Optical reconstruction of detailed animatable human body models

Carsten Stoll, Jürgen Gall, Edilson de Aguiar, Hans-Peter Seidel, Sebastian Thrun, Christian Theobalt

MPI–I–2009–4–006 August 2009

Authors' Addresses

Carsten Stoll
Max-Planck-Institut für Informatik
Saarbrücken, Germany

Jürgen Gall
ETH Zürich
Zürich, Switzerland

Edilson de Aguiar
Disney Research
Pittsburgh, PA, USA

Hans-Peter Seidel
Max-Planck-Institut für Informatik
Saarbrücken, Germany

Sebastian Thrun
Stanford University
Stanford, CA, USA

Christian Theobalt
Stanford University / Max-Planck-Institut für Informatik
Stanford, CA, USA / Saarbrücken, Germany

Acknowledgements

We would like to thank our performer Tatjana Feldmann as well as Derek D. Chan for his help with video production. This work has been developed within the Max-Planck-Center for Visual Computing and Communication (MPC VCC) collaboration.

Abstract

We present a new performance capture approach that reconstructs a rigged fully-animatable virtual double of a real person from multi-view video recordings. Our algorithm is even capable of reconstructing models of people in natural wide everyday apparel, such as a wavy skirt, and only requires a minimum of manual interaction. Without the use of optical markers in the scene, our algorithm first reconstructs skeleton motion and detailed time-varying surface geometry of a real person from a reference video sequence. These captured reference performance data are then analyzed to automatically identify non-rigidly deforming pieces of apparel on the animated geometry. For each piece of apparel, parameters of a physically-based cloth simulation model are estimated, and surface geometry of occluded body regions is approximated. The reconstructed character model comprises a skeleton-based representation for the actual body parts and a physically-based simulation model for the apparel. In contrast to previous performance capture methods, we cannot only play back the captured data but create in real-time new animations with new body motion and cloth deformation that look as realistic as in the reference sequence.

Keywords

Performance Capture, Multi-View Video Analysis, Cloth Simulation

Contents

1 Introduction

Increasingly powerful computing hardware will soon make it feasible to render characters of ever higher visual complexity and realism even in real-time applications like games or networked virtual worlds. Certain visual effects that were previously only seen in feature films, such as detailed cloth animation or body models of high geometry and texture detail, now also slowly find their way into consumer market applications. The recent advent of hardware accelerated physics simulation has expedited this trend, and more advanced simulated characters will soon become part of many computer games.

The ability to render more complex characters also makes the animation design process a more challenging one. Each design element of a character requires greater attention to detail, resulting in an overall more labor-intensive and costly process. More detailed surface geometry will have to be designed or scanned, more detailed textures will have to be painted or photographed, more detailed motion will have to be keyframed or captured, and cloth simulation models will have to be properly tuned to enable real-time rendering. Various acquisition technologies exist to support the animator in parts of the design process, such as motion capture systems to get skeleton motion, 3D scanners to measure static geometry, or specialized measurement devices to analyze material parameters. Unfortunately, using such tools can be very cumbersome and time-consuming, and therefore often complicates animation design rather than to accelerate it. Capture technology would be much more practical if all aspects of an animation that were mentioned above could be obtained from real subjects in a simple, fast, and cost-effective unified process. Being able to start off such rich new captured performance models, animation professionals would not only save valuable time during initial scene design. They would also be given the possibility to easily create complete virtual doubles of real actors and their apparel even under tight budget and time constraints. Additionally, the creative freedom of animators is greatly enhanced since they can either directly work with unmodified captured scenes or freely modify any aspect of a captured performance in

Figure 1.1: Our method allows us to reconstruct fully-animatable character models from video data that comprise a skeleton, surface geometry with skinning weights, and a cloth simulation model for wide apparel. The image shows some poses of newly created animations of two animatable models with different skirts. The sequences were created by simply specifying new skeletal motion parameters.

their favorite animation tool.

The research that has come closest to achieving this goal are so-called performance capture approaches which enable the reconstruction of detailed motion and time-varying geometry of entire human actors or pieces of apparel from multi-view video recordings [5, 29, 8]. Unfortunately, performances captured with these approaches can merely be played back and it is impossible to create entirely new motion sequences with similar lifelike surface deformation. We therefore propose in this paper a new marker-less method to capture performances from multi-view video that are *fully-animatable*. Our algorithm requires only minimal manual interaction and can create fully-rigged virtual doubles even of people wearing wide and wavy clothing, such as a skirt or a dress. Our method begins by capturing the performance of the actor, i.e. spatio-temporally coherent dynamic shape and skeletal motion, in a reference sequence.

This reference performance is automatically divided into mostly rigid segments of the body and non-rigidly deforming attire. For each piece of apparel, the parameters of a physics-based forward simulation model are estimated from the reference sequence. Potentially occluded geometry is approximated to properly handle collisions of the apparel with the body. Our main contributions are:

- A new approach to capture lifelike fully-animatable virtual humans from sparse video recordings of real world actors that comprise detailed surface geometry, a fully-rigged skeleton, and a cloth model for pieces of apparel.

- An algorithm to automatically segment the body geometry into cloth and non-cloth areas, and to approximate collision geometry that is occluded by the attire.

- An algorithm to estimate suitable cloth simulation parameters from the entire apparel of a subject in motion, and not only from small sample patches.

- A method to easily create realistic arbitrary new animations of the reconstructed animatable performance model, without having to do a new recording.

To the best of our knowledge, this is the first method in the literature to capture such rich performance models. Skeletal motion can be freely modified to create new animations where cloth and surface deformations look as lifelike as in the reference data. We demonstrate this on a variety of example sequences and also confirm this observation in a user study.

2 Related Work

In our research, we can build up on previous work in many different disciplines ranging from the huge body of work on dynamic scene reconstruction, over work on physically-based animation to parameter estimation. In the following, we limit our overview to a representative subset of most related works.

A building block of our automatic animatable human character reconstruction method is a performance capture approach, i.e. an algorithm that reconstructs detailed dynamic shape and motion of real world actors performing in front of cameras. Today's performance capture algorithms are based on marker-less motion capture algorithms from computer vision [22]. Most of these original marker-less capturing methods only work for people with skin-tight clothing, merely provide skeletal motion parameters and thus fail to recover detailed dynamic shape models, let alone cloth simulation parameters. One exception is the work by Rosenhahn et al. [24] who perform marker-less skeleton-based tracking of a person wearing a skirt. Their goal was accurate motion capture under occlusion, and not detailed surface capture. In consequence, the skirt moves rather rigidly and does not exhibit natural cloth behavior.

In the fields of image-based rendering and 3D video, several methods for dynamic geometry reconstruction have been proposed [18, 33, 30], with the final goal of rendering novel viewpoints of the scene. Unfortunately, the level of detail in the captured geometry is typically not sufficient for reconstruction of an animatable model. Furthermore, most methods fall short in recovering actual motion parameters, and if they do, such as in the case of model-based 3D video [6], people in arbitrary loose clothing cannot be handled. One step towards reconstructing an animatable representation of a human was taken by Allen et al. [1] who learn a deformation model of the naked shoulder and torso from body scans in different postures. Sand et al. [25] use a similar idea and learn pose-dependent body deformations by using marker-based motion capture and silhouette matching. Similarly, the SCAPE model learns variations in human body shape and surface deformation from laser scans of people [2]. [21] learn a data-driven model of skin and muscle deformation from dense marker-based motion capture data. All these methods are suitable for more or less naked people, but none of them can handle people in wide apparel.

Recently, new performance capture algorithms have been proposed that promise to reconstruct detailed scene geometry and surface appearance from unmodified multi-view video streams. Starck and Hilton [28] use a combination of shape-from-silhouette, multi-view stereo, and spatio-temporal cross-parameterization to capture detailed dynamic scene geometry of moving people. To facilitate reconstruction of coherent geometry of people in arbitrary clothing de Aguiar et al. propose to use deformable meshes created from static full-body laser scans for tracking [9, 8]. The method by Vlasic et al. [29] also reconstructs spatio-temporally coherent geometry of people in everyday apparel by fitting a skeleton to visual hulls and non-rigidly deforming a surface mesh to capture time-varying surface geometry. As part of our method, we use the performance capture approach presented by Gall et al. [12]. They also employ a kinematic skeleton, but use a different tracking strategy that combines local and global optimization with robust image and silhouette features. Since skeleton-based animation is the predominant character animation paradigm in real-time applications, we decided to use a skeleton-based model representation. However, with only few additional steps in our pipeline, we could also use pure surface data captured with the mesh-based approaches cited above.

The biggest disadvantage of all performance capture approaches to date is that the captured result can merely be played back. It is impossible for an animator to create arbitrary new pose sequences in which the actor's geometry would deform like in the reference set, and the cloth would wrinkle in the same plausible way. Mesh sequence editing techniques exist [16, 32] that allow for a certain degree of modification. But they neither allow the same level of control as our fully-rigged character, nor can they create plausible cloth animations in new poses at a level of quality comparable to our approach.

