Marker-free Kinematic Skeleton Estimation from Sequences of Volume Data

Christian Theobalt, Edilson de Aguiar, Marcus A. Magnor,
Holger Theisel and Hans-Peter Seidel
MPI Informatik
Stuhlsatzenhausweg 85
66123 Saarbrücken, Germany
{theobalt,edeaguia,magnor,theisel,hpseidel}@mpi-sb.mpg.de

ABSTRACT

For realistic animation of an artificial character a body model that represents the character's kinematic structure is required. Hierarchical skeleton models are widely used which represent bodies as chains of bones with interconnecting joints. In video motion capture, animation parameters are derived from the performance of a subject in the real world. For this acquisition procedure too, a kinematic body model is required. Typically, the generation of such a model for tracking and animation is, at best, a semi-automatic process. We present a novel approach that estimates a hierarchical skeleton model of an arbitrary moving subject from sequences of voxel data that were reconstructed from multi-view video footage. Our method does not require a-priori information about the body structure. We demonstrate its performance using synthetic and real data.

Categories and Subject Descriptors

I.3.7 [**Computer Graphics**]: Three-Dimensional Graphics and Realism—*Virtual Reality*; I.4.8 [**Image Processing and Computer Vision**]: Scene Analysis—*Motion, Tracking, Time-varying Imagery*; I.5.1 [**Pattern Recognition**]: Models—*Structural*; I.2.10 [**Artificial Intelligence**]: Vision and Scene Understanding—*Motion, Video Analysis*

General Terms

Algorithms, Measurement

Keywords

Model Reconstruction, Kinematic Skeleton, Tracking, Volume Processing, Learning, Motion Capture

VRST'04, November 10-12, 2004, Hong Kong.

1. INTRODUCTION

The generation of realistic artificial characters has always been a challenging problem in computer animation and in the development of 3D virtual worlds. Not only the design of a realistic physical appearance but also the generation of life-like motion is a prerequisite for a convincing visual impression. Many techniques have been developed that assist the animator in the latter task spanning from key-framing, over physics-based animation to motion capture. All these techniques have in common, that they rely on a kinematic skeleton model of the body which represents the character as a chain of bones and interconnecting joints. In motion capture, animation parameters are derived from the performance of a moving subject in the real-world. Many systems have been developed for capturing humans, but only the marker-based optical ones are general enough to be applied to a broader category of subjects, e.g. animals. The captured parameters define the pose in terms of the configurations of the joints in the skeleton. The employed model has to be designed manually before the capturing session starts or it can be learned in a semi-automatic procedure [26]. Automatic construction of models for arbitrarily shaped bodies has been difficult so far.

We have developed a novel approach that enables the automatic construction of a kinematic skeleton model of an arbitrary moving subject. Our method does with practically no a-priori information about the body structure. The inputs to our system are sequences of voxel volumes of a moving subject that can be reconstructed from multi-view video streams by means of a non-intrusive shape-from-silhouette approach. The system is flexible enough to derive the body structure of any type of subject that can be modeled as a linked kinematic chain, such as humans, most animals and several mechanical structure. We expect this approach to be a helpful tool for people working in computer animation and motion analysis. Although our method is mainly a tool for reconstructing skeletons, we can also perform basic marker-less optical motion tracking. We demonstrate our system using volume sequences acquired in the real world, as well as synthetic voxel data created with a 3D animation package.

2. RELATED WORK

Commercial human motion capture systems can be classified as mechanical, electromagnetic, or optical systems [21].

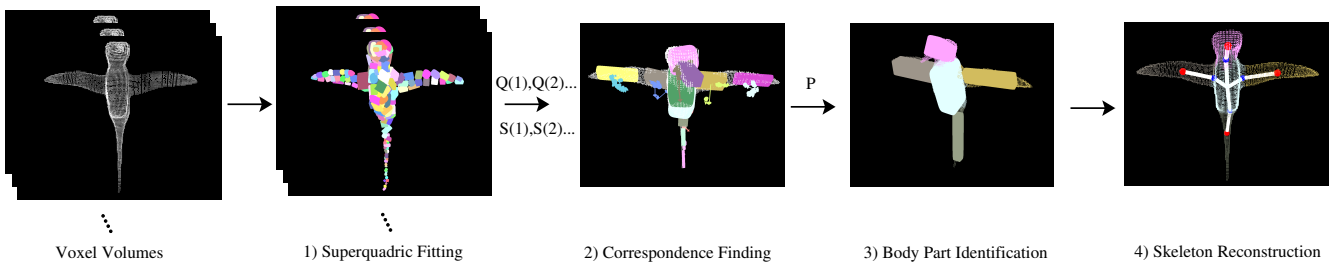


Figure 1: Visualization of our algorithm's workflow.

Video-based systems used in the industry typically require the person to wear optical markers on the body to whose 3D locations a kinematic skeleton is fitted [26]. Since in many application scenarios no visual intrusion into the scene is desired, researchers in computer vision have investigated marker-free optical methods [12]. Some of these methods work in 2D and represent the body by a probabilistic region model [30] or a stick figure [18]. More advanced algorithms employ a kinematic skeleton assembled of simple shape primitives, such as cylinders [25], ellipsoids [7], or superquadrics [13]. Inverse kinematics approaches linearly approximate the non-linear mapping from image to parameter space [3, 31] to compute model parameters directly. Analysis-through-synthesis methods search for optimal body parameters that minimize the misalignment between image and projected model. To assess the goodness-of-fit, features, such as image discontinuities, are typically extracted from the video frames [13]. A force field exerted by multiple image silhouettes aligns a 3D body model in Ref. [10]. In Ref. [23] a combination of stereo and silhouette fitting is used to estimate human motion. A hardware-accelerated silhouette-based motion estimation is described in Ref. [5], and in Ref. [11] a particle filter is applied to estimate body pose parameters from silhouette views.

