Multi-Layer Skeleton Fitting for Online Human Motion Capture

Christian Theobalt Marcus Magnor Pascal Schüler
Hans-Peter Seidel

Max-Planck-Institut für Informatik
Stuhlsatzenhausweg 85, Saarbrücken, Germany
{theobalt|magnor|schueler|hpseidel}@mpi-sb.mpg.de

Abstract

This paper describes a new approach to fitting a kinematic model to human motion data which is a component of a marker-free optical human motion capture system. Efficient vision-based feature tracking and volume reconstruction by shape-from-silhouette are applied to raw image data obtained from several synchronized cameras in real-time. The combination of both sources of information enables the application of a new method for fitting a sophisticated multi-layer humanoid skeleton. We present results with real video data that demonstrate that our system runs at 1-2 fps.

1 Introduction

The field of human motion capture is an example for the coalescence of computer vision and computer graphics. The acquisition of human motion data is a prerequisite for the control of artificial characters in Virtual Reality and Augmented Reality applications as well as in computer animation and video games. The analysis of human motion, e.g. gesture recognition, can be used for intelligent user interfaces and automatic monitoring applications [5]. For animation, detailed skeletal body models are commonly applied. Existing optical motion capture systems using such models only work in a very constrained scene setup which makes necessary optical markers or similar scene-intrusive devices [7, 9]. Increasing computing power of off-the shelf computing hardware makes possible the first marker free vision-based motion capture systems. Previous approaches in this field [10, 6, 19] use features extracted from video frames to fit simple kinematic skeleton models to human body poses. The simultaneous recovery of pose and body shape from video streams [17] has also been considered. Optical flow and probabilistic body part models were used to fit a hierarchical skeleton to walking sequences [2]. None of the above approaches runs in real-time or comes close to interactive performance, however. If real-time performance is to be achieved, comparably simple models, such as probabilistic region representations and probabilistic filters for tracking [24], or the combination of feature tracking and dynamic appearance models [8] are used. Unfortunately, these approaches fail to support sophisticated body models.

New methods for the acquisition and efficient rendering of volumetric scene representations obtained from multiple camera views, known as shape from silhouette or the visual hull [12, 20, 18, 4], have been presented. Recent research shows that it is possible to acquire and render polyhedral visual hulls in real-time [15]. An image-based approach to visual hull construction samples and textures visual hulls along a discrete set of viewing rays [16]. State-of-the-art graphics hardware can be used to render acquired volume data interactively [13].

Only recently, new methods have been presented that use shape-from-silhouette to capture human motion. These approaches reconstruct the volume of a moving person at interactive frame rates and fit a comparably simple ellipsoid model to the volumes [3], or compute the motion parameters for a kinematic structure by means of a force-field exerted by the volume elements [14]. In [23], an iterative closest point method is used to fit a human model to volume data.

In previous work, efficient optical feature tracking and volume reconstruction were hardly considered simultaneously for the acquisition of human motion. In this paper we present a new method for marker-free motion capture which uses efficient

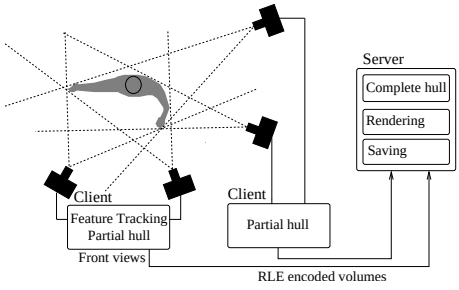

Figure 1: Online system architecture

color-based feature tracking to determine the 3D locations of salient body features over time. At the same time, a voxel-based reconstruction of the person's visual hull from multiple camera views is performed. The combination of both sources of information enables fitting of a multi-layer kinematic skeleton to the acquired motion data.

