Machine Vision and Applications (2007) 18:25–40
DOI 10.1007/s00138-006-0046-y

ORIGINAL PAPER

A system for articulated tracking incorporating a clothing model

**Bodo Rosenhahn · Uwe Kersting · Katie Powell ·
Reinhard Klette · Gisela Klette · Hans-Peter Seidel**

Received: 3 April 2006 / Accepted: 21 August 2006 / Published online: 19 October 2006

Abstract In this paper an approach for motion capture of dressed people is presented. A cloth draping method is incorporated in a silhouette based motion capture system. This leads to a simultaneous estimation of pose, joint angles, cloth draping parameters and wind forces. An error functional is formalized to minimize the involved parameters simultaneously. This allows for reconstruction of the underlying kinematic structure, even though it is covered with fabrics. Finally, a quantitative error analysis is performed. Pose results are compared with results obtained from a commercially available marker based tracking system. The deviations have a magnitude of three degrees which indicates a reasonably stable approach.

Keywords Motion capture · Cloth draping · Pose estimation

1 Introduction

Classical motion capture (MoCap) comprises techniques for recording the movements of real objects such as humans or animals [36]. In biomechanical settings, it is aimed at analyzing captured data to quantify the movement of body segments, e.g., for clinical studies, diagnostics of orthopaedic patients or to help athletes to understand and improve their performances. It has also grown increasingly important as a source of motion data for computer animation. Surveys on existing methods for MoCap can be found in [16,26]. Well known and commercially available marker based tracking systems exist, e.g., those provided by Motion Analysis, Vicon or Simi [25]. The use of markers comes along with intrinsic problems, e.g., incorrect identification of markers, tracking failures, the need for special laboratory environments and lighting conditions and the fact that people may not feel comfortable with markers attached to the body. This can lead to unnatural motion patterns. As well, marker based systems are designed to track the motion of the markers themselves, and thus it must be assumed that the recorded motion of the markers is identical to the motion of the underlying human segments. Since human segments are not truly rigid this assumption may cause problems, especially in highly dynamic movements typically seen in sporting activities. For these reasons, marker-less tracking is an important field of research that requires knowledge in biomechanics, computer vision and computer graphics.

Typically, researchers working in the area of computer vision prefer simplified human body models for MoCap, e.g., stick, ellipsoidal, cylindric or skeleton models [4,5,15,18,24]. In computer graphics, advanced object modeling and texture mapping techniques for human motions are well known [8,10,22,37], but image processing or pose estimation techniques (if available) are often simplified.

B. Rosenhahn (✉) · H.-P. Seidel
Max Planck Center for Visual Computing and
Communication, 66123 Saarbrücken, Germany
e-mail: rosenhahn@mpi-sb.mpg.de

U. Kersting · K. Powell
Department of Sports and Exercise Science,
The University of Auckland,
Auckland, New Zealand

G. Klette · R. Klette
Department of Computer Science,
The University of Auckland,
Auckland, New Zealand

In [13] a shape-from-silhouettes approach is applied to track human beings and incorporates surface point clouds with skeleton models. One of the subjects even wears a pair of shorts, but the cloth is not explicitly modeled and simply treated as rigid component. Furthermore, the authors just perform a quantitative error analysis on synthetic data, whereas in the present study a second (commercial) marker based tracking system is used for comparison.

A recent work of us [32] combines silhouette based pose estimation with more realistic human models: these are represented by free-form surface patches. Local morphing along the surface patches is applied to gain a realistic human model within silhouette based MoCap. Also a comparison with a marker based system is performed indicating a stable system.

In this setup, the subjects have to wear a body suit to ensure an accurate matching between the silhouettes and the surface models of the legs. Unfortunately, body suits may be uncomfortable to wear in contrast to loose clothing (shirts, shorts, skirts, etc.). The subjects also move slightly different in body suits compared to being in clothes since all body parts (even unfavorite ones) are clearly visible. The incorporation of cloth models would also simplify the analysis of outdoor scenes and arbitrary sporting activities. It is for these reasons that we are interested in a MoCap system which also incorporates cloth models.

Cloth draping [17,19,23,35] is a well known research topic in computer graphics. Virtual clothing can be moved and rendered so that it blends seamlessly with motion and appearance in movie scenes. The motion of fabrics is determined by bending, stretching and shearing parameters, as well as external forces, aerodynamic effects and collisions. For this reason the estimation of cloth simulation parameters is essential and can be done by video [3,11,29] or range data [21] analysis. Existing approaches can be roughly divided into geometrically or physically based ones. Physical approaches model cloth behavior by using potential and kinetic energies. The cloth itself is often represented as a particle grid in a spring–mass scheme or by using finite elements [23]. Geometric approaches [35] model cloths by using other mechanics theories which are often determined empirically. These methods can be very fast computationally but are often criticized as being not very appealing visually.

This contribution starts with a summary of the silhouette based MoCap system [32]. It continues with introducing a kinematically motivated cloth draping model and a wind model, which allows deformation of the particle mesh of a cloth with respect to oncoming external forces. The proposed draping method belongs to the class of geometric approaches [35] for cloth draping. The reason for choosing this class is twofold: firstly, we need a model which supports time efficiency, since cloth draping is needed in one of the innermost loops for minimization of the used error functional. Secondly, it should be easy to implement and based on the same parametric representation as the used free-form surface patches. This allows a direct integration into the MoCap system. In section four we explain how to minimize the cloth draping and wind parameters within an error functional for silhouette based MoCap. This allows us to determine joint positions of the legs even if they are partially occluded (e.g., by skirts). We present MoCap results of a subject wearing a skirt and perform a quantitative error analysis. Section five concludes with a summary.

1.1 Contributions

In this paper we inform about the following main contributions:

1. A so-called kinematic cloth draping method is proposed. It belongs to the class of geometric cloth draping methods and is well suited to be embedded in a MoCap system due to the use of a joint model.
2. The cloth draping is extended by including a wind model which allows to adapt the cloth draping to external forces, the scene dynamics or speed of movement.
3. The main contribution is to incorporate the cloth draping algorithm in a silhouette based MoCap system. This allows for determining the joint configurations even when parts of the person are covered with fabrics (see Fig. 1).
4. Finally, we perform a quantitative error analysis. This is realized by comparing the MoCap results with a (commercially available) marker based tracking system. The analysis shows that we receive stable results and can compete with the error range of marker based tracking systems.

2 Foundations

This section describes the modules of the MoCap system presented in [32]. It starts with mathematic and algorithmic foundations, image segmentation based on level set functions and the used model representation. These foundations are needed in subsequent sections.

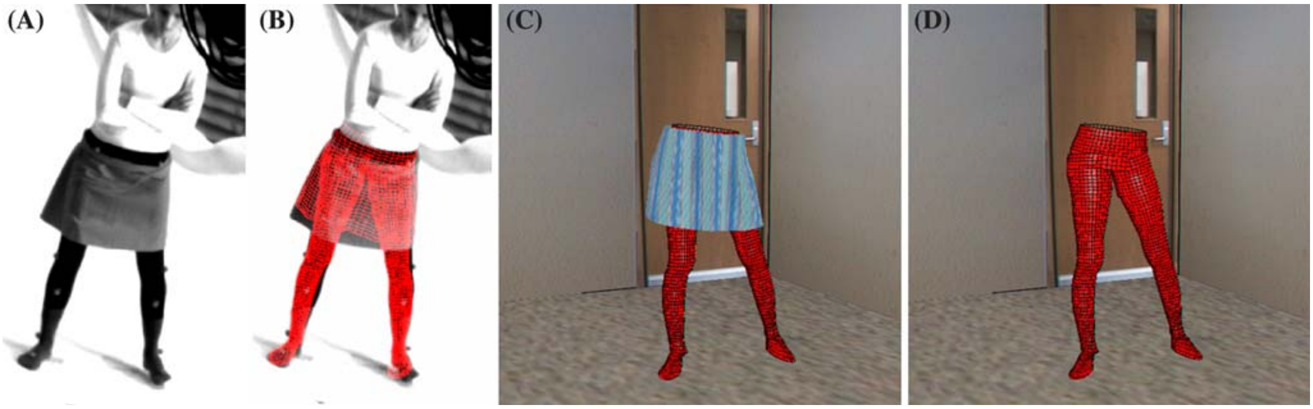


Fig. 1 **a** *Input*: A multi-view image sequence (four cameras, one cropped image is shown). **b** The algorithm determines the cloth parameters and joint configuration of the underlined leg model. **c** Cloth and leg configuration in a virtual environment. **d** Plain leg configuration

2.1 Pose estimation

The pose estimation algorithm requires a set of point correspondences (X_i, x_i), with 4D (homogeneous) model points X_i and 3D (homogeneous) image points x_i. Each image point defines a 3D Plücker line $L_i = (n_i, m_i)$ (a projective ray), with a (unit) direction n_i and moment m_i [27,31].