Recently, sequences of shape-from-silhouette (visual hull) models have been considered as input data for human motion estimation. Ellipsoidal body models [7], kinematic skeletons [20], or skeleton models with attached volume samples [29] are fitted to the volume data. Other visual hull-based approaches fit a pre-defined kinematic model with triangular mesh surface representation [2] to the volumes, or employ a Kalman Filter and primitive shapes for tracking [22].

All previously mentioned marker-free techniques rely on some form of pre-designed body model or require a significant amount of a-priori knowledge to generate the model from the data in a semi-automatic procedure. In contrast, we present an approach that estimates the moving subject's kinematic structure from the motion of individual rigid body parts that were automatically identified. We achieve this by combining a volume decomposition technique based on superquadric shells with a motion tracking of these primitive shapes. The derived model may then serve as a representation for motion tracking.

The idea of characterizing 3D point clouds by means of fitting primitive shapes is a common approach in 3D shape analysis (see [19] for a survey) where it is typically applied to static data. In Ref. [8], multiple superquadric shapes are used to decompose 3D point data into primitive sub-shapes. The same category of geometric primitives is used in computer vision for object recognition, range map segmentation [17] and analysis of medical data sets [1]. A method for clustering triangle meshes which can also extract shape skeletons is described in [15].

Most similar to our approach is the work by Cheung et al. [6], where a person's skeleton and motion are estimated from visual hulls, and the work by Kakadiaris et al. [14] where body models are estimated from multiple silhouette images. Our method differs from these approaches in that it does not require a dedicated initialization phase where prescribed motion sequences are to be performed with each limb separately. Thus, our method requires far less a-priori information about the tracked subject.

In contrast to our previous work [9], we now employ superquadrics, a class of shape primitives that can approximate many volumes more accurately. Thus, we designed a novel split and merge approach, a novel method for rigid body classification and a new criterion for joint localization.

3. OVERVIEW

Fig. 1 illustrates the main algorithmic workflow of our method. The system expects a voxel volume $V(t)$ for each time step t of a motion sequence as input (Sect. 4). In step 1, the Superquadric Fitting step, each $V(t)$ is packed with superquadric shells using a split and merge approach (Sect. 5). The result is a set of fitted shape primitives $Q(t)$ and a list of associated voxel subsets $S(t)$ for each time instant. The correspondences between superquadrics at different time instants are established by means of a dynamic programming method in step 2, the Correspondence Finding step (Sect. 6). The result of step 2 is a path for each primitive shape that describes its motion over time. Together, all superquadric paths form the path set P. Knowing their motion, the primitives are clustered into separate rigid bodies in step 3, the Body Part Identification step (Sect. 7). After step 3, the motion of each rigid body over time is known, and joint locations between neighboring bodies can be estimated in step 4, the Skeleton Reconstruction step (Sect. 8).

4. INPUT DATA

It is our intent to demonstrate that the presented method is capable of reliably estimating kinematic body models from

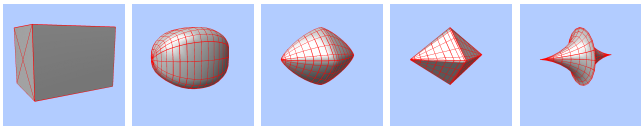


Figure 3: Different superquadric shapes obtained with different ϵ_1 and ϵ_2.

multi-view video data acquired in the real world. Unfortunately, it turns out to be difficult to find decent test subjects for the acquisition of multi-view video sequences of anything else but humans. Thus, in order to complement the human motion data that we recorded in our multi-camera studio, we created several synthetic data sets to demonstrate the flexibility of our approach. The synthetic sequences were generated with 3D Studio Max^{TM} by placing animation skeletons into the surface meshes of a bird, a snowman and a monster. Animations with these models were created via key-framing. For each time frame of animation, a separate surface voxel set was exported.

The video footage acquired in the real word was recorded in our multi-view video studio [28]. Eight synchronized cameras are placed in a convergent setup around the center of the scene. Each camera records at a resolution of 320x240 pixels and at a frame rate of 15 fps which is the technical limit for external synchronization. The cameras are metrically calibrated into a common coordinate system. From image silhouettes we reconstruct the voxel-based volume of the object in the foreground by means of a space-carving approach [16]. In addition to simple shape-from-silhouette reconstruction, this method employs a color-consistency criterion over multiple camera views to enhance the reconstruction quality. In our experiments, we carve surface voxel sets out of volume blocks of 256^3 volume elements.