Our method of identifying which parts of the moving geometry are likely to be part of the actual body and which are likely to be cloth is inspired by motion segmentation methods that identify approximately rigidly moving sections in mesh animations [14, 10].

Our final animatable model uses the position-based dynamics approach by Mueller et al. [20] for forward simulation of cloth. A variety of alternative cloth simulation methods exist in the literature (see [7] for an overview of some methods) but the work by Mueller et al. represents a good compromise between speed, simulation accuracy, and parameter complexity. We estimate parameters of the cloth model from captured performance data. Many previous approaches for cloth estimation from real world samples reconstruct deforming cloth geometry only, but not a physically-based forward simulation model. Some methods use multi-view feature and stereo for vision-based deforming geometry capture of square cloth samples [23]. Other algorithms rely on special marker patterns printed on the fabric, as well as on a priori geometry model of a piece of apparel, to measure time-varying cloth geometry from multi-view video [26, 31]. The latter approach also learns a simple data-driven deformation model which can approximate the wrinkling of fabric in new poses. Recently, Bradley et al. [5] proposed a new approach to capture deforming cloth geometry from multi-view video without explicit markers in the scene. So far only few algorithms estimated parameters of a physics-based cloth model. Bhat et al. [4] learn such parameters from waving square fabric samples that were recorded with a real-time structured light system. Hasler et al. [13] use a similar approach but rely solely on texture information from videos. Conceptually related is the approach by Shi et al. [27] who estimate parameters for simulating secondary skin deformation. In contrast, our method does not require an a priori shape model for individual pieces of apparel, does not require any form of visual pattern in the scene, and does not require off-line physical material measurements from fabric samples. Our method fully-automatically identifies all cloth regions on the entire moving human and recovers plausible forward simulation parameters.

3 Overview

It is our goal to reconstruct a fully-animatable human character in general clothing from unmodified multi-view video streams, without requiring much manual intervention by the user. Our algorithm begins by recording a so-called *reference sequence* of the person whose model is to be reconstructed (Fig. 3.1(a)). In this sequence, the person walks around for a few seconds in the same attire that should be part of the *animatable model*. From this reference sequence, we reconstruct a so-called *reference performance* which comprises a deforming mesh sequence that represents the detailed dynamic surface geometry of the moving actor over time, as well as a sequence of joint angle parameters of the underlying skeleton (Fig. 3.1(b)). The reference performance is captured by a marker-less performance capture technique that requires no manual intervention, even if the input shows difficult motion of a person in loose attire, Chapter 4.

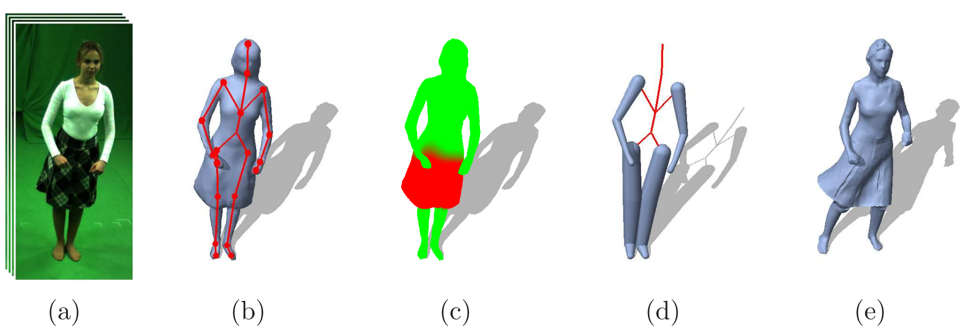


Figure 3.1: Overview of our processing pipeline: (a) multi-view video sequence of a reference performance; (b) performance capture result: skeleton motion + deforming surface; (c) cloth segmentation (red=regions of loose apparel); (d) estimated collision proxies; (e) After optimal cloth simulation parameters are found, arbitrary new animations can be created.

The dynamic surface geometry of the reference performance is automatically decomposed into approximately rigid body parts and non-rigidly deforming pieces of attire (Fig. 3.1(c)). In the same stage of the algorithm, we reconstruct approximate collision proxy geometry for both occluded and non-occluded parts of the body (Fig. 3.1(d)). These proxies are needed for accurate and fast cloth simulation, Chapter 5. After segmentation, we fit to each region of the deforming fabric a physics-based cloth simulation model, Chapter 6. The parameters of this model are numerically optimized to best reproduce the cloth motion in the reference sequence.

Given an animation model comprising of a skeleton with surface skinning for non-cloth regions, collision proxies, and a physics-based simulation model for apparel, arbitrary new animations of the character with realistic dynamic surface appearance can be created by simply changing skeletal motion parameters, Fig. 3.1(e) and Chapter 7.

4 Capturing the Reference Performance

The performance of an actor is captured by $c = 8$ synchronized and calibrated cameras running at 40 fps and providing 1004×1004 pixel resolution. This yields multi-view video frames $I_{c,t}$ for each camera and each frame $t = 1, \ldots, N$. The image silhouettes are extracted by chroma-keying, yielding N silhouette images $S_{c,t}$. Prior to recording, we acquire a model of the subject that comprises two components, a surface mesh $\mathcal{M}$ and an underlying bone skeleton $\mathcal{P}$. To obtain the initial geometry, we take a static full-body laser scan $\mathcal{M}_{high}$ of the actor wearing the attire for the reference sequence, which we then decimate to roughly 5000 triangles to obtain $\mathcal{M}$. The decimation is necessary to keep both performance capture and, later, performance simulation times in reasonable bounds. The skeleton with 36 degrees-of-freedom is inserted into the surface mesh by manually marking the joint positions. Thereafter, the method of [3] is employed to assign each vertex a weight that is used to deform the surface according to the skeleton pose. The latter paper also proposes a fully-automatic skeleton insertion approach which could be used instead of our manual procedure. For the deformation, we rely on quaternion blend skinning [15] that approximates the non-linear surface deformation better than linear blend skinning methods.

Please note that the original laser scan may contain holes, some of them due to bad measurements and some of them due to occlusion from the scanner, such as at the underneath of the skirt. We first automatically fill in all holes by means of Poisson surface reconstruction to create a closed surface. Some originally occluded regions may have been filled in a “semantically” wrong way, such as the sheet of triangles connecting the lower rim of skirt and the legs. We have developed a semi-automatic tool that proposes regions of “semantically” incorrect triangles. The user can optionally refine this proposal manually. Implausible triangles are thereby marked as invalid. They are weighted down in the reference performance capture and excluded from the animatable model estimation.

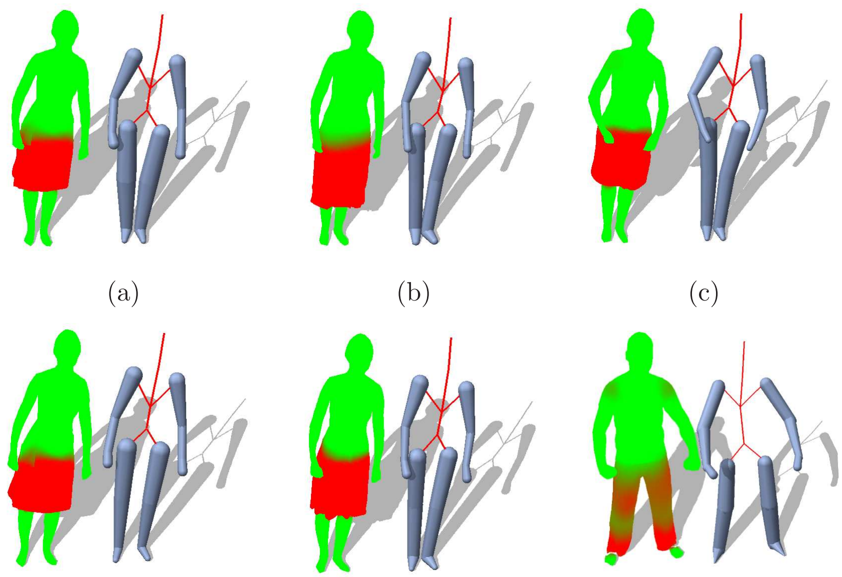
After pre-processing, the reference performance is captured without additional manual interaction using the method of [12]. In the following we will only briefly outline the main aspects of this approach. For each frame t of the reference sequence, the surface geometry in terms of the mesh $\mathcal{M}(t)$ and the joint motion parameters $\mathcal{P}(t)$ are estimated. To find the body pose in the current frame, the skeletal pose $\mathcal{P}(t)$ is optimized and quaternion blend skinning deforms the detailed surface $\mathcal{M}(t-1)$ of the previous time step into the current time step. Once converged, a Laplacian deformation framework is employed to estimate the fine surface deformation $\mathcal{M}(t)$ without limiting the deformation to comply with the skeleton. We estimate the globally optimal skeleton pose in two phases: The first phase searches for the nearest local minimum of an energy functional that assesses the model-to-image alignment based on silhouettes and texture features. To this end, the whole articulated skeleton is optimized locally. Subsequently, misaligned bones are detected by evaluating the energy of each body part and the skeleton is traversed to label all affected bones. The labeled bones are then re-estimated by a particle-based global optimization similar to [11] that is also used to initialize the tracker.

