International Journal of Image and Graphics

COMBINING 2D FEATURE TRACKING AND VOLUME RECONSTRUCTION FOR ONLINE VIDEO-BASED HUMAN MOTION CAPTURE

CHRISTIAN THEOBALT
MARCUS A. MAGNOR
PASCAL SCHÜLER
HANS-PETER SEIDEL

Max-Planck-Instituf für Informatik,
Stuhlsatzenhausweg 85, 66123 Saarbrücken, Germany
{*theobalt,magnor,schueler,hpseidel*}*@mpi-sb.mpg.de*

Received (Day Month Year)
Revised (Day Month Year)
Accepted (Day Month Year)

The acquisition of human motion data is of major importance for creating interactive virtual environments, intelligent user interfaces, and realistic computer animations. Today's performance of off-the-shelf computer hardware enables marker-free non-intrusive optical tracking of the human body. In addition, recent research shows that it is possible to efficiently acquire and render volumetric scene representations in real-time. This paper describes a system to capture human motion without the use of markers or scene-intruding devices. Instead, a 2D feature tracking algorithm and a silhouette-based 3D volumetric scene reconstruction method are applied directly to the image data. A person is recorded by multiple synchronized cameras, and a multi-layer hierarchical kinematic skeleton is fitted to each frame in a two-stage process. The pose of a first model layer at every time step is determined from the tracked 3D locations of hands, head and feet. A more sophisticated second skeleton layer is fitted to the motion data by applying a volume registration technique. We present results with a prototype system showing that the approach is capable of running at interactive frame rates.

Keywords: Marker-less Human Motion Capture; Feature Tracking; Volume Reconstruction; Shape-from-Silhouette; Kinematic Body Model.

1. Introduction

In recent years, the task of human motion capture has brought together researchers from computer vision and computer graphics. Motion capture is the process of finding a mathematical description of observed motion in terms of an underlying model. Finding such a mathematical description of human motion has long been of scientific interest.

At the end of the 19th century, Edward Muybridge already undertook studies in human and animal locomotion from camera images [1]. In the 1970's Johannson

examined the psychophysical aspects behind visual motion perception of humans by examining so-called Moving Light Displays, light emitting markers on the body of a person [2]. Nowadays, the number of scientific disciplines interested in accurate human motion acquisition and the range of possible applications for motion capture systems is manifold:

Realistic animation of virtual characters is an important choreographic element in today's feature films and computer games [3]. In order to animate characters in a natural looking way, accurate motion data of real actors performing are needed. In sports and bio-mechanics research, human motion capture can be a great support for the analysis of gait anomalies or inefficient motion cycles in athletic movements [4].

Another important application area is surveillance, where image sequences of moving people are to be interpreted automatically [5]. Possible applications are automatic crime detection and general monitoring scenarios where a high-level interpretation of the video footage is needed.

The design of more user-friendly machines has been a field of intensive research for years. Many scientists aim at equipping computers with means to visually perceive the environment. The first step towards vision-based user interfaces are gesture recognition and interpretation systems enabling a more intuitive way of human computer interaction [6,7,8].

The advent of new media technology has opened new challenging application areas. The ongoing development of video broadcasting technology, as well as video-on-demand and teleconferencing makes necessary efficient encoding algorithms for image data transmission [9]. Since many video sequences are centered around human actors, model-based encoding schemes that are based on transmitting a 3D model of the person and its motion parameters instead of the full video stream can help to significantly reduce the bandwidth needs [10,11]. The MPEG-4 standard meets the demands by defining so-called body-animation parameters for 3D shape models of humans [12,13].

The challenge, however, lies in the fact that in video-based encoding and many other applications the motion parameters have to be estimated without using any form of intrusion into the scene. Unfortunately, existing commercial human motion capture systems fail to fulfill this requirement.

Commercial human motion capture systems existing today can be mainly classified into the following categories [3]: Mechanical, electromagnetic and optical systems.

Mechanical systems consist of an exoskeleton structure that needs to be attached to the body of the performing actor and the angles of the joints are measured directly. Electromagnetic systems require the person to wear emitters or receivers whose positions and orientations are measured.

This paper describes a video-based motion capture system for full body motion that works without the use of optical markers. The presented method is based on and extends our work that has been previously published in [14] and [15].

The described system enables fitting of a complex kinematic human body model to human motion based on multi-view video footage depicting a moving person. A

color-based feature tracking algorithm is combined with a fast silhouette-based volume reconstruction and registration method. The tracking algorithm is applied in the image planes of two of the available camera views to track the locations of salient body parts such as the head, the hands, and the feet. The 3D locations of theses parts are computed via triangulation. This information is sufficient to fit a first layer kinematic skeleton to the motion data at each time frame. The configuration of elbows, knees and the torso are found by employing a volumetric reconstruction of the person at each time step by means of a shape-from-silhouette approach. The voxel-based volume of the person is used to fit a more complex second layer kinematic skeleton. The result of the algorithm are the joint parameters of the kinematic skeleton model for every frame of the observed motion sequence.

The rest of the paper begins with a review of relevant related work in Sec. 2. An overview of the motion capture system architecture is given in Sec. 3. The applied silhouette computation method is described in Sec. 5, and the color-based feature tracking is explained thereafter in Sec. 6. In Sec. 7 the shape-from-silhouette approach for the computation of the visual hulls is illustrated. The applied multi-layer body model and the fitting of the model layers to the volume data is presented in Sec. 8. Results with our prototype system are described in Sec. 9, and the paper concludes in Sec. 10.

2. Related work

Commercial optical motion capture systems only work in a very constrained scene setup. The person to be tracked has to wear markers, and many cameras have to observe the scene from different viewpoints to prevent occlusions [16,17]. The first marker-free vision-based motion capture systems have only recently become feasible thanks to increasing computational power of off-the-shelf hardware [5,18].

