on discrete cosine transform (DCT). The advantage of using this descriptor is the capability of selecting proper frequency regions in various tasks, which results in better tracking and poses matching. For example, we use low and middle frequency in tracking for intensity as well as edge tracking. Also, we pass up color of clothes in database matching by avoiding low-frequency information.

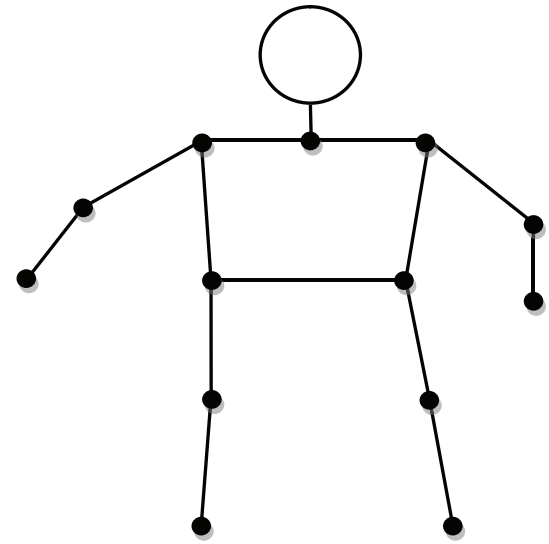


Figure 1: Human body model which is utilized in the proposed algorithm.

The paper is organized as follows. In the next section, we review the human model utilized in the proposed algorithm. Section 3 discusses the proposed algorithm for tracking and 3D reconstruction of human motion using sequences of images acquired by a single video camera. Experimental results appear in Section 4, and we conclude the paper in Section 5.

2. Human Body Model

Human skeleton system is treated as a series of jointed links (segments), which can be modeled as rigid bodies. In the motion reconstruction applications, it is common to use a simple skeleton system for modeling the important segments. We describe the body as a stick model consisting of a set of thirteen joints (plus the head), which are connected by thirteen segments as shown in Figure 1.

The algorithm needs the knowledge of relative lengths of the segments for the 3D reconstruction purpose, which can be obtained from anthropometric data, which is shown in Table 1.

With known 2D position and using the knowledge of length of the segments and enforcing some constraints such as dynamic smoothing, we can reconstruct 3D human model.

3. Proposed Algorithm

Figure 2 shows the block scheme of the proposed algorithm. In the proposed method, we track 2D joints position using a static and uncalibrated monocular video and use them to estimate 3D skeletal configuration. Since not enough information is available from monocular video for 3D reconstruction; we save several 2D exemplars of various body poses in the database and use them to correct tracked points. In this algorithm, joint tracking is based on the $n*n$

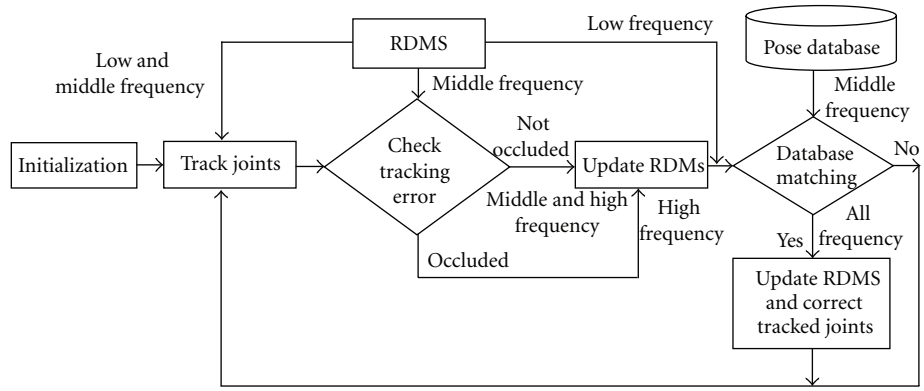


Figure 2: Block scheme of the proposed algorithm.

DC	1	5	6	14	15	27	28
2	4	7	13	16	26	29	42
3	8	12	17	25	30	41	43
9	11	18	24	31	40	44	53
10	19	23	32	39	45	52	54
20	22	33	38	46	51	55	60
21	34	37	47	50	56	59	61
35	36	48	49	57	58	62	63

Figure 3: DCT coefficients and different frequency regions for an 8∗8 DCT block.

Table 1: Relative lengths of the human body segments [27].

Segment	Relative Length (MC) (cm)	Relative Length (L) (unit)
Height	175	8i
Lower Arm	35	2i
Upper Arm	25	1.5i
Neck-Head	25	1.25i
Shoulder Girdle	44	2i
Torso	53	2.5i
Pelvic Girdle	30	1.5i
Upper leg	46	2i
Lower leg	52	2i
Foot	22	1i

block of DCT coefficients (descriptor matrix). Algorithm starts by background subtraction and 2D joints' positions are initialized by the user in the first frame. Then, the descriptor matrix is calculated and saved as "reference descriptor matrix" for each joint. In the next stage, all joints are tracked using their own RDMs. After finding joint positions in the subsequent frames, RDMs are updated based on DCT block frequency regions considering occlusion problem and tracking errors. The advantage of using RDMs is the capabilities of selecting proper frequency regions in various tasks, which results in an efficient tracking and pose-matching algorithms. When the human pose is estimated in the current frame, it is compared with different poses in the database based on middle-frequency information. In the case of correspondence, joint positions are corrected and RDMs are updated. We use the information of middle-frequency regions for this purpose to remove clothing color (low frequency) and body deformation details (high frequency).