5. SUPERQUADRIC FITTING

5.1 Fitting a Superquadric to Voxel Data

A superquadric is a closed curve defined as the solution of the implicit equation

$$((\frac{x}{a_1})^{\frac{2}{\epsilon_2}} + (\frac{y}{a_2})^{\frac{2}{\epsilon_2}})^{\frac{\epsilon_2}{\epsilon_1}} + (\frac{z}{a_3})^{\frac{2}{\epsilon_1}} = 1 \qquad (1)$$

In Eq. 1 a_1, a_2 and a_3 are the radii along the three main axes, and ϵ_1 and ϵ_2 are roundness parameters. All points that fulfill this equation lie by definition on the surface of the superquadric. If one considers the left-hand-side of Eq. 1 being a function $F(x, y, z)$, a simple test for deciding if a point (x, y, z) lies inside $(F < 1)$, on the surface of $(F = 1)$, or outside $(F > 1)$ the primitive shape is feasible. Depending on the roundness parameters, the shape of a superquadric shell mediates between circular and rectangular, enabling a variety of intermediate shapes (see Fig. 3). A superquadric in a general position is described by three additional rotation parameters (R_x, R_y, R_z) and three translation parameters (T_x, T_y, T_z) with respect to the world origin. Thus, in order to fit a superquadric $\mathcal{Q}$ to a set of N 3D points (in our case surface voxels) such that its surface comes as close as possible to all points, 11 shape parameters $\mathcal{Q} = [a_1, a_2, a_3, \epsilon_1, \epsilon_2, R_x, R_y, R_z, T_x, T_y, T_z]$ need to be determined. The optimal parameters of a superquadric approximating a 3D voxel set are found by numerically minimizing an error function that measures the distance between the shape's surface and the volume elements.

The choice of a good error function is essential for the quality of the final fit. We have run experiments with several different distant measures and found the following one to produce the best results:

$$D = \frac{a_1 a_2 a_3}{N} \sum_{i=1}^{N} (F(x_i, y_i, z_i)^{\epsilon_1} - 1)^2 \qquad (2)$$

In Eq. 2 N is the number of voxels and $d = F(x_i, y_i, z_i)^{\epsilon_1} - 1$ is an approximation to the distance of a volume element to the superquadric surface as proposed in Ref. [17]. The factor $\frac{a_1 a_2 a_3}{N}$ is included in order to prevent a shape primitive from growing too much in one direction or uniformly in all directions. We have evaluated several non-linear optimization schemes on test voxel sets to identify the most appropriate minimizer. We achieved best results with the LBFGS-B method [4], which is a quasi-Newton algorithm that permits the specification of bound constraints on the parameters. Results with other numerical optimization schemes such as Amoeba (a downhill-simplex variant), Powell's method (a direction set method), and the often used Levenberg-Marquardt optimizer were significantly worse (see Ref. [24] for information on these methods). This is mainly due to the fact that these methods don't allow for constrains on the parameter space, and thus it happens frequently that the roundness parameters become negative which corresponds to inappropriate superquadric shapes.

A good initial set of parameters to start the minimization with is found by fitting a regular ellipsoid to the voxel data (a regular ellipsoid can be expressed as a superquadric by setting $\epsilon_1 = \epsilon_2 = 1$). The ellipsoid's position T_i_x, T_i_y, T_i_z coincides with the voxel set's center of gravity, the directions of its three main axes are identical to the directions of the eigenvectors of the voxel set's covariance matrix [7]. The initial radii a_i_1, a_i_2, a_i_3 along the main axes are found as $a\_i_j = 2 \cdot \sqrt{\lambda_j}$, λ_j being the eigenvalue corresponding to eigenvector j [1]. The initial rotation R_i_x, R_i_y, R_i_z is also derived from the directions of the eigenvectors.

5.2 Split and Merge

Using the method described in Sect. 5.1 for each time step, we fill the voxel volumes with superquadric shells. We achieve this by applying a hierarchical *split and merge* approach [8]. The procedure starts with a split stage, approximating the whole voxel volume first by one superquadric which is subdivided into two superquadrics if this reduces the overall fitting error (Fig. 2). The split stage recursively processes each newly created superquadric in the same way, thereby producing a hierarchical decomposition of the voxel set. The split stage is performed for each voxel volume $V(t)$ individually.

The merge stage follows the split stage and improves the fitting result by merging pairs of neighboring superquadrics into one. It is performed only for the voxel volume $V(1)$ of the first time step.

In the following the individual steps of the split stage and the merge stage are detailed.

5.2.1 Split Stage

For each $V(t)$:

1 The whole set of 3D voxels $V(t)$ is approximated by one superquadric $\mathcal{Q}$.

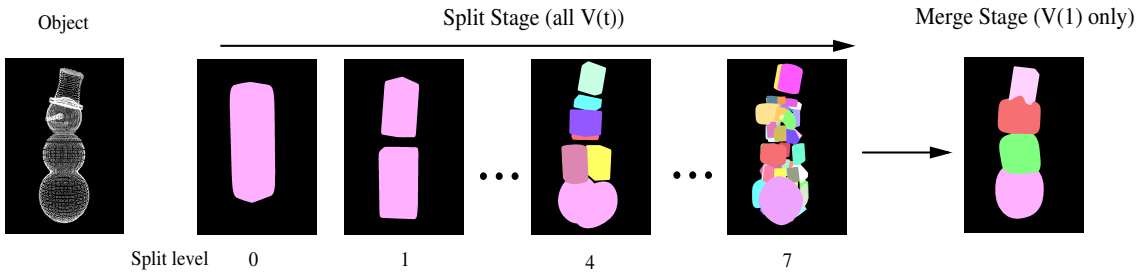


Figure 2: Illustration of the split and merge procedure using the snowman model as an example.

2 The set of 3D voxels is split into two subsets S_1 and S_2 along the plane $\mathcal{P}$ orthogonal to the major inertia axis of the voxel set (Note that $\mathcal{P}$ contains the centroid of the set).

3 S_1 and S_2 are approximated individually by one superquadric each. For each subset, the procedure is repeated from step 2.