Non-real time approaches [19,20,21] use features extracted from video frames to fit simple kinematic skeleton models with volumetric limb representations to human body poses. Image differencing [22] and silhouette skeletonization [23] are also used to fit simple kinematic models to video streams. The use of TV image sequences for the acquisition of articulated motion is presented in [24]. In [25] an implicit-surface human body model is fitted to video material. More recently, Bregler et. al. use the combination of optical flow, a probabilistic region model, and the twist parameterization for human body joints to fit a kinematic model to video footage [26]. Existing real-time systems use comparably simple models, such as probabilistic region representations and probabilistic filters for tracking [27], or combine feature tracking and dynamic appearance models [28]. Unfortunately, these approaches cannot support sophisticated human body models like kinematics skeletons or dynamic body representations.

At the same time, a new method for the acquisition and efficient rendering of volumetric scene representations obtained from multiple camera views, known as shape-from-silhouette or the visual hull [29], has been proposed. Early approaches in

the field construct discrete three-dimensional grids of volume elements (voxels) from a set of silhouette images of a scene, a method known as voxel carving or volume intersection [30,31,32]. More recently, it was shown that a polyhedral representation of the visual hull can be acquired and rendered in real-time [33]. An image-based approach to visual hull construction samples and textures visual hulls along a discrete set of viewing rays [34]. State-of-the-art graphics hardware can be used to accelerate the construction of slices of the visual hull [35]. Most work focuses on improving the quality of the reconstructed scene [36].

Only recently, methods have been presented that use the reconstructed visual hulls of a person to capture human motion. In [37] ellipsoids are used to represent the torso, the arms and the legs in a human body model. An expectation-maximization-like algorithm is applied to fit these ellipsoidal shells to the voxel volume of the person in real-time. Luck et al. [38] use a kinematic skeleton without an accompanying surface representation that is fitted to the volume data by means of forces exerted from the volume elements to the bones. Their system is also capable of running in real-time.

Mikić et al. [39] fit a body model that consists of simple shape primitives to volumes of a moving person in an off-line process. Compared to the previous approach, the employed body model is more detailed. It explicitly enforces the connectivity of body parts and the limb segments are modeled in more detail. An Extended Kalman Filter is used in the tracking algorithm.

A kinematic skeleton parameterized by 32 joint parameters and consisting of 15 segments covered with a triangle mesh surface representation is used in [40]. The body model is fitted to the visual hull data by minimizing a distance metric.

Weik and Liedtke [41] fit a kinematic skeleton with attached surface patches to the visual hull data of a moving person by applying a hierarchical iterative closest point procedure.

Unfortunately, most of the previously mentioned volume-based human motion capture methods fail to robustly resolve body poses in which the limbs are positioned very close to the torso. Our new method handles these situations more robustly by combining the information originating from a fast feature tracking algorithm with the information stemming from the reconstructed visual hulls of a person.

3. System Overview

3.1. *Software architecture*

In Fig. 1 the architecture of our motion capture system is illustrated. Currently, there are up to 3 clients, each of which is running on a 1 GHz single processor Athlon^{TM} PC. One client computer controls two Sony^{TM} DFW-V500 IEEE1394 video cameras that run at a resolution of 320x240 in color mode. Each client performs a background subtraction (Sec. 5), as well as the computation of a partial visual hull, i.e. a visual hull that is reconstructed from the two connected cam-

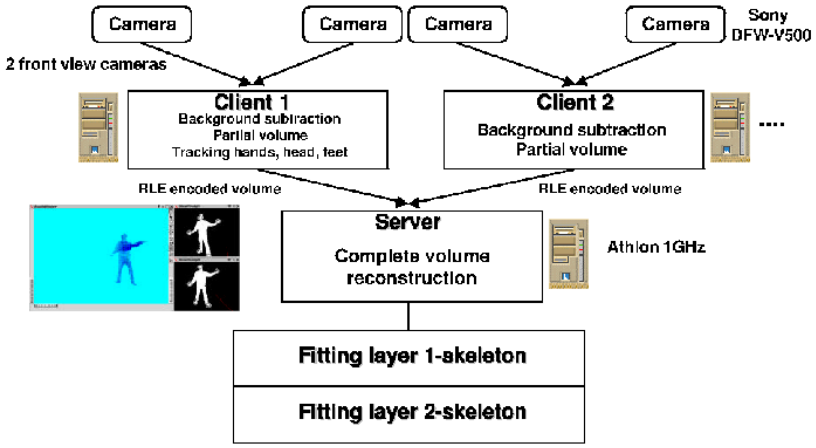


Fig. 1. Motion capture system architecture.

era views only, in real-time. Additionally, the client controlling the two front view cameras identifies and tracks the positions of hand, head and feet at interactive frame rates (see Sec. 5 and Sec. 6). The partial visual hulls from both clients are transferred to a server which builds the complete visual hull and renders it using OpenGL. The server also sends the trigger signals to the cameras for synchronization. The software architecture scales easily to more cameras and more clients by employing a hierarchical network structure. In the following, the previously described client-server components are referred to as the online system. The model fitting is currently implemented as a separate application which works with recorded visual hulls and 3D locations acquired with the online system.

3.2. *Scene setup*

The person to be tracked is supposed to move inside a confined volume. The scene is observed from up to six synchronized cameras (variable setups using 4-6 cameras are possible) that are arranged in a convergent setup around the center of the scene. We require that two of these cameras are observing the person from nearby positions in front (Fig. 2). The person moves barefooted and needs to face these cameras allowing only limited rotation around the vertical body axis. The cameras are calibrated using Tsai's method [42].

4. Initialization

In the first frame, the person is supposed to stand in an initialization position, facing the two frontal cameras, with both legs next to each other and spreading the

arms horizontally away to the side at maximal extent. The model fitting application (see Fig. 1) takes visual hulls and 3D feature locations that are saved by the online system as input.

The dimensions of the kinematic skeleton need to be adjusted to the body dimensions of the moving person. This is either done by manually measuring the limb lengths and loading them into the application, or by an interactive step. In this step the user marks shoulder, hip, elbow and knee locations in the two camera frames showing the person in the initialization position from front. Together with the tracked positions, the 3D locations of all joints can be computed and the lengths of the body segments are derived. The thicknesses of the arms and legs are set by the user.

