Cloth X-Ray: MoCap of People Wearing Textiles *

B. Rosenhahn[1], U. G. Kersting[2], Katie Powell[2] and Hans-Peter Seidel[1]

[1] Max Planck Institute for Informatics, Stuhlsatzhausenweg 85,
D-66123 Saarbrücken, Germany
rosenhahn@mpi-inf.mpg.de
[2]Department of Sport and Exercise Science
The University of Auckland, New Zealand

Abstract. The contribution presents an approach for motion capturing (MoCap) of dressed people. A cloth draping method is embedded in a silhouette based MoCap system and an error functional is formalized to minimize image errors with respect to silhouettes, pose and kinematic chain parameters, the cloth draping components and external wind forces. We report on various experiments with two types of clothes, namely a skirt and a pair of shorts. Finally we compare the angles of the MoCap system with results from a commercially available marker based tracking system. The experiments show, that we are basically within the error range of marker based tracking systems, though body parts are occluded with cloth.

1 Introduction

Marker-less motion capturing is a highly challenging topic of research and many promising approaches exist to tackle the problem [12, 5, 1, 10, 4, 7]. In most setups it is required that the subjects have to wear either body suits, to be naked or at least to wear clothing which stresses the body contours (e.g. swim suits). Such clothing is often uncomfortable to wear in contrast to loose clothing (shirts or shorts). The analysis of outdoor sport events also requires to take clothing into account. On the other hand, cloth draping is a well established field of research in computer graphics and virtual clothing can be moved and rendered so that it blends seamlessly with motion and appearance in movie scenes [6, 8, 9, 17]. Existing approaches can be roughly divided in 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 [9]. Geometric approaches [17] model clothes by using other mechanics theories which are often determined empirically. These methods can be very fast computationally but are often criticized as being visually unappealing.

The motivation of this work is to combine a cloth draping algorithm with a marker-less MoCap system. The key idea is to use the appearance of the cloth and the visible parts of the human being to determine the underlying kinematic structure, though it might be heavily occluded.

* We gratefully acknowledge funding by the Max-Planck Center for visual computing and communication.

2 Foundations: Silhouette based MoCap

This work is an extension of a previously developed marker-less MoCap system [16]. In this system, the human being is represented in terms of free-form surface patches, joint indices are added to each surface node and the joint positions are assumed. This allows to generate arbitrary body configurations, steered through joint angles. The assumed corresponding counterparts in the images are 2D silhouettes: These are used to reconstruct 3D ray bundles and a spatial distance constraint is minimized to determine the position and orientation of the surface mesh and the joint angles. In this section we will give a brief summary of the MoCap system. These foundations are needed later to explain concisely, where and how the cloth draping approach is incorporated.

2.1 Silhouette extraction

Image segmentation usually means to estimate boundaries of objects in an image. It is an important step for data abstraction, but the task can become very difficult due to noise, shading, occlusion or texture transitions between the object and the background. Our approach is based on image segmentation based on level sets [3, 14, 2].

Fig. 1. Silhouette extraction based on level set functions. Left: Initial segmentation. Right: Segmentation result.

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. On a discrete image, the level set functions are modeled through a distance transform from the contour line to the inner and outer region with negative and positive distance values, respectively. Both regions are analyzed with respect to the probabilities of image features (e.g. gray value distributions, color or texture channels). Now the key idea is to evolve the contour line, to maximize the probability density functions with respect to each other. Furthermore, the boundary between both regions should be as small as possible. This can be expressed by adding a

smoothness term. Both parts lead to the following energy functional that is sought to be minimized:

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

where $\nu > 0$ is a weighting parameter and $H(s)$ is a regularized version of the Heaviside function, e.g. the error function. 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 and its standard deviation σ. Figure 1 shows on the left an example image with an initialization of the region as rectangle. The right image shows the estimated (stationary) contour after 50 iterations. As can be seen, the legs and the skirt are well extracted, but there are some deviations in the feet region, due to shadows. Such inaccuracies can be compensated through the pose estimation procedure.