Every 3D rigid motion can be represented in exponential form:

$$M = \exp(\theta\hat{\xi}) = \exp\begin{pmatrix} \hat{\omega} & v \\ 0_{3\times1} & 0 \end{pmatrix}, \tag{1}$$

where $\theta\hat{\xi}$ is the matrix representation of a twist $\xi \in \mathrm{se}(3) = \{(v, \hat{\omega}) | \, v \in \mathbb{R}^3, \hat{\omega} \in \mathrm{so}(3)\}$, with $\mathrm{so}(3) = \{A \in \mathbb{R}^{3\times3} | \, A = -A^T\}$. The Lie algebra $so(3)$ is the tangential space of the 3D rotations. Its elements are (scaled) rotation axes, which can either be represented as a 3D vector

$$\theta\omega = \theta\begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix}, \text{ with } \|\omega\|_2 = 1, \tag{2}$$

or as a screw symmetric matrix

$$\theta\hat{\omega} = \theta\begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix}. \tag{3}$$

In fact, M is an element of the one-parametric Lie group SE(3), known as the group of direct affine isometries. A main result of Lie theory is that to each Lie group there exists a Lie algebra which can be found in its tangential space, by derivation and evaluation at its origin; see [27] for more details. The corresponding Lie algebra to SE(3) is denoted as se(3). A twist contains six parameters and can be scaled to $\theta\xi$ with a unit vector ω. The parameter $\theta \in \mathbb{R}$ corresponds to the motion velocity (i.e., the rotation velocity and pitch). For varying θ, the motion can be identified as screw motion around an axis in space. The six twist components can either be represented as a 6D vector

$$\begin{aligned} \theta\xi &= \theta(\omega_1, \omega_2, \omega_3, v_1, v_2, v_3)^{\mathrm{T}} \\ \text{with } \|\omega\|_2 &= \|(\omega_1, \omega_2, \omega_3)^{\mathrm{T}}\|_2 = 1, \end{aligned} \tag{4}$$

or as a 4×4 matrix

$$\theta\hat{\xi} = \theta\begin{pmatrix} 0 & -\omega_3 & \omega_2 & v_1 \\ \omega_3 & 0 & -\omega_1 & v_2 \\ -\omega_2 & \omega_1 & 0 & v_3 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \tag{5}$$

To reconstruct a group action $M \in \mathrm{SE}(3)$ from a given twist, the exponential function $\exp(\theta\hat{\xi}) = \sum_{k=0}^{\infty} \frac{(\theta\hat{\xi})^k}{k!} = M \in \mathrm{SE}(3)$ must be computed. This can be done efficiently by using the Rodriguez formula [27].

For pose estimation the reconstructed Plücker lines are combined with the screw representation for rigid motions and applied within a fix point iteration scheme: incidence of the transformed 3D point X_i with the 3D ray $L_i = (n_i, m_i)$ can be expressed as

$$(\exp(\theta\hat{\xi})X_i)_{3\times1} \times n_i - m_i = 0. \tag{6}$$

Indeed, X_i is a homogeneous 4D vector, and after multiplication with the 4×4 matrix $\exp(\theta\hat{\xi})$ the homogeneous component (which is one) is neglected to evaluate the cross product with n_i. Now the equation is linearized by using $\exp(\theta\hat{\xi}) = \sum_{k=0}^{\infty} \frac{(\theta\hat{\xi})^k}{k!} \approx I + \theta\hat{\xi}$, with I as identity matrix. This results in

$$((I + \theta\hat{\xi})X_i)_{3\times1} \times n_i - m_i = 0 \tag{7}$$

and can be reordered into an equation of the form $A\xi = b$. Collecting a set of such equations (each is of rank two) leads to an overdetermined system of equations, which can be solved using, for example, the Householder algorithm. The Rodriguez formula can be applied to reconstruct the group action M from the estimated twist ξ. Then the 3D points can be transformed and the process is iterated until the algorithm converges.

Joints are expressed as special screws with no pitch of the form $\theta_j\hat{\xi}_j$ with known $\hat{\xi}_j$ (the location of the rotation axes as part of the model representation) and unknown joint angle θ_j. The constraint equation of a jth joint has the form

$$(\exp(\theta_j\hat{\xi}_j)\cdots\exp(\theta_1\hat{\xi}_1)\exp(\theta\hat{\xi})X_i)_{3\times 1}\times n_i - m_i = 0, \quad (8)$$

which is linearized in the same way as the rigid body motion itself. It leads to three linear equations with the six unknown pose parameters and j unknown joint angles. Collecting a sufficient number of equations leads to an overdetermined system of equations.

Note that the algorithm uses reconstructed 3D lines. Therefore it is possible to gain equations for different cameras (calibrated with respect to the same world coordinate system), to put them together in one system of equations and solve them simultaneously. This is the key idea to deal with partial occlusions: A joint that is not visible in one camera must be visible in another one to get a solvable system of equations. A set of four cameras around the subject covers a large range and allows the analysis of complicated motion patterns.

2.2 Image segmentation

Image segmentation usually means estimating boundaries of objects in an image. This task can become very difficult since noise, shading, occlusion or texture transitions between the object and the background may distort the segmentation or even make it impossible. Our approach for image segmentation is based on level sets [7,9,14,28].

A level set function $\Phi \in \Omega \mapsto \mathbb{R}$ splits the image domain Ω into two regions Ω_1 and Ω_2 with $\Phi(x) > 0$ if $x \in \Omega_1$ and $\Phi(x) < 0$ if $x \in \Omega_2$. The zero-level line thus marks the boundary between both regions. The segmentation should maximize the total a-posteriori probability given the probability densities p_1 and p_2 of Ω_1 and Ω_2, i.e., pixels are assigned to the most probable region according to the Bayes rule. Ideally, the boundary between both regions should be as small as possible. This can be expressed by the following energy functional that is sought to be minimized:

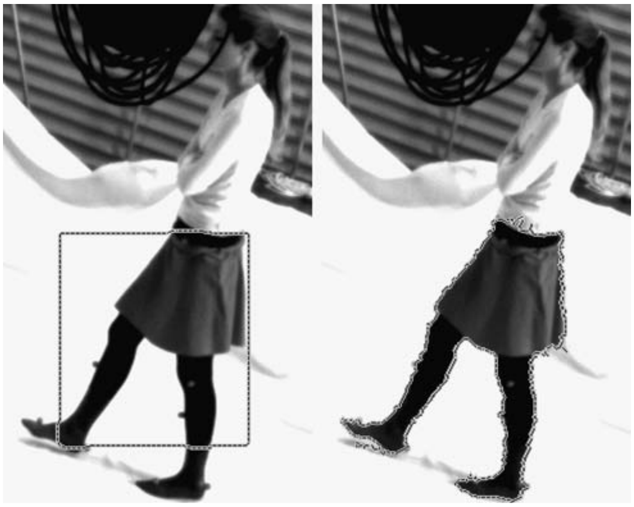


Fig. 2 Silhouette extraction based on level set functions. *Left*: Initial segmentation. *Right*: Segmentation result

$$E(\Phi,p_1,p_2) = -\int_{\Omega} \big(H(\Phi(x))\log p_1 \\ +(1-H(\Phi(x)))\log p_2 + \nu|\nabla H(\Phi(x))|\big)\,\mathrm{d}x,$$

where $\nu > 0$ is a weighting parameter and $H(s)$ is a regularized version of the Heaviside function, e.g., the error function. Minimization with respect to the region boundary represented by Φ can be performed according to the gradient descent equation

$$\partial_t\Phi = H'(\Phi)\left(\log\frac{p_1}{p_2} + \nu\,\mathrm{div}\left(\frac{\nabla\Phi}{|\nabla\Phi|}\right)\right), \quad (9)$$

where $H'(s)$ is the derivative of $H(s)$ with respect to its argument. The probability densities p_i are estimated according to the *expectation–maximization principle*. Having the level set function initialized with some contour, the probability densities within the two regions are estimated by the gray value histograms smoothed with a Gaussian kernel K_σ having standard deviation σ.

Figure 2 shows on the left an example image with an initialization of the region as a rectangle. The right image shows the estimated (stationary) contour after 50 iterations. The legs and skirt are well extracted, but there are some deviations due to shadows. Such inaccuracies can be compensated through the pose estimation procedure. For our algorithm we make a tracking assumption. Therefore, we initialize the silhouette with the pose of the last frame which greatly reduces the number of iterations needed.

2.3 Shape registration

The goal of shape registration can be formulated as follows: given a certain distance measure, the task is to determine one transformation that leads to the minimum distance between shapes. A very popular shape matching method working on such representations is the iterated closest point (ICP) algorithm [2]. Given two finite sets P and Q of points, the (original) ICP algorithm calculates a rigid transformation T and attempts to ensure $\mathrm{TP} \subseteq Q$.

1. Nearest point search: For each point $p \in P$ find the closest point $q \in Q$.
2. Compute registration: Determine the transformation T that minimizes the sum of squared distances between pairs of closest points (p, q).
3. Transform: Apply the transformation T to all points in set P.
4. Iterate: Repeat steps 1–3 until the algorithm converges.

This algorithm converges to the next local minimum of the sum of squared distances between closest points. A good initial estimate is required to ensure convergence to the sought solution. Unwanted solutions may be found if the sought transformation is too large, e.g., many shapes have a convergence radius in the area of 20° [12], or if the point sets do not provide sufficient information for a unique solution.