5. Silhouette Segmentation

The segmentation step consists of two parts. First, the person's silhouette is separated from the background in each camera perspective. Then, the silhouettes obtained from the front-view cameras are segmented in order to identify hand, feet and head. The former step is performed for every time step, the latter is performed for the initial frame only.

Separating the person from the background is done by using a background distribution for each camera perspective consisting of a mean image $\mu(x, y)$ and a standard-deviation image $\sigma(x, y)$. These are generated from several consecutive video frames of the static background scene. For the silhouette extraction a method originally proposed in [37] is used which proves to be robust against shadows cast by the person on the floor and the walls. If a pixel $p(x, y)$ differs in at least one color channel by more than an upper T_u threshold from the background distribution

$$\|p(x, y) - \mu(x, y)\| > T_u \cdot \sigma(x, y) \quad , \tag{1}$$

it is classified as foreground. If its difference from the background statistics is smaller than the lower threshold T_l in all channels it is surely a background pixel. All pixels

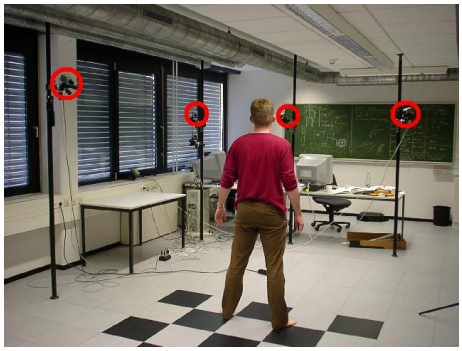


Fig. 2. Scene setup: Camera studio, four visible cameras marked with circles.

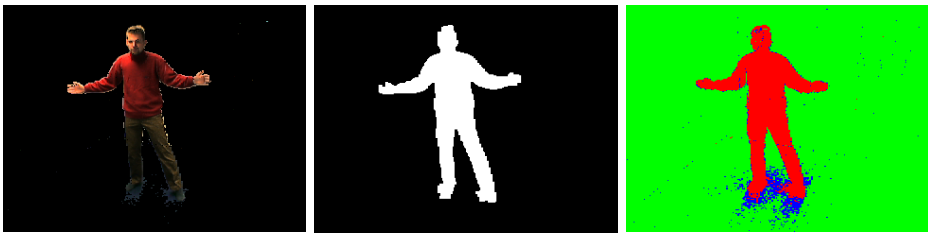


Fig. 3. Video frame after background subtraction (l) and the corresponding silhouette (m). Shadows cast by the person onto the floor are identified by the background subtraction (r).

which fall in between these thresholds are possibly in shadow areas. Shadow pixels are classified by a large change in intensity but only small change in hue. If $p(x, y)$ is the color vector of the pixel to be classified, and $\mu(x, y)$ is the corresponding background pixel color vector, their difference in hue is

$$\Delta = \cos^{-1}\left(\frac{p(x,y)\cdot\mu(x,y)}{\|p(x,y)\|\|\mu(x,y)\|}\right) \quad . \tag{2}$$

If $\Delta > T_{angular}$ the pixel is classified as foreground, else as shadow. A 0/1-silhouette image for each camera is computed this way.

The binary silhouette images of the person standing in the initialization position seen from the two front view cameras are segmented using a Generalized Voronoi Diagram (GVD) decomposition (see Fig. 4). Often used in free space segmentation of cognitive topological maps of mobile robots [43,44,45], 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 critical points, i.e. points locally minimizing the clearance to the silhouette boundary. These points are used as centers for border lines between adjacent regions in the silhouette. These lines connect the two boundary pixels closest to the critical point (Fig. 4). Since in the silhouette the boundaries to the head, hand and feet are identified by constrictions, the algorithm nicely segments these parts from the rest of the body. This way, the position and the regional extent of these body parts are extracted.

The connectivity of the recovered silhouette regions can be represented by a graph connecting the region centers. For the case of the human silhouette in the initialization position, the five terminating nodes in the connectivity graph correspond to the head, the hands and the feet of the person.

6. Tracking head, hands and feet

To track the motion of body parts in 2D, we implemented a fast tracking strategy. We use a continuously adaptable mean-shift algorithm which is capable of tracking

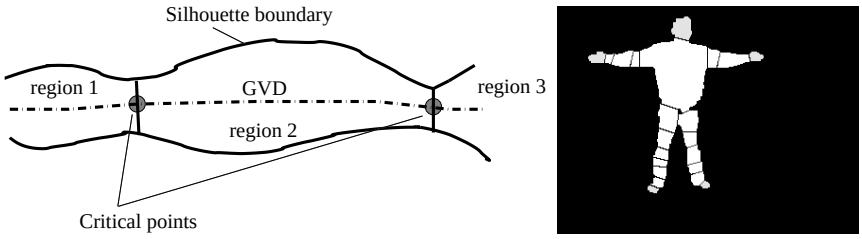


Fig. 4. GVD with critical points (l). Silhouette segmented by Generalized Voronoi Diagram decomposition (r).

the mean of dynamically changing probability distributions, originally developed for face tracking [46,47]. From the segmentation step, it is known which pixels belong to the head, the hands and the feet for both front camera views at $t = 0$. The HSV color is the principal cue used for tracking body parts. The color range of human skin in the camera view is different depending on lighting conditions and camera adjustment. Since the locations and extents of the head, the hands and the feet in the image planes are known, average skin colors for each tracked feature can be computed. These values are used to define tolerance intervals in color space. For the colors in these intervals, color histograms are computed based ont the video frames with the person in initialization position.

After the first video frames, the algorithm proceeds as follows. For each new frame and for each tracked feature, an intermediate gray-scale image is computed that contains for each pixel an approximation to the probability of belonging to the body part under consideration. This can be done by back-projecting the appropriate color histogram into the corresponding video frames after background subtraction. Alternatively, we can simply filter out all pixels in the allowed color interval and set all pixels passing the test to the maximum pixel value. In practice, this leads to fast convergence of the tracking algorithm.