We obtain a set of shapes $Q_{split}(t)$ and a set of corresponding voxel subsets $S_{split}(t)$ that approximate the voxel model $V(t)$. After a sufficient number of subdivisions (in our case typically 7), there is a high likelihood that all points in one voxel subset belong to the same rigid body of the moving subject's kinematic skeleton. Nonetheless, it is still possible that more than one superquadric is fitted to one rigid body (e.g. four superquadrics to the upper arm).

5.2.2 *Merge Stage*

For $V(1)$ only:

1 For each subset of voxels $S_i \in S_{split}$, we determine the list $K_i = \{S_{n1}, .., S_{nk}\}$ of neighboring voxel subsets $(S_{n1}, .., S_{nk} \in S_{split})$.

2 For each possible pairing of the voxel set S_i and one neighboring voxel set $S_j \in K_i$, a merged voxel set M_j is created. A novel superquadric is fitted to each M_j and the fitting error D_j is computed (Eq. 2). From all paired superquadrics whose D_j is smaller than the sum of fitting errors of the superquadrics it was created from, the one with the lowest D_j is chosen to replace the two primitives it emerged from.

3 A new set of superquadrics is obtained. The procedure is repeated from step 1. It terminates when no further reduction of the fitting error is possible.

We perform the merging step only on the first voxel volume $V(1)$. If we were considering voxel volumes from different time steps independently and merging superquadrics only due to structural criteria, it would not be possible to prevent erroneous merges across rigid body boundaries. The resulting set of shapes is the starting point for the correspondence finding step (Sect 6) which exploits the temporal dimension to prevent merging across boundaries of separate bodies.

The result of the split and merge process is a set of superquadrics $Q(t)$ and a set of voxel subsets $S(t)$ for each $V(t)$.

6. CORRESPONDENCE FINDING

After subdividing each voxel volume using primitive shapes, a set of correspondences $C(t, t+1)$ between each pair of superquadric sets $Q(t)$ and $Q(t+1)$ at subsequent time steps is computed. The set of correspondences describes for each shape primitive in $Q(t)$ to which member of $Q(t+1)$ it is related. For every superquadric, the correspondences indicate from which 3D location at t to which position at $t+1$ it moves.

Assuming that we can keep the number of superquadrics constant for all time instants, the correspondences enable the reconstruction of a complete motion path for each individual shape primitive over the duration of the whole input sequence. The correspondence finding procedure looks at each pair of superquadric sets $Q(t)$ and $Q(t+1)$ at subsequent time instants separately.

Since the number of shape primitives in the sets $Q(t)$ and $Q(t+1)$ may differ, we employ a two-stage procedure to establish the correspondences and to reorganize the superquadrics such that their number at each time instant is constant. This way we establish a bijective correspondence mapping between superquadrics at subsequent time steps.

Technically, the correspondences from t to $t+1$ are established by searching for correspondences from $t+1$ to t which are, in the end, inverted. In the first stage, a correspondence for each individual shape primitive is established to a superquadric at the preceding time instant by means of a dynamic programming approach [27]. The error function used in this optimization procedure is the Euclidean distance between the superquadric centers.

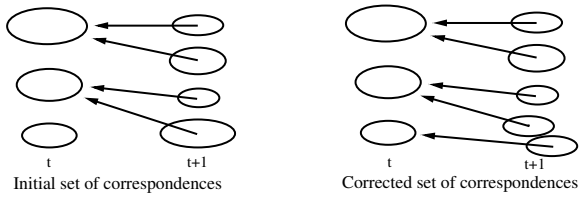
Dynamic programming establishes a first set of correspondences. After the first stage, two cases of degenerate correspondences may occur that need to be corrected in a second stage in order to establish a bijective mapping.

The first case, the *unmatched superquadric* (Fig. 4 A), occurs if there exists a superquadric $\mathcal{Q}_1$ at time t to which no superquadric from time $t+1$ established a correspondence. To solve this problem, the superquadric $\mathcal{Q}_2 \in Q(t+1)$ closest to $\mathcal{Q}_1$ according to the Euclidean distance is selected. The voxel subset associated to $\mathcal{Q}_2$ is split in two and two new superquadrics $\mathcal{Q}_3$ and $\mathcal{Q}_4$ are fitted to the newly created voxel subsets. $\mathcal{Q}_3$ inherits the original correspondence to time t from $\mathcal{Q}_2$, $\mathcal{Q}_4$ establishes a new correspondence to $\mathcal{Q}_1$.

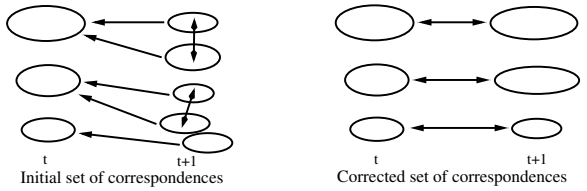
The second case, the *multi-match* (Fig. 4 B), arises if more than one superquadric from $Q(t+1)$ found the same partner in $Q(t)$. We solve this problem by merging all the superquadrics at $t+1$ corresponding to the same superquadric at t. This is achieved by merging all the associated voxel subsets and fitting a new shape primitive.

The two degenerate cases are corrected subsequently. After stage two of the correspondence finding, the correspondence directions are inverted. By this means, for each primitive in $Q(t)$ exactly one partner from $Q(t+1)$ is found.