2.2 Registration, Pose estimation

Assuming an extracted image contour and the silhouette of the projected surface mesh, the closest point correspondences between both contours are used to define a set of corresponding 3D lines and 3D points. Then a 3D point-line based pose estimation algorithm for kinematic chains is applied to minimize the spatial distance between both contours: For point based pose estimation each line is modeled as a 3D Plücker line $L_i = (n_i, m_i)$, with a (unit) direction n_i and moment m_i [13]. The 3D rigid motion is expressed as 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 $\xi \in se(3) = \{(v, \hat{\omega}) | v \in \mathbb{R}^3, \hat{\omega} \in so(3)\}$, with $so(3) = \{A \in \mathbb{R}^{3\times3} | A = -A^T\}$. 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 or screw symmetric matrix,

$$\theta\omega = \theta\begin{pmatrix}\omega_1\\ \omega_2\\ \omega_3\end{pmatrix}, \text{ with } \|\omega\|_2 = 1 \quad \text{or} \quad \theta\hat{\omega} = \theta\begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix}. \tag{2}$$

A twist ξ contains six parameters and can be scaled to $\theta\xi$ for 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 or as a 4×4 matrix,

$$\theta\xi = \theta(\omega_1, \omega_2, \omega_3, v_1, v_2, v_3)^T, \|\omega\|_2 = 1, \quad \theta\hat{\xi} = \theta\begin{pmatrix} 0 & -\omega_3 & \omega_2 & v_1 \\ \omega_3 & 0 & -\omega_1 & v_2 \\ -\omega_2 & \omega_1 & 0 & v_3 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \tag{3}$$

To reconstruct a group action $M \in SE(3)$ from a given twist, the exponential function $\exp(\theta\hat{\xi}) = \sum_{k=0}^{\infty} \frac{(\theta\hat{\xi})^k}{k!} = M \in SE(3)$ must be computed. This can be done efficiently by using the Rodriguez formula [13].

For pose estimation the reconstructed Plücker lines are combined with the screw representation for rigid motions: Incidence of the transformed 3D point X_i with the 3D ray $L_i = (n_i, m_i)$ can be expressed as

$$(\exp(\theta\hat{\xi})X_i)_{3\times 1} \times n_i - m_i = 0. \tag{4}$$

Since $\exp(\theta\hat{\xi})X_i$ is a 4D vector, the homogeneous component (which is 1) is neglected to evaluate the cross product with n_i. Then the equation is linearized and iterated, see [16].

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 is part of the model) and unknown joint angle θ_j. The constraint equation of an ith point on a jth joint has the form

$$(\exp(\theta_j\hat{\xi}_j)\ldots\exp(\theta_1\hat{\xi}_1)\exp(\theta\hat{\xi})X_i)_{3\times 1} \times n_i - m_i = 0 \tag{5}$$

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.

3 Kinematic cloth draping

For our set-up we decided to use a geometric approach to model cloth behavior. The main reason is that 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. A cloth draping algorithm in the area of seconds would require hours to calculate the pose of one frame and weeks for a whole sequence. We decided to model the skirt as a string-system with underlined

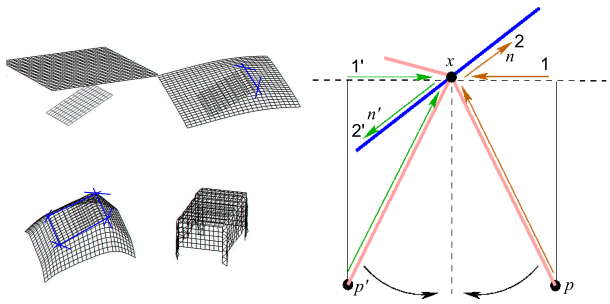

Fig. 2. The cloth draping principle. Joints are used to deform the cloth while draping on the surface mesh.

kinematic chains: The main principle is visualized on the left in Figure 2 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 point 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 on the left

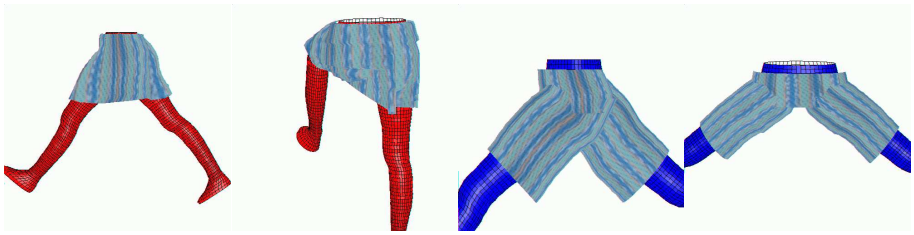

Fig. 3. Cloth draping of a skirt and shorts in a simulation environment.