The original ICP algorithm has been modified in order to improve the rate of convergence and to register partially overlapping sets of points. Zhang [38] uses a modified cost function based on robust statistics to limit the influence of outliers. Other approaches aim at the avoidance of local minima during registration subsuming the use of Fourier descriptors [33], color information [20] or curvature features [34].

The advantages of ICP algorithms are obvious: they are easy to implement and will provide good results, if the sought transformation is not too large [12]. For our tracking system we compute correspondences between points on image silhouettes to the surface mesh, with the ICP algorithm presented in [33].

2.4 Silhouette based motion capturing

Human models are often represented in a layered structure based on skeletons, meta-balls or polygonal surface patches (for the bones, muscles and the skin, respectively) [10]. It is further common to add local deformation operators onto model joints [1]. These representations are well suited for certain applications in computer graphics, but for this specific task we prefer a unified representation: we decided to model objects in terms of two-parametric free-form surface patches. For the leg model we use three patches for the hip, left leg and right leg, respectively. We further add joint indices onto each surface node so that we can directly determine the joints of the corresponding kinematic chain for each node on the legs.

Once a contour has been extracted from the image, see Sect. 2.2, points on this contour must be matched to 3D points on the object surface. This is done by an ICP procedure, see Sect. 2.3. Firstly, one determines those points from the object surface that are part of the object silhouette resulting in the 3D object rim contour. The projection of each of these points is then matched to the closest point of the extracted contour. In this way, one obtains a 2D–3D point correspondence for each 3D mesh point that is part of the object silhouette. Based on the correspondences, equations are generated following Sect. 2.1. These are solved by using the Householder algorithm and are iterated until the overall pose converges.

In order to deal with larger motions during pose tracking a simple sampling method is used to apply the pose estimation algorithm to different neighboring (random) starting positions. From all results the one with the smallest error between the extracted silhouette and the projected surface mesh is chosen. This is important in order to avoid local minima during tracking.

This approach minimizes the spatial distance of the 3D surface rim to the 2D image silhouettes. In a multi-view set-up, it leads to equations which are sufficient for a unique solution of the pose and kinematic chain parameters. After pose estimation, a new rim contour is determined from the new pose and the process is iterated until the overall pose converges. An example for silhouette based motion capturing is shown in Fig. 3. The pose result is visualized by projecting the transformed meshes onto the images. Experiments proved that the algorithm can handle self-occlusions, even for highly dynamic movements.

Though it is not the main contribution of this work, it is important to point out that the combined use of level set based image segmentation, 3D shape and motion priors (see [6]) leads to stable tracking results even in highly noisy image sequences as shown in Fig. 4; in a walking sequence we replaced 25% of all pixels by a uniform random value. Additionally, we added heavy occlusions to all camera views by randomly distributing box-shaped occlusions of random size and gray value across the images.

Figure 16 also shows a corrupted frame of a knee-bending sequence. Our approach is able to handle such outliers, and still leads to reasonably good tracking results.

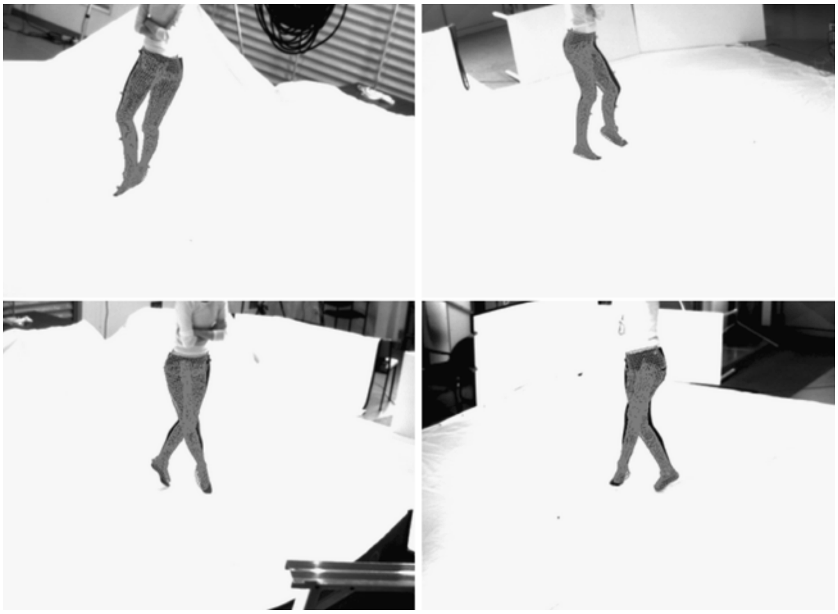


Fig. 3 Pose result of the silhouette based MoCap system for four synchronized cameras. The pose is visualized by projecting the transformed surface mesh onto the image data

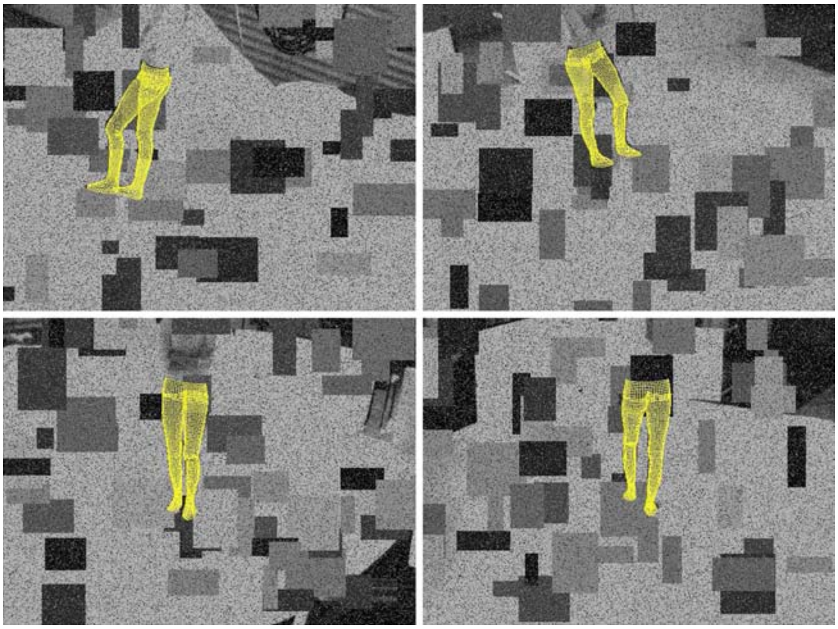


Fig. 4 Segmentation and pose estimation in highly noisy image sequences. We replaced 25% of all pixels by a uniform random value. Additionally, we randomly distributed box-shaped occlusions of random size and gray value across the images

3 Kinematic cloth draping

Cloth draping is a highly interesting discipline in computer graphics. It deals with the realistic simulation and modeling of a cloth which is moving and falling on objects, e.g., a human being. As mentioned in Sect. 1, existing approaches can roughly be divided into geometrically or physically based ones. Physical approaches model the cloth behavior by using potential and kinetic energies [23]. Geometric approaches [35] model clothes by using other mechanics theories which are often determined empirically. For our set-up we decided to use a geometric approach to model cloth behavior. The reason is two fold: firstly, cloth draping is needed in one of the innermost loops for pose estimation and segmentation. Therefore it must be very fast. In our case we need around 400 iterations for each frame to converge to a solution. An assumed cloth draping algorithm which would need a few seconds would result in hours to calculate the pose of one frame and weeks for a whole sequence. The second reason is that we are interested in a model representation which fits as well as possible to our free-form surface based representation.