After all time steps have been processed in this way, each superquadric set contains the same number of shapes as the set $Q(1)$. Note that in order to establish correct correspondences $C(t, t+1)$ the superquadric sets are modified as well. For each shape primitive in $Q(1)$ a complete motion path over the whole sequence can be identified by linking subsequent correspondences. The so-created set of paths P contains for each $\mathcal{Q}_i \in Q(1)$ a path P_i, P_i being an ordered set of 3D coordinates $P_i = \{(x_i(t), y_i(t), z_i(t)) \mid t \text{ valid time step}\}$ of the superquadric center at time t. Fig. 5 (1) shows example paths of individual superquadrics that we found with our approach.



(A) Unmatched Superquadric



(B) Multi−match

Figure 4: Handling degenerate cases during correspondence finding.

7. BODY PART IDENTIFICATION

The paths of P provide all necessary information we need to identify separate rigid bodies in the kinematic skeleton of the moving subject. In case we are analyzing volume data of a human, this means that the paths enable us to identify, for example, the upper arm segment or the lower leg segment. Implicitly, we make the simplifying assumption that individual kinematic elements can be represented as rigid structures that do not undergo strong deformations.

In order to identify individual rigid bodies, we make use of the fact that the mutual Euclidean distance between any two points on the same body does not change while the skeleton is moving. Thus, if the mutual distance between the motion paths of two superquadrics over time is subject to significant variations, it is most likely that the two primitives do not lie inside the same rigid body.

This criterion gives us a procedure at hand which enables clustering individual superquadrics into separate kinematic elements of the whole body. We employ a voting-based test that analyzes the curve of Euclidean distances between superquadric paths over time. The value of the distance curve $d_{i,j}(t)$ between the paths of two superquadrics $\mathcal{Q}_i \in Q(1)$ and $\mathcal{Q}_j \in Q(1)$ at time t is defined as the Euclidean distance between their respective positions on the paths at t. In order to decide if $\mathcal{Q}_i$ and $\mathcal{Q}_j$ lie on the same rigid body we look for the presence of two features in the distance curves.

The first feature is a significant change in the first derivative of $d_{i,j}$ at some time step t. For each t at which $d'_{i,j}(t) > T_{deriv}$, T_{deriv} being a derivative threshold, a voting counter $vc(i,j)_{deriv}$ is increased by one.

The second feature arises at every time step for which the value of the distance curve differs by more than a threshold from the initial distance value $d_{i,j}(1)$. Thus, for each t with $\|d_{i,j}(t) - d_{i,j}(1)\| > T_{diff}$, T_{diff} being a difference threshold, a second voting counter $vc(i,j)_{diff}$ is increased by one.

The final vote $vc(i,j)$ is the sum of the two previously mentioned voting counters $vc(i,j) = vc(i,j)_{deriv} + vc(i,j)_{diff}$. If this final vote is larger than a threshold T_{vote}, the distance curve fails the test and the superquadrics are considered to be on different rigid bodies.

To eliminate spurious peaks in a distance curve due to noise, a median filter is applied to it before applying the distance criterion. By means of our voting-based scheme and appropriate thresholds (found through experiments) it is possible to perform robust path comparison even in the presence of measurement noise.

We apply the voting-based test to classify individual rigid bodies as follows:

1 A seed superquadric $\mathcal{Q}_{seed} \in E(1)$ is selected and a distance curve $d_{seed,k}$ with each superquadric $\mathcal{Q}_k \in E(1) \setminus \{\mathcal{Q}_{seed}\}$ is computed.

2 For each $\mathcal{Q}_k$ the voting-based test is applied to $d_{seed,k}$, and $\mathcal{Q}_k$ is classified as lying on the same rigid body if the test is passed.

3 The procedure iterates by restarting from step 1 and selecting a new seed from all superquadrics that have not yet been assigned to a rigid body.

The seed $\mathcal{Q}_{seed}$ in the first iteration is the superquadric nearest to the center of gravity (COG) of the voxel set $V(1)$. In the subsequent iterations, the selected seed is the superquadric nearest to the COG of the body part that was found in the preceding iteration. This seed selection criterion is a heuristics which enables the construction of a hierarchy of rigid bodies in the moving character. The rigid body detected first is considered to be the root of the skeleton hierarchy. Each subsequently detected rigid body is considered to be on the next lower hierarchy level, and to be connected to the root. The whole classification procedure is recursively applied to each individual rigid body on the next lower hierarchy level, thereby further refining the set of detected body parts.

In case of a human subject this strategy leads to the identification of one rigid body for the torso and one for each arm, each leg and the head in the first iteration. Now the procedure is repeated for each limb which produces the final correct subdivision into body parts.

For each $V(t)$ it is now known which voxel subsets form a rigid body and how the rigid bodies move over time. Fig. 5 shows individual body parts as they were found in some of our test data sets.

8. SKELETON RECONSTRUCTION

In the final step we use the detected rigid bodies and their motion to estimate the 3D locations of joints in the skeleton hierarchy. Joint finding can be performed for each time

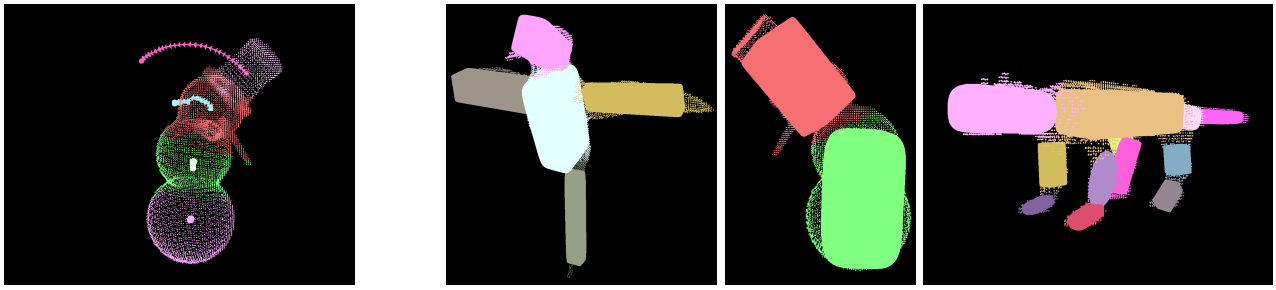


Figure 5: Motion paths of individual superquadrics as they were found by our method (1). Individual body parts detected in different test subjects (2),(3),(4).