We use a separate continuously adaptable mean shift tracker for each of the five body parts in both front views that takes the intermediate gray-scale images as input. The algorithm iteratively repositions the center of a rectangular search window to the mean of the pixel values within the search area. The tracking algorithm terminates if no further change in search window position is performed (see [46] for details). Starting with the mean position in the previous frame, the center of the search window after convergence is taken as the new body part position in the current frame. At time step $t = 0$ the trackers are initialized with the center positions found during the Voronoi decomposition step.

The whole procedure runs for each pair of video frames acquired from the two front view cameras. Figure 5 shows a screen-shot of our system where the tracked body parts are marked by circles. We assume that the colors of the head, the hands and the feet are sufficiently different from the colors of the clothes that the person

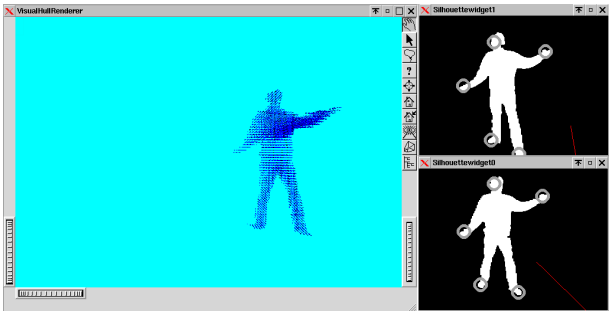


Fig. 5. A screen-shot of the server application showing the visual hull (l) and silhouettes with tracked feature locations (r).

wears. Head, hands and feet colors need to be similar in HSV space for our method to work properly. Requiring that the person moves barefooted is the easiest way to fulfill this constraint. The drawback of the method is that in case of overlapping body parts, the trackers can be mislead.

Once their locations in the front camera views are determined, the 3D positions of the body parts are computed via triangulation. We assume that the tracked centroids of the hands correspond to the projected wrist joint locations, the centroids of the feet to the ankle joint locations, and the centroid of the head to the model root joint.

7. Volume reconstruction

From the silhouettes of the moving person, we reconstruct a voxel-based approximation to the visual hull [29] at every time step. Our approach adapts the voxel carving method and is similar to the algorithms presented in [37] and [38].
The box in space in which the person is allowed to move is subdivided into a regular grid of volume elements.

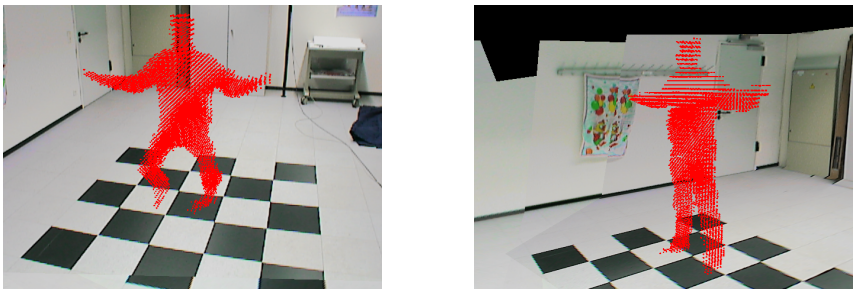


Fig. 6. Visual hulls reconstructed from 4 camera views, each voxel is drawn as small box. The volumes are rendered into a model of the acquisition room.

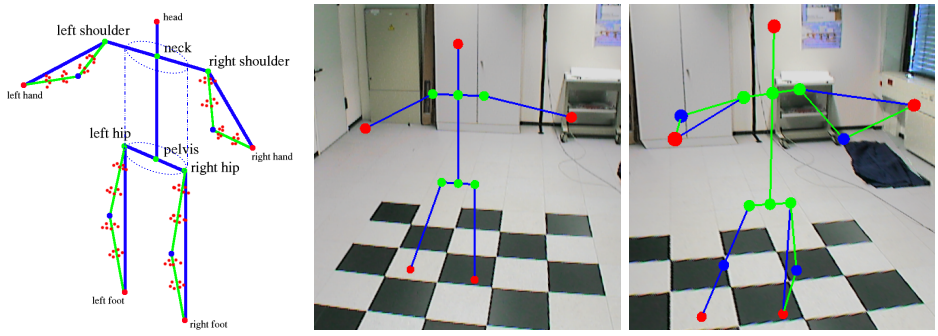


Fig. 7. The left image shows skeleton layer 1 and layer 2 with the attached cylinder samples and the cylindrical torso area. Skeleton layer 1 (m) and skeleton layer 2 (r) rendered into a model of the camera room.

From the camera calibration, the camera matrices are know which enables the computation of image plane locations of world space points for each camera view. In our distributed implementation, each voxel is simultaneously projected into the views of the two cameras connected to one client computer. If it projects into the silhouettes of the person in both views, it is classified as occupied space. This way, each client computes a partial visual hull that is reconstructed from two camera views only. The partial hulls from each client i, $\mathcal{V}_i$, are run-length-encoded and transferred to the server application via LAN. On the server, the complete visual hull $\mathcal{VH}$ is constructed by intersecting the volumes, $\mathcal{VH} = \bigcap_i V_i$. The intersection can be efficiently implemented using bitwise boolean operators. The voxel projections can be precomputed for each static camera view. Two example visual hulls reconstructed from four camera views can be seen in Fig. 6.

8. 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 , as well as the model parameters in the previous time step $t-1$ as input (Fig. 8). In a three-step procedure, the orientation of the torso is estimated first, then the layer-1 skeleton is fitted, and as a last step, the refined layer-2 skeleton is adapted to the body pose at time t. The joint parameters for time $t = 0$ are known since the person is required to stand in an initialization position.

8.1. *The Skeleton*

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. 7). 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. 7). 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 (Fig. 8). The lengths of the additional layer-2 segments are constant and known from initialization ($l_{forearm}$ and $l_{upperarm}$ in Fig. 8), the lengths of the attached layer-1 segments vary during the fitting of layer 1 (l_{whole} in Fig. 8). Together with the corresponding layer-1 leg and arm segments, triangles are formed in 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 Sec. 8.3). 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 (Sec. 8.4).

