Performance Capture from Sparse Multi-view Video

Edilson de Aguiar* Carsten Stoll* Christian Theobalt† Naveed Ahmed* Hans-Peter Seidel* Sebastian Thrun†

*MPI Informatik, Saarbruecken, Germany
†Stanford University, Stanford, USA

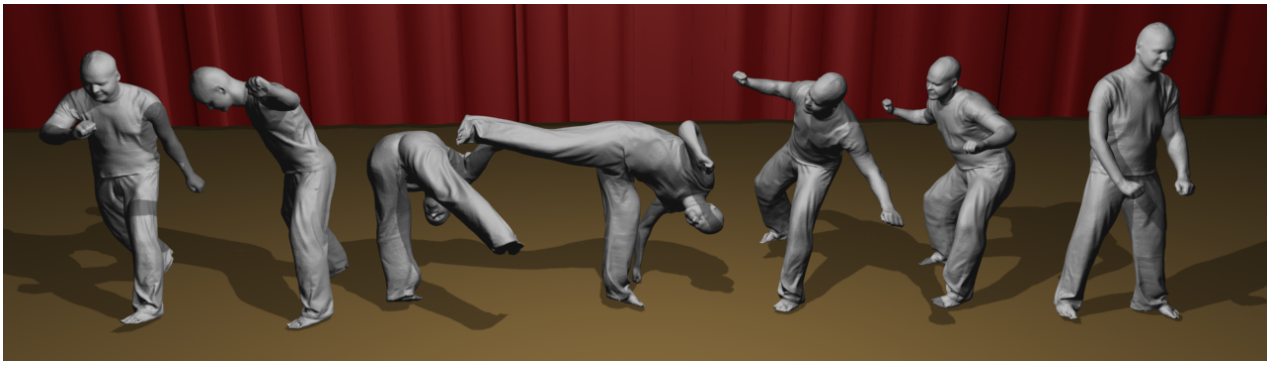


Figure 1: *A sequence of poses captured from eight video recordings of a capoeira turn kick. Our algorithm delivers spatio-temporally coherent geometry of the moving performer that captures both the time-varying surface detail as well as details in his motion very faithfully.*

Abstract

This paper proposes a new marker-less approach to capturing human performances from multi-view video. Our algorithm can jointly reconstruct spatio-temporally coherent geometry, motion and textural surface appearance of actors that perform complex and rapid moves. Furthermore, since our algorithm is purely mesh-based and makes as few as possible prior assumptions about the type of subject being tracked, it can even capture performances of people wearing wide apparel, such as a dancer wearing a skirt. To serve this purpose our method efficiently and effectively combines the power of surface- and volume-based shape deformation techniques with a new mesh-based analysis-through-synthesis framework. This framework extracts motion constraints from video and makes the laser-scan of the tracked subject mimic the recorded performance. Also small-scale time-varying shape detail is recovered by applying model-guided multi-view stereo to refine the model surface. Our method delivers captured performance data at high level of detail, is highly versatile, and is applicable to many complex types of scenes that could not be handled by alternative marker-based or marker-free recording techniques.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism; I.4.8 [Image Processing and Computer Vision]: Scene Analysis

Keywords: performance capture, marker-less scene reconstruction, multi-view video analysis

ACM Reference Format
de Aguiar, E., Stoll, C., Theobalt, C., Ahmed, N., Seidel, H., Thrun, S. 2008. Performance Capture from Sparse Multi–view Video. *ACM Trans. Graph. 27*, 3, Article 98 (August 2008), 10 pages. DOI = 10.1145/1360612.1360697 http://doi.acm.org/10.1145/1360612.1360697.

1 Introduction

The recently released photo-realistic CGI movie Beowulf [Paramount 2007] provides an impressive foretaste of the way how many movies will be produced as well as displayed in the future. In contrast to previous animation movies, the goal was not the creation of a cartoon style appearance but a photo-realistic display of the virtual sets and actors. Today it still takes a tremendous effort to create authentic virtual doubles of real-world actors. It remains one of the biggest challenges to capture human performances, i.e. motion and possibly dynamic geometry of actors in the real world in order to map them onto virtual doubles. To measure body and facial motion, the studios resort to marker-based optical motion capture technology. Although this delivers data of high accuracy, it is still a stopgap. Marker-based motion capture requires a significant setup time, expects subjects to wear unnatural skin-tight clothing with optical beacons, and often makes necessary many hours of manual data cleanup. It therefore does not allow for what both actors and directors would actually prefer: To capture human performances *densely in space and time* - i.e. to be able to jointly capture accurate dynamic shape, motion and textural appearance of actors in arbitrary everyday apparel.

In this paper, we therefore propose a new marker-less dense performance capture technique. From only eight multi-view video recordings of a performer moving in his normal and even loose or wavy clothing, our algorithm is able to reconstruct his motion *and* his spatio-temporally coherent time-varying geometry (i.e. geometry with constant connectivity) that captures even subtle deformation detail. The abdication of any form of optical marking also makes simultaneous shape and texture acquisition straightforward.

Our method achieves a high level of flexibility and versatility by explicitly abandoning any traditional skeletal shape or motion parametrization and by posing *performance capture* as *deformation capture*. For scene representation we employ a detailed static laser scan of the subject to be recorded. Performances are captured in a multi-resolution way, i.e. first global model pose is inferred using a lower-detail model, Sect. 5, and thereafter smaller-scale shape and motion detail is estimated based on a high-quality model, Sect. 6.

Global pose capture employs a new analysis-through-synthesis procedure that robustly extracts from the input footage a set of position constraints. These are fed into an efficient physically plausible shape deformation approach, Sect. 4, in order to make the scan mimic the motion of its real-world equivalent. After global pose recovery in each frame, a model-guided multi-view stereo and contour alignment method reconstructs finer surface detail at each time step. Our results show that our approach can reliably reconstruct very complex motion exhibiting speed and dynamics that would even challenge the limits of traditional skeleton-based optical capturing approaches, Sect. 7.

To summarize, this paper presents a new video-based performance capture method

- that passively reconstructs spatio-temporally coherent shape, motion and texture of actors at high quality;
- that draws its strength from an effective combination of new skeleton-less shape deformation methods, a new analysis-through-synthesis framework for pose recovery, and a new model-guided multi-view stereo approach for shape refinement;
- and that exceeds capabilities of many previous capture techniques by allowing the user to record people wearing loose apparel and people performing fast and complex motion.

2 Related Work

Previous related work has largely focused on capturing subelements of the sophisticated scene representation that we are able to reconstruct.

Marker-based optical motion capture systems are the workhorses in many game and movie production companies for measuring motion of real performers [Menache and Manache 1999]. Despite their high accuracy, their very restrictive capturing conditions, that often require the subjects to wear skin-tight body suits and reflective markings, make it infeasible to capture shape and texture. Park et al. [2006] try to overcome this limitation by using several hundred markers to extract a model of human skin deformation. While their animation results are very convincing, manual mark-up and data cleanup times can be tremendous in such a setting and generalization to normally dressed subjects is difficult. In contrast, our marker-free algorithm requires a lot less setup time and enables *simultaneous* capture of shape, motion and texture of people wearing everyday apparel.

Marker-less motion capture approaches are designed to overcome some restrictions of marker-based techniques and enable performance recording without optical scene modification [Moeslund et al. 2006; Poppe 2007]. Although they are more flexible than intrusive methods, it remains difficult for them to achieve the same level of accuracy and the same application range. Furthermore, since most approaches employ kinematic body models, it is hard for them to capture motion, let alone detailed shape, of people in loose everyday apparel. Some methods, such as [Sand et al. 2003] and [Balan et al. 2007] try to capture more detailed body deformations in addition to skeletal joint parameters by adapting the models closer to the observed silhouettes, or by using captured range scan data [Allen et al. 2002]. But both algorithms require the subjects to wear tight clothes. Only few approaches, such as the work by [Rosenhahn et al. 2006], aim at capturing humans wearing more general attire, e.g. by jointly relying on kinematic body and cloth models. Unfortunately, these methods typically require handcrafting of shape and dynamics for each individual piece of apparel, and they focus on joint parameter estimation under occlusion rather than accurate geometry capture.