step individually, but we usually regard the skeleton reconstructed for the first time instant as the reference model. The rigid body hierarchy, and thus the information which rigid bodies are connected, has already been determined in the Body Part Identification step (Sect. 7). For each pair of connected adjacent rigid bodies B_a and B_b the joint location is estimated relative to the boundary voxels between the voxel subsets associated with B_a and B_b. The joint location at time t is estimated as the center of gravity of the set of voxels which contains all those voxels from both voxel subsets that have at least one adjacent voxel from the other voxel subset, respectively. This is a simple but efficient heuristic approach which produces good results for our test data.

The primary goal of our system is to reconstruct a kinematic skeleton model. Nonetheless, since we are able to build such a model for each time step of a motion sequence, approximate motion tracking of the moving subject is also feasible. Although applying our joint localization scheme to each time step of video is not tracking in a strict sense since we do not apply the same body model in each frame, it is still possible to obtain a first rough estimate of the motion parameters. In the future, we plan to further evolve our system into a complete motion tracking approach that employs the same body model at each time step of video.

9. RESULTS AND DISCUSSION

We evaluated the performance of our system using synthetic and real data sets. The synthetic data sets we used were the moving snowman (on avg. 8000 voxels per time step), the bird (on avg. 11000 voxels per time step), and the monster (on avg. 14000 voxels per time step). Motion sequences with these models were created using 3D Studio Max^{TM}. The snowman was animated using one point of articulation at the neck, the derived skeleton and the correctly detected two body parts are shown in Fig. 6. In order to create the bird data set we animated 4 joints in a kinematic skeleton, one at the neck, one at the tail and two at the roots of the wings. The skeleton which was found by our method nicely coincides with the actual kinematic model we used for animation (Fig. 6). Our most complex data set is the monster, a lizard-like four-legged creature. In total, we used 15 joints for animating its motion, 2 in the tail, 3 in each leg and 1 at the neck. The skeleton of the creature that we estimated is shown in Fig. 6. In the monster data set, it was hard to identify the feet as separate rigid bodies since their motion is only very marginal compared to the rest of the body. In general, it is difficult to find decent segmentation

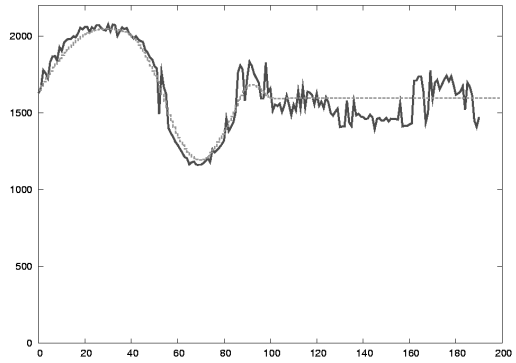


Figure 7: Plot of reconstructed (dashed) against ground truth y-coordinate of one joint in the snowman skeleton for each time step of the input sequence.

thresholds if the relative motion between two body parts is hardly noticeable.

Since for the synthetic sequences we know the ground truth joint positions, we can provide an estimate of the accuracy of our approach. For visual illustration we plot in Fig. 7 the reconstructed y-coordinate against the true y-coordinate of one joint in the snowman skeleton for each time step of the input sequence. With the exception of some outliers, the difference in y-coordinates is small (mostly below 2% with respect to the length of the body, 5% in the worst case).

We also ran experiments with video footage of a moving person that was recorded in our multi-view video studio. From the multi-view silhouette frames shape-from-silhouette voxel models were reconstructed. Although the space carving approach eliminates most of the typical artifacts in shape-from-silhouette volumes that are due to insufficient visibility, some noise still appears in the form of bulky arms and legs. In our tests we analyzed a sequence of 40 frames, roughly 22500 voxels each, in which the person is only moving the arms and the head. This way it is possible to nicely demonstrate the working principle of our method. Since no motion is performed with the lower extremities, the torso and the legs are classified as belonging to the same rigid body (Fig. 6). Even though the sequence is very short, the kinematic structure of the arms, the leg and the head are correctly found. Little inaccuracies in the detected locations of the elbow joints can be observed. This is mainly due to the fact that the sequence is very short and that the person wears comparably wide cloths.