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

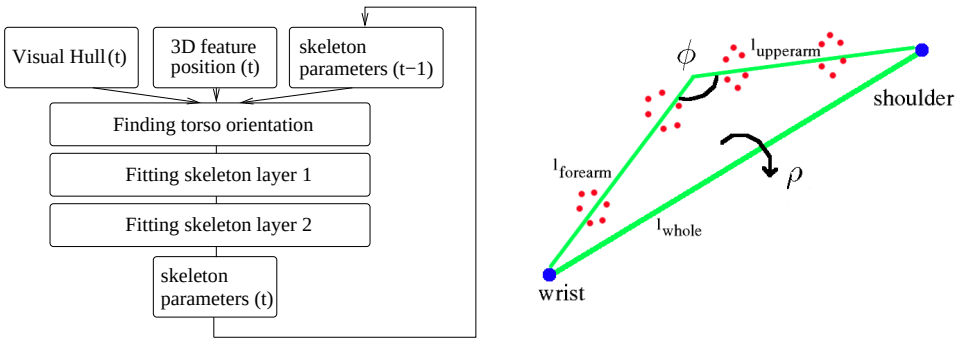


Fig. 8. Skeleton fitting steps overview (l). Arm structure of model layer 2 (r).

8.2. *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. 9). For segmenting out the torso voxels, we make use of the skeleton model. A cylindrical volume around the spine axis (Fig. 7) 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.

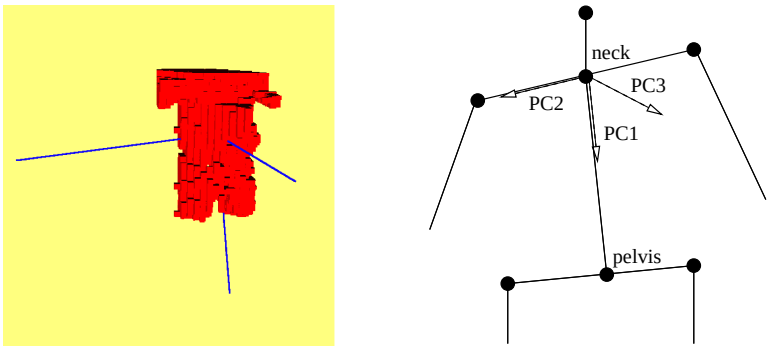


Fig. 9. The principal components of all the voxels inside the torso (l). Aligning the skeleton with the recovered torso orientation (r).

8.3. *Fitting the first skeleton layer*

The 2D feature tracking (Sec. 6) reports a set of 3D goal locations for the head, the hands and the feet. From the initialization, the skeleton dimensions are known. The neck bone is assumed to be upright at all time steps, so that the 3D location of the neck joint in world coordinates is known from the 3D location of the head. The model root located at the head is translated to match the triangulated 3D head position at t.

The principal components of the torso voxels define the goal orientation for the neck joint local coordinate system at time step t (Fig. 9). The corresponding neck joint rotation for time t is directly available by using the PC vectors as the column vectors of the rotation matrix $\mathcal{R}_{neck}$. To keep the hip bones parallel to the floor level, the pelvis joint rotation is set to the inverse neck rotation.

The locations of shoulder and hip joint in world coordinates as well as the locations of hands and feet are known. The distances between the left and right

shoulder and hand as well as the left and right hip and foot are computed, and the lengths of the corresponding layer-1 segments are adapted to these values.

The skeleton is represented as a hierarchical kinematic chain. Each joint corresponds to a rotation matrix $\mathcal{R}_j$ and a translation $\vec{t}_j$, which can be represented as a combined matrix $\mathcal{A}(R_j, \vec{t}_j)$ in homogeneous coordinates. To find the rotation transformation $\mathcal{R}_{shoulder}$ of the shoulder and hip joints, the following procedure is applied which is illustrated using an arm as an example. The layer-1 arm segment is assumed to be aligned with the x-axis of the coordinate frame spun by the shoulder joint. Knowing all preceding joint transformations in the skeleton hierarchy and assuming that $\mathcal{R}_{shoulder}$ is the identity rotation $\mathcal{I}$, the position of the hand in the shoulder coordinate frame is computed. The actual rotation of the shoulder joint $\mathcal{R}_{shoulder}$ at time t is the rigid body transform that aligns the tip of the arm segment with the hand position. This rotation is straightforward to compute (Fig. 10). Since the translation component of the complete shoulder joint transform is known from the skeleton structure, $\mathcal{A}(R_{shoulder}, \vec{t}_{shoulder})$ is completely determined. The same procedure applies to the leg segments.

8.4. *Fitting the second skeleton layer*

Once the model parameters are found for the first skeleton layer, the additional degrees of freedom of the second model layer are recovered by using the visual hull information. During the fitting step of model layer 1, the lengths of arm and leg segments are recomputed for each time step. Knowing the lengths of the additional two segments of arms and legs enables computing the elbow and knee joint angles (ϕ in Fig. 8) directly using the cosine theorem. In order to find the additional angle $\rho(t)$ of the layer-1 arm and leg segments (see also 8.1), a maximal overlap between the set of cylinder samples attached to the layer-2 model and the voxel data obtained from the visual hull is searched. The search procedure is as follows, using the arm segment as an example:

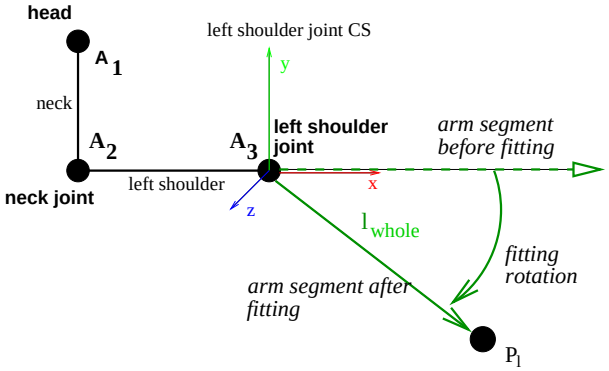


Fig. 10. Illustration of layer-1 fitting using the left arm segment as an example.