The output reference performance captures the geometry of the actor in each pose, and also captures the true look of both coarse and medium scale folds in clothing. At the same time, the bone skeleton configurations are reliably captured, even under occlusion by loose apparel, Fig. 3.1(b).

5 Cloth Segmentation and Approximation of Collision Geometry

Although human motion is mainly controlled by a kinematic skeleton, even the surface of a naked person never deforms entirely like a collection of rigid bodies. In practice, the non-rigid surface deformation of naked skin or areas with rather tight clothing, which is often due to muscle bulging, can be plausibly approximated in real-time by surface skinning methods. This does, however, not hold true for surface regions which represent loose pieces of apparel, since fabric exhibits starkly different non-rigid deformation behavior.

We therefore analyze the deforming surface geometry of the reference performance and automatically segment it into cloth and non-cloth regions, Fig. 5.1. To identify cloth regions, we search for surface areas that deform mainly non-rigidly. If a body part deforms approximately rigidly, mutual distances of vertices on that part hardly change over time. We take this insight and derive the following simplified but efficient rule to decide if a vertex $\mathbf{v_i}$ lies on a cloth segment: In the first mesh pose $\mathcal{M}(1)$ of the reference sequence, a ray is shot from vertex $\mathbf{v_i}$ in the negative local normal direction and intersected with the mesh surface on the opposite side. Vertex $\mathbf{v_i}$ is paired with the intersection point $\mathbf{p_i}$, which is described on the intersected face at sub-triangulation accuracy using barycentric coordinates. We now compute the standard deviation $\sigma(\mathbf{v_i}) = \sigma(d(\mathbf{v_i}, \mathbf{p_i}))$ of the Euclidean distance $d(\mathbf{v_i}, \mathbf{p_i})$ over the entire reference sequence. If this standard deviation exceeds a threshold t_{rigid} we conclude that $\mathbf{v_i}$ lies on a non-rigidly deforming surface area likely to be cloth. In other words, we find non-rigidly deforming surface regions by searching for sections with non-persistent cross-sectional areas. This is a weaker criterion than testing preservation of mutual distances between all points on a segment, but is much more efficient to compute and performs well in practice.



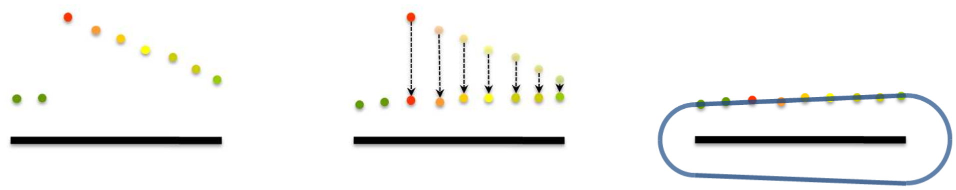
(a) (b) (c)

(d) (e) (f)

Figure 5.1: Cloth segmentation results and proxies found for several reference sequences showing different clothing styles (red=cloth, green=approximately rigid): (a) skirt 1, dancing sequence; (b) skirt 2, walking sequence; (c) skirt 3, dancing sequence; (d) skirt 1, walking sequence; (e) dress, dancing sequence; (f) capoeira sequence - cloth regions and transition areas at boundaries are reliably identified. Note that in the skirt 1 result the dent in the skirt region boundary is a woolen cord that was automatically excluded from the cloth simulation region since it stays mostly rigid.

Please note that we could also search for non-rigid segments by finding regions with starkly varying vertex to bone distances over time. However, in practice we found that this leads to frequent misclassification of actual body parts, in particular the arms, as cloth. The reason for this is that the skeleton of the actor does not provide all anatomical degrees of freedom (e.g. all shoulder and spine joints) which makes the least squares poses of the bones deviate from the true physiological positions. This deviation can lead to unintended bone distance variations which would result in wrong classification, in particular in the arms. Our purely surface-based decision criterion does not suffer from this problem and it also works for performances captured with skeleton-free measurement approaches [8].

Since our final animatable performance model combines physically-based and skeleton-based animation, we need to make sure that the transition between the two modalities is seamless. We therefore transform the $\sigma(\mathbf{v_i})$ values



(a) (b) (c)

Figure 5.2: 1-D proxy fitting to segmented data: (a) vertices around a bone with weights (green = high, red = low); (b) moving vertices closer according to their distance standard deviation σ; (c) fitting the conical section and end-caps.

into "*clothness weights*" $\delta(\mathbf{v_i}) \in [0, 1]$:

$$\delta(\mathbf{v_i}) = cl\left(\frac{\sigma(\mathbf{v_i}) - t_{rigid}}{t_{cloth} - t_{rigid}}\right) . \tag{5.1}$$

Here, the function cl clamps the weights to $[0, 1]$. t_{cloth} represents a threshold on $\sigma(\mathbf{v_i})$ above which vertex motion is fully determined by the physics simulation. The motion of a vertex with $\delta(\mathbf{v_i}) = 0$ is fully-controlled by skeleton motion + skinning, a vertex with $\delta(\mathbf{v_i}) = 1$ is fully cloth model controlled. Vertices with $\delta(\mathbf{v_i}) \in]0, 1[$ lie in the transition zone between cloth and skeleton simulation (e.g. the rim of the skirt and parts of the pants in Fig. 5.1). In Chapter 6.3 we explain how to blend the two animation types in those areas. To create a smooth clothness distribution and fill in potential holes in the estimation, we also perform a diffusion of the δ's on the surface.

For both the estimation of cloth simulation parameters, as well as for creating new animations it is essential that collisions of the fabric with the body are plausibly simulated. Unfortunately, the true shape of body geometry under wide attire is not directly visible and is thus neither part of the captured mesh sequence nor the skeleton model itself. However, by analyzing the reference performance and by capitalizing on general assumptions about human anatomy it is feasible to approximate even occluded body geometry.

We therefore take a different route and approximate collision geometry with capped conic collision proxies for which collision tests are computationally very efficient. One conic collision proxy can be fitted to each bone in the body. Depending on the sequence captured, however, it is often sufficient to only estimate proxies for arms and legs, such as in our test data with different skirts. The dimensions of each collision proxy are estimated from the segmented reference performance. We base our estimation procedure on the assumption that in areas with strong cloth motion collision proxies are most likely further away from the outer surface, whereas in mostly rigid areas the collision geometry should coincide with the surface of the captured mesh. We therefore use the following efficient 1-D fitting procedure to fit rotationally symmetric conic proxies to the first captured pose of the surface mesh, Fig. 5.2: Given a bone, all mesh vertices closest to that bone are transformed into local cylindrical coordinates and the rotational dimension around the bone is discarded. The parameters of the conic in 1-D are fitted to these vertices by means of a weighted linear regression. Since non-cloth vertices are closer to the true collision surface than cloth vertices, each vertex is weighted by the reciprocal of its deviation. Also, the cylindrical distance of each vertex from the bone is reduced by its standard deviation $\sigma(\mathbf{v_i})$ that was computed to find its clothness value. Finally, we post-process all the fitted proxies, and average the conic shapes of the corresponding bones on joints and on the left and right body halves to enforce symmetry.

6 Estimating a Cloth Model

6.1 Physically-based Cloth Model

For cloth simulation we use the position-based dynamics approach presented in [20]. Many previous cloth simulation methods do not use vertex positions explicitly as part of the simulation state, but compute them by integrating velocities. In contrast, in position-based dynamics vertex positions are explicit members of the simulation state which allows for direct position manipulation during simulation. The position-based dynamics framework bears a couple of advantages in our problem setting: it allows for fast real-time simulation, it is very stable, and it allows us to formulate our simulation constraints using the same mathematical framework. Here, we only briefly review the main concepts of our approach and refer the interested reader to the original paper by Müller et al. for details on the physics-based simulation.

Each piece of cloth is geometrically represented by the respective surface mesh region found by our segmentation approach (Chapter 5). The simulated geometry comprises vertices with positions $\mathbf{v_i}$, and velocities $\mathbf{u_i}$. Masses m_i for each vertex are approximated by the respective areas on the surface (i.e. $\frac{1}{3}$ of the area of the adjacent triangles). We set the masses of all vertices with $\delta(\mathbf{v_i}) = 0$ to infinity which guarantees simulation stability at boundaries between cloth and skeleton animation. In addition, the solver expects a set of n_c constraints C_j acting on the positions of the mesh vertices, as well as a set of external forces acting on the fabric $\mathbf{f}$. In analogy to the original paper by Müller et al., we represent each constraint C_j by means of a function defined over a subset of vertex position. The strength of each constraint is further controlled by a stiffness parameter $k_j \in [0, 1]$.