Other related work explicitly reconstructs highly-accurate geometry of moving cloth from video [Scholz et al. 2005; White et al. 2007]. However, these methods require visual interference with the scene in the form of specially tailored color patterns on each piece of garment which renders simultaneous shape and texture acquisition infeasible.

A slightly more focused but related concept of performance capture is put forward by *3D video* methods which aim at rendering the appearance of reconstructed real-world scenes from new synthetic camera views never seen by any real camera. Early shape-from-silhouette methods reconstruct rather coarse approximate 3D video geometry by intersecting multi-view silhouette cones [Matusik et al. 2000; Gross et al. 2003]. Despite their computational efficiency, the moderate quality of the textured coarse scene reconstructions often falls short of production standards in the movie and game industry. To boost 3D video quality, researchers experimented with image-based methods [Vedula et al. 2005], multi-view stereo [Zitnick et al. 2004], multi-view stereo with active illumination [Waschbüsch et al. 2005], or model-based free-viewpoint video capture [Carranza et al. 2003]. In contrast to our approach, the first three methods do not deliver spatio-temporally coherent geometry or 360 degree shape models, which are both essential prerequisites for animation post-processing. At the same time, previous kinematic model-based 3D video methods were unable to capture performers in general clothing. [Starck and Hilton 2007] propose a combination of stereo and shape-from-silhouette to reconstruct performances from video. They also propose a spherical reparameterization to establish spatio-temporal coherence during postprocessing. However, since their method is based on shape-from-silhouette models which often change topology due to incorrect reconstruction, establishing spatio-temporal coherence may be error-prone. In contrast, our prior with known connectivity handles such situations more gracefully.

Data-driven 3D video methods synthesize novel perspectives by a pixel-wise blending of densely sampled input viewpoints [Wilburn et al. 2005]. While even renderings under new lighting can be produced at high fidelity [Einarsson et al. 2006], the complex acquisition apparatus requiring hundreds of densely spaced cameras makes practical applications often difficult. Further on, the lack of geometry makes subsequent editing a major challenge.

Recently, new animation design [Botsch and Sorkine 2008], animation editing [Xu et al. 2007], deformation transfer [Sumner and Popović 2004] and animation capture methods [Bickel et al. 2007] have been proposed that are no longer based on skeletal shape and motion parametrizations but rely on surface models and general shape deformation approaches. The explicit abandonment of kinematic parametrizations makes performance capture a much harder problem, but bears the striking advantage that it enables capturing of both rigidly and non-rigidly deforming surfaces with the same underlying technology.

Along this line of thinking, the approaches by [de Aguiar et al. 2007a] and [de Aguiar et al. 2007b] enable mesh-based motion capture from video. At a first look, both methods also employ laser-scanned models and a more basic shape deformation framework. But our algorithm greatly exceeds their methods' capabilities in many ways. First, our new analysis-through-synthesis tracking framework enables capturing of motion that shows a level of complexity and speed which would have been impossible to recover with previous flow-based or flow- and feature-based methods. Secondly, we propose a volumetric deformation technique that greatly increases robustness of pose recovery. Finally, in contrast to previ-

ous methods, our algorithm explicitly recovers small-scale dynamic surface detail by applying model-guided multi-view stereo.

Related to our approach are also recent animation reconstruction methods that jointly perform model generation and deformation capture from scanner data [Wand et al. 2007]. However, their problem setting is different and computationally very challenging which makes it hard for them to generate the visual quality that we achieve by employing a prior model. The approaches proposed in [Stoll et al. 2006] and [Shinya 2004] are able to deform mesh-models into active scanner data or visual hulls, respectively. Unfortunately, neither of these methods has shown to match our method's robustness, or the quality and detail of shape and motion data which our approach produces from video only.

3 Video-based Performance Capture

Prior to video-recording human performances we take a full-body laser scan of the subject in its current apparel by means of a Vitus Smart^{TM} laser scanner. After scanning, the subject immediately moves to the adjacent multi-view recording area. Our multi-view capturing apparatus features $K = 8$ synchronized geometrically and photometrically calibrated video cameras running at 24 fps and providing 1004x1004 pixels frame resolution. The cameras are placed in an approximately circular arrangement around the center of the scene (see video for visualization of input). As part of pre-processing color-based background subtraction is applied to all video footage to yield silhouette images of the captured performers.

Once all of the data has been captured, our automatic performance reconstruction pipeline commences which only requires a minimum of manual interaction during pre-processing. To obtain our computational model of shape and motion, we first transform the raw scan into a high-quality surface mesh $\mathcal{T}_{tri} = (\mathbf{V}_{tri}, \mathbf{T}_{tri})$ with n_s vertices $\mathbf{V}_{tri} = \{\mathbf{v}_1 \dots \mathbf{v}_{n_s}\}$ and m_s triangles $\mathbf{T}_{tri} = \{\mathbf{t}_1 \dots \mathbf{t}_{m_s}\}$ by employing the method of [Kazhdan et al. 2006] (see Fig. 2(l)). Additionally, we create a coarser tetrahedral version of the surface scan $\mathcal{T}_{tet} = (\mathbf{V}_{tet}, \mathbf{T}_{tet})$ (comprising of n_t vertices $\mathbf{V}_{tet}$ and m_t tetrahedrons $\mathbf{T}_{tet}$) by applying a quadric error decimation and a subsequent constrained Delaunay tetrahedralization (see Fig. 2(r)). Typically, $\mathbf{T}_{tri}$ contains between 30000 and 40000 triangles, and the corresponding tet-version between 5000 and 6000 tetrahedrons. Both models are automatically registered to the first pose of the actor in the input footage by means of a procedure based on iterative closest points (ICP). Since we asked the actor to strike in the first frame of video a pose similar to the one that she/he was scanned in, pose initialization is greatly simplified, as the model is already close to the target pose.

Our capture method explicitly abandons a skeletal motion parametrization and resorts to a deformable model as scene representation. Thereby, we are facing a much harder tracking problem, but gain an intriguing advantage: we are now able to track non-rigidly deforming surfaces (like wide clothing) in the same way as rigidly deforming models and do not require prior assumptions about material distributions or the segmentation of a model.

The first core algorithmic ingredient of mesh-based performance capture is a fast and reliable shape deformation framework that expresses the deformation of the whole model based on a few point handles, Sect. 4. We capture performances in a multi-resolution way to increase reliability. First, an analysis-through-synthesis method based on image and silhouette cues estimates the global pose of an actor at each frame on the basis of the lower-detail tetrahedral input model, Sect. 5. The sequence of processing steps is designed to enable reliable convergence to plausible poses despite the highly multi-modal solution space of optimization-based mesh deformation. Once global poses are found, the high-frequency aspect of performances is captured. For instance, the motion of folds in a skirt is recovered in this step. To this end the global poses are transferred to the high-detail surface scan, and surface shape is refined by enforcing contour alignment and performing model-guided stereo, Sect. 6.

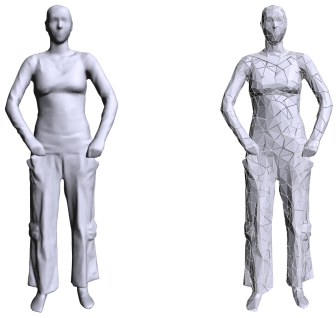


Figure 2: *A surface scan $\mathbf{T}_{tri}$ of an actress (l) and the corresponding tetrahedral mesh $\mathbf{T}_{tet}$ in an exploded view (r).*

The output of our method is a dense representation of the performance in both space and time. It comprises of accurately deformed spatio-temporally coherent geometry that nicely captures the liveliness, motion and shape detail of the original input.