Section 2 gives an overview of the complete motion capture system architecture described in detail in [21]. The algorithms which are part of the real-time component are described in sections 4 to 6. The focus of this paper, the actual fitting of the skeleton, is described in Sect. 7. Results with real video streams are presented in Sect. 8. The paper concludes with a summary and the presentation of future work in Sect. 9

2 System Overview

The person to be tracked moves in a confined volume of space and is observed from multiple synchronized and calibrated cameras. Currently up to 6 Sony™ DFW-V500 IEEE1394 can record the scene in color mode and at a resolution of 320×240 pixels. Two cameras at a time are connected to one PC. On each PC a client application is running which performs a background segmentation and a volume reconstruction of the moving subject for the two connected camera views. In addition, the PC controlling the two front camera views tracks the locations of hands, head and feet and computes their 3D locations via triangulation. The so-constructed partial visual hulls are compressed and transferred to a sever PC which reconstructs the complete volume and displays it. These steps can be performed in real-time (see Fig. 1).

The saved volumes and 3D feature locations are used to fit a multi-layer skeleton model to the motion data in a separate step (Fig. 4). The software architecture is modular and enables easy extension of the approach to an arbitrary number of cameras using a hierarchical structure of the client-server network. For development efficiency reasons, the volume reconstruction and model-fitting components were implemented separately. The results, however, clearly show that an integrated system runs at near interactive frame rates.

3 Initialization

The cameras are calibrated in a common global coordinate system using Tsai's method [22]. In the first frame, the person is supposed to stand in an initialization position, facing the two front cameras, with both legs next to each other and spreading the arms horizontally away to the side at maximal extent. The person moves barefooted and needs to face these cameras allowing finite rotation and bending of the upper body part. For acquisition of a background several video frames without a moving subject are recorded with each camera.

4 Segmentation

The person's silhouette has to be separated from the background in each camera perspective and for each video frame. Additionally, the silhouettes showing the person in the initialization position seen from the front cameras is subdivided to find the initial image plane locations of head, hands and feet.

The separation of the moving person from the background is done by means of a statistical background model based on the color distribution of each pixel in the static background scene.

Figure 2: Video frame after background subtraction (l). GVD segmentation of silhouette in initialization position (r)

The method proposed in [3] is adapted which enables robust elimination of shadows cast by the person on the floor and the walls [21]. This way, a binary image for each camera is computed.

To identify the initial locations of head, hands and feet, the two front view silhouettes of the person in the initialization position are subdivided by means of a Generalized Voronoi Diagram (GVD) decomposition. Often used in motion planning for mobile robots [11], the Generalized Voronoi Diagram is the set of all points in the silhouette which is equidistant to at least two silhouette boundary points.

The GVD point set can be used to segment the silhouette into distinct regions by searching for points locally minimizing the clearance to the silhouette boundary. Lines are constructed through these points to separate neighboring regions. The boundaries to the head, hands and feet regions in the silhouettes are marked by constrictions. Fig. 2 shows a silhouette decomposed by this algorithm.

A graph encoding the connectivity between neighboring regions is built whose terminating nodes correspond to the searched feature locations (see [21] for details).

5 Tracking head, hands and feet

The client controlling the front views tracks the locations of hand, head and feet over time in both cameras. Several continuously adaptable mean shift trackers are used which follow the mean of dynamically changing pixel color probability distributions [1]. From the segmentation step, the initial locations and regional extents of the head, the hands and the feet regions in both front camera perspectives are known. The HSV mean colors of human skin are computed for each of these regions. Color intervals around these mean colors are defined. For every tracked feature in each video frame, pixel-wise region membership probabilities are approximated by color histogram back-projection. A separate continuously adaptable mean-shift tracker is used for each of the five body parts in each front camera view. Within a limited search window, every tracker uses gradient information for convergence to the mean of the region-membership probability distribution (see [1] for details). The whole procedure is run for each video frame acquired from the two front view cameras.

The 3D positions of the tracked body parts are computed by triangulation using the recovered 2D image plane locations in each front camera view [21].