Making use of the temporal coherence, we start with the rotation of the arm in the previous frame, $\rho(t-1)$, and rotate the arm segment to ν equidistant angles ξ_l in the interval $[\rho(t-1)-s, \rho(t-1)+s]$, with s defining the search neighborhood size. For each such orientation, ξ_l, a quality measure for the overlap between the cylinder samples and the visual hull, $match_l$, is computed which is the higher the better the model fits to the voxel set. For each cylinder sample, the corresponding voxel it lies in is computed (see Fig. 11) . If n is the number of these voxels which belong to the visual hull (i.e. are filled), then n^k is the overlap match score for the current configuration ξ_l, where a value of $k = 4$ is used for best performance.
Using the set of ν match scores, the final rotation $\rho(t)$ of the arm segment is found by computing the center of gravity of the set $\Xi = \{\xi_l \times match_l \mid l = 1, \ldots, \nu\}$, the set of angles ξ_l each multiplied by its corresponding match score

$$\rho(t) = \frac{1}{\sum_{l=0}^{\nu-1} match_l} \sum_{l=0}^{\nu-1} \xi_l \times match_l \quad . \tag{3}$$

This particular match function is a heuristic which exaggerates good overlap scores. The procedure for the leg segments is the same. Although the difference between match scores for neighboring ξ_l can be very small, this approach still allows us to recover small changes in rotation from $t-1$ to 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.

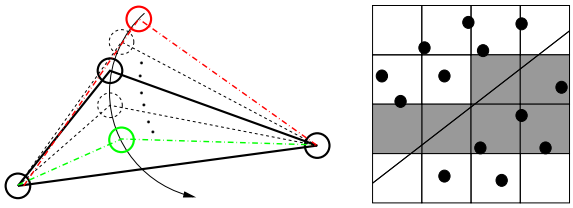


Fig. 11. (l) Testing rotations between search interval bounds (stippled lines), (r) slice through voxel volume showing overlap between samples (black dots) and voxels (gray boxes). A volume sample overlaps with a voxel if it is contained within the voxel's cubic volume.

The fitting step for a layer-2 segment is only performed if the corresponding knee or elbow joint is sufficiently bended. If this is not the case, the rotation angle $\rho(t-1)$ from the previous time step is used for this segment at time t.

For each video frame, the 2-step fitting procedure results in a set of model parameters describing the body pose. These parameters can be easily used to animate any artificial character based on a similarly structured skeleton.

Table 1. Timing results.

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

9. Results

The system is tested on several sequences of a person moving in front of the camera setup shown in Fig. 2. Figure 12 shows sample frames taken from an example motion sequence to which the skeleton model was fitted. From the different viewing positions it can be seen that the complete human skeleton is nicely fitted to the volumes of the moving person. The orientations of the shoulders and the torso are also correctly recovered over the whole sequence.

The number and positions of the cameras are crucial for the quality of the visual hull. Typical reconstruction errors produced by shape-from-silhouette approaches are visibility artifacts observable as arms or legs that are too thick, also known as phantom volumes. In the case of the visual hulls of humans, these artifacts arise in the from of voxel planes around the arms or legs in which the skeleton must lie (Fig. 12). Our approach can still recover the correct arm and leg configurations in the presence of these visual hull errors. A camera looking at the scene from the top is not required, and even with as few as four cameras looking from the side, robust fitting is possible.

The combination of feature tracking and volume reconstruction makes the system more robust. The knowledge of correct head, hands and feet positions enables correct model fitting even in cases that are problematic for pure volume-based motion capture approaches [37,40]. For instance, if the arms are very close to the body the feature tracking prevents them from getting stuck in the torso volume.

The combined visual hull reconstruction, background subtraction, feature tracking and visual hull rendering can run at approximately 6-7 fps for a 64^3 voxel volume using 2 client computers and a server. Measurements show that currently feature tracking consumes over 30% of total computation time. Furthermore, we experience a network overhead in our current implementation, since the frame rates of one client running independently without sending data to the server can reach up to 19 fps (measured using the internal camera trigger). The performance of the model fitting depends on the chosen parameters, such as the number of cylinder samples and angular search steps. For an average motion sequence a model fitting frame rate of 1-2 fps is achieved. In Table 1, the timings that we obtained while using 256 cylinder samples in total and 15 angular search steps are summarized.

With the current implementation, the recovery of a single layer-2 arm or leg segment rotation is by far the computationally most expensive step. Fitting the layer-1 model can be done at almost no cost. Higher frame rates can be achieved if less cylinder samples and less search steps are used.

More results including videos of the system in action can be found at http://www.mpi-sb.mpg.de/∼theobalt/VisualHullTracking.