The simulation loop comprises two steps. First, an explicit Euler integration computes new vertex positions based on the current velocities and external forces. In a second step, a Gauss-Seidel like solver modifies the vertex position estimates such that the constraints are satisfied. This constraint projection step works locally, i.e. it handles all constraints separately from each other. This allows for simple handling of even non-linear constraints. Within the overall simulation loop, the projection of all constraints is iterated several times.

We employ four types of simulation constraints, collision constraints, stretching and bending constraints, as well as additional so-called blend constraints that enable us to seamlessly interpolate between cloth and skeleton simulation, as explained in more detail in Chapter 6.2. Stretching constraints take the form

$$C_{stretch}(\mathbf{v_a}, \mathbf{v_b}) = \|\mathbf{v_a} - \mathbf{v_b}\| - l_{a,b}\,, \tag{6.1}$$

where $l_{a,b}$ is the rest length of the edge between $\mathbf{v_a}$ and $\mathbf{v_b}$, which is the length of the respective edge in the decimated body scan $\mathcal{M}$. The bending constraint evaluates to

$$C_{bend}(\mathbf{v_a}, \mathbf{v_b}, \mathbf{v_c}, \mathbf{v_d}) = acos(\mathbf{n_l} \cdot \mathbf{n_r}) - \phi\,, \tag{6.2}$$

where $\mathbf{v_a}$, $\mathbf{v_b}$, $\mathbf{v_c}$, and $\mathbf{v_d}$ are the positions of vertices of two adjacent triangles that share a common edge, and $\mathbf{n_l}$ and $\mathbf{n_r}$ are the respective triangle normals. ϕ is the rest-angle between the two triangle normals, which in our case is set to 0 everywhere. For general pieces of apparel, the rest angle may be different from 0 and should actually be estimated from the reference sequence as well. However, in order to keep the estimation problem tractable, we exclude the rest angles from the optimization and set them to a constant conservative value.

We use spatial hashing to quickly find collisions and self-intersections, yielding additional collision constraints with fixed stiffness $k_{coll} = 1.0$. Finally, we include so-called blend constraints, which are defined for each vertex with a $\delta(\mathbf{v_a}) \in]0, 1[$ and read

$$C_{blend}(\mathbf{v_a}) = \mathbf{p_a}\,, \tag{6.3}$$

where $\mathbf{p_a}$ is the position the vertex should reach. The stiffness of this constraint, $k_{blend}(\mathbf{v_a})$, is vertex-specific and explained in more detail in Chapter 6.3.

External forces comprise gravity $\mathbf{f_g}$ and drag $\mathbf{f_d}$. Unlike the original paper by Mueller et al., we use a more sophisticated model for air-resistance similar to Bhat et al. [4]. The air drag force depends quadratically on the velocity in direction of the surface normal, and linearly in tangential direction:

$$\mathbf{f_d}(\mathbf{v_i}) = -0.5A(\mathbf{v_i})h(d_n\|\mathbf{u_{i,n}}\|^2\mathbf{n} + d_t\mathbf{u_{i,t}})\,. \tag{6.4}$$

Here, $A(\mathbf{v_i})$ is the area of the surface patch around the vertex $\mathbf{v_i}$ (in our case equal to m_i), and $h = 0.005$ the drag constant. $\mathbf{u_{i,n}}$ and $\mathbf{u_{i,t}}$ are the components of $\mathbf{u_i}$ in direction of normal $\mathbf{n}$ and tangential to it, respectively. The constants d_n and d_t are the drag coefficients.

Friction is handled by directly manipulating velocities of colliding vertices. To this end, vertex velocities are damped in the direction perpendicular to the collision normal, and reflected in the direction of the collision normal.

The step size of our simulation is set to coincide with the capture rate of our system, i.e. to 40 Hz. This allows easy synchronization between the data. However, it would be a straightforward process to decouple simulation and animation framerates by expressing fractional steps using linear blending between two updates, as is usually done in real-time simulations.

For each cloth region on the model, we therefore have to determine the three parameters $\rho = (d_n, d_t, k_{bend})$, i.e. the two drag coefficients, as well as the bending stiffness. Since our simulation is aimed at real-time performance the cloth exhibits a certain amount of stretching, even if the stretching stiffness is set to its maximum. This inherent flexibility is higher than that of all types of cloth we recorded in our experiments. Therefore, we use a fixed stretching stiffness of $k_{stretch} = 1$. In theory, the number of simulation steps n_{iter} per time frame of video could be considered a free parameter of our optimization problem as well. However, we aim for a character that can be simulated in real-time, and in real-time applications the number of iterations is usually fixed in order to balance resources. We therefore employ the same number of iterations ($n_{iter} = 16$) during both parameter optimization and creation of novel animations.

6.2 Combining Cloth and Skeleton Animation

Before explaining how to estimate cloth parameters, we describe how new poses of our final fully-animatable performance model are created. Positions of vertices with $\delta(\mathbf{v_i}) = 0$ are solely determined by the current joint parameters of the skeleton and dual quaternion skinning. Similarly, the positions of all pure cloth vertices with $\delta(\mathbf{v_i}) = 1$ are determined by the physical simulation described in Chapter 6.1. For blend vertices, $\delta(\mathbf{v_i}) \in]0, 1[$, the new pose is jointly determined by cloth and skeleton simulation. To this end, the position of $\mathbf{v_i}$ according to skeleton motion, $\mathbf{p_i}$, is included as a blend constraint into the physics simulation (Chapter 6.1) with a stiffness of $k_{blend}(\mathbf{v_i}) = (1 - \delta(\mathbf{v_i}))$. At each time step, collisions of the cloth with collision proxies are tested and included into the simulation.

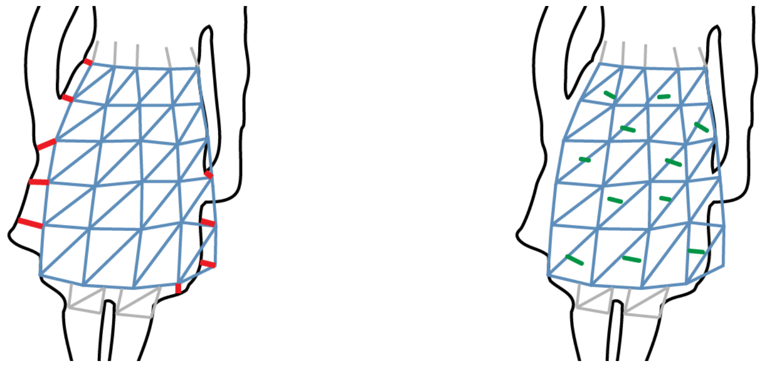


Figure 6.1: Two components of the error function on the cloth section (light blue) shown in 2D. (left) silhouette distance error: red lines between reprojected and measured silhouette points; (right) SIFT distance: green lines between predicted SIFT feature locations and measured SIFT feature locations.

6.3 Cloth Parameter Estimation

The parameters are found by running a numerical optimization that strives to minimize cloth deformation in the reference sequence. Our energy function measures two properties of the simulated animation: **1)** the alignment of the cloth region's silhouette edges with silhouette boundaries in all input images over time, $E_{sil}(\rho, t)$, and **2)** the alignment of the reprojected cloth with robust features in the interior of the fabric in all camera views, $E_{sift}(\rho, t)$. Both error terms are evaluated over the entire reference sequence and their contributions are combined, yielding the overall fitting error

$$E_{fit}(\rho) = \frac{1}{N} \sum_{t=1}^{N} (\alpha E_{sil}(\rho, t) + \beta E_{sift}(\rho, t)) \,. \tag{6.5}$$

Here, the values $\alpha = 1$ and $\beta = 10$ are empirically determined weights that are kept constant for all our estimations.

For each time step of video t and each input camera view c, a certain set of vertices on the cloth segment under consideration should project onto the silhouette boundary of the respective input frame. This set can be easily identified and we call it the set of silhouette rim vertices $\mathcal{V}_{c,t}$, Fig. 6.1(left). The silhouette error then evaluates to

$$E_{sil}(\rho, t) = \frac{1}{n_{sil}} \sum_{c} \sum_{\mathbf{v} \in \mathcal{V}_{\mathbf{c},\mathbf{t}}} d^2_{im}(\mathbf{q}_{S_{c,t}}(\mathbf{v}), \mathbf{q}'_{S_{c,t}}(\mathbf{v})) \,, \tag{6.6}$$

where $d_{im}(\mathbf{q}_{S_{c,t}}(\mathbf{v}), \mathbf{q}'_{S_{c,t}}(\mathbf{v}))$ is the image space distance between the reprojection $\mathbf{q}'(\mathbf{v})$ of silhouette rim vertex $\mathbf{v}$ into silhouette image $S_{c,t}$ and the closest boundary point of the measured silhouette in the same image, $\mathbf{q}_{S_{c,t}}(\mathbf{v})$. n_{sil} is the number of all rim vertices from all $\mathcal{V}_{c,t}$ in this frame.