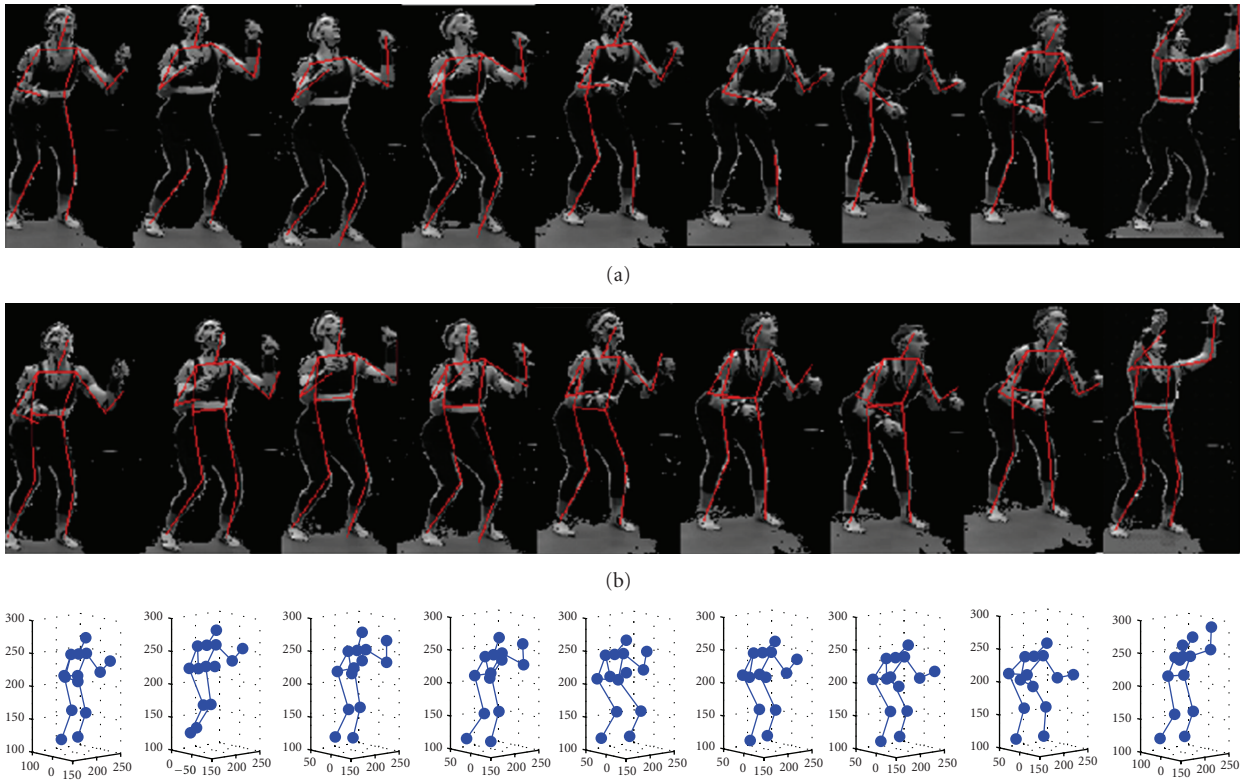
A major problem that may be encountered in the algorithm is the occlusion of joints. To handle the problem, we detect occluded joints and mark them as "occluded." When an "occluded" joint appears again, its positions are corrected by interpolation.

As it is shown in Figure 2, we utilize different frequency regions for various tasks in the proposed algorithm. Table 2 summarizes various tasks in the proposed algorithm and the utilized frequency regions.

Given the 2D joint locations, the 3D body configuration is estimated using Taylor's algorithm [12].

Table 2: The required frequency regions for different tasks in the proposed algorithm.

Tasks	Low frequency	Middle frequency	High frequency
Frequency regions for matching process		×	
Frequency region for database matching process	×	×	
Calculation of tracking errors		×	
Updating RDMs for non-occluded joints	×	×	×
Updating RDMs for occluded joints			×
Updating RDMs if database matching occur		×	

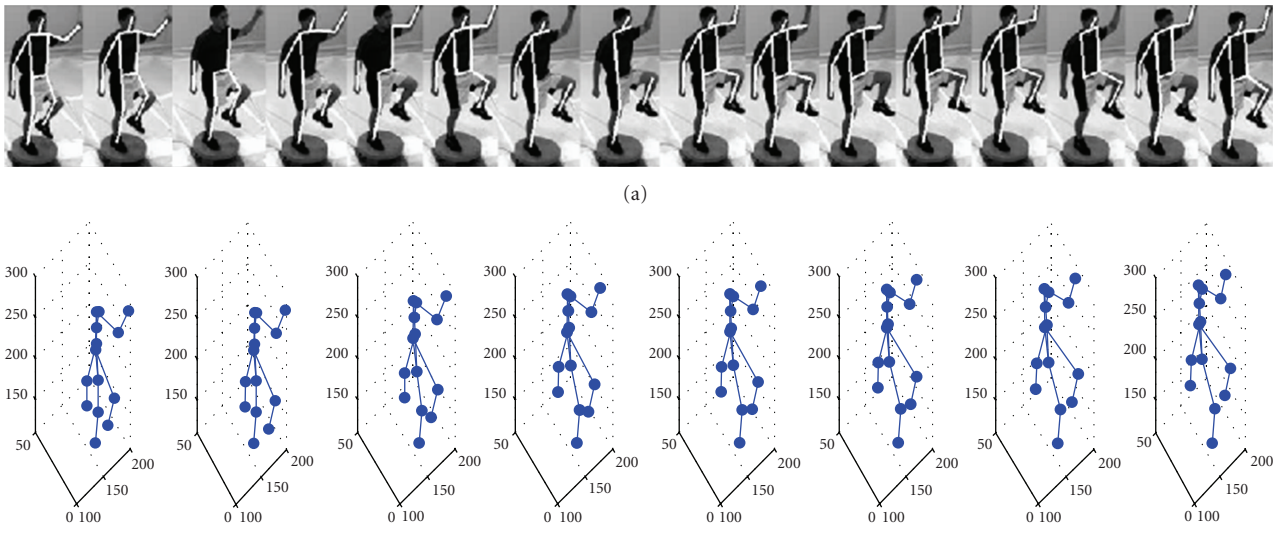


(a)

(b)

(c)

Figure 4: Reconstruction results for some frames of a typical video with 300 frames. (a) shows the results of the proposed tracking algorithm before the interpolation of some joints of the video sequence labeled as occluded. (b) shows the results of the proposed tracking algorithm after the interpolation of occluded joints. (c) shows 3D reconstruction results.



(a)

(b)

Figure 5: Reconstruction results for some frames of a typical video using the proposed algorithm. (a) shows 2D joint tracking. (b) shows 3D reconstruction results.