4 A Deformation Toolbox

Our performance capture technique uses two variants of Laplacian shape editing. For low-frequency tracking, we use an iterative volumetric Laplacian deformation algorithm which is based on our tetrahedral mesh $\mathcal{T}_{tet}$, Sect. 4.1. This method enables us to infer rotations from positional constraints and also implicitly encodes prior knowledge about shape properties that we want to preserve, such as local cross-sectional areas. For recovery of high-frequency surface details, we transfer the captured pose of $\mathcal{T}_{tet}$ to the high-resolution surface scan, Sect. 4.2. Being already roughly in the correct pose, we can resort to a simpler non-iterative variant of surface-based Laplacian deformation to infer shape detail from silhouette and stereo constraints, Sect. 4.3.

4.1 Volumetric Deformation

It is our goal to deform the tetrahedral mesh $\mathcal{T}_{tet}$ as naturally as possible under the influence of a set of position constraints $\mathbf{v}_j \approx \mathbf{q}_j$, $j \in \{1, \dots, n_c\}$. To this end, we iterate a linear Laplacian deformation step and a subsequent update step, which compensates the (mainly rotational) errors introduced by the nature of the linear deformation. This procedure minimizes the amount of non-rigid deformation each tetrahedron undergoes, and thus exhibits qualities of an elastic deformation. Our algorithm is related to the approach by [Sorkine and Alexa 2007]. However, we decide to use a tetrahedral construction rather than their triangle mesh construction, as this allows us to implicitly preserve certain shape properties, such as cross-sectional areas, after deformation. The latter greatly increases tracking robustness since non-plausible model poses (e.g. due to local flattening) are far less likely.

Our deformation technique is based on solving the tetrahedral Laplacian system $\mathbf{L}\mathbf{v} = \delta$ with

$$\mathbf{L} = \mathbf{G}^T\mathbf{D}\mathbf{G}\,, \tag{1}$$

and

$$\delta = \mathbf{G}^T\mathbf{D}\mathbf{g}\,, \tag{2}$$

where $\mathbf{G}$ is the discrete gradient operator matrix for the mesh, $\mathbf{D}$ is a $4m_t \times 4m_t$ diagonal matrix containing the tetrahedra's volumes, and $\mathbf{g}$ is the set of tetrahedron gradients, each being calculated as $\mathbf{g}_j = \mathbf{G}_j\mathbf{p}_j$ (see [Botsch and Sorkine 2008] for more detail). Here, $\mathbf{p}_j$ is a matrix containing the vertex coordinates of tetrahedron $\mathbf{t}_j$. The constraints $\mathbf{q}_j$ can be factorized into the matrix $\mathbf{L}$ by eliminating the corresponding rows and columns in the matrix and incorporating the values into the right-hand side δ.

We now iterate the following steps :

- *Linear Laplacian deformation:* By solving the above system we obtain a set of new vertex positions $\mathbf{V}'_{tet} = \{\mathbf{v}'_1 \dots \mathbf{v}'_{n_t}\}$. Due to the linear formulation, this deformed model exhibits artifacts common to all simple Laplacian techniques, i.e. the local elements do not rotate under constraints but rather simply scale and shear to adjust to the desired pose.
- *Rotation extraction:* We now extract a transformation matrix $\mathbf{T}_i$ for each tetrahedron which brings $\mathbf{t}_i$ into configuration $\mathbf{t'}_i$. These transformations can be further split up into a rigid part $\mathbf{R}_i$ and a non-rigid part $\mathbf{S}_i$ using polar decomposition. Keeping only the rotational component removes the non-rigid influences of the linear deformation step from the local elements.
- *Differential update:* We finally update the right hand side δ using Eq. (2) by applying the rotations $\mathbf{R}_i$ to the gradients of the tetrahedron.

Iterating this procedure minimizes the amount of non-rigid deformation $\mathbf{S}_i$ remaining in each tetrahedron. Henceforth we will refer to this deformation energy as E_D. While our subsequent tracking steps would work with any physically plausible deformation or simulation method such as [Botsch et al. 2007; Müller et al. 2002], our technique has the advantages of being extremely fast, of being very easy to implement, and of producing plausible results even if material properties are unknown.

4.2 Deformation Transfer

To transfer a pose from $\mathcal{T}_{tet}$ to $\mathcal{T}_{tri}$, we express the position of each vertex $\mathbf{v}_i$ in $\mathcal{T}_{tri}$ as a linear combination of vertices in $\mathcal{T}_{tet}$. These coefficients $\mathbf{c}_i$ are calculated for the rest pose and can be used afterwards to update the pose of the triangle mesh.

We generate the linear coefficients $\mathbf{c}_i$ by finding the subset $\mathbf{T}_r(\mathbf{v}_i)$ of all tetrahedra from $\mathcal{T}_{tet}$ that lie within a local spherical neighborhood of radius r (in all our cases r was set to 5% of the mesh's bounding box diagonal) and contain a boundary face with a face normal similar to that of $\mathbf{v}_i$. Subsequently, we calculate the (not necessarily positive) barycentric coordinate coefficients $\mathbf{c}_i(j)$ of the vertex with respect to all $\mathbf{t}_j \in \mathbf{T}_r(\mathbf{v}_i)$ and combine them into one larger coefficient vector $\mathbf{c}_i$ as

$$\mathbf{c}_i = \frac{\sum_{\mathbf{t}_j \in \mathbf{T}_r(\mathbf{v}_i)} \mathbf{c}_i(j)\phi(\mathbf{v}_i, \mathbf{t}_j)}{\sum_{\mathbf{t}_j \in \mathbf{T}_r(\mathbf{v}_i)} \phi(\mathbf{v}_i, \mathbf{t}_j)} .$$

$\phi(\mathbf{v}_i, \mathbf{t}_j)$ is a compactly supported radial basis function with respect to the distance of $\mathbf{v}_i$ to the barycenter of tetrahedron $\mathbf{t}_j$. This weighted averaging ensures that each point is represented by several tetrahedra and thus the deformation transfer from tetrahedral mesh to triangle mesh will be smooth. The coefficients for all vertices of $\mathcal{T}_{tri}$ are combined into a matrix $\mathbf{B}$. Thanks to the smooth partition of unity definition and the local support of our parametrization, we can quickly compute the mesh in its transferred pose $\mathcal{V}'_{tri}$ by multiplying the current vertex positions of the current tetrahedral mesh $\mathcal{V}_{tet}$ with $\mathbf{B}$.

4.3 Surface-based Deformation

Our surface-based deformation relies on a simple least-squares Laplacian system as it has been widely used in recent years [Botsch and Sorkine 2008]. Given our triangle mesh $\mathcal{T}_{tri}$ we apply a discrete least-squares Laplacian using cotangent weights to deform the surface under the influence of a set of position constraints $\mathbf{v}_j \approx \mathbf{q}_j, j \in \{1, \dots, n_c\}$. This can be achieved by minimizing the energy

$$\underset{\mathbf{v}}{\operatorname{argmin}} \left\{ \|\mathbf{L}\mathbf{v} - \delta\|^2 + \|\mathbf{C}\mathbf{v} - \mathbf{q}\|^2 \right\}. \quad (3)$$

Here, $\mathbf{L}$ is the cotangent Laplacian matrix, δ are the differential coordinates, and $\mathbf{C}$ is a diagonal matrix with non-zero entries $\mathbf{C}_{j,j} = w_j$ only for constrained vertices $\mathbf{v}_j$ (where w_j is the weight of the additional entry). This formulation uses the Laplacian as a regularization term for the deformation defined by our constraints.