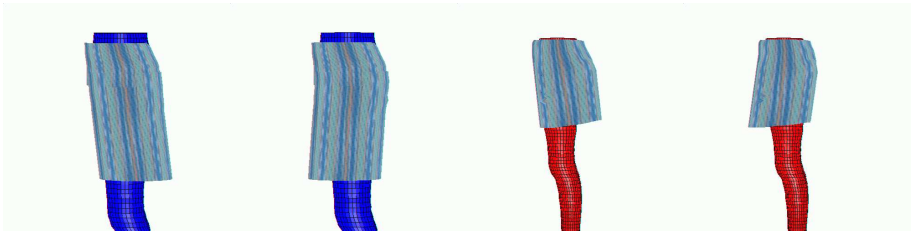

Fig. 4. Wind model on the shorts (left) and the skirt (right). Visualized is frontal and backward wind.

of Figure 2. Now the remaining points are not allowed to *fall* downwards any more. 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 Figure 2. The image on the right in Figure 2 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 around 90 degrees (2), leading to the rotation axis n. The point pair $(n, x \times n)$ are the components of the twist, see equation (3). 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 natural swinging draping pattern. The draping velocity is steered through a rotation velocity θ, which is set to 2 degrees during 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 on the left in Figure 2 show the ongoing draping and the final result.

Figure 3 shows example images of a skirt and a pair of shorts falling on the leg model. The skirt is modeled as a 2-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 second sampling dimension. For this reason, the skirt needs to be re-constrained after draping. If a stretching parameter is exceeded, the particles are re-constrained 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, especially for certain leg configurations. Figure 3 shows that even the creases are maintained. In this case, shorts are simpler since they are modeled as cylinders, transformed together with the legs and then draped.

To improve the dynamic behavior of clothing during movements, we also add a wind-model to the cloth draping. We continue with the cloth-draping in the following way: dependent on the direction of wind 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 4 shows examples of the shorts and skirt with frontal or backward wind. The wind force and direction are later part of the minimization function during pose tracking. 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 measurements of the cloth.

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 known joint positions along the legs. Furthermore we assume the wearing of a skirt

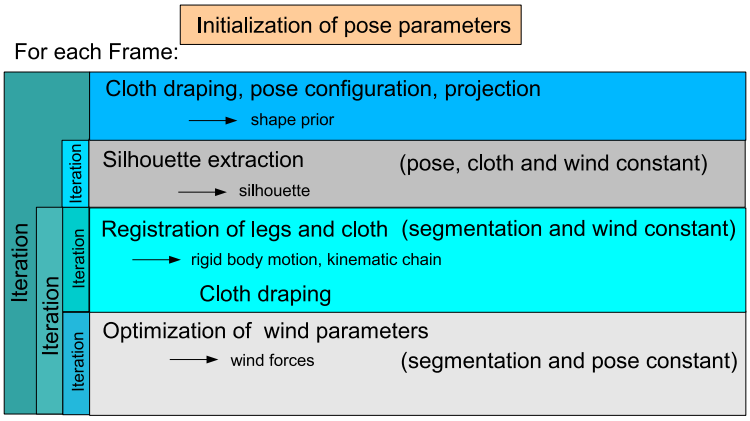

Fig. 5. The basic algorithm for combined cloth draping and motion capturing

or shorts with known measures. The person is walking or stepping in a four-camera setup. These cameras are triggered and calibrated with respect to one world coordinate system. 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 to 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:

$$E(\Phi, p_1, p_2, \theta\xi, \theta_1, \ldots, \theta_n, c, w) = \underbrace{-\int_\Omega \left(H(\Phi)\log p_1 + (1-H(\Phi))\log p_2 + \nu|\nabla H(\Phi)|\right) dx}_{\text{segmentation}}$$

$$\underbrace{+\lambda \int_\Omega (\Phi - \Phi_0(\underbrace{\theta\xi, \theta_1, \ldots, \theta_n}_{\text{pose and kinematic chain,}}, \underbrace{c, w}_{\text{wind parameters}}))dx}_{\text{shape error}}$$

Due to the large number of parameters and unknowns we decided for an iterative minimization scheme, see Figure 5: 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 (section 2.1). 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 (section 2.2). Finally, different wind directions and wind forces are sampled to refine the pose result (section 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.

5 Experiments

Fig. 6. Example sequences for tracking clothed people. **Top row**: walking, leg crossing, knee bending and knee pulling with a skirt. **Bottom row**: walking, leg crossing, knee bending and knee pulling with shorts. The pose is determined from 4 views (just one of the views is shown, images are cropped).