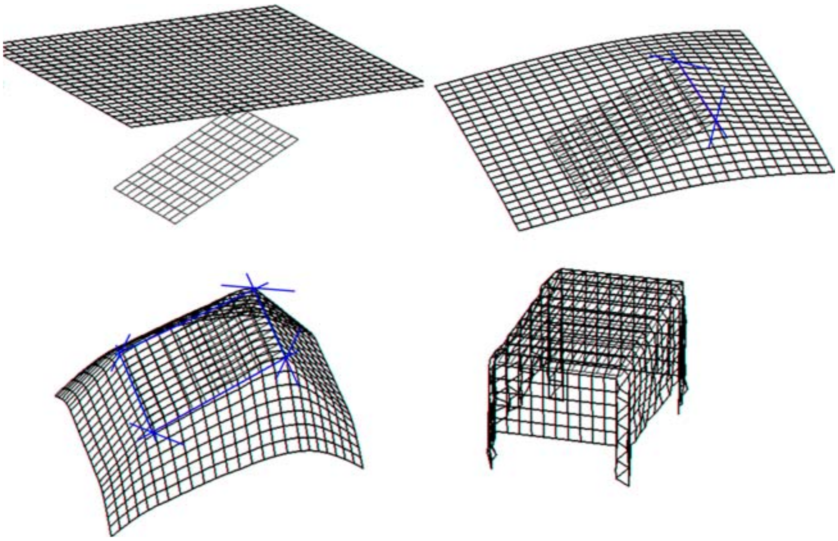


Fig. 5 A piece of cloth falling on a tilted plane

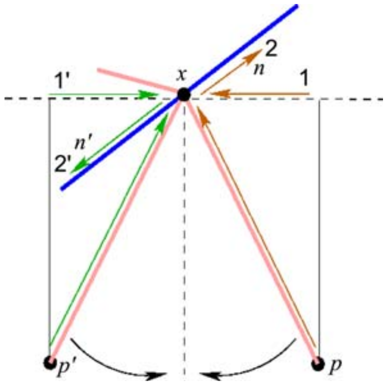


Fig. 6 Basic principle of kinematic cloth draping

Due to these limitations, we decided to model the skirt as a string-system based on kinematic chains: The main principle is visualized in Fig. 5 for a piece of cloth falling on a plane. The piece of cloth is represented as a particle grid, a set of points with known topology. While lowering the cloth, the distance of each cloth pointing to the ground plane is determined. If the distance between one point on the cloth to the surface is below a threshold, the point is set as a fixed point; see the top-right image of Fig. 5. Now the remaining points are not allowed to "fall" downwards anymore. Instead, for each point, the nearest fixed point is determined and a joint (perpendicular to the particle point) is used to rotate the free point along the joint axis through the fixed point. The used joint axes are marked as blue lines in Fig. 5. The image in Fig. 6 shows the geometric principle to determine the twist for rotation around a fixed point: The blue line represents a mesh of the rigid body, x is the fixed point and the (right) pink line segment connects x to a particle p of the cloth. The direction between both points is projected onto the y-plane of the fixed point (1). The direction is then rotated about 90° (2), leading to the rotation axis n. The point pair $(n, x \times n)$ defines the components of the twist, see Eq. (5). While lowering the cloth, free particles not touching a second rigid point will swing below the fixed point (e.g., p'). This leads to an opposite rotation [indicated with (1′), (2′) and n'] and the particle swings back again, resulting in a naturally swinging draping pattern. The draping velocity is steered through a rotation velocity θ, which is set to 2° during the iteration. Since all points either become fixed points, or result in a stationary configuration while swinging backwards and forwards, we constantly use 50 iterations to drape the cloth. The remaining images in Fig. 5 show the ongoing draping and the final result.

Figure 7 shows examples of a skirt falling on the leg model. The skirt is modeled as a two-parametric mesh model. Due to the use of general rotations, the internal distances in the particle mesh cannot change with respect to one of these dimensions, since a rotation maintains the distance between the involved points. However, this is not the case for the other sampling dimension. For this reason, the skirt needs to be reconstrained after draping. This is visualized in Fig. 8; if a stretching parameter is exceeded, the particles are reconstrained to minimal distance to each other. This is only done for the non-fixed points (i.e., for those which are not touching the skin). It results in a better appearance. Figure 8 shows that even the creases are maintained.

The cloth draping algorithm is only suited for non-moving (static) objects, but during tracking also cloth dynamics appear (e.g., swinging effects). To improve the dynamic behavior of clothing during movements, we further add a wind model to the cloth draping.

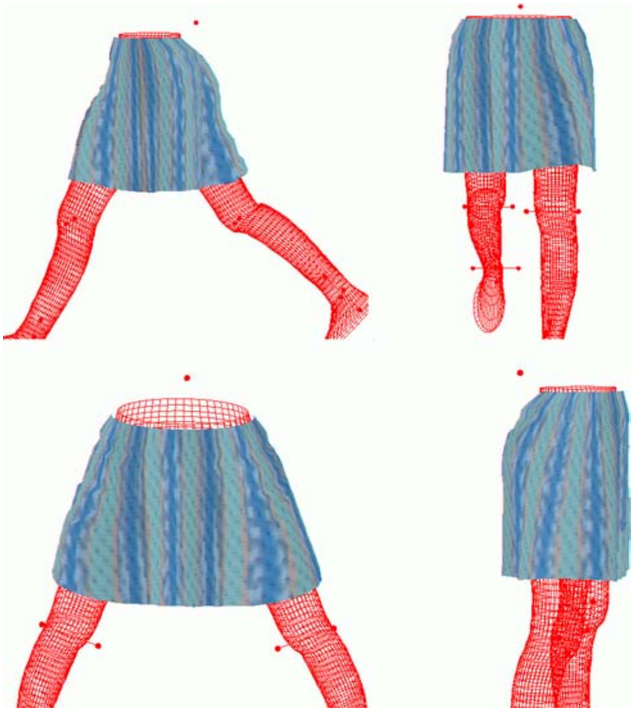


Fig. 7 Skirt draping on leg model

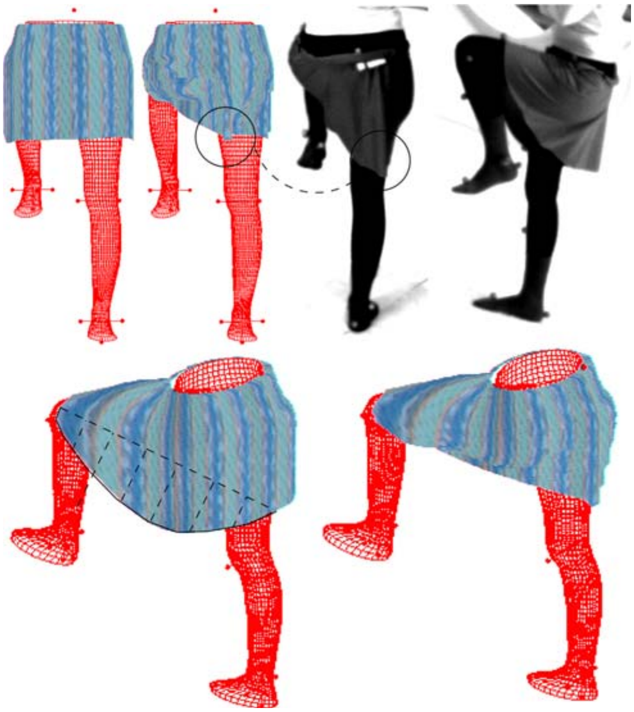


Fig. 8 Reconstraining the skirts' length

We extend the cloth-draping in the following way: dependent on the direction of a wind force we determine a joint on the nearest fixed point for each free point on the surface mesh with the joint direction being perpendicular to the wind direction. Now we rotate the free point around this axis dependent on the wind force (expressed as an angle) or until the cloth is touching the underlying surface. Figure 9 shows examples of the cloth with no, frontal or backward wind. The wind force and direction are later part of the minimization function during pose tracking.

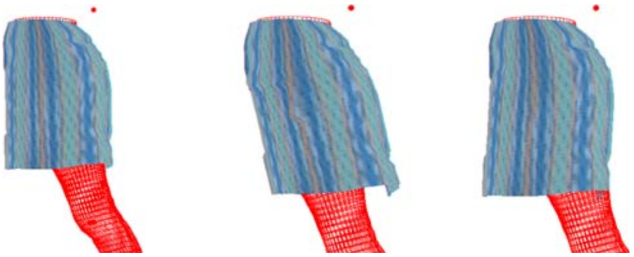


Fig. 9 Wind model on skirt. *Left*: no wind, *middle*: frontal wind, *right*: backwards wind

Cloth dynamics during movements are modeled with the help of such external forces: the walking person causes a relative wind force acting on its body during movement. Also the swinging dynamics of the cloth during tracking are estimated in terms of such external wind forces. Figure 10 visualizes the effect of the used wind model.

Since the motion dynamics of the cloth are determined dynamically, we need no information about the cloth type or weight since they are implicitly determined from the minimized cloth dynamics in the image data. We only need the measures of the cloth (in this case of a skirt).

4 Combined cloth draping and MoCap

The assumptions are as follows: we assume the representation of a subject's lower torso (i.e., for the hip and legs) in terms of free-form surface patches. We also assume

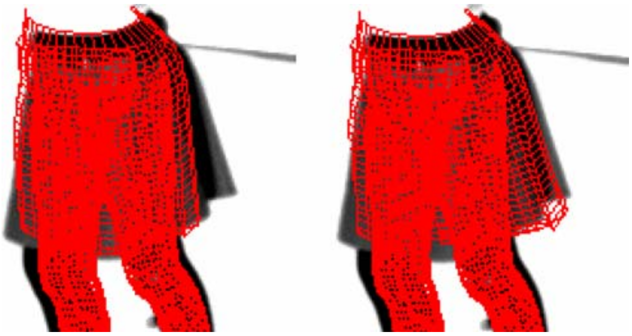


Fig. 10 *Left*: Overlaid pose result without wind model. *Right*: Overlaid pose result including wind model