5 Capturing the Global Model Pose

Our first step aims at recovering for each time step of video a global pose of the tetrahedral input model that matches the pose of the real actor. In a nutshell, our global pose extraction method computes deformation constraints from each pair of subsequent multi-view input video frames at times t and $t+1$. It then applies the volumetric shape deformation procedure to modify the pose of $\mathbf{T}_{tet}$ at time t (that was found previously) until it aligns with the input data at time $t+1$. In order to converge to a plausible pose under this highly multi-modal goodness-of-fit criterion, it is essential that we extract the right types of features from the images in the right sequence and apply the resulting deformation constraints in the correct order.

To serve this purpose, our pose recovery process begins with the extraction of 3D vertex displacements from reliable image features which brings our model close to its final pose even if scene motion is rapid, Sect. 5.1. The distribution of 3D features on the model surface is dependent on scene structure, e.g. texture, and can, in general, be non-uniform or sparse. Therefore, the resulting pose may not be entirely correct. Furthermore, potential outliers in the correspondences make additional pose update steps unavoidable. We therefore subsequently resort to two additional steps that exploit silhouette data to fully recover the global pose. The first step refines the shape of the outer model contours until they match the multi-view input silhouette boundaries, Sect. 5.2. The second step optimizes 3D displacements of key vertex handles until optimal multi-view silhouette overlap is reached, Sect. 5.3. Conveniently, the multi-view silhouette overlap can be quickly computed as an XOR operation on the GPU.

We gain further tracking robustness by subdividing the surface of the volume model into a set R of approximately 100-200 regions of similar size during pre-processing [Yamauchi et al. 2005]. Rather than inferring displacements for each vertex, we determine representative displacements for each region as explained in the following sections.

5.1 Pose Initialization from Image Features

Given two sets of multi-view video frames $I_1(t), \dots, I_k(t)$ and $I_1(t+1), \dots, I_k(t+1)$ from subsequent time steps, our first processing step extracts SIFT features in each frame [Lowe 1999] (see Fig. 3). This yields for each camera view k and either time step a list of $\ell(k) = 1, \dots, L_k$ 2D feature locations $u_{k,t}^{\ell(k)}$ along with their SIFT feature descriptors $dd_{k,t}^{\ell(k)}$ – henceforth we refer to each such list as $LD_{k,t}$. SIFT features are our descriptors of choice, as

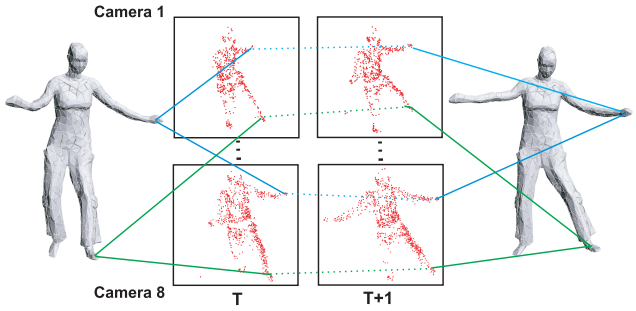


Figure 3: *3D correspondences are extracted from corresponding SIFT features in respective input camera views at t and t+1. These 3D correspondences, two of them illustrated by lines, are used to deform the model into a first pose estimate for $t+1$.*

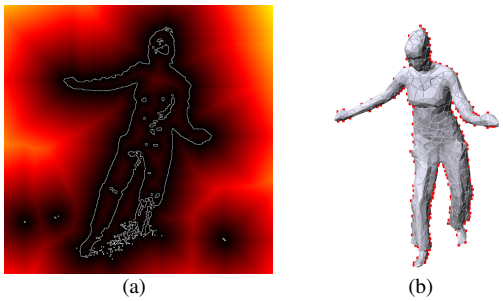


Figure 4: *(a) Color-coded distance field from the image silhouette contour shown for one camera view. (b) Rim vertices with respect to one camera view marked in red on the 3D model.*

they are largely invariant under illumination and out-of-plane rotation and enable reliable correspondence finding even if the scene motion is fast.

Let $\mathcal{T}_{tet}(t)$ be the pose of $\mathcal{T}_{tet}$ at time t. To transform feature data into deformation constraints for vertices of $\mathcal{T}_{tet}(t)$, we first need to pair image features from time t with vertices in the model. We therefore first associate each $\mathbf{v}_i$ of $\mathcal{T}_{tet}(t)$ with that descriptor $dd^i_{k,t}$ from each $I_k(t)$ that is located closest to the projected location of $\mathbf{v}_i$ in this respective camera. We perform this computation for all camera views and discard a feature association if $\mathbf{v}_i$ is not visible from k or if the distance between the projected position of $\mathbf{v}_i$ and the image position of $dd^i_{k,t}$ is too large. This way, we obtain a set of associations $A(\mathbf{v}_i,t) = \{dd^{j1}_{1,t}, \cdots, dd^{jK}_{K,t}\}$ for a subset of vertices that contains at most one feature from each camera. Lastly, we check the consistency of each $A(\mathbf{v}_i,t)$ by comparing the pseudo-intersection point $\mathbf{p}_i^{INT}$ of the reprojected rays passing through $u^{j1}_{1,t}, \ldots, u^{jK}_{K,t}$ to the 3D position of $\mathbf{v}_i$ in model pose $\mathcal{T}_{tet}(t)$. If the distance $\|\mathbf{v}_i - \mathbf{p}_i^{INT}\|$ is greater than a threshold $\mathcal{E}_{DIST}$ the original feature association is considered implausible and $\mathbf{v}_i$ is removed from the candidate list for deformation handles.

The next step is to establish temporal correspondence, i.e. to find for each vertex $\mathbf{v}_i$ with feature association $A(\mathbf{v}_i,t)$ the corresponding association $A(\mathbf{v}_i,t+1)$ with features from the next time step. To this end, we preliminarily find for each $dd^j_{k,t} \in A(\mathbf{v}_i,t)$ a descriptor $dd^f_{k,t+1} \in LD_{k,t+1}$ by means of nearest neighbor distance matching in the descriptor values, and add $dd^f_{k,t+1}$ to $A(\mathbf{v}_i,t+1)$. In practice, this initial assignment is likely to contain outliers, and therefore we compute the final set of temporal correspondences by means of robust spectral matching [Leordeanu and Hebert 2005]. This method efficiently bypasses the combinatorial complexity of the correspondence problem by formulating it in closed form as a spectral analysis problem on a graph adjacency matrix. Incorrect matches are eliminated by searching for an assignment in which both the feature descriptor values across time are consistent, and pairwise feature distances across time are preserved. Fig. 3 illustrates a subset of associations found for two camera views. From the final set of associations $A(\mathbf{v}_i,t+1)$ we compute the predicted 3D target position $\mathbf{p}_i^{EST}$ of vertex $\mathbf{v}_i$ again as the virtual intersection point of reprojected image rays through the 2D feature positions.

Each vertex $\mathbf{v}_i$ for which a new estimated position was found is a candidate for a deformation handle. However, we do not straightforwardly apply all handles to move directly to the new target pose. We rather propose the following step-wise procedure which, in practice, is less likely to converge to implausible model configurations: We resort to the set of regions R on the surface of the tet-mesh (as described above) and find for each region $r_i \in R$ one best handle from all candidate handles that lie in r_i. The best handle vertex $\mathbf{v}_i$ is the one whose local normal is most collinear with the difference vector $\mathbf{p}_i^{EST} - \mathbf{v}_i$. If no handle is found for a region, we constrain the center of that region to its original 3D position in $\mathcal{T}_{tet}(t)$. This prevents unconstrained surface areas from arbitrary drifting. For each region handle, we define a new intermediate target position as $\mathbf{q}'_i = \mathbf{v}_i + \frac{\mathbf{p}_i^{EST} - \mathbf{v}_i}{\|\mathbf{p}_i^{EST} - \mathbf{v}_i\|}$. Typically, we obtain position constraints $\mathbf{q}'_i$ for around 70% to 90% of the surface regions R that are then used to change the pose of the model. This step-wise deformation is repeated until the multi-view silhouette overlap error $SIL(\mathcal{T}_{tet}, t+1)$ cannot be improved further. The overlap error is computed as the XOR between input and model silhouette in all camera views.