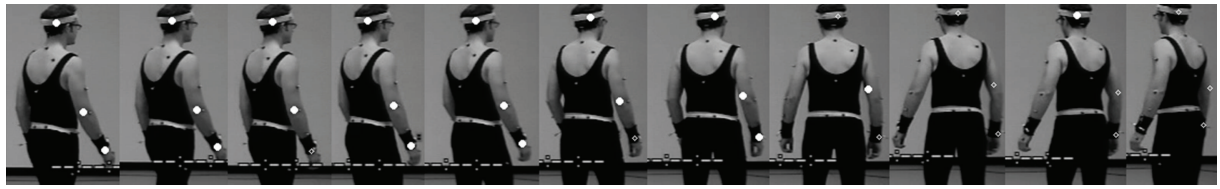
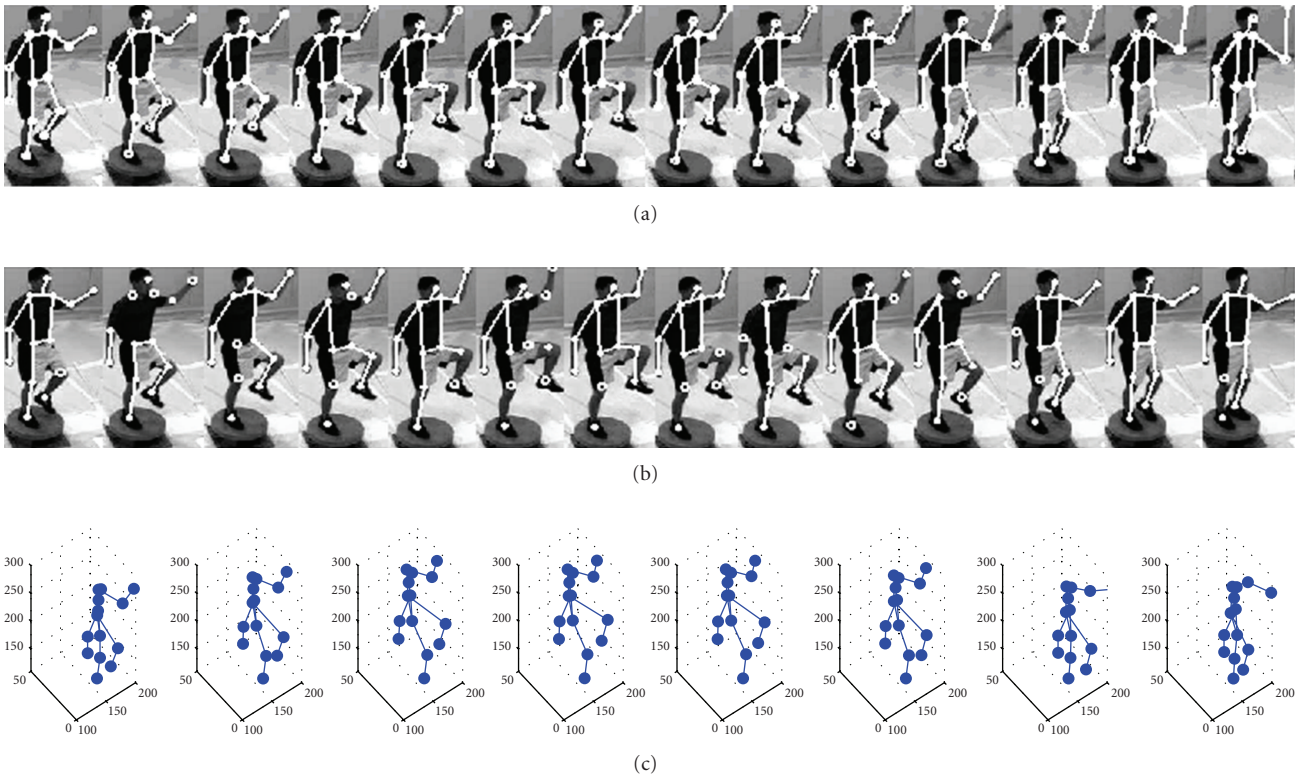


Figure 6: The tracking result for head and right hand joints. Larger circle is used to show true tracking limb and smaller circle is used to show occluded limb detected by the algorithm.



(a)

(b)

(c)

Figure 7: Comparison of the proposed descriptor with the shape context descriptor. (a) Tracking using the shape context descriptor, (b) Tracking using the proposed descriptor, (c) 3D reconstruction using the proposed descriptor.

3.1. Descriptor Matrix. In this article, we use DCT-based descriptors for the tracking and matching purposes. Descriptor Matrix (DM) for a point p_i is an $n*n$ DCT coefficients matrix. By utilizing the image window of fixed size $(n*n)$ centered on point p_i, a descriptor matrix for the point p_i is calculated as follows:

$$F(u,v) = C(u)C(v) \sum_{x=p_x-n/2}^{x=p_x+n/2} \sum_{y=p_y-n/2}^{y=p_y+n/2} f(x,y) \cos\left[\frac{(2x+1)u\pi}{2n}\right] \times \cos\left[\frac{(2y+1)v\pi}{2n}\right], \quad (1)$$

where (p_x, p_y) is the coordinate of central point p_i, and $C(x)$ is calculated using the following equation:

$$C(x) = \begin{cases} \frac{1}{\sqrt{n}}, & \text{if } x = 0 \\ \sqrt{\frac{2}{n}}, & \text{otherwise.} \end{cases} \quad (2)$$

There are n^2 coefficients in each DM matrix divided into three frequency regions according to Figure 3. White region is the low-frequency region, gray region is the middle-frequency region, and black region is high-frequency region. We use these three frequency regions for tracking and matching joints. We use matrix distance as a method to measure the similarity between two descriptor matrixes.