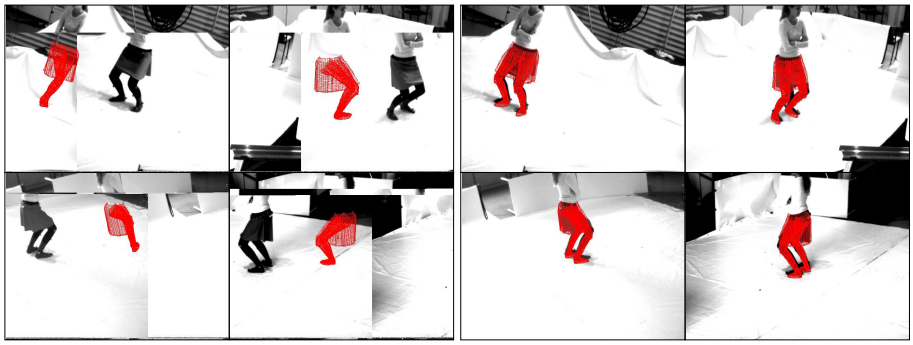

Fig. 7. Error during grabbing the images

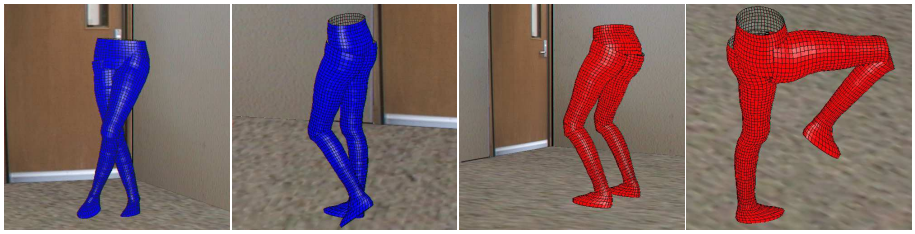

Fig. 8. Example leg configurations of the sequences. The examples are taken from the subject wearing the shorts (blue) and the skirt (red) (leg crossing, walking, knee bending, knee pulling).

For the experiments we used a four-camera set up and grabbed image sequences of the lower torso with different motion patterns: The subject was asked to wear the skirt and the shorts while performing walking, leg crossing and turning, knee bending and walking with knees pulled up. We decided on these different patterns, since they are not only of importance for medical studies (e.g. walking), but they are also challenging for the cloth simulator, since the cloth is partially stretched (knee pulling sequence) or hanging down loosely (knee bending). The turning and leg crossing sequence is interesting due to the higher occlusions. Figure 6 shows some pose examples for the subject wearing the skirt (top) and shorts (bottom). The pose is visualized by overlaying the projected surface mesh onto the images. Just one of the four cameras is shown. Each sequence consists of 150-240 frames. Figure 7 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 image on the right) the pose is correct. Figure 8 shows leg configurations in a virtual environment. The position of the body and the joints reveal a natural configuration.

Finally, the question about the stability arises. To answer this question, we attached markers to the subject and tracked the sequences simultaneously with the commercially available Motion Analysis system [11]. The markers are attached to the visible parts of the leg and are not disturbed by the cloth. We then compare joint angles for different sequences with the results of the marker based system, similar to [16]. The overall

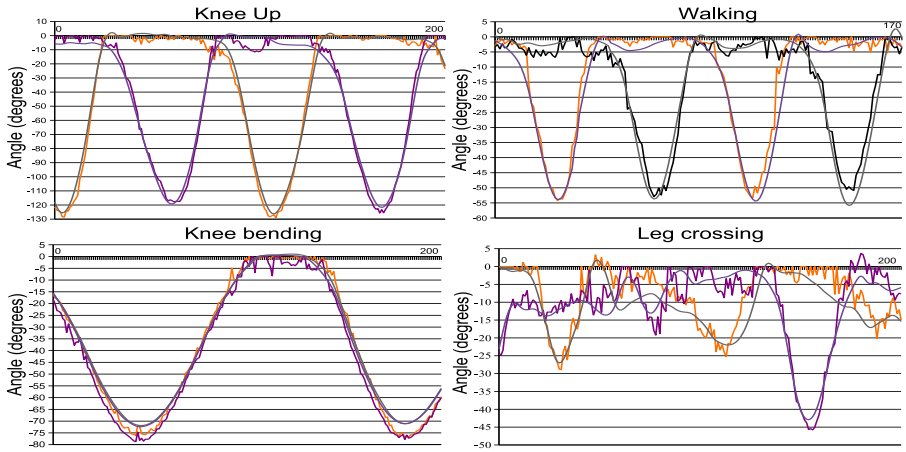

Fig. 9. Left: Knee angles from sequences wearing the shorts. **Right**: Knee angles from sequences wearing the skirt. **Top left**: Angles of the knee up sequence. **Bottom left**: Angles of the knee bending sequence. **Top right**: Angles of the walking sequence. **Bottom right**: Angles of the leg crossing sequence.

errors for both types of cloth varies between 1.5 and 4.5 degrees, which indicates a stable result.