We would like to remark that we do not require tracking of features across the entire sequence which greatly contributes to the reliability of our method. The output of this step is a feature-based pose estimate $\mathcal{T}^F_{tet}(t+1)$.

5.2 Refining the Pose using Silhouette Rims

In image regions with sparse or low-frequency textures, only few SIFT features may have been found. In consequence, the pose of $\mathcal{T}^F_{tet}(t+1)$ may not be correct in all parts. We therefore resort to another constraint that is independent of image texture and has the potential to correct for such misalignments. To this end, we derive additional deformation constraints for a subset of vertices on $\mathcal{T}^F_{tet}(t+1)$ that we call *rim vertices* $\mathbf{V}_{RIM}(t+1)$, see Fig. 4(b). In order to find the elements of $\mathbf{V}_{RIM}(t+1)$, we first calculate contour images $C_{k,t+1}$ using the rendered volumetric model silhouettes. A vertex $\mathbf{v}_i$ is considered a rim vertex if it projects into close vicinity of the silhouette contour in (at least) one of the $C_{k,t+1}$, and if the normal of $\mathbf{v}_i$ is perpendicular to the viewing direction of the camera k.

For each element $\mathbf{v}_i \in \mathbf{V}_{RIM}(t+1)$ a 3D displacement is computed by analyzing the projected location $u_{k,t+1}$ of the vertex into the camera k that originally defined its rim status. The value of the distance field from the contour at the projected location defines the total displacement length in vertex normal direction, Fig. 4(a). This way, we obtain deformation constraints for rim vertices which we apply in the same step-wise deformation procedure that was already used in Sect. 5.1. The result is a new model configuration $\mathcal{T}^R_{tet}(t+1)$ in which the projections of the outer model contours more closely match the input silhouette boundaries.

5.3 Optimizing Key Handle Positions

In the majority of cases, the pose of the model in $\mathcal{T}_{tet}^{R}(t+1)$ is already close to a good match. However, in particular if the scene motion was fast or the initial pose estimate from SIFT was not entirely correct, residual pose errors remain. We therefore perform an additional optimization step that corrects such residual errors by globally optimizing the positions of a subset of deformation handles until good silhouette overlap is reached.

Instead of optimizing the position of all $1000-2000$ vertices of the volumetric model, we only optimize the position of typically 15-25 key vertices $\mathbf{V}_k \subset \mathbf{V}_{tet}$ until the tetrahedral deformation produces optimal silhouette overlap. Tracking robustness is increased by designing our energy function such that surface distances between key handles are preserved, and pose configurations with low distortion energy E_D are preferred. We ask the user to specify key vertices manually, a procedure that has to be done only once for every model. Typically, key vertices are marked close to anatomical joints, and in case of model parts representing loose clothing, a simple uniform handle distribution produces good results.

Given all key vertex positions $\mathbf{v}_i \in \mathbf{V}_k$ in the current model pose $\mathcal{T}_{tet}^{R}(t+1)$, we optimize for their new positions $\mathbf{p}_i$ by minimizing the following energy functional:

$$E(\mathbf{V}_k) = w_S \cdot SIL(\mathbf{T}_{tet}(\mathbf{V}_k), t+1) + w_D \cdot E_D + w_C \cdot E_C \,. \quad (4)$$

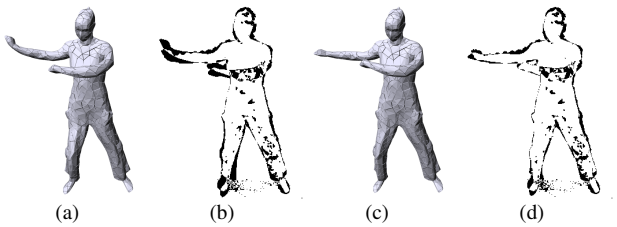
Here, $SIL(\mathbf{T}_{tet}(\mathbf{V}_k), t+1)$ denotes the multi-view silhouette overlap error of the tet-mesh in its current deformed pose $\mathbf{T}_{tet}(\mathbf{V}_k)$ which is defined by the new positions of the $\mathbf{V}_k$. E_D is the deformation energy as defined in Sect. 4.1. Implicitly we reason that low energy configurations are more plausible, see Sect. 4.1. E_C penalizes changes in distance between neighboring key vertices. All three terms are normalized and the weights w_S, w_D, and w_C are chosen in a way such that $SIL(\mathbf{T}_{tet}(\mathbf{V}_k), t+1)$ is the dominant term. We use a Quasi-Newton LBFGS-B method to minimize Eq. (4) [Byrd et al. 1995].

Fig. 5 illustrates the improvements in the new output pose $\mathcal{T}_{tet}^{O}(t+1)$ that are achieved through key handle optimization.

5.4 Practical Considerations

The above sequence of steps is performed for each pair of subsequent time instants. Surface detail capture, Sect. 6, commences after the global poses for all frames were found.

Typically the rim step described in Sect. 5.2 is performed once more after the last silhouette optimization steps which, in some cases, leads to a better model alignment. We also perform a consistency check on the output of low frequency pose capture to correct potential self-intersections. To this end, for every vertex lying inside another tetrahedron, we use the volumetric deformation method to displace this vertex in outward direction along its normal until the intersection is resolved.



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

Figure 5: *Model (a) and silhouette overlap (b) after the rim step; slight pose inaccuracies in the leg and the arms appear black in the silhouette overlap image. (c),(d) After key vertex optimization, these pose inaccuracies are removed and the model strikes a correct pose.*

6 Capturing Surface Detail

Once global pose has been recovered for each frame the pose sequence of $\mathcal{T}_{tet}$ is mapped to $\mathcal{T}_{tri}$, Sect. 4.2. In the following, the process of shape detail capture at a single time step is explained.

6.1 Adaptation along Silhouette Contours

In a first step we adapt the silhouette rims of our fine mesh to better match the input silhouette contours. As we are now working on a surface mesh which is already very close to the correct configuration, we can allow a much broader and less smooth range of deformations than in the volumetric case, and thereby bring the model in much closer alignment with the input data. At the same time we have to be more careful in selecting our constraints, since noise in the data now has more deteriorating influence.

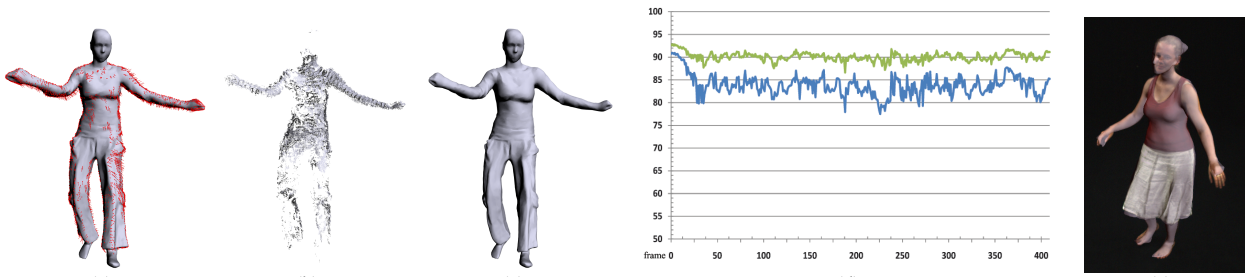
Similar to Sect. 5.2 we calculate rim vertices, however on the high-resolution surface mesh, Fig. 6(a). For each rim vertex the closest 2D point on the silhouette boundary is found in the camera view that defines its rim status. Now we check if the image gradient at the input silhouette point has a similar orientation to the image gradient in the reprojected model contour image. If this is the case, the back-projected input contour point defines the target position for the rim vertex. If the distance between back-projection and original position is smaller than threshold $\mathcal{E}_{RIM}$ we add it as constraint to Eq. (3). Here we use a low weight (between 0.25 and 0.5 depending on the quality of the segmentation) for the rim constraint points. This has a regularizing and damping effect on the deformation that minimizes implausible shape adaptation in the presence of noise. After processing all vertices, we solve for the new surface. This rim projection and deformation step is iterated up to 20 times or until silhouette overlap can not be improved further.