6 Volume Reconstruction

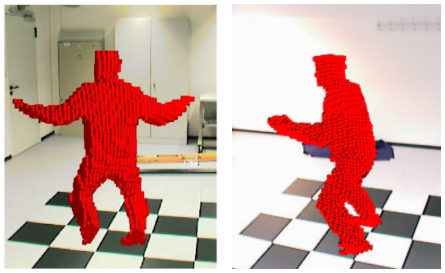

Figure 3: Example volume back-projected into 2 camera views

The visual hull of the moving subject is reconstructed in real-time using a shape-from-silhouette method comparable to those presented in [3] and [14]. The confined volume of space in which the person is allowed to move is regularly subdivided into volume elements (voxels).

Each client (Fig. 1) projects every volume element back into the silhouette images of both controlled camera views. If a voxel back-projects into the silhouette of the person in both views, it is classified as occupied space. The clients run-length-encode their partial visual hulls and transfer them to the server via LAN. The server reconstructs the complete visual hull by forming the intersection of the partial hulls from each client. A considerable speedup is achieved by precomputing the image plane coordinates of each re-projected voxel in every static camera view. Two example visual hulls reconstructed from four camera views can be seen in Fig. 3.

7 Skeleton Fitting

The skeleton fitting algorithm estimates the joint parameters of a multi-layer kinematic model for each time step t of a recorded motion sequence.

It uses the stored volume models and 3D location data of head, hands and feet from the online system (Fig. 1), as well as the model parameters in the previous time step $t-1$ as input (Fig. 4). The joint parameters for time $t=0$ are known since the person is required to stand in an initialization position.

The dimensions of the body model are adjusted to the dimensions of the moving person. This is either done by manually measuring the limb lengths and loading them into the application or by interactively marking shoulder, hip, elbow and knee locations in the two front camera views showing the person in initialization position. The lengths of all body segments can then be automatically derived. The thicknesses of the volumes attached to arms and legs are set by the user.

The human body is modeled as a 2-layer kinematic skeleton. The first layer of the model consists of a structure of 10 bone segments and 7 joints. Each joint spans a local coordinate frame which is defined by a rotation matrix $\mathcal{R}$ and a translation vector $\vec{t}$ relative to the preceding joint in the skeleton hierarchy.
The second layer refines the layer-1 structure by upper arm and forearm segments, as well as thigh and lower leg segments (Fig. 5). The volumetric extents of the corresponding limbs are modeled by means of point samples taken from cylindrical volumes centered around the segments, henceforth called cylinder samples (Fig. 5). Every pair of these new segments is connected via a 1-DOF revolute joint which serves as a simplified model of the elbow or knee joint. The lengths of the additional layer-2 segments are constant and known from initialization. Together with the corresponding layer-1 leg and arm segments, triangles are formed in

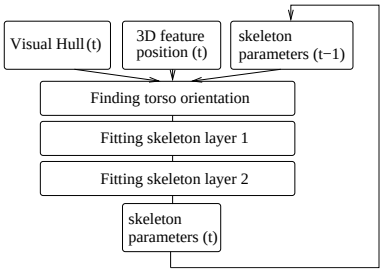

Figure 4: Skeleton fitting overview

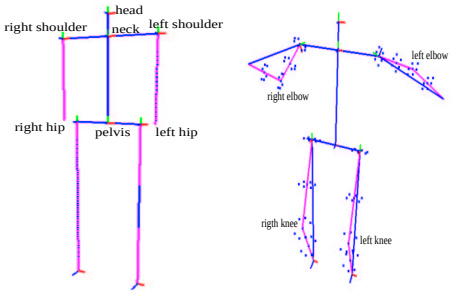

Figure 5: Skeleton layer 1 (l), Skeleton layer 2 (r)

which the lengths of the first layer bones vary during model fitting. The bending angles of the elbow and knee (henceforth denoted by ϕ) at each time step t are fully determined by the cosine theorem (see Sect. 7.2). The additional rotational degree of freedom (henceforth denoted by ρ) of the layer-2 arm and leg constructions around the corresponding layer-1 segment in each time step t is found using the cylinder samples and the visual hull voxels (Sect. 7.3).