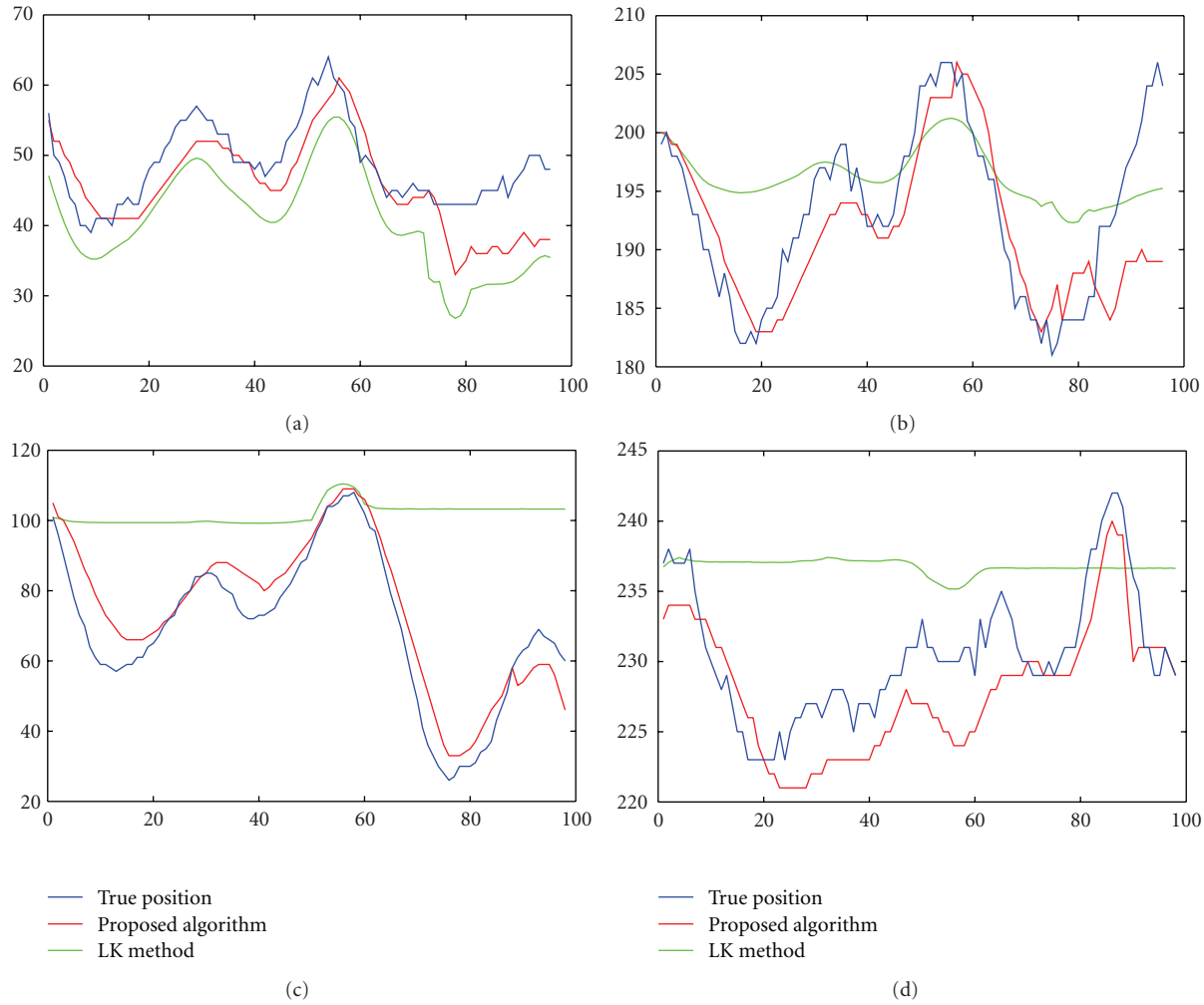


FIGURE 8: Tracking of head and hand joints using the proposed algorithm and LK method [26]. (a) Head joint position in x direction, (b) head joint position in y direction, (c) hand joint position in x direction, (d) hand joint position in y.

Matrix distance for two descriptor matrixes M and N, in the specified frequency region of R, is calculated as

$$M_{\text{dis}}(M,N) = \sqrt{\sum_{f\in R}\left(M_f - N_f\right)^2}. \tag{3}$$

3.2. Reference Descriptor Matrix (RDM). RDMs store the required information for the tracking of joints. To find the location of a joint in the current frame, RDMs of joints in the previous frame as well as the information of database is employed. RDMs are generated for different joints of the body independently and are specified by $\text{RDM}_1, \ldots, \text{RDM}_{13}$. Reference descriptor matrix for joint j (RDM_j) is loaded from the descriptor matrix for joint j after the initialization of joints by the user and updated after finding the location of joints in the subsequent frames. Updating routine is different for each frequency region as follows.

Low-Frequency Region. This region consists of general shape and intensity information of the tracked joint, so it changes gradually in successive frames. Tracking process may lose the tracked joint for several reasons such as occlusion problem or large distortion. Therefore, the tracked joint information may be incorrect in the current frame. For safekeeping of the general joint information, we leave the low frequency coefficients unchanged during the tracking. This region is updated only when a correspondence is found in the database.

Middle-Frequency Region. This region consists of general edge information. Because the individual limbs are deformable due to moving muscle and clothing, we update middle frequency coefficients during tracking only if the tracked joint is not occluded. Furthermore, this region is updated when a correspondence is found in the database.

High-Frequency Region. This region consists of joints' details. The region is updated frame by frame without any restriction.