6.2 Model-guided Multi-view Stereo

Although the silhouette rims only provide reliable constraints on outer boundaries, they are usually evenly distributed on the surface. Hence, the deformation method in general nicely adapts the shape of the whole model also in areas which don't project on image contours. Unless the surface of the actor has a complicated shape with many concavities, the result of rim adaptation is already a realistic representation of the correct shape.

However, in order to recover shape detail of model regions that do not project to silhouette boundaries, such as folds and concavities in a skirt, we resort to photo-consistency information. To serve this purpose, we derive additional deformation constraints by applying the multi-view stereo method proposed by [Goesele et al. 2006]. Since our model is already close to the correct surface, we can initialize the stereo optimization from the current surface estimate and constrain the correlation search to 3D points that are at most ± 2 cm away from $\mathcal{T}_{tri}$.

As we have far less viewpoints of our subject than Goesele et al. and our actors can wear apparel with little texture, the resulting depth maps (one for each input view) are often sparse and noisy. Nonetheless, they provide important additional cues about the object's shape. We merge the depth maps produced by stereo into a single point cloud $\mathcal{P}$, Fig. 6(b), and thereafter project points from $\mathbf{V}_{tri}$ onto $\mathcal{P}$ using a method similar to [Stoll et al. 2006]. These projected points provide additional position constraints that we can



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

Figure 6: Capturing small-scale surface detail*: (a) First, deformation constraints from silhouette contours, shown as red arrows, are estimated. (b) Additional deformation handles are extracted from a 3D point cloud that was computed via model-guided multi-view stereo. (c) Together, both sets of constraints deform the surface scan to a highly accurate pose.* – **Evaluation***: (d) per-frame silhouette overlap in per cent after global pose estimation (blue) and after surface detail reconstruction (green). (e) Blended overlay between an input image and the reconstructed model showing the almost perfect alignment of our result.*

use in conjunction with the rim vertices in the surface-based deformation framework, Eq. (3). Given the uncertainty in the data, we solve the Laplace system with lower weights for the stereo constraints.

7 Results and Applications

Our test data were recorded in our acquisition setup described in Sect. 3 and comprise of 12 sequences that show four different actors and that feature between 200 and 600 frames each. To show the large application range of our algorithm, the captured performers wore a wide range of different apparel, ranging from tight to loose, and made of fabrics with prominent texture as well as plain colors only. Also, the recovered set of motions ranges from simple walks, over different dance styles, to fast capoeira sequences. As the images in Figs. 1,7 and 8, as well as the results in the accompanying video demonstrate, our algorithm faithfully reconstructs this wide spectrum of scenes. We would also like to note that, although we focused on human performers, our algorithm would work equally well for animals provided that a laser scan can be acquired.

Fig. 1 shows several captured poses of a very rapid capoeira sequence in which the actor performs a series of turn kicks. Despite the fact that in our 24 fps recordings the actor rotates by more than 25 degrees in-between some subsequent frames, both shape and motion are reconstructed at high fidelity. The resulting animation even shows deformation details such as the waving of the trouser legs (see video). Furthermore, even with the plain white clothing that the actor wears in the input and which exhibits only few traceable SIFT features, our method performs reliably as it can capitalize on rims and silhouettes as additional sources of information. Comparing a single moment from the kick to an input frame confirms the high quality of our reconstruction, Fig. 7(b) (Note that input and virtual camera views differ slightly).

The video also shows the captured capoeira sequence with a static checkerboard texture. This result demonstrates that temporal aliasing, such as tangential surface drift of vertex positions, is almost not noticeable, and that the overall quality of the meshes remains highly stable.

In Fig. 7(a) we show one pose from a captured jazz dance performance. As the comparison to the input in image and video shows, we are able to capture this fast and fluent motion. In addition, we can also reconstruct the many poses with complicated self-occlusions, such as the inter-twisted arm-motion in front of the torso, like in Fig. 7(a).

Fig. 8 shows one of the main strengths of our method, namely its ability to capture the full time-varying shape of a dancing girl wearing a skirt. Even though the skirt is of largely uniform color, our results capture the natural waving and lifelike dynamics of the fabric (see also the video). In all frames, the overall body posture, and also the folds of the skirt were recovered nicely without the user specifying a segmentation of the model beforehand. We would also like to note that in these skirt sequences (one more in the video) the benefits of the stereo step in recovering concavities are most apparent. In the other test scenes, the effects are less pronounced and we therefore deactivated the stereo step (Sect. 6.2) there to reduce computation time. The jitter in the hands that is slightly visible in some of the skirt sequences is due to the fact that the person moves with an opened hand but the scan was taken with hands forming a fist. In general, we also smooth the final sequence of vertex positions to remove any remaining temporal noise.

Apart from the scenes shown in the result images, the video contains three more capoeira sequences, two more dance sequences, two more walking sequences and one additional skirt sequence.

7.1 Validation and Discussion

Table 1 gives detailed average timings for each individual step in our algorithm. These timings were obtained with highly unoptimized single-threaded code running on an Intel Core Duo T2500 Laptop with 2.0 GHz. We see plenty of room for implementation improvement, and anticipate that parallelization can lead to a significant run time reduction.

So far, we have visually shown the high capture quality, as well as the large application range and versatility of our approach. To formally validate the accuracy of our method, we have compared the silhouette overlap of our tracked output models with the segmented input frames. We use this criterion since, to our knowledge, there is no gold-standard alternative capturing approach that would provide us with accurate time-varying 3D data. The re-projections of our final results typically overlap with over 85% of the input silhouette pixels, already after global pose capture only (blue curve in Fig. 6(d)). Surface detail capture further improves this overlap to more than 90% as shown by the green curve. Please note that this measure is slightly negatively biased by errors in foreground segmentation in some frames that appear as erroneous silhouette pixels. Visual inspection of the silhouette overlap therefore confirms the almost perfect alignment of model and actual person silhouette. Fig. 6(e) shows a blended overlay between the rendered model and an input frame which proves this point.

Our algorithm robustly handles even noisy input, e.g. due to typically observed segmentation errors in our color-based segmenta-

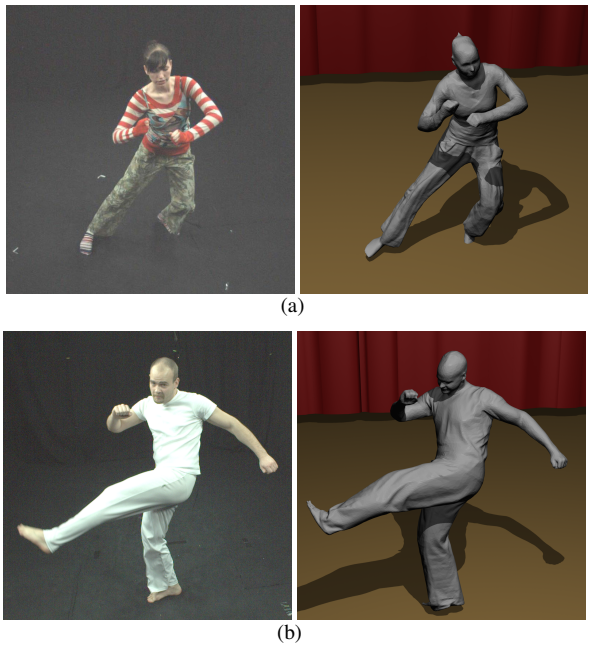


Figure 7: *(a) Jazz dance posture with reliably captured intertwisted arm motion. (b) One moment from a very fast capoeira turn kick (Input and virtual viewpoints differ minimally).*

tion (see video). All 12 input sequences were reconstructed fully-automatically after only minimal initial user input. As part of preprocessing, the user marks the head and foot regions of each model to exclude them from surface detail capture. Even slightest silhouette errors in these regions (in particular due to shadows on the floor and black hair color) would otherwise cause unnatural deformations. Furthermore, for each model the user once marks at most 25 deformation handles needed for the key handle optimization step, Sect. 5.3.