The layer-1 model has 24 degrees of freedom in total. Layer 2 extends this by 4 degrees of freedom.

7.1 Finding the torso orientation

Pure optical tracking of the shoulder positions is difficult due to the lack of detectable salient features. However, the reconstructed volume can be used to find the shoulder position and torso orientation. The voxel positions are interpreted as a 3-dimensional data set with coordinate origin in its center of gravity. For this set a 3×3 covariance matrix $\mathcal{C}$ is computed. The 3 eigenvectors of the symmetric matrix $\mathcal{C}$, the principal components (PCs), denote the directions of strongest variance in the data and are mutually orthogonal. If the data is limited to the voxels corresponding to the torso of the person, the first principal component lies along the spine segment direction, the second along the connection between the shoulders, and the third is orthogonal to these two (see Fig. 6). For segmenting out the torso voxels, we make use of the skeleton model. A cylindrical volume around the spine axis (Fig. 6) is used to constrain the PC computation to the torso part. The algorithm to find the upper body orientation makes use of temporal coherence :

The parameterization of the skeleton model is known from the previous time step $t-1$. Assuming that the change in body orientation is small from time $t-1$ to time t, the position and orientation of the cylindrical volume at time $t-1$ are used to separate the torso part from the complete visual hull at time step t. The principal components of the torso volume at time t can now be computed.

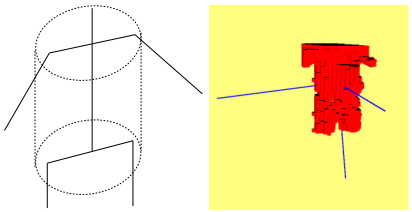

Figure 6: torso search volume (l) and the principal components of all the voxels inside the torso (r)

7.2 Fitting the first skeleton layer

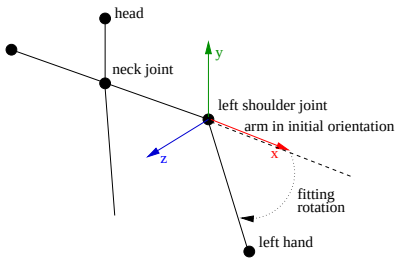

Figure 7: Fitting layer-1 arm segment

The skeleton dimensions are obtained from the initialization (Sect. 5). From feature tracking, the 3D locations of the head, the hands and the feet are known at each time step. Together with the torso orientation (Sect. 7.1), this enables fitting the layer-1 skeleton. The root of the model (located at the head) is translated to the known 3D head position. The neck bone is upright with respect to the global coordinate system at all times, hence the relative position of head joint center to neck joint center in world coordinates is fixed. The principal components from the torso orientation computation define the goal orientation for the neck joint local coordinate system at time step t. The corresponding neck joint rotation for time t is directly available by using the PC vectors as the column vectors of the rotation matrix $\mathcal{R}_{neck}(t)$. To keep the hip bones parallel to the floor level, the pelvis joint rotation is set to the inverse neck rotation. This constrains the set

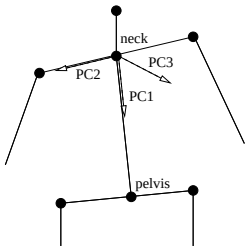

Figure 8: Torso aligned with principal component orientation

of allowed motions but enables quick model fitting which proves to be sufficient in all our experiments.

Now, the locations of the shoulder and hip joint are known for time step t. The distances between shoulders and wrists, as well as between the hips and the ankles are computed, and the lengths of the corresponding layer-1 segments are appropriately rescaled. Knowing the positions of the hands in the shoulder coordinate systems and the feet in the hip coordinate systems, the correct shoulder and hip rotations are straightforward to compute (Fig. 7). Fitting the layer-1 skeleton to the video footage is performed in real-time (Sect. 8).