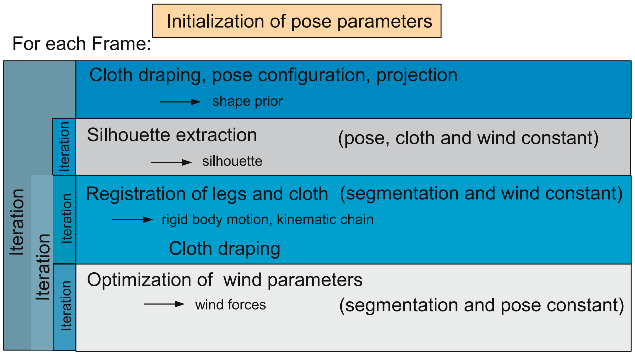

Initialization of pose parameters
For each Frame:
Cloth draping, pose configuration, projection
shape prior
Silhouette extraction
(pose, cloth and wind constant)
silhouette
Registration of legs and cloth
(segmentation and wind constant)
rigid body motion, kinematic chain
Cloth draping
Optimization of wind parameters
wind forces
(segmentation and pose constant)
Iteration
Iteration
Iteration
Iteration
Iteration

Fig. 11 The basic algorithm for combined cloth draping and motion capturing

known joint positions along the legs. Furthermore, we assume the wearing of a skirt with known measures. The person is walking or stepping within a calibrated volume of a four-camera setup. These cameras are synchronized by an external trigger signal. The task is to determine the pose of the model and the joint configuration. For this we minimize the image error between the projected surface meshes and the extracted image silhouettes. The unknowns are the pose, kinematic chain and the cloth parameters (wind forces, cloth thickness, etc.). The task can be represented as an error functional as follows:

$$E(\Phi,p_1,p_2,\theta\xi,\theta_1,\dots,\theta_n,c,w) = \underbrace{-\int_\Omega \big(H(\Phi)\log p_1 + (1-H(\Phi))\log p_2 + \nu|\nabla H(\Phi)|\big)\,\mathrm{d}x}_{\text{segmentation}} + \lambda \underbrace{\int_\Omega (\Phi - \Phi_0(\underbrace{\theta\xi,\theta_1,\dots,\theta_n}_{\text{pose and kinematic chain}}, \underbrace{c,w}_{\text{wind parameters}}))\mathrm{d}x}_{\text{shape error}}$$

Due to the large number of parameters and unknowns we decided for an iterative minimization scheme, see Fig. 11; Firstly, the pose, kinematic chain and wind parameters are kept constant, while the error functional for the segmentation (based on Φ, p_1, p_2) is minimized (Sect. 2.2). Then the segmentation and wind parameters are kept constant while the pose and kinematic chain are determined to fit the surface mesh and the cloth to the silhouettes (Sects. 2.1 and 2.3). Note that after cloth draping, the skirt is treated as a kinematic chain during pose estimation. Therefore, the cloth contributes with equations to the overall pose. After each iteration, there is a need to re-drap (and adapt) the cloth. Finally, different wind directions and wind forces are sampled to refine the pose result (Sect. 3). Since all parameters influence each other, the process is iterated until a steady state is reached. In our experiments, we always converged to a local minimum. The computation time is approximately 5 min per four-camera frame on a standard (3 GHz) linux machine. Figure 12 shows results of pose estimation and cloth draping by superimposing the surface patches with the image data. The analysis of various image sequences revealed that it is possible to recover the underlying kinematic structure though body parts are heavily occluded by fabrics. We analyzed different image sequences, with a subject walking, dancing, bending the knees or pulling them up. The algorithm can handle partial occlusions, e.g., caused by crossing legs.

Figure 13 shows pose results animated in a virtual environment. The images on the left show pose results with the cloth model, whereas the images on the right just show the estimated leg configuration. In the sequence, the body is turning around 180°, legs are crossing and the skirt is swinging. Due to the dynamics of the sequence, a static cloth draping (without taking into account external forces) would cause (and has caused in the first version of our program) major problems for tracking. But with the included wind model applied to the skirt, the tracking is successful and stable.

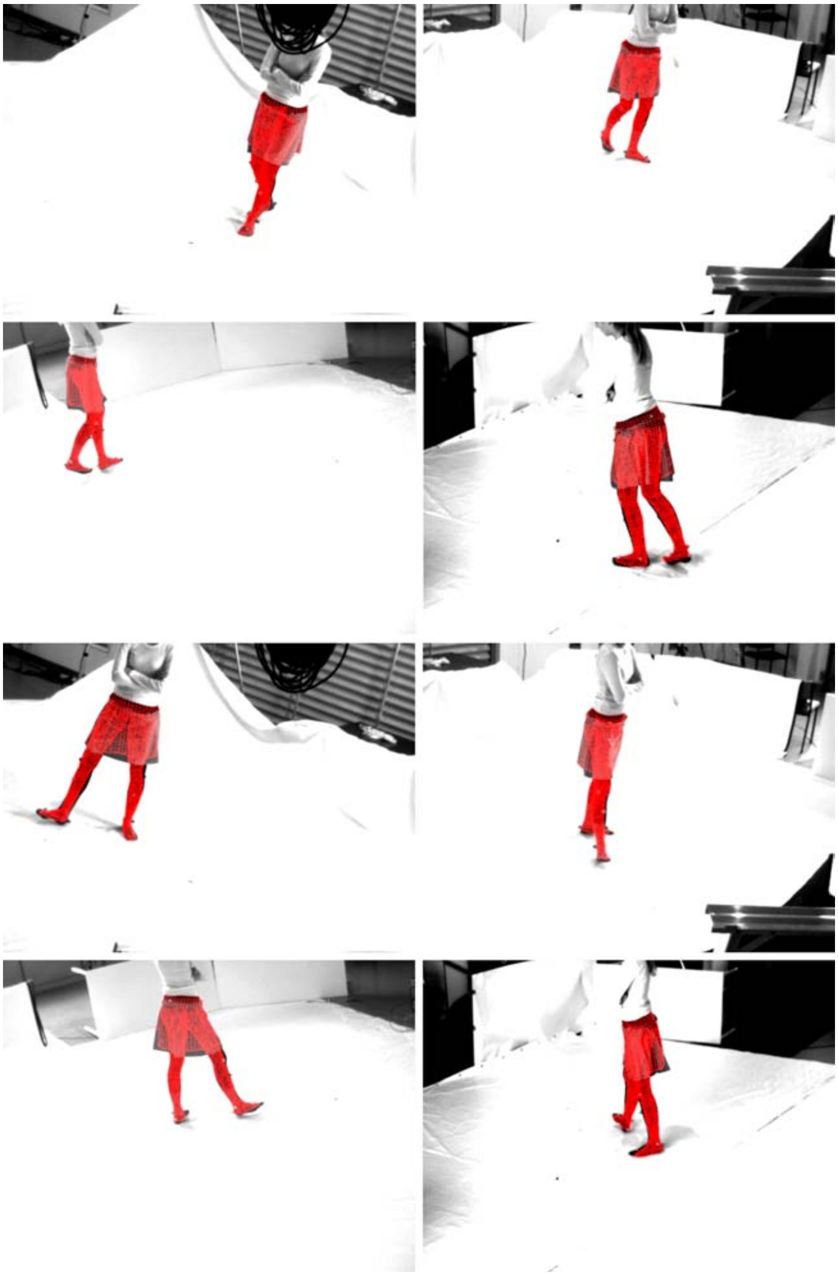
Figure 14 shows an example frame of a sequence with the subject pulling up the knees and Fig. 15 shows a pose result where the subject is bending the knees. Both images are interesting since in the first example the skirt is stretched whereas in the second example the skirt is partially hanging down loosely. It can be seen that the leg configurations are (according to subjective visual judgment) accurately recovered.

Figure 16 visualizes the stability of our approach: while grabbing the images, a couple of frames were stored completely wrong. These sporadic outliers can be compensated from our algorithm, and a few frames later (see the images below) the pose is correct. Due to the use of level set functions and the incorporated shape prior, the segmentation is close to the last pose configuration leading to an incorrect, but not completely wrong pose. Once the image data is useful again, the segmentation automatically relies more on the image data and converges to the real image silhouette.

4.1 Quantitative error analysis

A lack of many studies so far (see e.g., [24]) is that the only feedback one receives is visual feedback of the pose provided by overlaying the pose with the image data or by visualizing them in a virtual environment. To enable a quantitative error analysis, we decided to use a commercial marker based tracking system for comparison. We use the motion analysis software [25] with eight Falcon cameras. For data capture we use the Eva

Fig. 12 Pose results of skirt draping and wind adaption during tracking



3.2.1 software and the motion analysis Solver Interface 2.0 for inverse kinematics computing. For this system the subject must have retro-reflective markers attached to specific anatomical landmarks. Around each camera is a strobe light led ring and a red-filter is in front of each lens. This gives very strong image signals of the markers in each camera. These are treated as point markers which are reconstructed in the eight-camera system. One of the cameras can be seen on the left in Fig. 17. The system is calibrated by using a wand-calibration method. Due to the filter in front of the images we had to use a second camera set-up which provides *real* image data. This camera system (consisting of four cameras) is calibrated by using a calibration cube. After calibration, both camera systems are calibrated with respect to each other. Then we generate a stick-model from the point markers including joint centers and orientations (Fig. 18). This results in a completely calibrated set-up which we use for a system comparison. The right image of Fig. 17 shows our test subject wearing the skirt and the retro-flective markers. Figure 19 shows two example frames of the sequence with the pose overlaid in one of the four cameras. The images on the right show the animated leg configuration.