3.3. Tracking. The tracking process is based on the matching techniques in the frequency domains. Tracking process aims

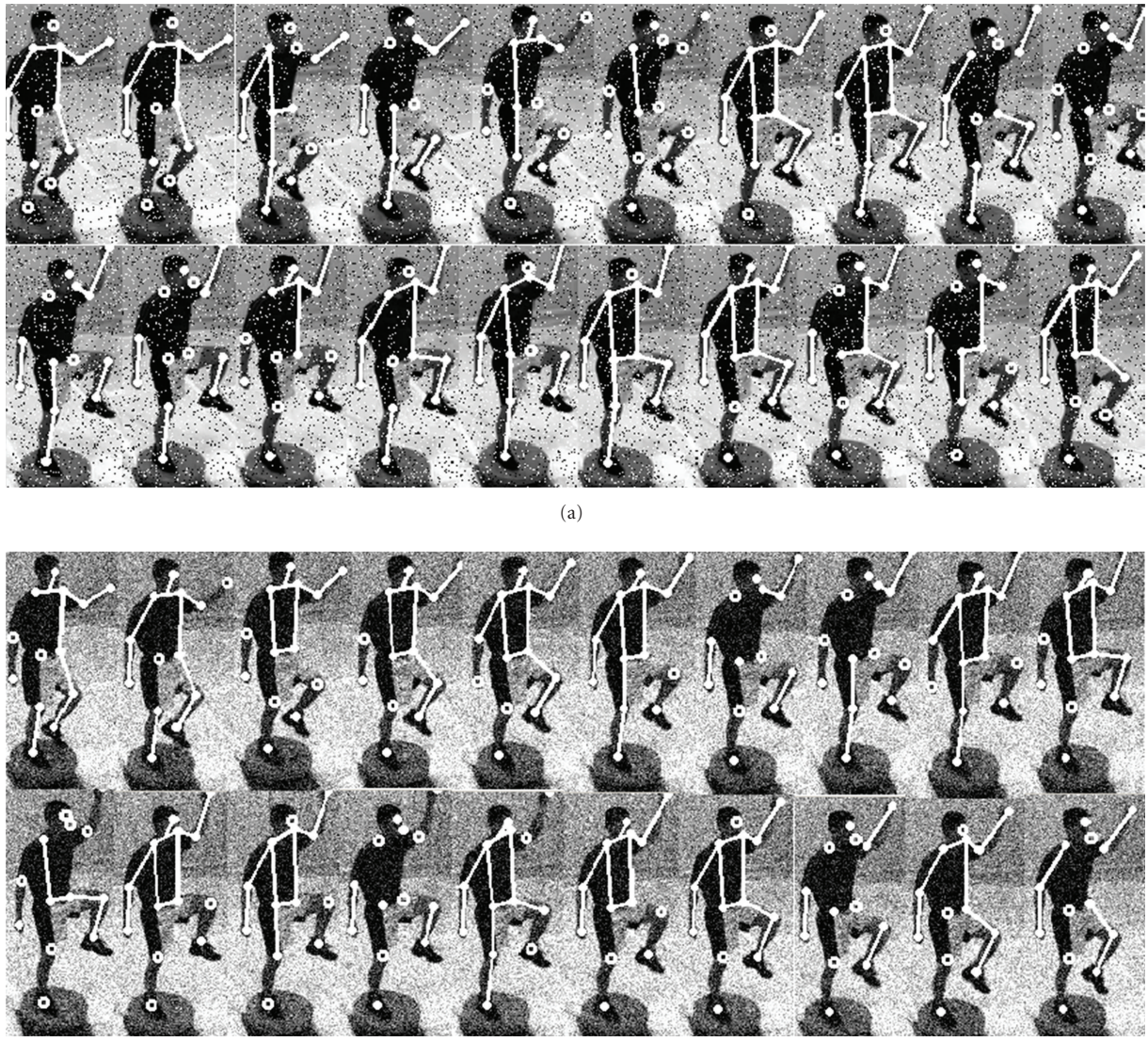


Figure 9: Tracking results of the proposed algorithm in the noisy environments. (a) Noisy image with 10 percent salt and pepper noise, (b) images with Gaussian noise of SNR = 10 dB.

to find body joints in successive frames. Because of temporal correspondences between subsequent frames, search for the corresponding joint is local. In two successive frames, limbs and joints have the same intensity and general shape, but they are different in details. So, we use low- and middle-frequency regions in tracking process.

The tracking process is based on DCT matching techniques. Its basic idea is to track joints through the sequence of frames by utilizing RDMs. For this purpose, descriptor matrices are computed for each pixel in the search window. The best match is found by selecting minimum matrix distance between low and middle frequencies of RDM_j and search window descriptor matrixes (SWDMs).

Assuming that the initial estimate of the pose has been given, the tracking algorithm can be summarized in two steps as follows.

(1) Generate descriptor matrices for all pixels in the search window at frame t (SWDMs).

(2) Determine best matching point in the search window by computing matrix distance between RDM_j and SWDMs.

As mentioned before, a major problem that may be encountered in the algorithm is the occlusion of joints. To handle the problem, we detect occluded joints and mark them as "occluded." In order to detect the occlusion of the tracked joint j at frame t, we calculate matrix distance in the middle-frequency region between descriptor matrix of