7.3 Fitting the second skeleton layer

The volume data is used to find the values of the four additional degrees of freedom in layer 2. By means of the cosine theorem the elbow and knee angles of the second layer (ϕ) are directly determined. The lengths of the layer-2 arm and leg segments are constants. The distances of shoulders and hands, as well as of hips and feet at time t are obtained from the layer-1 skeleton. As an example, the distance d between shoulder and hand is depicted in Fig. 9. In order to find the additional rotation angle of the layer-1 arm and leg segments (ρ in Fig. 9), a maximal overlap of the visual hull and the layer-2 cylinder samples is computed. This is done only if there is a noticeable bending of elbows and knees, i.e., if a layer-1 segment is significantly shorter than the sum

of the attached layer-2 segment lengths. Otherwise, $\rho(t-1)$ is passed on.

The shapes of the arms and legs on layer 2 are modeled using point samples taken from cylindrical volumes centered around the layer-2 arm and leg segments (Fig. 5). The computation of the best layer-2 rotation $\rho(t)$ is a problem of optimally registering a set of these cylinder samples against the voxels in the visual hull volume. This is done by transforming the problem into searching a maximum of a goodness-of-fit function. For a given rotation ρ, let n be the number of cylinder samples that lie inside visual hull voxels. The goodness-of-fit function is defined as $match(\rho) = n^4$. It is sampled for ν equally spaced rotation angles ξ_l ($l = 0, \ldots, \nu-1$) which are taken from an interval $[\rho(t-1) - s, \rho(t-1) + s]$ centered around $\rho(t-1)$, where s determines the interval size. This is based on the assumption that the change of ρ is only small from $t-1$ to t.

The final rotation $\rho(t)$ of the arm segment is found by

$$\rho(t) = \frac{1}{\sum_{l=0}^{\nu-1} match(\xi_l)} \sum_{l=0}^{\nu-1} \xi_l \times match(\xi_l).$$

This particular match function is a heuristic which exaggerates good overlap scores. The method quickly converges towards good registrations at sub-voxel resolution, overcoming the quantization inaccuracy of the visual hull volume.

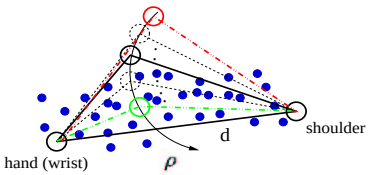

Figure 9: (L) Testing rotations between bounds of search interval. Small spheres represent visual hull voxels, cylinder samples are not drawn.

The model fitting routine reports the correct rotations for each model joint at time step t. Currently, this is done in form of a rotation matrix $\mathcal{R}$ for each joint. In addition, the elbow and knee angles, layer-2 rotation angles and the correct model root translation are reported for each time t. The accumulation of model fitting errors on layer 2 is prevented by searching for the best fit in a search interval at every time step. This way, false estimates of the magnitude of the search interval size can be corrected. The parameterization of arms and legs on two different layers cannot directly be applied to standard skeletons used in animation systems(e.g HAnim models). If the presented method is to be used to control virtual characters, an additional step has to be taken. The rotation matrices defined by the shoulder and hip joints have to be multiplied by matrices rotating the layer-1 arm and leg segments onto the corresponding upper arm and leg segments in the local coordinate system. This transform is straightforward to compute.

8 Results

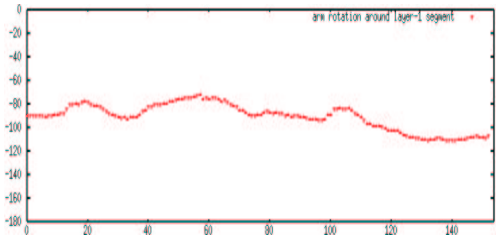

Figure 10: rotation angles for layer-2 arm segment

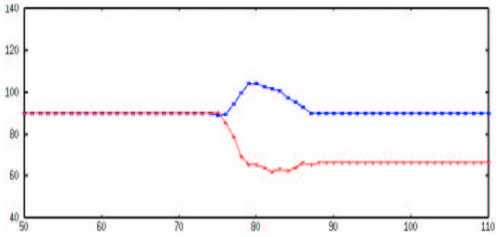

Figure 11: rotation angles for layer-2 leg segments while prostrating

The model fitting quality of the presented method is evaluated using test sequences obtained with our motion capture setup. The complete online system performing background subtraction, visual hull reconstruction, feature tracking and visual hull rendering can run at approximately 6-7 fps for a 64^3