In individual frames of two out of three capoeira turn kick sequences (11 out of around 1000 frames), as well as in one frame of each of the skirt sequences (2 frames from 850 frames), the output of global pose recovery showed slight misalignments in one of the limbs. Please note that, despite these isolated pose errors, the method always recovers immediately and tracks the whole sequence without drifting – this means the algorithm can run without supervision and the results can be checked afterwards. All observed pose misalignments were exclusively due to oversized silhouette areas because of either motion blur or strong shadows on the floor. Both of this could have been prevented by better adjustment of lighting and shutter speed, and more advanced segmentation schemes. In either case of global pose misalignment, at most two deformation handle positions had to be slightly adjusted by the user. At none of the over 3500 input frames we processed in total, it was necessary to manually correct the output of surface detail capture (Sect. 6).

Step	Time
SIFT step (Sect. 5.1)	∼34s
Global rim step (Sect. 5.2)	∼145s
Key handle optimization (Sect. 5.3)	∼270s
Contour-based refinement (Sect. 6.1)	∼27s
Stereo, 340 × 340 depth maps (Sect. 6.2)	∼132s

Table 1: *Average run times per frame for individual steps.*

For comparison, we implemented two related approaches from the literature. The method by [de Aguiar et al. 2007a] uses surface-based deformation and optical flow to track a deformable mesh from multi-view video. As admitted by the authors, optical flow fails for fast motions like our capoeira kicks, which makes tracking with their approach infeasible. In contrast, our volumetric deformation framework, in combination with the multi-cue analysis-through-synthesis approach, captures this footage reliably. The method proposed in [de Aguiar et al. 2007b] solves the slightly different problem of capturing continuous 3D feature trajectories from multi-view video without 3D scene geometry. However, as shown in their paper, the trajectories can be employed to deform a surface scan to move like the actor in video. In our experiments we found that it is hard for their method to maintain uninterrupted trajectories if the person moves sometimes quickly, turns a lot, or strikes poses with complex self-intersections. In contrast, our method handles these situations robustly. Furthermore, as opposed to both of these methods, we perform a stereo-based refinement step that improves contour alignment and that estimates true time-varying surface detail and concavities which greatly contribute to the naturalness of the final result.

Despite our method's large application range, there are a few limitations to be considered. Our current silhouette rim matching may produce erroneous deformations in case the topological structure of the input silhouette is too different from the reprojected model silhouette. However, in none of our test scenes this turned out to be an issue. In future, we plan to investigate more sophisticated image registration approaches to solve this problem entirely. Currently, we are recording in a controlled studio environment to obtain good segmentations, but are confident that a more advanced background segmentation will enable us to handle outdoor scenes.

Moreover, there is a resolution limit to our deformation capture. Some of the high-frequency detail in our final result, such as fine wrinkles in clothing or details of the face, has been part of the laser-scan in the first place. The deformation on this level of detail is not actually captured, but it is "baked in" to the deforming surface. Consequently, in some isolated frames small local differences in the shape details between ground-truth video footage and our deformed mesh may be observed, in particular if the deformed mesh pose deviates very strongly from the scanned pose. To illustrate the level of detail that we are actually able to reconstruct, we generated a result with a coarse scan that lacks fine surface detail. Fig. 9 shows an input frame (l), as well as the reconstructions using the detailed scan (m) and the coarse model (r). While, as noted before, finest detail in Fig. 9(m) is due to the high-resolution laser scan, even with a coarse scan, our method still captures the important lifelike motion and the deformation details, Fig. 9(r). To further support this point, the accompanying video shows a side-by-side comparison between the final result with a coarse template and the final result with the original detailed scan.

Also, in our system the topology of the input scanned model is preserved over the whole sequence. For this reason, we are not able to track surfaces which arbitrarily change apparent topology over time (e.g. the movement of hair or deep folds with self-collisions). Further on, although we prevent self-occlusions during global pose capture, we currently do not correct them in the output of surface detail capture. However, their occurrence is rather seldom. Manual or automatic correction by collision detection would also be feasible.

Our volume-based deformation technique essentially mimics elastic deformation, thus the geometry generated by the low-frequency tracking may in some cases have a rubbery look. For instance, an arm may not only bend at the elbow, but rather bend along its entire length. Surface detail capture eliminates such artifacts in gen-

Figure 8: *Side-by-side comparison of input and reconstruction of a dancing girl wearing a skirt (input and virtual viewpoints differ minimally). Body pose and detailed geometry of the waving skirt, including lifelike folds and wrinkles visible in the input, have been recovered.*

eral, and a more sophisticated yet slower finite element deformation could reduce this problem already at the global pose capture stage.

Despite these limitations we have presented a new non-intrusive approach to spatio-temporally dense performance capture from video. It deliberately abandons traditional motion skeletons to reconstruct a large range of real-world scenes in a spatio-temporally coherent way and at a high level of detail.

7.2 Applications

In the following, we briefly exemplify the strengths and the usability of our algorithm in two practical applications that are important in media production.

3D Video Since our approach works without optical markings, we can use the captured video footage and texture the moving geometry from the input camera views, for instance by using the blending scheme from [Carranza et al. 2003]. The result is a 3D video representation that can be rendered from arbitrary synthetic views (see video and Fig. 10(l),(m)). Due to the highly-detailed underlying scene geometry the visual results are much better than with previous model-based or shape from silhouette-based 3D video methods.

Figure 9: *Input frame (l) and reconstructions using a detailed (m) and a coarse model (r). Although the fine details on the skirt are due to the input laser scan (m), even with a coarse template, our method captures the folds and the overall lifelike motion of the cloth (r).*

Figure 10: *(l),(m) High-quality 3D Video renderings of the dancer wearing a skirt. (r) Fully-rigged character automatically estimated from a capoeira turn kick output.*

Reconstruction of a fully-rigged character Since our method produces spatio-temporally coherent scene geometry with practically no tangential distortion over time, we can reconstruct a fully-rigged character, i.e. a character featuring an animation skeleton, a surface mesh and associated skinning weights, Fig. 10(r), in case this is a suitable parametrization for a scene. To this end we feed our result sequences into the automatic rigging method proposed in [de Aguiar et al. 2008] that fully-automatically learns the skeleton and the blending weights from mesh sequences. Although not the focus of this paper, this experiment shows that the data captured by our system can optionally be converted into a format immediately suitable for modification with traditional animation tools.

8 Conclusion

We have presented a new approach to video-based performance capture that produces a novel dense and feature-rich output format comprising of spatio-temporally coherent high-quality geometry, lifelike motion data, and optionally surface texture of recorded actors. The fusion of efficient volume- and surface-based deformation schemes, a multi-view analysis-through-synthesis procedure, and a multi-view stereo approach enables our method to capture performances of people wearing a wide variety of everyday apparel and performing extremely fast and energetic motion. The proposed method supplements and exceeds the capabilities of marker-based optical capturing systems that are widely used in industry, and will provide animators and CG artists with a new level of flexibility in acquiring and modifying real-world content.

Acknowledgements

Special thanks to our performers Maria Jacob, Yvonne Flory and Samir Hammann, as well as to Derek D. Chan for helping us with the video. This work has been developed within the Max-Planck-Center for Visual Computing and Communication (MPC VCC) collaboration.

References

ALLEN, B., CURLESS, B., AND POPOVIĆ, Z. 2002. Articulated body deformation from range scan data. *ACM Trans. Graph. 21*, 3, 612–619.