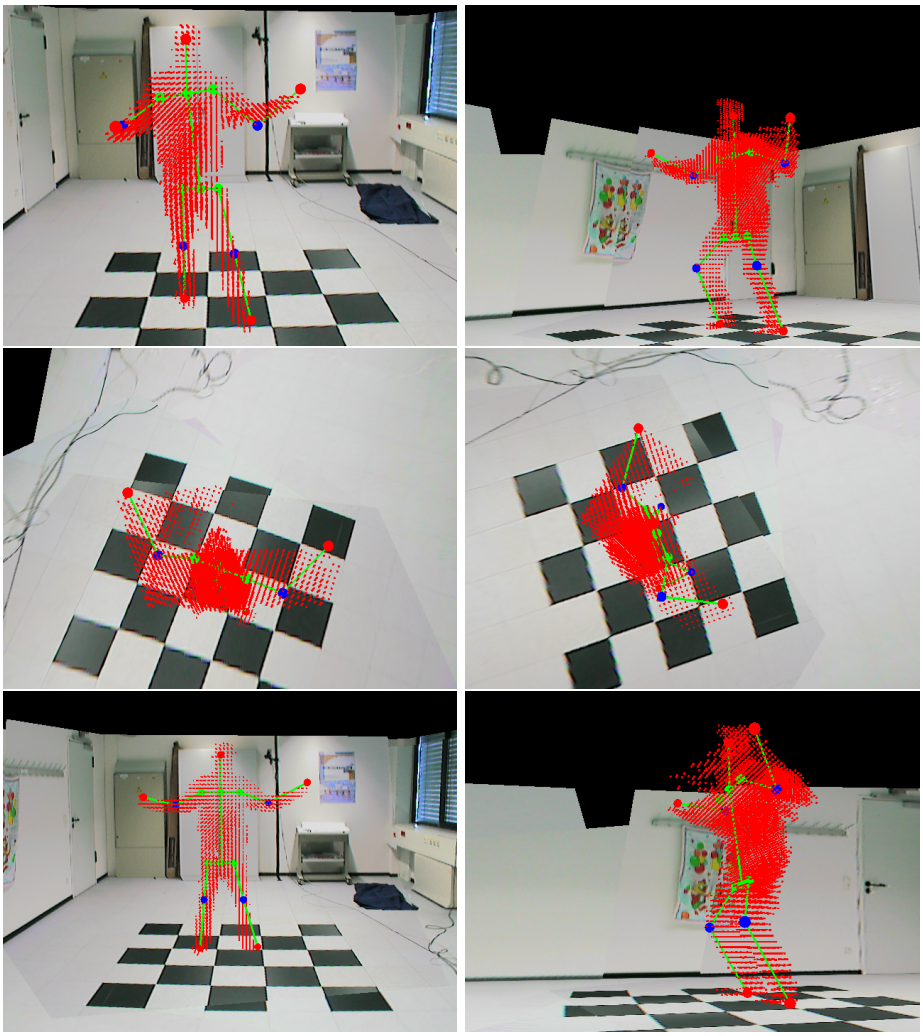


Fig. 12. Skeleton fitted to visual hulls (rendered as point sets) of a moving person. In this sequence, the person is recorded from four camera views. In the middle row reconstruction errors in the visual hull arising as arms which are too thick can be seen. These "phantom volumes" are due too insufficient visibility from the input camera views and are typical for visual hull methods. The fitting method can nonetheless correctly recover the body pose.

10. Conclusion

In this paper we present a method that combines color-based feature tracking and 3D scene reconstruction from silhouettes for human motion capture. The algorithm enables fast fitting of a kinematic skeleton model to the video footage recorded simultaneously from multiple video cameras. The feature tracking enables fitting of a simplified skeleton to the motion data. The special multi-layer parameterization enables the alignment of a more complex skeleton with the body poses in a second step. This layer-2 skeleton features a special representation for arm and leg segments including cylinder samples attached to the skeleton. The presented method uses the reconstructed volumetric visual hull to find the correct configuration of the kinematic skeleton at every time step by means of a volume registration technique.

Results of a prototype implementation capturing the motion of a human performer demonstrate the system's ability to fit the skeleton in real-time and a more detailed skeleton at near interactive frame rates. This hybrid approach of combining feature tracking and volume reconstruction is found to be capable of correctly finding human body configurations even in the presence of typical visibility artifacts 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.

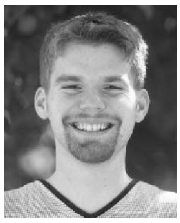
In the future, the model fitting step and the visual hull reconstruction will be integrated into one real-time motion capture and character control application. The use of a dynamic motion model for feature tracking is also another area of our research. Furthermore, extending the method to handle a higher range of body orientations by pose-dependent selection of cameras for tracking is currently considered.

References

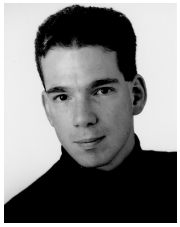
1. E. Muybridge. *Animal Locomotion*. 1887.
2. G. Johannson. Visual perception of biological motion and a model for its analysis. *Perception and Psychophysics*, 14(2):201–211, 1973.
3. A. Menache. *Understanding Motion Capture for Computer Animation and Video Games*. Morgan Kaufmann, 1995.
4. T. Calvert and M. Chapman. *Analysis and Synthesis of Human Movement*, pages 432–474. Academic Press, 1994.
5. D.M. Gavrila. The visual analysis of human movement. *Computer Vision and Image Understanding*, 73(1):82–98, January 1999.
6. V.I. Pavlovic, R. Sharma, and T.S. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. *Pattern Analysis and Machine Intelligence*, 19(7):677–695, July 1997.
7. T. Starner, J. Weaver, and A.P. Pentland. Real-time american sign language recognition using desk and wearable computer based video. *Pattern Analysis and Machine Intelligence*, 20(12):1371–1375, December 1998.