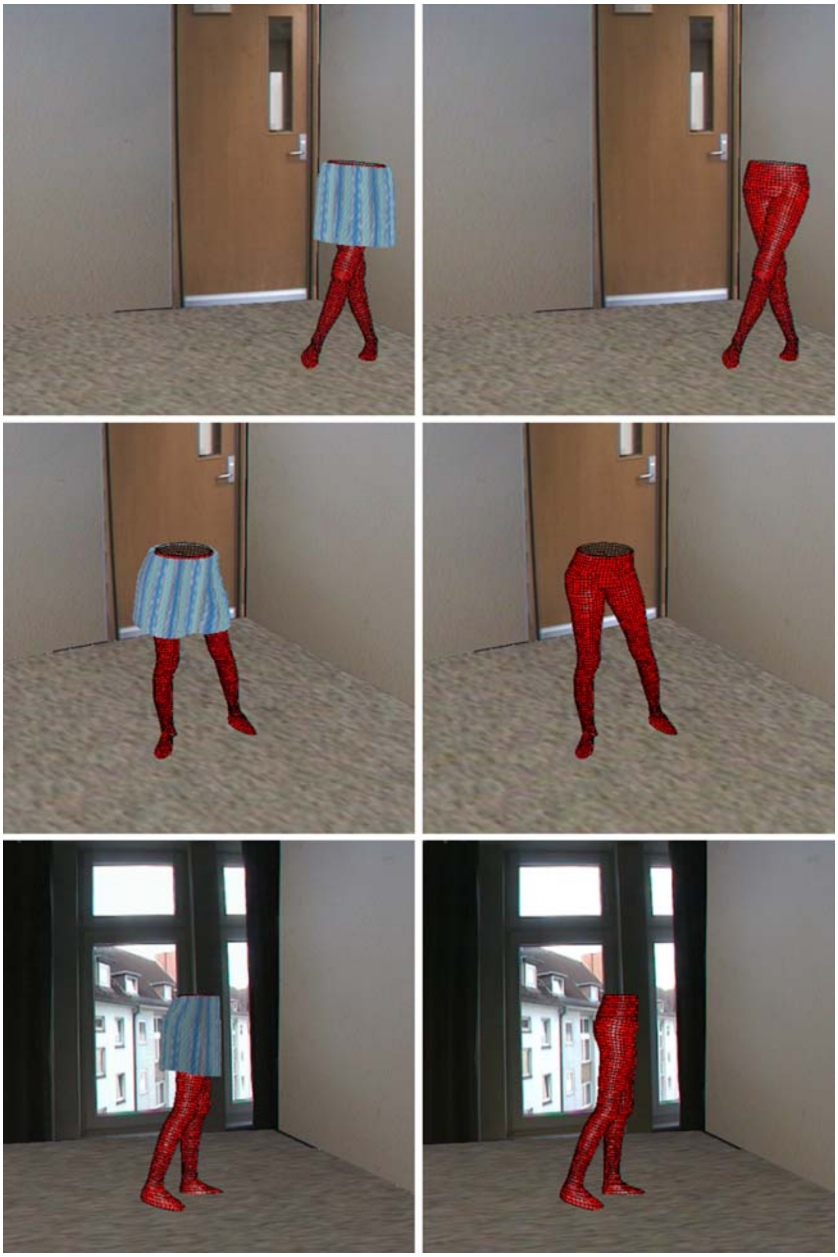


Fig. 13 Pose results animated in a virtual environment. Each image pair shows the pose result with the estimated cloth parameters on the left and the plain leg configuration on the right

The diagram in Fig. 20 shows the left and right knee of a plain walking sequence. The (smoothed) motion analysis data are overlaid with the silhouette based results. The mean absolute difference is 2.72° for the left knee and 3.1° for the right knee. Maximum errors occur when the knees are nearly parallel to each other and overlapping in the image data. Table 1 summarizes deviations of the left and right knee for different motion sequences (each of 240 frames).

In [30], eight biomechanical measurement systems are compared (including the used motion analysis system). The author also performed a rotation experiment which shows that root mean square errors are typically within 3°. This shows that the errors are in the error range of marker based tracking systems. We consider this as a good result and indication of a stable tracking algorithm.

Table 1 Deviations between marker based and marker-less tracking systems

Sequence	Left knee	Right knee
Dancing (see Figs. 12, 13)	3.42	2.95
Knee-up (see Fig. 14)	3.22	3.43
Knee bending (see Figs. 15, 16)	3.33	3.49
Walking (see Figs. 19, 20)	2.72	3.1

Evaluated are the left and right knee for different motion sequences (each of 240 frames)

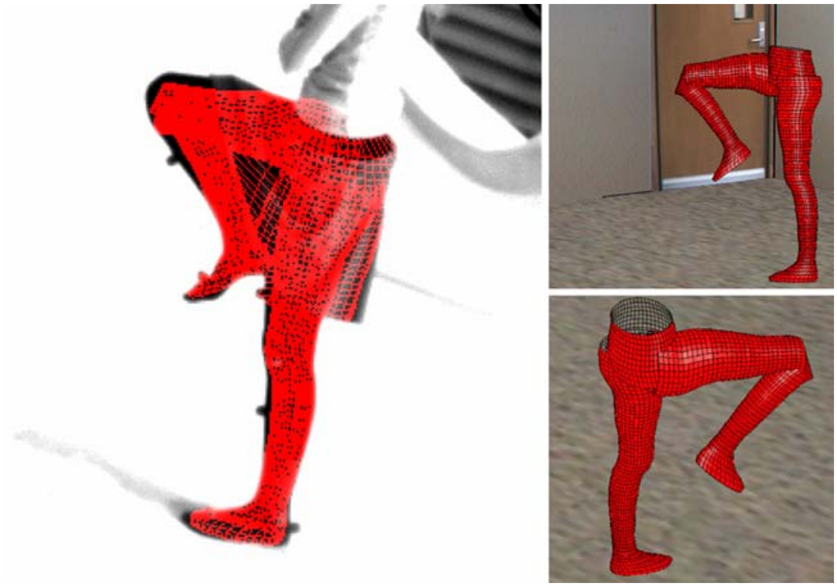


Fig. 14 *Left*: Pose result of the walking sequence, where the knees are pulled up. *Right*: The leg configuration in a virtual environment (from two different viewing angles)

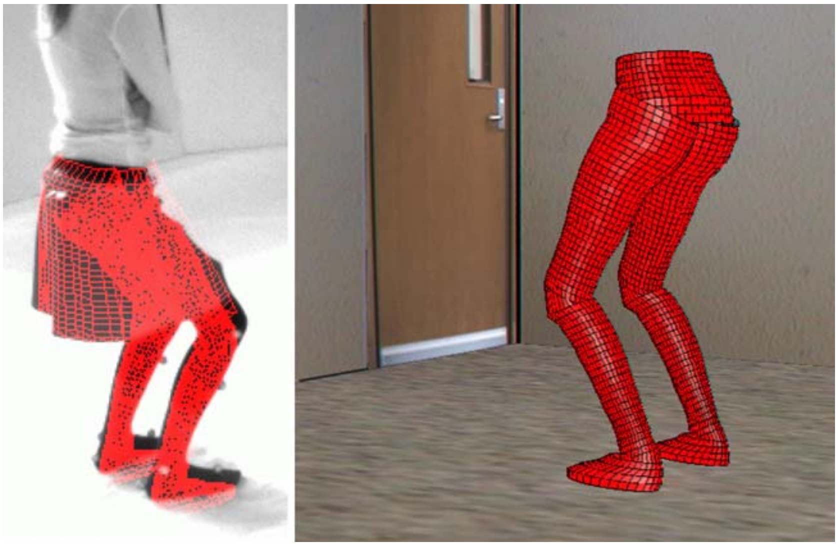


Fig. 15 *Left*: Pose results of a knee-bending sequence, where the knees are bent. *Right*: The leg configuration in a virtual environment

5 Summary

The contribution presents an approach for motion capture of clothed people. To achieve this we extend a silhouette based motion capture system, which relies on image silhouettes and free-form surface patches of the body with a cloth draping procedure. Due to the limited time constraints for cloth draping we decided for a geometric approach based on kinematic chains. We call this cloth draping procedure kinematic cloth draping. This model is very well suited to be embedded in a motion capture system since it allows us to minimize the cloth draping parameters (and wind forces) within the same error functional such as the segmentation and pose estimation algorithm. Due to the number of unknowns for the segmentation, pose estimation, joints and cloth parameters, we decided for an iterative solution. The experiments show that the formulated problem can be solved: we are able to determine joint configurations and pose parameters of the kine-

