Dealing with Self-occlusion in Region Based Motion Capture by Means of Internal Regions*

Christian Schmaltz[1], Bodo Rosenhahn[2], Thomas Brox[3], Joachim Weickert[1],
Lennart Wietzke[4], and Gerald Sommer[4]

[1] Mathematical Image Analysis Group, Faculty of Mathematics and Computer Science,
Building E1.1, Saarland University, 66041 Saarbrücken, Germany
{schmaltz,weickert}@mia.uni-saarland.de
[2] Max-Planck Institute for Informatics, 66123 Saarbrücken, Germany
rosenhahn@mpi-sb.mpg.de
[3] Faculty of Computer Science, Dresden University of Technology, 01187 Dresden, Germany
brox@inf.tu-dresden.de
[4] Institute of Computer Science, Christian-Albrecht-University, 24098 Kiel, Germany
{lw,gs}@ks.informatik.uni-kiel.de

Abstract. Self-occlusion is a common problem in silhouette based motion capture, which often results in ambiguous pose configurations. In most works this is compensated by a priori knowledge about the motion or the scene, or by the use of multiple cameras. Here we suggest to overcome this problem by splitting the surface model of the object and tracking the silhouette of each part rather than the whole object. The splitting can be done automatically by comparing the appearance of the different parts with the Jensen-Shannon divergence. Tracking is then achieved by maximizing the appearance differences of all involved parts and the background simultaneously via gradient descent. We demonstrate the improvements with tracking results from simulated and real world scenes.

Keywords: Pose estimation, tracking, kinematic chain, computer vision, human motion analysis.

1 Introduction

Capturing the motion of articulated objects, particularly humans, has been a popular research field for many years. Hundreds of papers have addressed this problem and we refer to [3], [6] and [7] for surveys on this topic.

Generally, pose tracking algorithms can be divided into 2-D approaches, which only track objects in the image plane, and 3-D approaches, which determine the object's pose, i.e., its 3-D position and orientation. Moreover, tracking methods can be classified by means of the tracked features. Two popular features are the object silhouette and local descriptors centered around feature points.

A major drawback of silhouette based 3-D pose tracking, particularly in case of articulated objects, is the problem of self-occlusion. This is, only the silhouette of parts of

* We acknowledge funding by the German Research Foundation under the projects We 2602/5-1 and SO 320/4-2, and by the Max-Planck Center for Visual Computing and Communication.

F.J. Perales and R.B. Fisher (Eds.): AMDO 2008, LNCS 5098, pp. 102–111, 2008.

the model is seen, leaving ambiguities in the pose of the remaining parts. For instance, if a person is seen from the front with a hand in front of its body, the contour of the hand and forearm is inside the object region and, hence, not part of the person's silhouette (see left image in Figure 4). As a consequence, there is no information to estimate the joint angles of the hand.

Approaches based on local patches do not suffer from these problems. They have other drawbacks, though. For example, feature detection might fail or produce too few features if there is not enough texture. Furthermore, as long as there is no matching to a reference frame, these features tend to introduce a drift. If such a matching is used, handling large rotations becomes difficult. Since neither method is clearly better than the other, both feature-based and silhouette-based approaches are still topics of open research. Here we focus on a silhouette-based approach that simultaneously computes the 3-D pose and the 2-D object contours seen in the images [8,9].

A common way to deal with ambiguities is by means of learned angle priors [12,15,1]. Based on the correlation of angles in the training samples, unresolved degrees of freedom are set to the most likely solution given the other, non-ambiguous angles. While this approach is reasonable and a last resort in case of body parts that are indeed occluded, the prior also has the tendency to introduce a bias towards the training motions, which is undesirable, e.g., in medical applications. Hence, it is beneficial to fully exploit the information provided by the images.

In this paper, we show how the information of internal silhouettes can be exploited. The main idea is to find components of the object model whose appearance differs from the surrounding model parts. Due to the difference in their appearance, the contours of these components can be extracted in the image and used as additional information for tracking. As a consequence, the tracking algorithm becomes more stable and can successfully track scenes that cannot be tracked with a conventional silhouette based method.

Related work. There are other human tracking approaches that decompose the model into several parts. Bottom-up approaches that learn the appearance of different parts from a training set are very common. The algorithm in [14] learns the appearance of each part modeled by a Gibbs distribution. Results are only given for multi-camera sequences, though. In [5] the appearance of each part is learned using AdaBoost. Another learning approach that considers different appearances of different object regions is explained in [17]. However, there is only 2-D tracking in these two approaches. In [2], average pixel intensities are computed inside parts of the object to estimate their appearance. This can be regarded as a parametric special case of the more general probability density functions we use for modeling the appearance of body parts. The tracking of multiple object parts also comprises similar ideas as the tracking of multiple objects in a scene, as proposed in [10].

Paper organization. In Section 2 we review a basic region based pose estimation algorithms. A new energy function for tracking with internal regions is introduced in Section 3, followed by an explanation how the internal regions used in this new approach can be found automatically. After showing and discussing some experiments in Section 4, we conclude with a summary in Section 5.