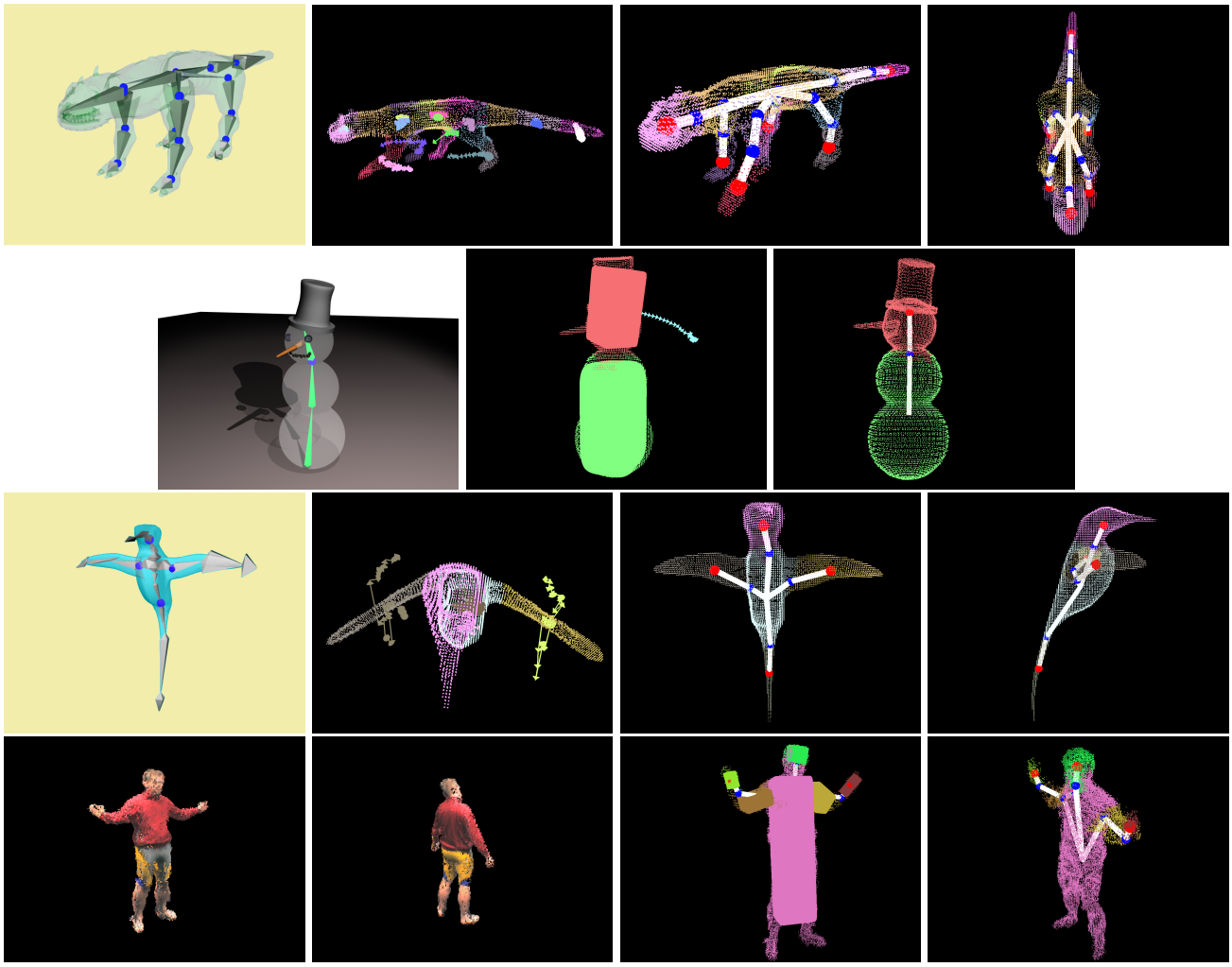


Figure 6: Top row: Monster with 3D Studio skeleton, animated joints are shown as spheres (1); motion of individual body parts (2); estimated skeleton within voxel set, joints are shown as (blue) spheres between bones, bones are shown in white (3),(4). Second row: Snowman with 3D Studio skeleton (1); estimated body parts (2); reconstructed skeleton (3). Third row: Bird with 3D Studio skeleton (1); estimated body parts and their motion (2); reconstructed skeleton (3),(4). Bottom row: Voxel sets from input sequence (1),(2); identified body parts and skeleton if only upper extremities move (3); estimated skeleton only (4).

Due to the hierarchical optimization the most time consuming components of our approach are the split and merge steps. Processing our test sequences on an Intel Xeon^{TM} 3.0 GHz we measured run-times of the splitting in the range of 120-160 s per time step of the input sequence, and of the merging in the range of 250-1000 s per time step. All the other processing steps in our algorithm run significantly faster. Correspondence finding takes 7-19 s for one time instant, skeleton reconstruction 0.3-0.5 s, and body part identification 3-6 s per time instant.

An important advantage of our method over related approaches is that it estimates the body structure of an arbitrary moving subject with a minimum of a-priori information. No special initialization motion is required to reconstruct the body model, any motion sequence is equally appropriate. Our experiments with real video data show that the method's performance does not significantly deteriorate if measurement noise is present in the volume data.

In its current state, the system is subject to a couple of limitations. Even though we don't prescribe an initialization motion, two different adjacent rigid body segments can only be discriminated if at least once in a sequence a relative motion between them can be observed. We consider this a principal problem of a non-informed motion analysis approach and not a limitation that is specific to our method. Furthermore, we expect that the system's performance will deteriorate if voxels of individual rigid bodies merge frequently with the rest of the volume (e.g. if the arms are often kept tight to the torso).

Although our method does not operate on the same accuracy level as marker-based approaches for skeleton derivation and motion tracking, it is nonetheless a useful tool in situations where visual interference with the captured scene is inappropriate and no information about the structure of a moving subject is available. In future, we will extend our approach to a complete skeleton learning and tracking method,

that uses the body structure learned in the first iteration to follow the motion without the use of optical markers.

10. CONCLUSIONS AND FUTURE WORK

We presented a novel approach for estimating a kinematic model of an arbitrarily structured moving body from sequences of voxel volumes reconstructed from video footage. We demonstrated that our algorithm is equally well-suited for the reconstruction of kinematic skeletons of animals and humans. In addition to estimating body models the approach can also perform a simple motion tracking.