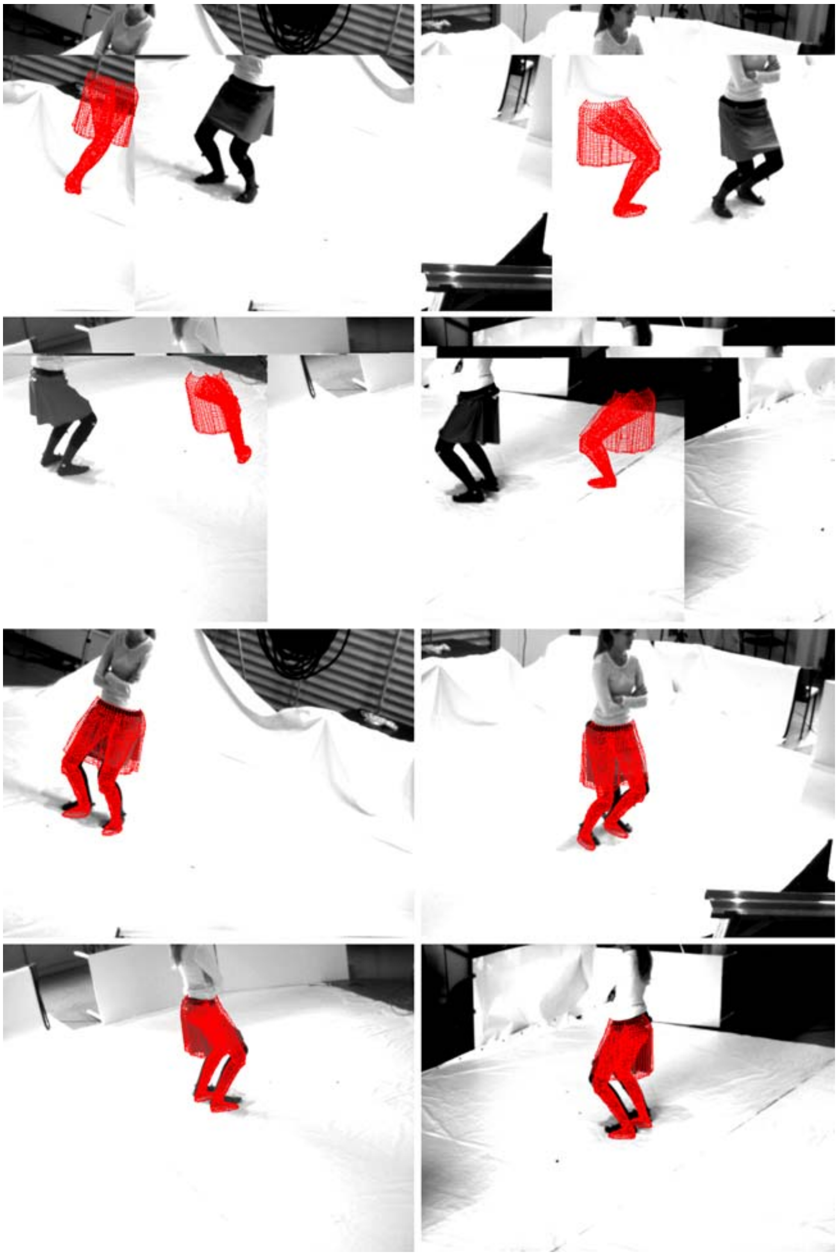


Fig. 16 *Top*: Error during grabbing the images. *Bottom*: Two frames later, the pose is correct again

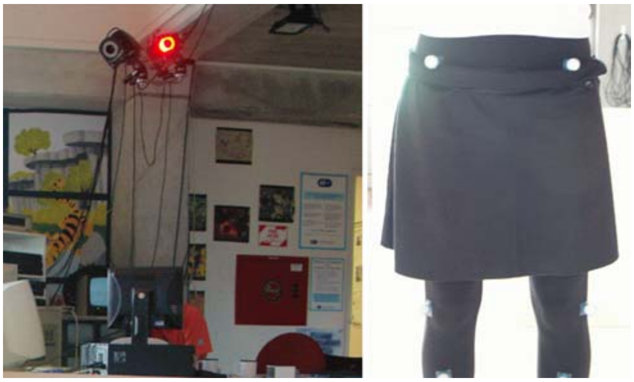


Fig. 17 The set-up for the quantitative error analysis. *Left*: The motion analysis system (and one of the strobe light cameras). *Right*: The subject with markers attached to the body

matic chains, though they are considerably covered with clothes. Indeed, we use the cloth draping appearance to recover the joint configuration and simultaneously determine wind dynamics of the cloth during walking and dancing sequences. We further performed a quantitative error analysis by comparing our method with a commercially available marker based tracking system. The experiments show that we are in the same error range as marker based tracking systems.

Applications are straight forward: the motion capture results can be used to animate avatars in computer animations, and the angle diagrams can be used for the analysis of sports movements or clinical studies. The possibility of wearing loose clothes is more comfortable for

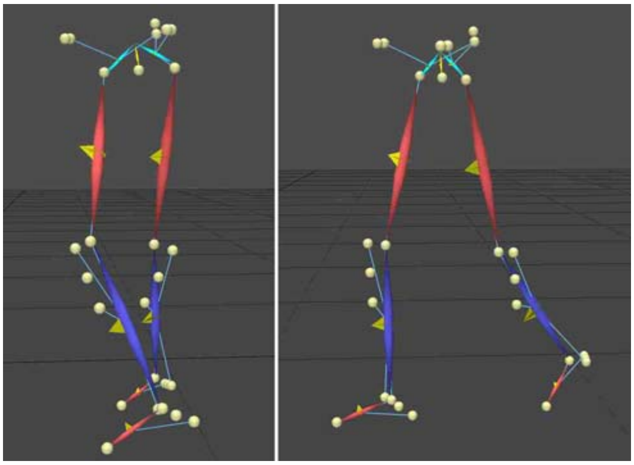


Fig. 18 The coordinate systems of the markers in the lab setup. The markers are used to generate a stick-figure model. The images show pose configurations of crossed legs and walking

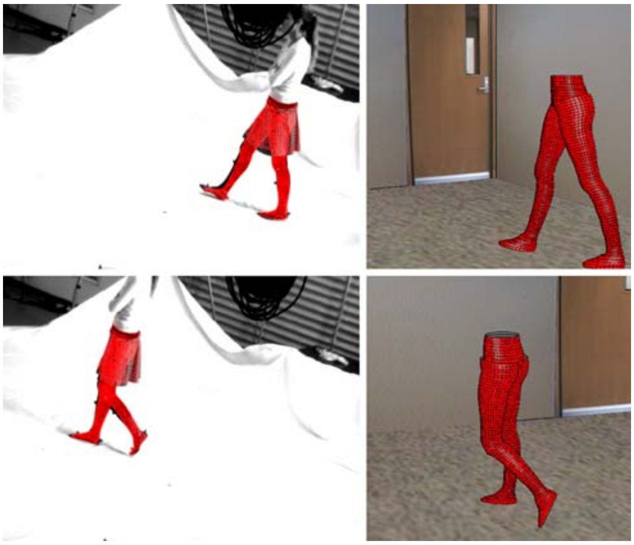


Fig. 19 *Left*: Pose results of the walking sequence. *Right*: The leg configuration in a virtual environment (from a slightly different viewing angle)

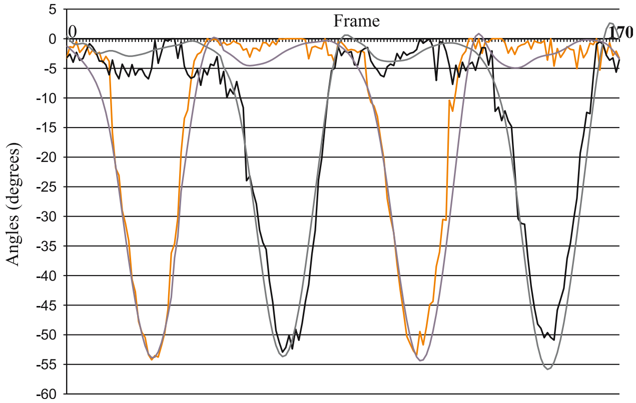


Fig. 20 Knee angles of a walking sequence. *Magenta/gray* left and right knee from the motion analysis system, *orange/black* left and right knee from the marker-less system (including cloth model)

many people and enables a more natural motion behavior. The presented extension also allows us to analyze outdoor activities, e.g., soccer or other team sports.

For future works we plan to extend the cloth draping model with more advanced ones [23] and we will compare different draping approaches and parameter optimization schemes in the motion capturing setup.

Acknowledgements We gratefully acknowledge funding by the DFG project RO2497/1 and the Max-Planck Center for visual computing and communication.