PCA computation	4 ms
Torso segmentation	5.5 ms
Layer-1 fitting	16 ms
Fitting 1 layer-2 segment	211 ms

Table 1: Timing results

voxel volume (see [21]). Fig. 12 shows the layer-2 skeleton fitted to two body poses of a motion sequence. Our algorithm correctly recovers the skeleton configuration, in particular the torso orientation. In Table 1 the performance of each part of the algorithm is summarized. The values are obtained from experiments on a 1 GHz Athlon PC. The volume size is 64^3, and 64 cylindrical volume samples are attached to each layer-2 arm and leg structure. With the current implementation, the recovery of a single arm or leg segment rotation (last row in Table 1) is by far the most computationally expensive step. For an average motion sequence, a fitting frame rate of 1-2 fps is achieved.

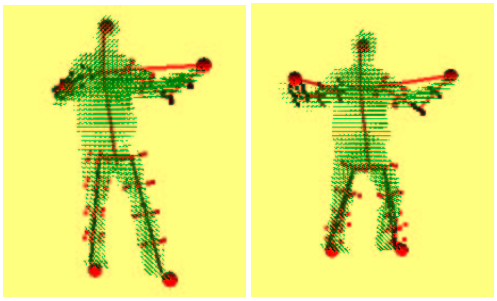

Figure 12: Layer-2 skeletons fitted to visual hulls. Red spheres mark tracked feature locations

A series of captured layer-2 arm segment rotations (ρ, see Sect. 7.3) for 153 consecutive time steps is shown in Fig. 10. A smooth sequence of angles is obtained even with a coarsely sampled voxel volume. Fig. 11 shows the change of angle ρ over time for both layer-2 leg segments while the observed person is prostrating. Since during this motion the knees are slightly moved outwards in opposite directions, the plot is symmetric. Flat parts in both plots mark the steps where no rotation angle computation for a layer-2 segment is performed since the limbs are almost fully stretched. A common problem in shape-from-silhouette approaches are shadow artifacts which arise if certain parts of the volume cannot be carved away since they are occluded in all camera views. When reconstructing human visual hulls, these artifacts typically exist in the form of overly voluminous arms and legs. Our approach can still recover the correct arm and leg configurations even in the presence of such reconstruction errors. A camera looking at the scene from the top is not required, and even with only 4 cameras looking from the side, robust fitting is possible. Further results can be found at http://www.mpi-sb.mpg.de/∼theobalt/VisualHullTracking.

9 Conclusion and future work

This paper presents a method to robustly fit a multi-layer kinematic skeleton to human motion simultaneously recorded from multiple camera views. The joint use of 3D feature tracking and shape-from-silhouette enables reliable fitting of a kinematic skeleton. The special multi-layer parameterization of this skeleton enables the use of an efficient fitting strategy based on volume registration. This method can correctly recover the arm and leg configurations even with only a few cameras and at a coarse voxel density in the visual hull. The feature tracking in the online system and constraints in the model parameterization currently limit the range of movements which can be captured. The fitting method itself, however, allows arbitrary rotations of the human actor around the vertical body axis.

Alternative registration techniques will be tested, and further improvement of the online system's tracking component will be considered. In the near future, the system will evolve into a real-time motion capture and character control application.

References

[1] G. Bradski. Computer vision face tracking as a component of a perceptual user interface. In *IEEE Workshop of Applications of Computer Vision*, pages 214–218, 1998.

[2] C. Bregler and J. Malik. Tracking people with twists and exponential maps. In *Computer Society Conference on Computer Vision and Pattern Recognition 98*, pages 8–15, 1998.