2 Pose Estimation from 2-D–3-D Point Correspondences

In this paper, we model humans as free-form surfaces embedded with kinematic chains, i.e., as a number of rigid body parts connected by joints and represented in a tree structure [6]. The n rotation axes ξ_i are part of the model. The joint angles $\Theta := (\theta_1, \dots \theta_n)$ are unknown and must be determined by the pose estimation algorithm. In total, the sought pose vector $\chi := (\xi, \Theta)$ consists of a 6-D vector $\xi \in se(3)$ corresponding to the rigid body motion of the whole model and the above-mentioned joint angle vector Θ.

We pursue a region-based tracking approach that is based on the work in [9]. It splits the image into two regions in such a way that the features in the foreground and background region are maximally different. In this sense, the approach has a lot in common with segmentation. However, instead of using a segmentation method as an intermediate step, we directly manipulate the pose parameters in order to optimize the partitioning. To this end, we consider the partitioning function $P : \mathbb{R}^{6+n} \times \Omega \ni (\chi, q) \mapsto \{0, 1\}$. It projects the body model with its current pose χ to the image plane Ω in order to determine if an image point q currently belongs to the object region $\{q \in \Omega | P(\chi, q) = 1\}$. The partitioning, and simultaneously the pose, are improved by minimizing the energy function

$$E(\chi) = -\int_{\Omega} \big(P(\chi, q) \log p_{in} + (1 - P(\chi, q)) \log p_{out}\big) dq \tag{1}$$

with a modified gradient descent. Here, p_{in} and p_{out} denote two probability density functions (pdfs) that represent the feature distribution in the object and background region, respectively. We use the three channels of the CIELAB color space but, in principle, any dense feature set can be used. The densities p_{in} and p_{out} are modeled by independent channel densities. We estimate each channel density either by a kernel density estimator or by a local Gaussian distribution [8]. It is worth noting that the estimated pdfs depend on the partitioning. Thus, they have to be recomputed when χ varies.

For approximating the gradient of E, we assume that $\nabla_\chi p_{in} \approx 0, \nabla_\chi p_{out} \approx 0$. These are reasonable assumptions, since the estimated pdfs only change slowly with varying χ. Furthermore, we assume that P was smoothed, e.g., by convolving it with a small Gaussian kernel. We obtain

$$\nabla E(\chi) = -\int_{\Omega} (\nabla P(\chi, q)(\log p_{in} - \log p_{out}))\, dq\ . \tag{2}$$

Thus, a modified gradient descent can be employed for minimizing (1). More precisely, each point on the contour of the projected model is moved either towards the interior or exterior of the object region depending on whether p_{in} or p_{out} is larger, respectively. This is illustrated in Figure 1. The displacement of contour points is transferred to the corresponding 3-D points on the surface model by using a point-based pose estimation algorithm [8]. In this way, a new rigid body motion and the joint angles are estimated and projecting the model with the new pose yields a refined partitioning. These steps are iterated until convergence.

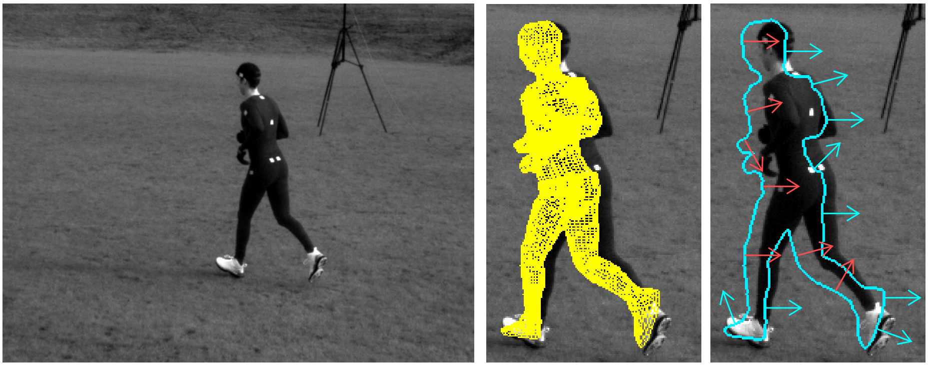

Fig. 1. This figure illustrates the movement applied to contour points by the tracking algorithm used. **Left:** Example input image of a scene. **Middle:** Object model projected in an inaccurate pose into the image plane (magnified). **Right:** Silhouette of the projected model and an example how some contour points might be adapted by the tracking algorithm (magnified). Cyan arrows indicate points that are moved towards the outside and red arrows indicates a movements towards the inside of the object.

3 Tracking Using a Split Model

The tracking algorithm explained so far works very well for rigid objects. However, in case of articulated objects, ambiguities may occur if the projection of a body part lies completely inside the object region and, consequently, yields not silhouette points. In such a situation, there is no cue for the pose of that part.

In this section, we explain how to overcome this problem by using internal silhouettes. To this end, the object model is split into different components and each of these components is projected separately to the image plane. We assume that there are some body parts that look different from other parts. This is reasonable since tracking cannot work if the structure to be tracked looks like the surrounding background. Even a human cannot follow an object that looks like the background after all.