For each time step of video and each camera view, we also compute a set of SIFT features [17], $\mathcal{F}_{c,t}$. In addition, we establish correspondences between features in two subsequent time steps for each camera view, Fig. 6.1(right). The feature-based energy term then reads

$$E_{sift}(\rho, t) = \frac{1}{n_{sift}} \sum_{c} \sum_{\mathbf{e} \in \mathcal{F}_{c,t}} d_{im}^2(\mathbf{o}_{I_{c,t}}(\mathbf{e}), \mathbf{o}'_{I_{c,t}}(\mathbf{e}))\,, \tag{6.7}$$

where $\mathbf{o}_{I_{c,t}}(\mathbf{e})$ is the 2D image position of feature $\mathbf{e}$ in image $I_{c,t}$. $\mathbf{o}'_{I_{c,t}}(\mathbf{e})$ is the predicted image position of feature $\mathbf{e}$ at time t and n_{sift} is the number of all features from all $\mathcal{F}_{c,t}$ in this frame. For each camera c, these predicted image positions are obtained from feature positions at the previous time step as follows: Each feature $\mathbf{e}$ at time step $t-1$ is projected back onto the final cloth surface at $t-1$ using the camera matrix of camera c. For each feature $\mathbf{e}$, this yields a 3D position on the surface of the mesh $\mathbf{p_{c,t-1}}(\mathbf{e})$, expressed in barycentric coordinates relative to the enclosing mesh triangle. The position at the current time step t, $\mathbf{p'_{c,t}}(\mathbf{e})$, is predicted from $\mathbf{p_{c,t-1}}(\mathbf{e})$ by the cloth simulation. The predicted image positions $\mathbf{o}'_{I_{c,t}}(\mathbf{e})$ are then obtained by reprojecting $\mathbf{p'_{c,t}}(\mathbf{e})$ back into each respective camera view.

When evaluating E_{fit}, both E_{sil} and E_{sift} are evaluated for each frame t after the new model pose is determined according to the method in Chapter 6.2, using $n_{iter} = 16$ cloth simulation iterations. Please note that we intentionally formulate our energy function in terms of image features of the original reference sequence, and not in terms of a 3D comparison to the tracked meshes from the reference performance. In this way we stay as close as possible to the measured data and prevent unintended fitting to potential inaccuracies in the tracked 3D reference performance. In addition, our formulations of E_{sil} and E_{sift} are memory efficient as they only require storage of 2D silhouette rim points and 2D feature locations for each input frame. Holding all image data of the reference sequence simultaneously in memory would be impossible, and we therefore refrain from more complex distance functions, e.g. distance-field based evaluations as in [8].

The combination of silhouette and SIFT features is essential for our goodness-of-fit measure. Silhouette data alone would not suffice since a lot of information on cloth behavior can be extracted from the inner regions of the cloth in all images. Silhouette information is important to asses the overall appearance of the cloth boundaries. We would also like to note that our feature handling with correspondence re-computation for each frame pair usually manages to make the majority of feature points found in all images

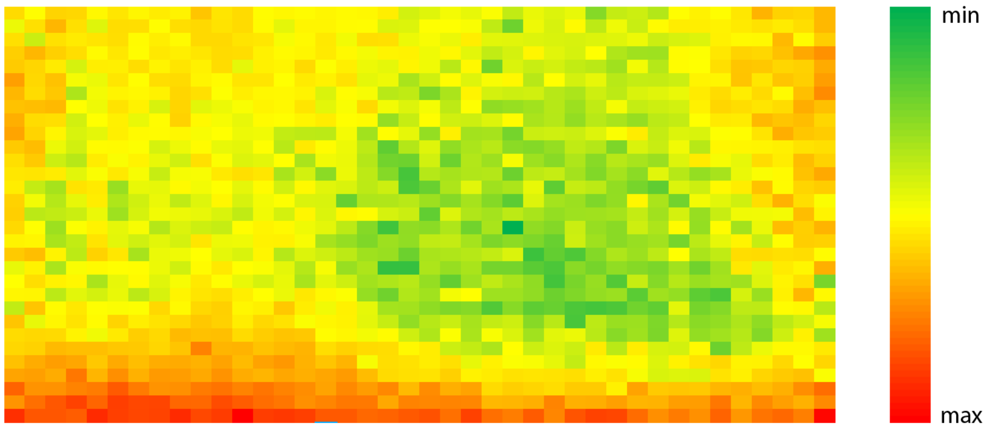


Figure 6.2: Plot of the energy function over k_{bend} on the horizontal and d_n on the vertical axis. Red color represents high energy, green low. A clear minimal energy area can be spotted near the lower right.

accessible to the optimization. Multi-frame correspondence estimation more frequently leads to lost feature tracks, in particular on motions with frequent turns and occlusions.

Unfortunately, the error function E_{fit} is non-convex and exhibits many local minima, as shown in Fig. 6.2 for two of the three simulation parameters (k_{bend} and d_n). This multi-modalness comes as no surprise since the cloth simulation behavior is highly non-linear causing potentially large changes in geometric cloth appearance for only small changes in ρ. Globally, there is a distinct minimal parameter region which we have to find. Any form of gradient-based optimization strategy would fail to achieve this goal. We therefore resort to a sampling-based minimization strategy which is suitable for such multi-modal error landscapes. In particular, we employ a greedy stochastic search similar to simulated annealing. The minimization is initialized by placing K^3 (typically $K = 4$) samples on regularly spaced grid positions of the space of simulation parameters, with $d_n \in [0, 10]$, $d_t \in [0, 50]$, and $k_{bend} \in [0, 1]$. The sample with the lowest error, $\tilde{\rho}$, is used as starting point for the greedy optimization $\rho_1 = \tilde{\rho}$. To this end, we first estimate a particle distribution from which to generate new samples. In our case, this distribution is a three-dimensional Gaussian $\mathcal{N}_n(\mu(n), \Gamma(n))$, whose mean $\mu(n)$ and diagonal covariance matrix $\Gamma(n)$ are dependent on the iteration n of the optimizer. Each diagonal entry of $\Gamma(1)$, namely $\sigma_{d_n}(1)$, $\sigma_{d_t}(1)$, and $\sigma_{k_{bend}}(1)$ is set to be the average distance between the grid samples used for initialization along the respective dimension. The covariance entries are reduced over time by multiplying with a linear scaling function $\Gamma(n) = g(n)\Gamma(1)$. Here $g(1) = 1$, and it linearly decreases to $g(n) = 0$.

Starting from $\tilde{\rho}$, the greedy optimizer generates one new sample per iteration by drawing from $\mathcal{N}_n$. In each iteration, $\mu(n)$ is set to the optimal particle from the previous iteration, ρ_{n-1}. A new particle is accepted if its error is lower than the error of ρ_{n-1}, otherwise $\rho_n = \rho_{n-1}$ and a new sample is drawn. After $n_{sam} = 192$ iterations, the optimizer returns the average of the ten last particles as optimum ρ_{max}. More advanced sampling strategies exist, but many of them, such as particle filter variants, are ruled out due to the much larger number of required particle evaluations that would lead to unreasonable running times.

7 Results and Discussion

The output of our reconstruction process is a fully-rigged character comprising of an animation skeleton, surface skinning weights, collision proxies, and a physically-based cloth model for each region with wavy apparel. In total, we captured 14 performances with a female test subject using the method described in Chapter 4, each comprising of between 580 and 1300 frames. In those sequences, two actors wear six different types of apparel: a skirt made of a fabric with medium thickness (s1), a long skirt with a rather light material (s2), a medium length skirt with comparably stiff material (s3), a dress (s4), loose track pants (s5), and skin-tight pants (s6). The capoeira sequences with clothing style s5 were kindly provided by [8]. For each attire, several different sequences were captured with motion styles ranging from walking over dancing to kicking and turning in the capoeira scene. Only for the first five clothing styles, we ran our full pipeline to estimate an animatable performance model. Clothing style s6 was only used to acquire motion data for new animations of the other performances. In the following we discuss several aspects of our approach. All final renderings in the paper and the accompanying video were done by real-time transfer of the pose of the decimated mesh $\mathcal{M}$ used for simulation to the high resolution body scan $\mathcal{M}_{high}$ using a method similar to [8].

7.1 Segmentation and Cloth Parameter Estimation

For each style of apparel, we choose one reference sequence to reconstruct an animatable performance from, Table 7.1. For skirt s1, we also reconstructed animatable models from two different reference motions, walking and dancing, to verify stability. Our segmentation approach was able to reliably identify the loose cloth areas in all reference scenes using fixed segmentation thresholds. The segmentation of s1 in two different reference sequences is almost identical demonstrating the stability of our method, Fig. 5.1(a,d). The segmentation of the pants in the capoeira sequence shows that our algorithm also faithfully handles tighter clothing which is less wavy than the skirts. The lower end of the pants (which is very wide in comparison to the leg) is marked as loose cloth, while segments further up on the leg receive an intermediate clothness weight since they are closer to the skin and thus less wavy. Here, our algorithm also correctly identifies the sleeves of the t-shirt as slightly wider apparel, Fig. 5.1(f).