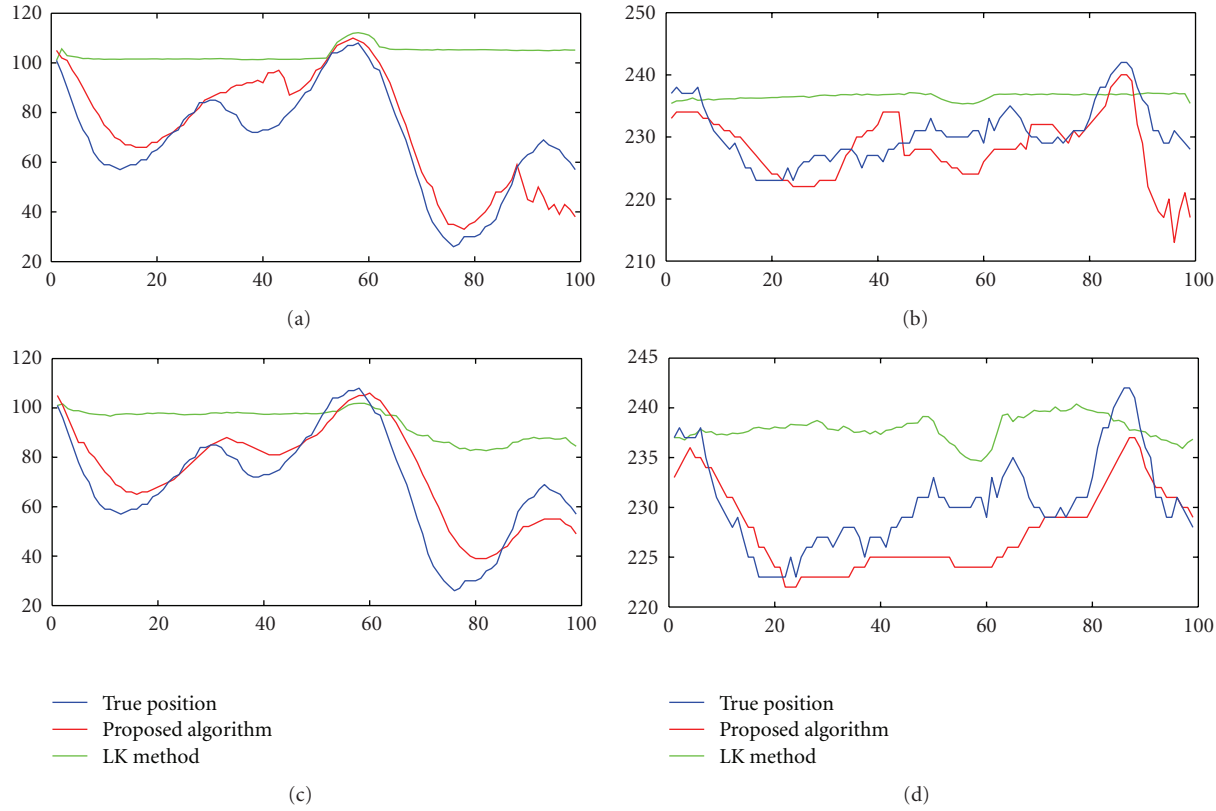


Figure 10: Tracking of left hand joint in noisy environment using the proposed algorithm and LK method [26]. (a) Hand joint position in x direction in images with Gaussian noise of SNR = 10 dB, (b) hand joint position in y direction in images with Gaussian noise of SNR = 10 dB, (c) hand joint position in x direction in images with salt and pepper noise with density of 10 percent, (d) hand joint position in y in images with salt and pepper noise with density of 10 percent.

tracked point ($\mathrm{DM}_j(t)$) and RDM_j. Then, we determine the occlusion of the joints based on the following equation:

$$M\mathrm{dis}_{\mathrm{middle}}\left(\mathrm{RDM}_j, \mathrm{DM}_j(t)\right) \begin{cases} < \Delta, & \text{joint is not occluded,} \\ > \Delta, & \text{joint is occluded.} \end{cases} \tag{4}$$

When an occluded joint appears again, its positions during the period of occlusion are estimated by linear interpolation using the positions of the joint before and after the occlusion.

3.4. Database Matching Process. The database consists of required information of different poses for video sequences of a number of subjects. This information includes body joints' positions and their descriptor matrixes in the middle frequency region as well as necessary labels for 3D reconstruction. Head position is used as the reference joint to calculate joints' positions. In other words, joints' positions are determined with respect to head.

To measure similarity between human pose in the current frame (p_f) and human pose in the database (p_d), we employ two kinds of the descriptor matrix: DDMs and FDMs, which will be defined later. If pose distance is smaller than a predefined threshold, correspondence occurs. In this case, joints' positions and middle frequency region of RDM are corrected. Human pose distance is defined by

$$P\mathrm{dis}_p\left(p_f, p_d\right) = \sum_{j=1}^{13} M\mathrm{dis}_{\mathrm{low,mid}}\left(\mathrm{DDM}_j, \mathrm{FDM}_j\right), \tag{5}$$

where database descriptor matrix (DDM) is generated using the low-frequency information of RDM (for intensity similarity of joints) and middle-frequency information of database (for edge similarity). DDM descriptor is defined as follows:

$$\mathrm{DDM}_f = \begin{cases} \mathrm{RDM}_f & f : \text{low frequency,} \\ \text{Database} & f : \text{middle frequency,} \\ 0 & f : \text{high frequency.} \end{cases} \tag{6}$$

Frame descriptor matrix (FDM) is also generated using the following algorithm.

(i) Search locally around the previous head position to find correspondence for $\mathrm{RDM}_{\mathrm{head}}$ point in the current frame.

(ii) Determine other joints in the current frame by adjusting the head position.

(iii) Generate descriptor matrices for each joint and save them as FDMs.

The algorithm to measure the similarity between human pose in the current frame (p_f) and human pose in the database (p_d) can be summarized as follows.

(1) Generate DDMs.

(2) Search locally to find the head position in the current frame.

(3) Determine other joints' positions in the current frame.

(4) Compute matrix distance for DDM and FDM in low and middle frequency regions.

(5) In the case of correspondence, correct joints' positions.

(6) Update RDMs.

As in the constant descriptor size, the descriptor matrices are not scale invariant. In the absence of substantial background clutter, scale invariance can be achieved by setting descriptor matrix size as a function of length for the body segments.

3.5. 3D Reconstruction. We use Taylor's method [12] to estimate the 3D configuration of the human body given the joints' positions. Taylor's method operates on a single 2D image, taken by an uncalibrated camera. It assumes a scaled orthographic projection model for the camera and need the following information.

(i) The image coordinates of joints (u, v).

(ii) The relative lengths of body segments connecting the joints.

(iii) The "closer endpoints" for body segments and joints.

In this paper, the image coordinates of joints are obtained using the proposed tracking and matching algorithms. The closer endpoints for segments are supplied by exemplars in the database, and automatically transferred to the input image after the matching process. The relative lengths of body segments are fixed in advance but can also be transferred from exemplars.

We use the same 3D kinematics model defined over joints as that in Taylor's work. We can solve for the 3D configuration of the body $\{(X_i, Y_i, Z_i) : i \in \text{joints}\}$ up to some ambiguity in scale s. The method considers the foreshortening of each body segment to construct the estimate of body configuration. For each pair of body segment's joints, we have the following equations:

$$\begin{aligned} l^2 &= (X_1 - X_2)^2 + (Y_1 - Y_2)^2 + (Z_1 - Z_2)^2, \\ (u_1 - u_2) &= s(X_1 - X_2), \\ (v_1 - v_2) &= s(Y_1 - Y_2), \\ dZ &= (Z_1 - Z_2), \\ dZ &= \sqrt{l^2 - \frac{\left((u_1 - u_2)^2 + (v_1 - v_2)^2\right)}{s^2}}. \end{aligned} \tag{7}$$

To estimate the configuration of a body, we first fix one joint as the reference point and then compute the positions of the others with respect to the reference point. Since we are using a scaled orthographic projection model, the X and Y coordinates are known up to the scale factor s. All that remains to compute relative depths of endpoints dZ. We compute the amount of foreshortening and use the user-supplied "closer endpoint" labels from the closest matching exemplar to solve for the relative depths.