BALAN, A. O., SIGAL, L., BLACK, M. J., DAVIS, J. E., AND HAUSSECKER, H. W. 2007. Detailed human shape and pose from images. In *Proc. CVPR*.

BICKEL, B., BOTSCH, M., ANGST, R., MATUSIK, W., OTADUY, M., PFISTER, H., AND GROSS, M. 2007. Multi-scale capture of facial geometry and motion. In *Proc. of SIGGRAPH*, 33.

BOTSCH, M., AND SORKINE, O. 2008. On linear variational surface deformation methods. *IEEE TVCG 14*, 1, 213–230.

BOTSCH, M., PAULY, M., WICKE, M., AND GROSS, M. 2007. Adaptive space deformations based on rigid cells. *Computer Graphics Forum 26*, 3, 339–347.

BYRD, R., LU, P., NOCEDAL, J., AND ZHU, C. 1995. A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. Comp. 16*, 5, 1190–1208.

CARRANZA, J., THEOBALT, C., MAGNOR, M., AND SEIDEL, H.-P. 2003. Free-viewpoint video of human actors. In *Proc. SIGGRAPH*, 569–577.

DE AGUIAR, E., THEOBALT, C., STOLL, C., AND SEIDEL, H.-P. 2007. Marker-less deformable mesh tracking for human shape and motion capture. In *Proc. CVPR*, IEEE, 1–8.

DE AGUIAR, E., THEOBALT, C., STOLL, C., AND SEIDEL, H. 2007. Marker-less 3d feature tracking for mesh-based human motion capture. In *Proc. ICCV HUMO07*, 1–15.

DE AGUIAR, E., THEOBALT, C., THRUN, S., AND SEIDEL, H.-P. 2008. Automatic conversion of mesh animations into skeleton-based animations. *Computer Graphics Forum (Proc. Eurographics EG'08) 27*, 2 (4), 389–397.

EINARSSON, P., CHABERT, C.-F., JONES, A., MA, W.-C., LAMOND, B., IM HAWKINS, BOLAS, M., SYLWAN, S., AND DEBEVEC, P. 2006. Relighting human locomotion with flowed reflectance fields. In *Proc. EGSR*, 183–194.

GOESELE, M., CURLESS, B., AND SEITZ, S. M. 2006. Multi-view stereo revisited. In *Proc. CVPR*, 2402–2409.

GROSS, M., WÜRMLIN, S., NÄF, M., LAMBORAY, E., SPAGNO, C., KUNZ, A., KOLLER-MEIER, E., SVOBODA, T., GOOL, L. V., LANG, S., STREHLKE, K., MOERE, A. V., AND STAADT, O. 2003. blue-c: a spatially immersive display and 3d video portal for telepresence. *ACM TOG 22*, 3, 819–827.

KANADE, T., RANDER, P., AND NARAYANAN, P. J. 1997. Virtualized reality: Constructing virtual worlds from real scenes. *IEEE MultiMedia 4*, 1, 34–47.

KAZHDAN, M., BOLITHO, M., AND HOPPE, H. 2006. Poisson surface reconstruction. In *Proc. SGP*, 61–70.

LEORDEANU, M., AND HEBERT, M. 2005. A spectral technique for correspondence problems using pairwise constraints. In *Proc. ICCV*.

LOWE, D. G. 1999. Object recognition from local scale-invariant features. In *Proc. ICCV*, vol. 2, 1150ff.

MATUSIK, W., BUEHLER, C., RASKAR, R., GORTLER, S., AND MCMILLAN, L. 2000. Image-based visual hulls. In *Proc. SIGGRAPH*, 369–374.

MENACHE, A., AND MANACHE, A. 1999. *Understanding Motion Capture for Computer Animation and Video Games*. Morgan Kaufmann.

MITRA, N. J., FLORY, S., OVSJANIKOV, M., GELFAND, N., AS, L. G., AND POTTMANN, H. 2007. Dynamic geometry registration. In *Proc. SGP*, 173–182.

MOESLUND, T. B., HILTON, A., AND KRÜGER, V. 2006. A survey of advances in vision-based human motion capture and analysis. *Comput. Vis. Image Underst. 104*, 2, 90–126.

MÜLLER, M., DORSEY, J., MCMILLAN, L., JAGNOW, R., AND CUTLER, B. 2002. Stable real-time deformations. In *Proc. of SCA*, ACM, 49–54.

PARAMOUNT, 2007. Beowulf movie page. http://www.beowulfmovie.com/.

PARK, S. I., AND HODGINS, J. K. 2006. Capturing and animating skin deformation in human motion. *ACM TOG (SIGGRAPH 2006) 25*, 3 (Aug.).

POPPE, R. 2007. Vision-based human motion analysis: An overview. *CVIU 108*, 1.

ROSENHAHN, B., KERSTING, U., POWEL, K., AND SEIDEL, H.-P. 2006. Cloth x-ray: Mocap of people wearing textiles. In *LNCS 4174: Proc. DAGM*, 495–504.

SAND, P., MCMILLAN, L., AND POPOVIĆ, J. 2003. Continuous capture of skin deformation. *ACM TOG 22*, 3.

SCHOLZ, V., STICH, T., KECKEISEN, M., WACKER, M., AND MAGNOR, M. 2005. Garment motion capture using color-coded patterns. *Computer Graphics Forum (Proc. Eurographics EG'05) 24*, 3 (Aug.), 439–448.

SHINYA, M. 2004. Unifying measured point sequences of deforming objects. In *Proc. of 3DPVT*, 904–911.

SORKINE, O., AND ALEXA, M. 2007. As-rigid-as-possible surface modeling. In *Proc. SGP*, 109–116.

STARCK, J., AND HILTON, A. 2007. Surface capture for performance based animation. *IEEE CGAA 27(3)*, 21–31.

STOLL, C., KARNI, Z., RÖSSL, C., YAMAUCHI, H., AND SEIDEL, H.-P. 2006. Template deformation for point cloud fitting. In *Proc. SGP*, 27–35.

SUMNER, R. W., AND POPOVIĆ, J. 2004. Deformation transfer for triangle meshes. In *SIGGRAPH '04*, 399–405.

VEDULA, S., BAKER, S., AND KANADE, T. 2005. Image-based spatio-temporal modeling and view interpolation of dynamic events. *ACM Trans. Graph. 24*, 2, 240–261.

WAND, M., JENKE, P., HUANG, Q., BOKELOH, M., GUIBAS, L., AND SCHILLING, A. 2007. Reconstruction of deforming geometry from time-varying point clouds. In *Proc. SGP*, 49–58.

WASCHBÜSCH, M., WÜRMLIN, S., COTTING, D., SADLO, F., AND GROSS, M. 2005. Scalable 3D video of dynamic scenes. In *Proc. Pacific Graphics*, 629–638.

WHITE, R., CRANE, K., AND FORSYTH, D. 2007. Capturing and animating occluded cloth. In *ACM TOG (Proc. SIGGRAPH)*.

WILBURN, B., JOSHI, N., VAISH, V., TALVALA, E., ANTUNEZ, E., BARTH, A., ADAMS, A., HOROWITZ, M., AND LEVOY, M. 2005. High performance imaging using large camera arrays. *ACM TOG 24*, 3, 765–776.

XU, W., ZHOU, K., YU, Y., TAN, Q., PENG, Q., AND GUO, B. 2007. Gradient domain editing of deforming mesh sequences. In *Proc. SIGGRAPH*, ACM, 84ff.

YAMAUCHI, H., GUMHOLD, S., ZAYER, R., AND SEIDEL, H.-P. 2005. Mesh segmentation driven by gaussian curvature. *Visual Computer 21*, 8-10, 649–658.

ZITNICK, C. L., KANG, S. B., UYTTENDAELE, M., WINDER, S., AND SZELISKI, R. 2004. High-quality video view interpolation using a layered representation. *ACM TOG 23*, 3, 600–608.