The diagrams in Figure 9 shows the overlay of the knee angles for two skirt and two shorts sequences. Due to space limits, we just show four sequences, the remaining four are available upon request. The two systems can be identified by the smooth curves from the Motion Analyis system and unsmoothed curves (our system).

6 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 on 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 on an iterative solution. The experiments with a skirt and shorts show that the formulated problem can be solved. We are able to determine joint configurations and pose parameters of the kinematic chains, though they are considerably covered with clothes. Indeed, we use the cloth draping appearance in images to recover the joint configuration and simultaneously determine wind dynamics of the cloth. 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 [15].

For future works we plan to extend the cloth draping model with more advanced ones [9] and we will compare different draping approaches and parameter optimization schemes in the motion capturing setup.

References

1. C. Bregler, J. Malik, and K. Pullen. Twist based acquisition and tracking of animal and human kinetics. *International Journal of Computer Vision*, 56(3):179–194, 2004.
2. T. Brox, M. Rousson, R. Deriche, and J. Weickert. Unsupervised segmentation incorporating colour, texture, and motion. In N. Petkov and M. A. Westenberg, editors, *Proc. Computer Analysis of Images and Patterns*, volume 2756 of *Lecture Notes in Computer Science*, pages 353–360. Springer, Berlin, 2003.
3. A. Dervieux and F. Thomasset. A finite element method for the simulation of Rayleigh–Taylor instability. In R. Rautman, editor, *Approximation Methods for Navier–Stokes Problems*, volume 771 of *Lecture Notes in Mathematics*, pages 145–158. Springer, Berlin, 1979.
4. P. Fua, R. Plänkers, and D. Thalmann. Tracking and modeling people in video sequences. *Computer Vision and Image Understanding*, 81(3):285–302, March 2001.
5. D.M. Gavrilla. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73(1):82–92, 1999.
6. J. Haddon, D. Forsyth, and D. Parks. The appearance of clothing. http://http.cs.berkeley.edu/haddon/clothingshade.ps, June 2005.
7. L. Herda, R. Urtasun, and P. Fua. Implicit surface joint limits to constrain video-based motion capture. In T. Pajdla and J. Matas, editors, *Proc. 8th European Conference on Computer Vision*, volume 3022 of *Lecture Notes in Computer Science*, pages 405–418, Prague, May 2004. Springer.
8. D.H. House, R.W. DeVaul, and D.E. Breen. Towards simulating cloth dynamics using interacting particles. *Clothing Science and Technology*, 8(3):75–94, 1996.
9. N. Magnenat-Thalmann and P. Volino. From early draping to haute cotoure models: 20 years of research. *Visual Computing*, 21:506–519, 2005.
10. I. Mikic, M. Trivedi, E. Hunter, and P. Cosman. Human body model acquisition and tracking using voxel data. *International Journal of Computer Vision*, 53(3):199–223, 2003.
11. MoCap-System. Motion analysis: A marker based tracking system. www.motionanalysis.com, June 2005.
12. T.B. Moeslund and E. Granum. A survey of computer vision based human motion capture. *Computer Vision and Image Understanding*, 81(3):231–268, 2001.
13. R.M. Murray, Z. Li, and S.S. Sastry. *Mathematical Introduction to Robotic Manipulation.* CRC Press, Baton Rouge, 1994.
14. S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations. *Computational Physics*, 79:12–49, 1988.
15. J. Richards. The measurement of human motion: A comparison of commercially available systems. *Human Movement Science*, 18:589–602, 1999.
16. B. Rosenhahn, U. Kersting, A. Smith, J. Gurney, T. Brox, and R. Klette. A system for marker-less human motion estimation. In W. Kropatsch, R. Sablatnig, and A. Hanbury, editors, *Pattern Recognition, 27th DAGM-symposium*, volume 3663 of *Lecture Notes in Computer Science*, pages 230–237, Vienna, Austria, September 2005. Springer.
17. J. Weil. The synthesis of cloth objects. *Computer Graphics (Proc. SigGraph)*, 20(4):49–54, 1986.