Moreover, Taylor notes that the minimum scale s can be estimated from the fact that dZ cannot be complex:

$$s \geq \frac{\sqrt{(u_1 - u_2)^2 + (v_1 - v_2)^2}}{l^2}. \tag{8}$$

This minimum value is a good estimate for the scale since one of the body segments is often perpendicular to the viewing direction.

4. Experimental Results

The proposed algorithm was applied for the reconstruction of human subjects from single-camera videos. The database consists of some poses of a number of subjects, performing different types of motions from the CMU MoCap database [28]. On this collection of poses, we manually determined joint locations of each pose and "closer endpoint" labels for each body segment, which are used in 3D reconstruction. Also, we save middle frequency of the descriptor matrix for each labeled joint.

Our experiments are divided into two parts: (i) reconstruction results for the sequences of real people with different motions in CMU MoCap database, and (ii) 2D tracking results in different video sequences.

4.1. Reconstruction Results. We tested the proposed algorithm on a variety of sequences of real human subjects performing various motions. To facilitate the tracking process, we utilized a background estimation algorithm based on temporal median filter. To make the proposed descriptor matrixes scale invariant, we set descriptor matrix size as a function of length of the body segments.

Figure 4 shows sample results of 2D body joint localization before and after interpolation and finally 3D reconstruction on the CMU dataset. Note that some joints are occluded or failed in 2D tracking. These joints are reconstructed by interpolation. Figure 5 shows sample results of another video, which its 3D reconstruction performed successfully.

4.2. Tracking Results. In this section, we investigate the robustness of the tracking algorithm in some video sequences consisting of occluded limbs and noise.

Figure 6 shows the robustness of the proposed algorithm for limb tracking and distinguishing the occluded limbs. We tracked head and right hand joints using the proposed algorithm. Bigger circles show the nonoccluded tracked joints. The occluded or falsely tracked joints, which are detected by (4), are shown by smaller circles. It is obvious that our tracking algorithm performs very well in tracking and detecting occluded limbs.

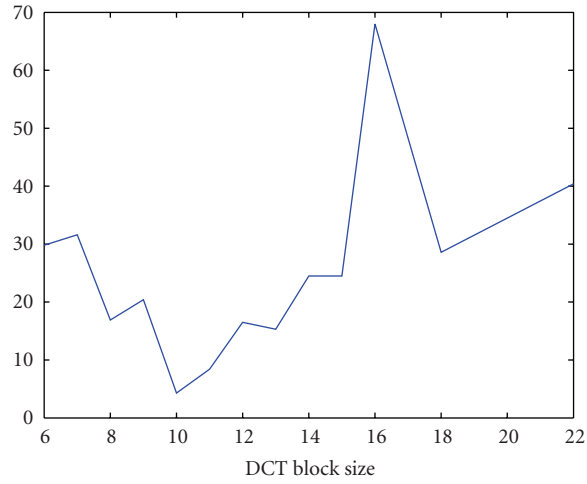


Figure 11: Average tracking errors of joints versus DCT block size for a typical video.

To show the efficiency of the proposed image descriptor, we compared the proposed descriptor with shape context descriptor [13, 29]. Figure 7 shows the results of joints tracking using the proposed descriptor as well as the shape context descriptor. The results of Figure 7 reveal that the proposed algorithm has tracked the joints more efficiently.

We also compared the proposed algorithm with a well-known tracking algorithm which tracks the feature points by optical flow and iterative Lucas-Kanade (LK) method in pyramids [26]. Figure 8 illustrates the true position of the head and hand joints as well as their positions tracked by the proposed algorithm and LK method. The figure shows that the LK method has lost the head and hand positions; however, the proposed algorithm successfully tracked it.

To show the efficiency of the proposed algorithm in the noisy environment, we tested the proposed algorithm with noisy images. Figure 9(a) shows the tracking results for the video sequence of Figure 7 corrupted with 10 percent salt and pepper noise. In Figure 9(b), the results of the proposed tracking algorithm are shown for the same video sequence corrupted with Gaussian noise of SNR = 10 dB. Solid circles in the figure are the joints that are tracked normally, and empty circles show the joints labeled as "occluded." Figure 10 shows the true position of the left-hand joint as well as its position tracked by the proposed algorithm and LK method for the noisy images of Figure 9. Figures 9 and 10 show the efficiency of the proposed algorithm in tracking videos in noisy environments.

We also investigated the effect of DCT block size on the efficiency of the tracking algorithm. Figure 11 shows the average tracking error for a typical video. As the figure show, the algorithm has the best output at the DCT block size of $10*10$. However, the efficiency of the algorithm does not change considerably for DCT block size of 8 to 14. Our experiments show that the optimal DCT block size depends on the height of the human body in pixels. For example, for human height of 130 pixels, the optimal block size is 8. By the increase of human height, the optimal block size linearly increases with the rate of 0.1 per pixel.

5. Conclusion

In this paper, a new method for 3D reconstruction of human motion from the image sequence of a single static and uncalibrated camera is described. In this method, 2D tracking is used for 3D reconstruction, which a database of selected frames is used for the correction of tracking process. We used DCT blocks as matrix descriptors, which are used in the matching process for finding appropriate pose in the database and tracking process. We used three frequency regions for different tasks to enhance the accuracy of the proposed algorithm. The algorithm can detect occluded joints and recover their positions by interpolation. The proposed algorithm was tested with several video sequences in noisy and noiseless environments, and experimental results showed the reliability of the algorithm. This method is robust in 2D tracking and holding the properties of each joint along tracking process.

We also investigated the effect of DCT block size on the efficiency of the tracking algorithm. To make the tracking system scale invariant, it is possible to use an adaptive block size based on the height of human in pixels.