[3] 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 2000 IEEE Conference on Computer Vision and Pattern Recognition (Computer Society Conference on Computer Vision and Pattern Recognition 2000)*, volume 2, pages 714 – 720, June 2000.

[4] P. Eisert, E. Steinbach, and B. Girod. Automatic reconstruction of stationary 3-D objects from multiple uncalibrated camera views. *IEEE Transactions on Circuits and Systems for Video Technology: Special Issue on 3D Video Technology*, 10(2):261–277, March 2000.

[5] D.M. Gavrila. The visual analysis of human movement. *Computer Vision and Image Understanding*, 73(1):82–98, January 1999.

[6] D.M. Gavrila and L.S. Davis. 3D model-based tracking of humans in action: A multi-view approach. In *Computer Society Conference on Computer Vision and Pattern Recognition 96*, pages 73–80, 1996.

[7] M. Gleicher. Animation from observation: Motion capture and motion editing. *Computer Graphics*, 4(33):51–55, November 1999.

[8] I. Haritaoglu, D. Harwood, and L.S. Davis. W4: Who? When? Where? What? A real time system for detecting and tracking people. In *Conference on Automatic Face and Gesture Recognition 98 (Tracking and Segmentation of Moving Figures)*, pages 222–227, 1998.

[9] L. Herda, P. Fua, R. Plaenkers, R. Boulic, and D. Thalmann. Skeleton-based motion capture for robust reconstruction of human motion. In *Proceedings of Computer Animation 2000*. IEEE CS Press, 2000.

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

[11] J. C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, 1991.

[12] A. Laurentini. The visual hull concept for silhouette-based image understanding. *Pattern Analysis and Machine Intelligence*, 16(2):150–162, February 1994.

[13] B. Lok. Online model reconstruction for interactive virtual environments. *Symposium on Interactive 3D Graphics, pp. 69-72, 2001*, 2001.

[14] J. Luck and D. Small. Real-time markerless motion tracking using linked kinematic chains. In *Proceedings of the International Conference on Computer Vision, Pattern Recognition and Image Processing 2002 (CVPRIP02), in cooperation with JCIS 2002*, 2002.

[15] W. Matusik, C. Buehler, and L. McMillan. Polyhedral visual hulls for real-time rendering. In *Proceedings of 12th Eurographics Workshop on Rendering*, pages 116–126, 2001.

[16] W. Matusik, C. Buehler, R. Raskar, S.J. Gortler, and L. McMillan. Image-based visual hulls. In *Siggraph 2000, Computer Graphics Proceedings*, pages 369–374, 2000.

[17] R. Plankers and P. Fua. Tracking and modeling people in video sequences. *Computer Vision and Image Understanding*, 81(3):285–302, March 2001.

[18] M. Potmesil. Generating octree models of 3D objects from their silhouettes in a sequence of images. *Computer Vision, Graphics and Image Processing*, 40:1–20, 1987.

[19] K. Rohr. Incremental recognition of pedestrians from image sequences. In *Computer Society Conference on Computer Vision and Pattern Recognition 93*, pages 8–13, 1993.

[20] R. Szeliski. Rapid octree construction from image sequences. *Computer Vision, Graphics, and Image Processing. Image Understanding*, 58(1):23–32, 1993.

[21] C. Theobalt, M. Magnor, P. Schüler, and H.-P. Seidel. Combining 2D feature tracking and volume reconstruction for online video-based human motion capture. In *Proceedings of 10th Pacific Conference on Computer Graphics and Applications 2002*, to appear.

[22] R.Y. Tsai. An efficient and accurate camera calibration technique for 3d machine vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'86)*, pages 364–374, June 1986.

[23] S. Weik and C.-E. Liedtke. Hierarchical 3D pose estimation for articulated human body models from a sequence of volume data. In *Robot Vision*, 2001.

[24] C.R. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfinder: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):780–785, 1997.