8. S. Malassiotis, N. Aifanti, and M.G. Strintzis. A gesture recognition system using 3D data. In *Proceedings of the 1st International Symposium on 3D Data Processing Visualization and Transmission (3DPVT-02)*, pages 190–193. IEEE Computer Society, 2002.
9. P. Eisert. *Very low bit-rate video coding using 3-D models*, volume 20 of *Berichte aus der Kommunikations- und Informationstechnik*. Shaker-Verlag, 2001.
10. N. Grammalidis, G. Goussis, G. Troufakos, and M.G. Strintzis. Estimating body animation parameters from depth images using analysis by synthesis. In *Proc. of Second International Workshop on Digital and Computational Video (DCV'01)*, page 93ff, 2001.
11. S. Weik and C.-E. Liedtke. Three-dimensional motion estimation for articulated human templates using a sequence of stereoscopic image pairs. In *VCIP99*, 1999.
12. T.K Capin and D. Thalmann. Controlling and efficient coding of mpeg-4 compliant avatars. In *Proc. IWSNHC3DI'99*, Santorini, Greece, 1999.
13. ISO/IEC. Overview of the MPEG-4 standard. *ISO/IEC JTC1/SC29/WG11 N2323, http://www.cselt.it/mpeg/standards/mpeg-4/mpeg-4.htm*, July 1998.
14. C. Theobalt, M. Magnor, P. Schueler, and H.-P. Seidel. Combining 2D feature tracking and volume reconstruction for online video-based human motion capture. In *Proceedings of Pacific Graphics 2002*, pages 96–103, 2002.
15. C. Theobalt, M. Magnor, P. Schueler, and H.-P. Seidel. Multi-layer skeleton fitting for online human motion capture. In *Proceedings of 7th International Fall Workshop on Vision, Modeling and Visualization*, pages 471–478, 2002.
16. M. Gleicher. Animation from observation: Motion capture and motion editing. *Computer Graphics*, 4(33):51–55, November 1999.
17. 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*, pages 77–85. IEEE CS Press, 2000.
18. J. K. Aggarwal and Q. Cai. Human motion analysis: A review. *CVIU*, 73(3):428–440, March 1999.
19. D. Hogg. Model-based vision : a program to see a walking person. *Image and Vision Computing*, 1(1):5–20, 1983.
20. 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.
21. K. Rohr. Incremental recognition of pedestrians from image sequences. In *Computer Society Conference on Computer Vision and Pattern Recognition 93*, pages 8–13, 1993.
22. Y. Kameda, M. Minoh, and K. Ikeda. Three dimensional motion estimation of a human body using a difference image sequence. In *Proceedings of the Asian Conference On Computer Vision '95*, pages 181–185, 1995.
23. Y. Guo, G. Xu, and S. Tsuji. Tracking human body motion based on a stick-figure model. *Journal of Visual Communication and Image Representation*, 5(1):1–9, 1994.
24. J.Y Zheng and S. Suezaki. A model based approach in extracting and generating human motion. In *Proceedings of the International Conference on Pattern Recognition*, pages 1201–1205, 1998.
25. R. Plankers and P. Fua. Tracking and modeling people in video sequences. *Computer Vision and Image Understanding*, 81(3):285–302, March 2001.
26. 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.
27. 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.
28. 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.
29. A. Laurentini. The visual hull concept for silhouette-based image understanding. *Pattern Analysis and Machine Intelligence*, 16(2):150–162, February 1994.
30. R. Szeliski. Rapid octree construction from image sequences. *Computer Vision, Graphics, and Image Processing. Image Understanding*, 58(1):23–32, 1993.
31. 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.
32. 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.
33. 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.
34. 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.
35. B. Lok. Online model reconstruction for interactive virtual environments. *Symposium on Interactive 3D Graphics, pp. 69-72, 2001*, 2001.
36. K. Kutulakos and S. Seitz. A theory of shape by space carving. In *Proceedings of the 7th IEEE International Conference on Computer Vision (ICCV-99)*, volume I, pages 307–314, Los Alamitos, CA, September 20–27 1999. IEEE.
37. 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 CVPR*, volume 2, pages 714 – 720, June 2000.
38. J. Luck, D. Small, and C.Q. Little. Hierarchical 3d pose estimation for articulated human body models from a sequence of volume data. In *Proceedings of the International Workshop on Robot Vision (RoboVis)*, pages 27–34, 2001.
39. I. Mikić, M. Triverdi, E. Hunter, and P. Cosman. Articulated body posture estimation from multicamera voxel data. In *Proc. of CVPR*, pages 455–462, 2001.
40. A. Bottino and A. Laurentini. A silhouette based technique for the reconstruction of human movement. *CVIU*, 83:79–95, 2001.
41. S. Weik and C.-E. Liedtke. Hierarchical 3d pose estimation for articulated human body models from a sequence of volume data. In *Robot Vision*, pages 27–34, 2001.
42. 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.
43. P.F. Rowat. *Representing the Spatial Experience and Solving Spatial Problems in a Simulated Robot Environment*. PhD thesis, University of British Columbia, 1979.
44. S. Thrun. Learning maps for indoor mobile robots. *Artificial Intelligence*, 99(1):21–71, 1998.
45. J.C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, 1991.
46. 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.
47. K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press, Boston, 1990.



Christian Theobalt received his M.Sc. degree in Artificial Intelligence from the University of Edinburgh, Scotland, and his Diplom (M.S.) degree in Computer Science from the Saarland University, Saarbrücken, Germany, in 2000 and 2001, respectively. Currently, he is a PhD student in the Computer Graphics Group at the Max-Planck-Institute for Computer Science, Saarbrücken, Germany. His research interests include motion analysis from video, 3D computer vision and image- and video-based rendering and reconstruction.



Marcus Magnor received his M.S. in Physics from the University of New Mexico, USA, in 1997 and his Ph.D. in Electrical Engineering from the University of Erlangen, Germany, in 2000. He was a Research Associate at Stanford University's Computer Graphics Lab, USA, before he joined the Max-Planck-Institute for Computer Science in Saarbrücken, Germany, where he is currently heading the Independent Research Group Graphics-Optics-Vision. His research interests include image- and video-based modeling and rendering, 3D imaging, and reverse rendering.

Pascal Schüler received his Diplom (M.S.) degree in Computer Science from the Saarland University, Saarbrücken, Germany in 2001. From 2001 to 2002 he worked as a research assistant in the Computer Graphics Group of the Max-Planck-Institute for Computer Science, Saarbrücken, Germany. He is now employed as a software engineer with Massen Machine Vision Systems, Konstanz, Germany.

Hans-Peter Seidel studied mathematics, physics and computer science at the University of Tübingen, Germany. He received his Ph.D. in mathematics in 1987, and his habilitation for computer science in 1989, both from University of Tübingen. From 1989, he was an assistant professor at the University of Waterloo, Canada. In 1992, he was appointed to the chair of Computer Graphics of the University of Erlangen-Nürnberg, Germany. Since 1999 he has been director of the Computer Graphics Group at the Max-Planck-Institute for Computer Science and honorary professor at Saarland University in Saarbrücken, Germany. In his research, Hans-Peter Seidel investigates algorithms for 3D Image Analysis and Synthesis. This involves the complete processing chain from data acquisition over geometric modeling to image synthesis. He was awarded the Leibniz prize 2003.