Table 7.1 lists the set of cloth simulation parameters which we estimated for the different clothing styles. According to our expectations, the stiffest skirt s3 (Fig. 7.1(middle)) was found to have the highest bending resistance, the dress s4 (Fig. 7.1(bottom)) made of fabric with medium density has a medium bending resistance, and the light long skirt s2 (Fig. 7.1(top)) a very low one. The visual draping characteristics of all simulated skirts closely match the draping behaviors of their real world counterparts.

When estimating parameters for the same skirt s1 from two different reference sequences (a simple walking and a dancing sequence), very similar simulation parameters were found (see parameter sets ρ_{1a} and ρ_{1b} in Table 7.1). The resulting animations with both parameter sets are, in turn, visually very similar. Although the reliability of the parameter estimation is certainly influenced by the contents of the reference sequence, this test shows that for reasonably long scenes with sufficient cloth motion, parameters can be faithfully found.

Our simulation model is physically plausible and allows realistic simulation, but it does not allow us to determine physical material parameters accurately. Comparison to measured material properties using material science test methods is therefore not straightforward. To nonetheless test the feasibility of estimated parameters, we performed a cross-validation of the computed optimal parameters for all clothing styles in terms of the cloth estimation error E_{fit}, Table 7.2. As one can see, the parameters found for each attire are the true optimal ones in comparison to all other reconstructed clothing styles.

7.2 User Study

Since our goal is realistic simulation of new performances, and realism can ultimately only be judged by a human observer, we performed a survey with 49 test persons. Each of the participants was shown a web page with one input camera view of the reference sequence of skirt s3, as well as three simulated performances from the same camera view (see accompanying additional

sequence	set name	d_n	d_t	k_{bend}	$frames$
s1a: dancing	ρ_{1a}	0.5	6.8	0.36	1300
s1b: walking	ρ_{1b}	1.3	5.8	0.26	800
s2: walking	ρ_2	1.7	44.1	0.003	1100
s3: dancing	ρ_3	0.15	12.0	0.98	1200
s4: dancing	ρ_4	0.5	0.08	0.43	800
s5: capoeira	ρ_5	2.3	2.8	0.03	400

Table 7.1: Estimated cloth simulation parameters for the skirts and the pants made of different fabrics. ρ_{1a} and ρ_{1b} were estimated for the same skirt from different reference sequences.

video). The simulated performances used the reference skeleton motion but different simulation parameters for the skirt: a medium light material (*video A*: $d_n = 2.0$, $d_t = 4.0$, $k_{bend} = 0.3$), a material exhibiting very high drag (*video B*: $d_n = 8.0$, $d_t = 4.0$, $k_{bend} = 0.8$), and the true estimated material parameters (*video C*: ρ_3). Please note that in terms of fitting error *video A* is closer to the truth than *video B*. The users were asked to rank the simulations according to the closeness of the simulated cloth behavior to the reference input (1=most similar, 3=most dissimilar). 39 (79.6%) participants ranked video C as 1. Out of these participants 32 (82.0%) also ranked video A in second place. From those users which did not rank video C correctly, 7 still ranked video B as the highest and video C as second highest. We therefore conclude that we can capture and simulate cloth behavior realistically, and that users are even able to rank simulations according to closeness of material parameters to the optimum.

7.3 Creating New Animations

Given a reconstructed animation model for a person, arbitrary new motions can be simulated by simply modifying the joint motion parameters of the skeleton. All motion sequences used for new animations were either captured by our performance capture algorithm from people wearing s6, or stem from a database of motion capture files. All new skeleton motions we employ are parts of the performances that were captured with the method from Chapter 4, but not used for reconstructing an animatable model. Note that data from any type of motion capture system or key-frame animations created by an animator would be equally feasible. In all new animations, both the overall appearance of the body and the deformation behavior of the cloth are very lifelike since even subtle deformation details are realistically reproduced,

	ρ_{1a}	ρ_2	ρ_3	ρ_4
s1	132.97	137.50	137.47	134.70
s2	218.90	214.37	224.89	226.37
s3	168.97	173.54	158.26	169.10
s4	142.39	145.18	145.79	137.62

Table 7.2: Cross-validation matrix: rows = clothing style, columns = parameter set - Entries are simulation parameter fitting errors: for one particular attire, the optimized parameters are also optimal in comparison to all other estimated parameter sets.

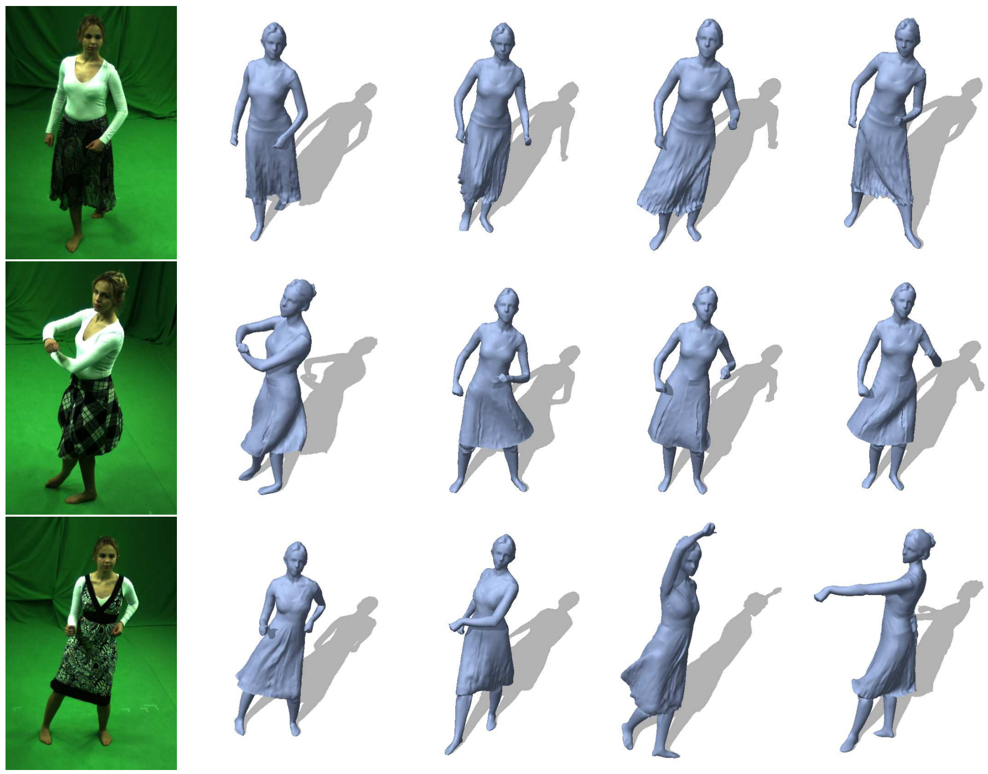
Fig. 7.1 and accompanying video.

Please note that the mesh pre-processing step described in Chapter 4 may lead to holes in the geometry, that may become visible during new animations. For instance, it may become visible that there is no real upper leg geometry under the skirt. For visualization, we therefore render the collision proxies as geometry approximation of the invisible parts of the upper leg to reduce this effect. For the long skirt s2, however, the proxy visualization does not look pleasing and we therefore refrain from it, exposing sometimes the missing leg geometry. In a practical animation setting, any standard 3D package could be used to create more sophisticated geometry in those regions.

7.4 Performance

It takes around 2.5 hours to capture a reference performance of 1000 frames on an Intel Core 2 Duo with 3.0 GHz. All sequences we captured, both reference sequences and skeleton motions only used for new animations, were tracked fully-automatically. Segmentation of cloth takes around 5 seconds. Estimation of cloth simulation parameters takes around 3-4 hours for a reference sequence consisting of roughly 1000 frames.

The actual computation of new poses and the transfer to the higher resolution mesh of about $25k$ triangles run in real-time with 60 fps on the same machine. Please note that our simulation code runs single-threaded and does not use any GPU acceleration for skinning, simulation, or detail transfer. The cloth simulation currently runs with the same step size as the animation input (i.e. 40 fps), but could easily be decoupled in both rendering of new animations as well as in the optimization procedure.



(a) (b) (c) (d) (e)

Figure 7.1: Results for skirts 2, 3, and the dress model (from top to bottom). Column (a) input frame from reference sequence; (b) the reconstructed animatable model simulating the same pose; (c)-(e) subsequent frames of a newly created animation for which only motion parameters of the underlying skeleton were given.

7.5 Discussion

Our approach is subject to a few limitations. First, since we start off with a closed full-body laser scan of a person, we are limited in what type of clothing we can handle. Without manual post-processing it will not be possible to handle apparel that shows difficult topology changes while moving, such as the opening up of a coat. However, we believe that additional user input would render our method capable of handling these situations, too. Further on, the quality of cloth reconstruction is dependent on the type of reference sequence captured. If the cloth does not move very much, we will not be able to identify pieces of apparel automatically. However, this is not a principal limitation of our method since in a general recording scenario it is fair to instruct the person to move in such a way that all pieces of apparel are exposed.