References

[1] P. Treleaven and J. Wells, "3D body scanning and healthcare applications," *Computer*, vol. 40, no. 7, pp. 28–34, 2007.

[2] P. Kelly, C. O. Conaire, J. Hodgins, and N. E. O'Connor, "Human motion reconstruction using wearable accelerometers," in *Proceedings of the ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA '10)*, Madrid, Spain, 2010.

[3] R. Poppe, "Vision-based human motion analysis: an overview," *Computer Vision and Image Understanding*, vol. 108, no. 1-2, pp. 4–18, 2007.

[4] Y. K. Wang and K. Y. Cheng, "A two-stage Bayesian network method for 3D human pose estimation from monocular image sequences," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, Article ID 761460, 2010.

[5] C. Barrón and I. A. Kakadiaris, "Estimating anthropometry and pose from a single uncalibrated image," *Computer Vision and Image Understanding*, vol. 81, no. 3, pp. 269–284, 2001.

[6] C. Sminchisescu and B. Triggs, "Kinematic jump processes for monocular 3D human tracking," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '03)*, pp. 69–76, Rhone-Alpes, France, 2003.

[7] C. Chen, Y. Zhuang, and J. Xiao, "Towards robust 3D reconstruction of human motion from monocular video," in *Lecture Notes in Computer Science: Advances in Artificial Reality and Tele-Existence*, Z. Pan, A. Cheok, M. Haller, R. Lau, H. Saito, and R. Liang, Eds., pp. 594–603, Springer, Berlin, Germany, 2006.

[8] G. Loy, M. Eriksson, J. Sullivan, and S. Carlsson, "Monocular 3D reconstruction of human motion in long action sequences," in *Lecture Notes in Computer Science: Computer Vision-ECCV*, T. Pajdla and J. Matas, Eds., vol. 3024, pp. 442–445, Springer, Berlin, Germany, 2004.

[9] G. Mori and J. Malik, "Estimating human body configurations using shape context matching," in *Lecture Notes in Computer Science: Computer Vision-ECCV*, A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, Eds., pp. 150–180, Springer, Berlin, Germany, 2002.

[10] A. Kanaujia, C. Sminchisescu, and D. Metaxas, "Semi-supervised hierarchical models for 3D human pose reconstruction," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '07)*, pp. 1–8, Minneapolis, Minn, USA, June 2007.

[11] R. Rosales and S. Sclaroff, "Learning and synthesizing human body motion and posture," in *Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 506–511, Grenoble, France, 2000.

[12] C. J. Taylor, "Reconstruction of articulated objects from point correspondences in a single uncalibrated image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '00)*, pp. 677–684, Hilton Head Island, SC, USA, June 2000.

[13] G. Mori and J. Malik, "Recovering 3D human body configurations using shape contexts.," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 7, pp. 1052–1062, 2006.

[14] M. J. Park, M. G. Choi, Y. Shinagawa, and S. Y. Shin, "Video-guided motion synthesis using example motions," *ACM Transactions on Graphics*, vol. 25, no. 4, pp. 1327–1359, 2006.

[15] I. Mikic, M. Trivedi, E. Hunter, and P. Cosman, "Articulated body posture estimation from multi-camera voxel data," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '01)*, pp. 455–460, Kauai, Hawaii, USA, 2001.

[16] S. Iwasawa, J. Ohya, K. Takahashi, T. Sakaguchi, S. Morishima, and K. Ebihara, "Human body postures from trinocular camera images," in *Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 326–331, Grenoble, France, 2000.

[17] A. Hilton, D. Beresford, T. Gentils, R. Smith, W. Sun, and J. Illingworth, "Whole-body modelling of people from multi-view images to populate virtual worlds," *Visual Computer*, vol. 16, no. 7, pp. 411–436, 2000.

[18] R. Plänkers and P. Fua, "Tracking and modeling people in video sequences," *Computer Vision and Image Understanding*, vol. 81, no. 3, pp. 285–302, 2001.

[19] R. Plankers, P. Fua, and N. D'Apuzzo, "Automated body modeling from video sequences," in *Proceedings of the IEEE International Workshop on Modelling People*, pp. 45–52, Kerkyra, Greece, 1999.

[20] G. K. M. Cheung, T. Kanade, J. Y. Bouguet, and M. Holler, "A real time system for robust 3D voxel reconstruction of human motions," in *Proceedings of the IEEE Conference onComputer Vision and Pattern Recognition*, pp. 714–720, Hilton Head Island, SC , USA, 2002.

[21] M. J. Park, M. G. Choi, and S. Y. Shin, "Human motion reconstruction from inter-frame feature correspondences of a single video stream using a motion library," in *Proceedings of the ASM SIGGRAPH Symposium on Computer Animation (SCA '02)*, pp. 113–120, July 2002.

[22] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[23] G. Qian and F. Guo, "Monocular 3D tracking of articulated human motion in silhouette and pose manifolds," *EURASIP Journal on Image and Video Processing*, vol. 2008, Article ID 326896, 2008.

[24] K. Rohr, "Towards model-based recognition of human movements in image sequences," *CVGIP: Image Understanding*, vol. 59, no. 1, pp. 94–115, 1994.

[25] N. Roodsarabi and A. Behrad, "3D human motion reconstruction using video processing image and signal processing," in *Proceedings of the 3rd international conference on Image and Signal Processing (ICISP '08)*, pp. 386–395, Cherbourg-Octeville, France, 2008.

[26] J. Y. Bouguet, "Pyramidal implementation of the lucas kanade feature tracker description of the algorithm," Intel Corporation, Microprocessor Research Labs, OpenCV Documents, 1999.

[27] F. Remondino and A. Roditakis, "3D reconstruction of human skeleton from single images or monocular video sequences," in *Lecture Notes in Computer Science: Pattern Recognition*, B. Michaelis and G. Krell, Eds., vol. 2781, pp. 100–107, Springer, Berlin, Germany, 2003.

[28] CMU Graphics Lab Motion Capture Database, http://mocap.cs.cmu.edu/.

[29] G. Mori, S. Belongie, and J. Malik, "Shape contexts enable efficient retrieval of similar shapes," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '01)*, pp. 723–730, Kauai, Hawaii, USA, December 2001.