References

1. Allen, B., Curless, B., Popovic, Z.: Articulated body deformation from range scan data. In: Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, pp. 612–619. San Antonio, Texas (2002)
2. Besl, P., McKay, N.: A method for registration of 3D shapes. IEEE Transa. Pattern Anal. Mach. Intell. **12**, 239–256 (1992)
3. Bhat, K., Twigg, C., Hodgins, J., Khosla, P., Popovic, Z., Seitz, S.: Estimating cloth simulation parameters from video. In: Breen, D., Lin,M. (eds.) Proceedings of ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 37–51 (2003)
4. Bregler, C., Malik, J.: Tracking people with twists and exponential maps. In: Proceedings of Computer Vision and Pattern Recognition, pp. 8–15. Santa Barbara, California (1998)
5. Bregler, C., Malik, J., Pullen, K.: Twist based acquisition and tracking of animal and human kinetics. Comput. Vis. **56**(3), 179–194 (2004)
6. Brox, T., Rosenhahn, B., Kersting, U., Cremers, D., Seidel, H.P.: Nonparametric density estimation for human pose tracking. In: Pattern Recognition, 28th DAGM-symposium. Lecture Notes in Computer Science. Springer, Berlin Heidelberg New York (2006)
7. Brox, T., Rousson, M., Deriche, R., Weickert, J.: Unsupervised segmentation incorporating colour, texture, and motion. In: Petkov, N., Westenberg, M.A. (eds.) Proceedings of Computer Analysis of Images and Patterns. Lecture Notes in Computer Science, vol. 2756, pp. 353–360. Springer, Berlin Heidelberg New York (2003)
8. Carranza, J., Theobalt, C., Magnor, M.A., Seidel, H.P.: Free-viewpoint video of human actors. In: Proceedings of SIGGRAPH 2003, pp. 569–577 (2003)
9. Caselles, V., Catté, F., Coll, T., Dibos, F.: A geometric model for active contours in image processing. Numer. Math. **66**, 1–31 (1993)
10. Chadwick, J., Haumann, D., Parent, R.: Layered construction for deformable animated characters. Comput. Graph. **23**(3), 243–252 (1989)
11. Chafri, H., Gagalowicz, A., Brun, R.: Determination of fabric viscosity parameters using iterative minimization. In: Gagalowicz, A., Philips, W. (eds.) Proceedings of Computer Analysis of Images and Patterns. Lecture Notes in Computer Science, vol. 3691, pp. 789–798. Springer, Berlin Heidelberg New York (2005)
12. Chetverikov, D., Stepanov, D., Krsek, P.: Robust Euclidean alignment of 3D point sets: The trimmed iterative closest point algorithm. Image Vis. Comput. **23**(3), 299–309 (2005)

13. Cheung, K., Baker, S., Kanade, T.: Shape-from-silhouette across time. Part ii. Applications to human modeling and markerless motion tracking. Comput. Vis. **63**(3), 225–245 (2005)
14. Dervieux, A., Thomasset, F.: A finite element method for the simulation of Rayleigh–Taylor instability. In: Rautman, R. (ed.) Approximation Methods for Navier–Stokes Problems. Lecture Notes in Mathematics, vol. 771, pp. 145–158. Springer, Berlin Heidelberg New York (1979)
15. Fua, P., Plänkers, R., Thalmann, D.: Tracking and modeling people in video sequences. Comput. Vis. Image Underst. **81**(3), 285–302 (2001)
16. Gavrilla, D.: The visual analysis of human movement: a survey. Comput. Vis. Image Underst. **73**(1), 82–92 (1999)
17. Haddon, J., Forsyth, D., Parks, D.: The appearance of clothing. http://http.cs.berkeley.edu/haddon/clothingshade.ps (2005)
18. Herda, L., Urtasun, R., Fua, P.: Implicit surface joint limits to constrain video-based motion capture. In: Pajdla, T., Matas, J. (eds.) Proceedings of the 8th European Conference on Computer Vision. Lecture Notes in Computer Science, vol. 3022, pp. 405–418. Springer, Prague (2004)
19. House, D., DeVaul, R., Breen, D.: Towards simulating cloth dynamics using interacting particles. Clothing Sci. Technol. **8**(3), 75–94 (1996)
20. Johnson, A.E., Kang, S.B.: Registration and integration of textured 3-D data. In: Proceedings of the International Conference on Recent Advances in 3-D Digital Imaging and Modeling, pp. 234–241. IEEE Computer Society (1997)
21. Jojic, N., Huang, T.: Estimating cloth draping parameters from range data. In: Proceedings of the International Workshop on Synthetic–Natural Hybrid Coding and 3-D Imaging, pp. 73–76. Greece (1997)
22. Magnenat-Thalmann, N., Seo, H., Cordier, F.: Automatic modeling of virtual humans and body clothing. Comput. Sci. Technol. **19**(5), 575–584 (2004)
23. Magnenat-Thalmann, N., Volino, P.: From early draping to haute cotoure models: 20 years of research. Vis. Comput. **21**, 506–519 (2005)
24. Mikic, I., Trivedi, M., Hunter, E., Cosman, P.: Human body model acquisition and tracking using voxel data. Comput. Vis. **53**(3), 199–223 (2003)
25. MoCap-Systems: Motion analysis, vicon, simi: marker based tracking systems. www.motionanalysis.com, www.vicon.com, www.simi.com/en/ (2005)
26. Moeslund, T., Granum, E.: A survey of computer vision based human motion capture. Comput. Vis. Image Underst. **81**(3), 231–268 (2001)
27. Murray, R., Li, Z., Sastry, S.: Mathematical Introduction to Robotic Manipulation. CRC Press, Baton Rouge (1994)
28. Osher, S., Sethian, J.A.: Fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulations. Comput. Phys. **79**, 12–49 (1988)
29. Pritchard, D., Heidrich, W.: Cloth motion capture. Eurographics **22**(3), 37–51 (2003)
30. Richards, J.: The measurement of human motion: a comparison of commercially available systems. Hum. Mov. Sci. **18**, 589–602 (1999)
31. Rosenhahn, B.: Pose estimation revisited. Technical Report TR-0308, Institute of Computer Science, University of Kiel, Germany (2003). Available at http://www.ks.informatik.uni-kiel.de
32. Rosenhahn, B., Kersting, U., Smith, A., Gurney, J., Brox, T., Klette, R.: A system for marker-less human motion estimation. In: Kropatsch, W.,Sablatnig, R., Hanbury, A. (eds.) Pattern Recognition, 27th DAGM-symposium. Lecture Notes in Computer Science, vol. 3663, pp. 230–237. Springer, Vienna (2005)
33. Rosenhahn, B., Sommer, G.: Pose estimation of free-form objects. In: Pajdla, T., Matas, J. (eds.) Computer Vision — Proceedings of the 8th European Conference on Computer Vision. Lecture Notes in Computer Science, vol. 3021, pp. 414–427. Springer, Berlin Heidelberg New York (2004)
34. Rusinkiewicz, S., Levoy, M.: Efficient variants of the ICP algorithm. In: Proceedings of the 3rd International Conference on 3-D Digital Imaging and Modeling, pp. 224–231 (2001)
35. Weil, J.: The synthesis of cloth objects. Comput. Graph. Proceedings of SigGraph **20**(4), 49–54 (1986)
36. Wikipedia: Motion capture. http://en.wikipedia.org/wiki/Motion_capture (2005)
37. You, L., Zhang, J.J.: Fast generation of 3d deformable moving surfaces. IEEE Trans. Syst. Man Cybern. B Cybern. **33**(4), 616–615 (2003)
38. Zhang, Z.: Iterative points matching for registration of free form curves and surfaces. Comput. Vis. **13**(2), 119–152 (1994)

Author's Biography

Dr. Bodo Rosenhahn gained his Ph.D. 2003 at the Institute of Computer Science, University Kiel, Germany. From 2003 to 2005 he was (DFG) PostDoc at the University of Auckland, New Zealand. Since November 2005 he is senior researcher at the Max-Planck center in Saarbücken, Germany. He is working on markerless motion capture, human model generation and animation, and image segmentation.

Dr. Uwe G. Kersting is a lecturer in biomechanics at the University of Auckland (New Zealand). He has been Director of the Biomechanics Laboratory at Tamaki Campus for the previous 2 years. He has published in various fields ranging from sports biomechanics to tissue adaptation. He has a broad experience in video based movement analysis techniques and is currently running several projects which require 3D motion capture.

Katie Powell currently studies physiotherapy at Boston University. During a research project she worked at the Gait Lab in Auckland and was involved in the Markerless MoCap project.

Dr. Reinhard Klette is professor of information technology in the department of computer science at The University of Auckland (New Zealand). His research interests are directed on theoretical and applied subjects in multimedia imaging and geometric algorithms. He has published about 300 journal and conference papers and books about image processing (with Piero Zamperoni), shape recovery based on visual information (with Karsten Schlüns and Andres Koschan) and digital geometry (with Azriel Rosenfeld).

Gisela Klette received her MS in Mathematics from Jena University, Germany. She worked at The University of Auckland at the Department of Computer Science in a combined teaching and research contract position. She supervised projects in biomedical image analysis and mathematical morphology. In 2006, she was a general member at IMA Minneapolis. Recently, she works on her Ph.D. in image analysis at the University of Groningen, The Netherlands.

Dr. Hans-Peter Seidel is the scientific director and chair of the computer graphics group at the Max-Planck-Institut (MPI) Informatik and a professor of computer science at the University of Saarbruecken, Germany.

Seidel has published some 200 technical papers in the field and has lectured widely on these topics. He has received grants from a wide range of organizations, including the German National Science Foundation (DFG), the German Federal Government (BMFBF), the European Community (EU), NATO and the German-Israel Foundation (GIF).

In 2003, Seidel was awarded the 'Leibniz Preis', the most prestigious German research award, from the German Research Foundation (DFG). Seidel is the first computer graphics researcher to receive this award. In 2004, he was selected as founding chair of the Eurographics Awards Programme.