The results of the cloth parameter optimization are dependent on the input sequence, too. If the recorded sequence does not demonstrate the behavior of the material in a reasonable way (for example when recording only a walking sequence without any turns), the optimization will most likely produce a different parameter set than for a highly dynamic scene of the same skirt. Again, properly instructing the actor remedies this problem.

We would also like to note that in a certain range different cloth simulation parameters may lead to very similar cloth behavior. To a certain extent one can see this in the error plot Fig. 6.2. For us, however, it is sufficient to find any such optimum within the globally low error region of the error landscape, since either of them will lead to plausible renderings.

The chosen cloth simulation method also imposes a few limitations. The maximal material stiffness we are able to simulate with the step size and number of iterations per frame we selected is limited. With our settings the cloth will always exhibit a certain amount of stretchiness that may not be there in the real fabric, even with maximal stiffness. Sometimes this is visible in our result animations. We could simulate higher stretching stiffnesses by employing hierarchical methods (as presented in [19]) or by using more sophisticated simulation methods. However, this would usually prevent us from rendering the cloth in real-time. Another consequence of the choice of simulation is that cloth simulation parameters are not independent of step size and number of iterations. This means that we always have to estimate parameters for exactly the same simulation settings that we also use for creating new animations and have to rerun the optimization if any of these settings change.

One might argue that instead of using our motion-driven cloth segmentation, one could identify loose apparel via distance thresholding between corresponding surface points of two laser scans of a subject, one with full attire and one with underwear only. However, this strategy would have several drawbacks. First, the two scans would need to be geometrically aligned. Scan alignment is a very difficult and potentially error prone process, in particular if the two scans exhibit pose differences, or if the apparel is very loose or made of very thick fabric yielding stark geometric differences between the scans. However, even if perfect alignment was achieved, one cannot easily differentiate regions of tight and wavy cloth with simple distance thresholding, let alone create a non-binary mask of clothness weights. Such a simple geometric strategy would fail to identify regions of loose apparel that are accidentally close to the skin in a single pose, but may move around significantly in other postures. Only with a motion-driven segmentation it is possible to identify those regions, for instance the upper seam of the skirt and the areas of different intermediate clothness weights in the track pants, Fig. 5.1. Admittedly, one could use a scan in underwear or fit a parametric body model like to SCAPE [2] to represent occluded geometry at more detail. In practice we found that any such additional step is unnecessary. Our proxies are very easy-to-fit, allow for faster collision handling than more detailed geometry, and fully suffice to faithfully simulate realistic cloth collisions with the body.

As we demonstrated, the level of detail that we can achieve in any of the newly created animations is fairly high, and it is trivial to create new sequences using the standard toolbox of the animator. As it was originally our intention, we anticipate that the reconstructed animatable actors will be used to create more realistic real-time characters for games and networked virtual environments. Admittedly, even though the models are fairly detailed, they may still be too coarse for application in feature films, where real-time rendering ability is not the primary concern. However, we are confident that a similar methodology with higher resolution sensors and more advanced manual post-processing will make a similar reconstruction approach also feasible for applications with highest production standards.

7.6 Conclusion

We have presented a new method for video-based reconstruction of a fully-animatable model of a human actor from a single multi-view reference sequence. The model comprises detailed surface geometry, a skeleton with surface skinning, cloth regions with physically-based simulation parameters, and approximations of occluded geometry. In contrast to previous performance capture methods, our results cannot only be played back, but animators can create completely new animations that show similarly realistic body motion and cloth behavior as the real subject in the reference video. Both visual inspection of newly created animations and the results of a user study confirmed, that we are able to capture animatable models of highest realism that can be rendered in real-time.

Bibliography

[1] B. Allen, B. Curless, Z. Popović, and A. Hertzmann. Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis. In *Proc. of SCA*, pages 147–156, 2006.

[2] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. SCAPE: Shape completion and animation of people. In *ACM TOG (Proc. SIGGRAPH '05)*, Los Angeles, California, August 2005.

[3] I. Baran and J. Popović. Automatic rigging and animation of 3d characters. *ACM TOG (Proc. SIGGRAPH '07)*, 2007.

[4] K. S. Bhat, C. D. Twigg, J. K. Hodgins, P. K. Khosla, Z. Z. Popović, and S. M. Seitz. Estimating cloth simulation parameters from video. In *Proc. of SCA*, 2003.

[5] D. Bradley, T. Popa, A. Sheffer, W. Heidrich, and T. Boubekeur. Markerless garment capture. *ACM TOG (Proc. SIGGRAPH '08)*, 2008.

[6] J. Carranza, C. Theobalt, M. Magnor, and H.-P. Seidel. Free-viewpoint video of human actors. In *ACM TOG (Proc. SIGGRAPH '03)*, 2003.

[7] K.-J. Choi and H.-S. Ko. Research problems in clothing simulation. *Computer-Aided Design*, 37(6):585–592, 2005.

[8] E. de Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H.-P. Seidel, and S. Thrun. Performance capture from sparse multi-view video. In *ACM TOG (Proc. SIGGRAPH'08)*, 2008.

[9] E. de Aguiar, C. Theobalt, C. Stoll, and H.-P. Seidel. Marker-less deformable mesh tracking for human shape and motion capture. In *Proc. IEEE CVPR*, 2007.

[10] E. de Aguiar, C. Theobalt, S. Thrun, and H.-P. Seidel. Automatic conversion of mesh animations into skeleton-based animations. *Proc. Eurographics EG'08*, 2008.

[11] J. Gall, B. Rosenhahn, T. Brox, and H.-P. Seidel. Optimization and filtering for human motion capture – a multi-layer framework. *International Journal of Computer Vision*, 2008.

[12] J. Gall, C. Stoll, E. de Aguiar, C. Theobalt, B. Rosenhahn, and H.-P. Seidel. Motion capture using simultaneous skeleton tracking and surface estimation. In *Proc. IEEE CVPR*, 2009.

[13] N. Hasler, M. Asbach, B. Rosenhahn, J.-R. Ohm, and H.-P. Seidel. Physically based tracking of cloth. In *In Proc. VMV 2006*, pages 49–56, 2006.

[14] D. L. James and C. D. Twigg. Skinning mesh animations. *ACM TOG (Proc. SIGGRAPH'05)*, 2005.

[15] L. Kavan, S. Collins, J. Žára, and C. O'Sullivan. Skinning with dual quaternions. In *Symposium on Interactive 3D graphics and games*, pages 39–46, 2007.

[16] S. Kircher and M. Garland. Editing arbitrarily deforming surface animations. In *ACM TOG (Proc. SIGGRAPH'06)*, 2006.

[17] D. G. Lowe. Object recognition from local scale-invariant features. In *Proc. ICCV*, volume 2, page 1150ff, 1999.

[18] W. Matusik, C. Buehler, R. Raskar, S. Gortler, and L. McMillan. Image-based visual hulls. In *ACM TOG (Proc. SIGGRAPH'00)*, 2000.

[19] M. Müller. Hierarchical position based dynamics. In *Proc. VRIPhys*, 2008.

[20] M. Müller, B. Heidelberger, M. Hennix, and J. Ratcliff. Position based dynamics. *J. Vis. Comun. Image Represent.*, 18(2):109–118, 2007.

[21] S. I. Park and J. K. Hodgins. Data-driven modeling of skin and muscle deformation. *ACM TOG (Proc. SIGGRAPH'08)*, 2008.

[22] R. Poppe. Vision-based human motion analysis: An overview. *CVIU*, 108(1-2):4–18, 2007.

[23] D. Pritchard and W. Heidrich. Cloth motion capture. In *Proc. Eurographics EG'03*, 2003.

[24] B. Rosenhahn, U. Kersting, K. Powell, R. Klette, G. Klette, and H.-P. Seidel. A system for articulated tracking incorporating a clothing model. *Machine Vision and Applications*, 18(1):25–40, 2007.

[25] P. Sand, L. McMillan, and J. Popović. Continuous capture of skin deformation. *ACM TOG (Proc. SIGGRAPH'03)*, 2003.

[26] V. Scholz, T. Stich, M. Keckeisen, M. Wacker, and M. Magnor. Garment motion capture using color-coded patterns. In *Proc. Eurographics EG'05*, 2005.

[27] X. Shi, K. Zhou, Y. Tong, M. Desbrun, H. Bao, and B. Guo. Example-based dynamic skinning in real time. *ACM TOG (Proc. SIGGRAPH '08)*, 2008.

[28] J. Starck and A. Hilton. Surface capture for performance based animation. *IEEE CGAA*, 27(3):21–31, 2007.

[29] D. Vlasic, I. Baran, W. Matusik, and J. Popović. Articulated mesh animation from multi-view silhouettes. *ACM TOG (Proc. SIGGRAPH '08)*, 2008.

[30] M. Waschbüsch, S. Würmlin, D. Cotting, F. Sadlo, and M. Gross. Scalable 3D video of dynamic scenes. In *Proc. Pacific Graphics*, pages 629–638, 2005.