In general, we believe that the method is an algorithmic component that can be used in combination with many non-intrusive motion estimation algorithms described in the literature. This combination creates a very powerful marker-free tracking system applicable to a large class of moving subjects. To demonstrate this in the future, we intend to further evolve the approach into a complete motion capture system by combining it with a volume-based motion tracking scheme.

11. REFERENCES

[1] F. Banégas, M. Jaeger, D. Michelucci, and M. Roelens. The ellipsoidal skeleton in medical applications. In *Proc. of the sixth ACM symposium on Solid modeling and applications*, pages 30–38. ACM Press, 2001.

[2] A. Bottino and A. Laurentini. A silhouette-based technique for the reconstruction of human movement. *CVIU*, 83:79–95, 2001.

[3] C. Bregler and J. Malik. Tracking people with twists and exponential maps. In *Proc. of CVPR 98*, pages 8–15, 1998.

[4] R. Byrd, P. Lu, J. Nocedal, and C. Zhu. A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. Comp.*, 16(5):1190–1208, 1995.

[5] J. Carranza, C. Theobalt, M. Magnor, and H.-P. Seidel. Free-viewpoint video of human actors. In *Proc. of SIGGRAPH'03*, pages 569–577, 2003.

[6] G. Cheung, B. S., and T. Kanada. Shape-from-silhouette of articulated objects and its use for human body kinematics estiamtion and motion capture. In *Proc. of CVPR*, 2003.

[7] K. Cheung, T. Kanade, J.-Y. Bouguet, and M. Holler. A real time system for robust 3D voxel reconstruction of human motions. In *Proc. of CVPR*, volume 2, pages 714–720, 2000.

[8] L. Chevalier, F. Jaillet, and B. A. Segmentation and superquadric modeling of 3D objects. In *Proc. of WSCG 2003*, 2003.

[9] E. de Aguiar, C. Theobalt, M. Magnor, H. Theisel, and H.-P. Seidel. M3 : Marker-free model reconstruction and motion tracking from 3d voxel data. In *Proc. of Pacific Graphics'04*. to appear, 2004.

[10] Q. Delamarre and O. Faugeras. 3D articulated models and multi-view tracking with silhouettes. In *Proc. of ICCV 99*, pages 716–721, 1999.

[11] B. Deutscher, A. Blake, and I. Reid. Articulated body motion capture by annealed particle filtering. In *Proc. of CVPR'00*, 2000.

[12] D. Gavrila. The visual analysis of human movement. *CVIU*, 73(1):82–98, January 1999.

[13] D. Gavrila and L. Davis. 3D model-based tracking of humans in action: A multi-view approach. In *Proc. of CVPR 96*, pages 73–80, 1996.

[14] I. Kakadiaris and D. Metaxas. 3D human body model acquisition from multiple views. In *Proc. of ICCV'95*, pages 618–623, 1995.

[15] S. Katz and A. Tal. Hierarchical mesh decomposition using fuzzy clustering and cuts. In *Proc. of SIGGRAPH'03*, pages 954–961, 2003.

[16] K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. *Int. J. Comput. Vision*, 38(3):199–218, 2000.

[17] A. Leonardis, A. Jaklic, and F. Solina. Superquadrics for segmenting and modeling range data. *IEEE PAMI*, 19(11):1289–1295, 1997.

[18] M. Leung and Y. Yang. First sight : A human body outline labeling system. *PAMI*, 17(4):359–379, 1995.

[19] S. Loncaric. A survey of shape analysis techniques. *Pattern Recognition*, 31(8):983–1001, 1998.

[20] J. Luck and D. Small. Real-time markerless motion tracking using linked kinematic chains. In *Proc. of CVPRIP02*, 2002.

[21] A. Menache. *Understanding Motion Capture for Computer Animation and Video Games*. Morgan Kaufmann, 1995.

[22] I. Mikić, M. Triverdi, E. Hunter, and P. Cosman. Articulated body posture estimation from multicamera voxel data. In *Proc. of CVPR*, 2001.

[23] R. Plaenkers and P. Fua. Tracking and modeling people in video sequences. *CVIU*, 81(3):285–302, March 2001.

[24] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes in C++*. Cambridge University Press, 2002.

[25] K. Rohr. Incremental recognition of pedestrians from image sequences. In *Proc. of CVPR*, pages 8–13, 1993.

[26] M.-C. Silaghi, R. Plaenkers, R. Boulic, P. Fua, and D. Thalmann. Local and global skeleton fitting techniques for optical motion capture. In *Modeling and Motion Capture Techniques for Virtual Environments*, number 1537 in LNAI, No1537, pages 26–40. Springer, 1998.

[27] M. Sniedovich. *Dynamic programming*. Marcel Dekker, Inc., 1992.

[28] C. Theobalt, M. Li, M. Magnor, and H.-P. Seidel. A flexible and versatile studio for synchronized multi-view video recording. In *Proc. of Vision, Video and Graphics*, pages 9–16, 2003.

[29] 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 *Proc. of Pacific Graphics 2002*, pages 96–103, 2002.

[30] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfinder: Real-time tracking of the human body. *PAMI*, 19(7):780–785, 1997.

[31] S. Yonemoto, D. Arita, and R. Taniguchi. Real-time human motion analysis and IK-based human figure control. In *Proc. of IEEE Workshop on Human Motion*, pages 149–154, 2000.