[31] R. White, K. Crane, and D. Forsyth. Capturing and animating occluded cloth. In *ACM TOG (Proc. SIGGRAPH'07)*, 2007.

[32] W. Xu, K. Zhou, Y. Yu, Q. Tan, Q. Peng, and B. Guo. Gradient domain editing of deforming mesh sequences. In *ACM TOG (Proc. SIGGRAPH '07)*, 2007.

[33] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski. High-quality video view interpolation using a layered representation. *ACM TOG (Proc. SIGGRAPH '04)*, 2004.

Below you find a list of the most recent technical reports of the Max-Planck-Institut für Informatik. They are available via WWW using the URL `http://www.mpi-inf.mpg.de`. If you have any questions concerning WWW access, please contact `reports@mpi-inf.mpg.de`. Paper copies (which are not necessarily free of charge) can be ordered either by regular mail or by e-mail at the address below.

Max-Planck-Institut für Informatik
Library
attn. Anja Becker
Stuhlsatzenhausweg 85
66123 Saarbrücken
GERMANY
e-mail: `library@mpi-inf.mpg.de`

MPI-I-2009-RG1-002	P. Wischnewski, C. Weidenbach	Contextual rewriting
MPI-I-2009-5-006	S. Bedathur, K. Berberich, J. Dittrich, N. Mamoulis, G. Weikum	Scalable phrase mining for ad-hoc text analytics
MPI-I-2009-5-004	N. Preda, F.M. Suchanek, G. Kasneci, T. Neumann, G. Weikum	Coupling knowledge bases and web services for active knowledge
MPI-I-2009-5-003	T. Neumann, G. Weikum	The RDF-3X engine for scalable management of RDF data
MPI-I-2008-RG1-001	A. Fietzke, C. Weidenbach	Labelled splitting
MPI-I-2008-5-004	F. Suchanek, M. Sozio, G. Weikum	SOFI: a self-organizing framework for information extraction
MPI-I-2008-5-003	F.M. Suchanek, G. de Melo, A. Pease	Integrating Yago into the suggested upper merged ontology
MPI-I-2008-5-002	T. Neumann, G. Moerkotte	Single phase construction of optimal DAG-structured QEPs
MPI-I-2008-5-001	F. Suchanek, G. Kasneci, M. Ramanath, M. Sozio, G. Weikum	STAR: Steiner tree approximation in relationship-graphs
MPI-I-2008-4-003	T. Schultz, H. Theisel, H. Seidel	Crease surfaces: from theory to extraction and application to diffusion tensor MRI
MPI-I-2008-4-002	W. Saleem, D. Wang, A. Belyaev, H. Seidel	Estimating complexity of 3D shapes using view similarity
MPI-I-2008-1-001	D. Ajwani, I. Malinger, U. Meyer, S. Toledo	Characterizing the performance of Flash memory storage devices and its impact on algorithm design
MPI-I-2007-RG1-002	T. Hillenbrand, C. Weidenbach	Superposition for finite domains
MPI-I-2007-5-003	F.M. Suchanek, G. Kasneci, G. Weikum	Yago : a large ontology from Wikipedia and WordNet
MPI-I-2007-5-002	K. Berberich, S. Bedathur, T. Neumann, G. Weikum	A time machine for text search
MPI-I-2007-5-001	G. Kasneci, F.M. Suchanek, G. Ifrim, M. Ramanath, G. Weikum	NAGA: searching and ranking knowledge
MPI-I-2007-4-008	J. Gall, T. Brox, B. Rosenhahn, H. Seidel	Global stochastic optimization for robust and accurate human motion capture
MPI-I-2007-4-007	R. Herzog, V. Havran, K. Myszkowski, H. Seidel	Global illumination using photon ray splatting
MPI-I-2007-4-006	C. Dyken, G. Ziegler, C. Theobalt, H. Seidel	GPU marching cubes on shader model 3.0 and 4.0
MPI-I-2007-4-005	T. Schultz, J. Weickert, H. Seidel	A higher-order structure tensor
MPI-I-2007-4-004	C. Stoll, E. de Aguiar, C. Theobalt, H. Seidel	A volumetric approach to interactive shape editing
MPI-I-2007-4-003	R. Bargmann, V. Blanz, H. Seidel	A nonlinear viseme model for triphone-based speech synthesis
MPI-I-2007-4-002	T. Langer, H. Seidel	Construction of smooth maps with mean value coordinates
MPI-I-2007-4-001	J. Gall, B. Rosenhahn, H. Seidel	Clustered stochastic optimization for object recognition and pose estimation

MPI-I-2007-2-001	A. Podelski, S. Wagner	A method and a tool for automatic veriication of region stability for hybrid systems
MPI-I-2007-1-003	A. Gidenstam, M. Papatriantafilou	LFthreads: a lock-free thread library
MPI-I-2007-1-002	E. Althaus, S. Canzar	A Lagrangian relaxation approach for the multiple sequence alignment problem
MPI-I-2007-1-001	E. Berberich, L. Kettner	Linear-time reordering in a sweep-line algorithm for algebraic curves intersecting in a common point
MPI-I-2006-5-006	G. Kasnec, F.M. Suchanek, G. Weikum	Yago - a core of semantic knowledge
MPI-I-2006-5-005	R. Angelova, S. Siersdorfer	A neighborhood-based approach for clustering of linked document collections
MPI-I-2006-5-004	F. Suchanek, G. Ifrim, G. Weikum	Combining linguistic and statistical analysis to extract relations from web documents
MPI-I-2006-5-003	V. Scholz, M. Magnor	Garment texture editing in monocular video sequences based on color-coded printing patterns
MPI-I-2006-5-002	H. Bast, D. Majumdar, R. Schenkel, M. Theobald, G. Weikum	IO-Top-k: index-access optimized top-k query processing
MPI-I-2006-5-001	M. Bender, S. Michel, G. Weikum, P. Triantafilou	Overlap-aware global df estimation in distributed information retrieval systems
MPI-I-2006-4-010	A. Belyaev, T. Langer, H. Seidel	Mean value coordinates for arbitrary spherical polygons and polyhedra in $\mathbb{R}^3$
MPI-I-2006-4-009	J. Gall, J. Potthoff, B. Rosenhahn, C. Schnoerr, H. Seidel	Interacting and annealing particle filters: mathematics and a recipe for applications
MPI-I-2006-4-008	I. Albrecht, M. Kipp, M. Neff, H. Seidel	Gesture modeling and animation by imitation
MPI-I-2006-4-007	O. Schall, A. Belyaev, H. Seidel	Feature-preserving non-local denoising of static and time-varying range data
MPI-I-2006-4-006	C. Theobalt, N. Ahmed, H. Lensch, M. Magnor, H. Seidel	Enhanced dynamic reflectometry for relightable free-viewpoint video
MPI-I-2006-4-005	A. Belyaev, H. Seidel, S. Yoshizawa	Skeleton-driven laplacian mesh deformations
MPI-I-2006-4-004	V. Havran, R. Herzog, H. Seidel	On fast construction of spatial hierarchies for ray tracing
MPI-I-2006-4-003	E. de Aguiar, R. Zayer, C. Theobalt, M. Magnor, H. Seidel	A framework for natural animation of digitized models
MPI-I-2006-4-002	G. Ziegler, A. Tevs, C. Theobalt, H. Seidel	GPU point list generation through histogram pyramids
MPI-I-2006-4-001	A. Efremov, R. Mantiuk, K. Myszkowski, H. Seidel	Design and evaluation of backward compatible high dynamic range video compression
MPI-I-2006-2-001	T. Wies, V. Kuncak, K. Zee, A. Podelski, M. Rinard	On verifying complex properties using symbolic shape analysis
MPI-I-2006-1-007	H. Bast, I. Weber, C.W. Mortensen	Output-sensitive autocompletion search
MPI-I-2006-1-006	M. Kerber	Division-free computation of subresultants using bezout matrices
MPI-I-2006-1-005	A. Eigenwillig, L. Kettner, N. Wolpert	Snap rounding of Bézier curves
MPI-I-2006-1-004	S. Funke, S. Laue, R. Naujoks, L. Zvi	Power assignment problems in wireless communication
MPI-I-2005-5-002	S. Siersdorfer, G. Weikum	Automated retraining methods for document classification and their parameter tuning
MPI-I-2005-4-006	C. Fuchs, M. Goesele, T. Chen, H. Seidel	An emperical model for heterogeneous translucent objects
MPI-I-2005-4-005	G. Krawczyk, M. Goesele, H. Seidel	Photometric calibration of high dynamic range cameras
MPI-I-2005-4-004	C. Theobalt, N. Ahmed, E. De Aguiar, G. Ziegler, H. Lensch, M.A. Magnor, H. Seidel	Joint motion and reflectance capture for creating relightable 3D videos
MPI-I-2005-4-003	T. Langer, A.G. Belyaev, H. Seidel	Analysis and design of discrete normals and curvatures
MPI-I-2005-4-002	O. Schall, A. Belyaev, H. Seidel	Sparse meshing of uncertain and noisy surface scattered data