3.1 Extending the Energy Function to Multiple Regions

Assume there are l model components $M_i, i = 1, \dots, l$ to track the body model M. These components can be chosen arbitrarily, e.g., some points of the model might be part of several components M_i, or of no component at all. This can be useful if a part of the object looks similar to the background, e.g., someone wearing a black T-shirt in front of a black wall. A component does not need to be connected, e.g., both arms may be handled as a single component.

Before introducing an energy function that can deal with such a multiple component model, we need to define some symbols: let $O_i(\chi,q)$ be the set of all 3-D points of the model component M_i with pose χ that are projected to the image point q. Furthermore, for the usual Euclidean metric d, let $d_i(\chi,q) := d(O_i(\chi,q),C) = \min_{x \in O_i(\chi,q)} \{d(x,C)\}$ be the minimal distance of the camera origin C to a 3-D point in the set $O_i(\chi,q)$. Finally, we define visibility functions $v_i : \chi \times \omega \mapsto \{0,1\}$ which are 1 if and only if the i-th object is visible in the image point q, given the current pose, i.e.,

$$v_i(\chi,q) := \begin{cases} 1 & \text{if } d_i(\chi,q) = \min_{j\in\{1,\dots,l\}}\{d_j(\chi,q)\}\ , \\ 0 & \text{else} \end{cases} . \tag{3}$$

These visibility functions are similar to those used in [10] for tracking multiple objects. However, that approach used different pdfs for the inside and outside region of each object, resulting in a total of $2k$ pdfs when tracking k objects. Here, we model each region M_i with a single pdf and one common pdf p_0 representing the background region. This yields a total of only $l+1$ pdfs. After defining the necessary functions for the background as $v_0(\chi,q) := \prod_{i=1}^{l}(1 - v_i(\chi,q))$ (the background is visible if no other object can be seen) and $P_0(\chi,q) := 1$ (ignoring visibility, the background covers the whole image), we can write the new energy function in a compact way:

$$E(\chi) = -\int_\Omega \sum_{i=0}^{l} \left[v_i(\chi,q)P_i(\chi,q)\log p_i\right] dq. \tag{4}$$

Note the difference between the energies (1) and (4): In (1) the pdfs came in pairs, i.e., only the distributions of foreground and background have been distinguished. Although that model can handle multiple colors (or other features) per region, the spatial content in each region is not available.

In contrast, a separate pdf per region is employed in (4). Since the proposed algorithm partitions the image into more regions, the generated pdfs are more accurate. Moreover, the distributions are separated spatially. This allows to track body parts that lie completely inside the object region.

3.2 Minimization

The minimization of (4) works in a similar way to earlier approaches. However, there are two important differences. Firstly, it is necessary to distinguish the different components M_i. Secondly, it is no longer possible to directly compare the pdfs of the interior and the exterior, since there is no pdf of the exterior of an object anymore.

The first step of the minimization is to project all object components M_i into the image plane to determine the image regions they occupy. To this end, occlusions have to be handled correctly. The visibility of points given the current pose can be computed with openGL [11]. Once it is clear in which image regions the object components M_i are visible, probability density functions for the interior of each component are estimated. Moreover, a single probability density function for the background is estimated from the remainder of the image domain.

After projecting the object components M_i, the 3-D points projected onto the 2-D silhouettes of each component M_i are used as new 2-D–3-D point correspondences. Similar to the basic algorithm, the 2-D parts of those points will be moved toward the interior or the exterior of the projected object component. To decide which of these two directions is appropriate, we evaluate the two pdfs next to that point.

That is, if the pdfs indicate that a 2-D point fits better to the neighboring component, the point is shifted in contour normal direction. These new correspondences are processed in the same way as the points from the original algorithm.

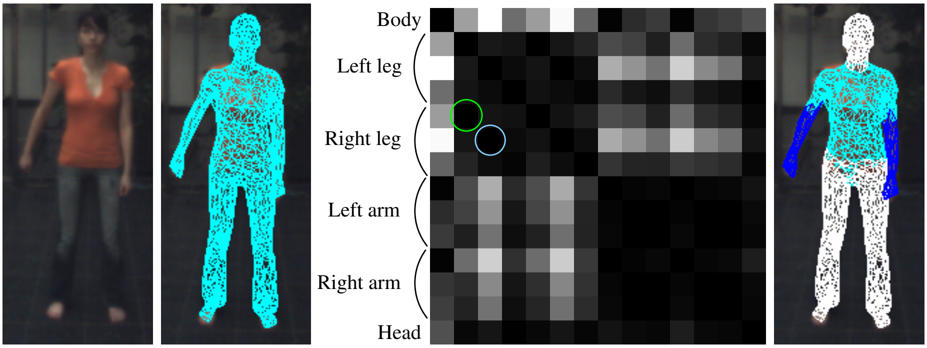

Fig. 2. This figure shows a result for the automatic splitting explained in Section 3.3. **Leftmost:** Input image (cropped). See Figure 4 to see the size of the whole image. **Left:** Object projected into the image in the initial pose. **Right:** Visualization of the similarity matrix computed for the first step of the automatic splitting algorithm. The green circle indicate the first two regions merged (the upper legs) and the blue circle the second pair of regions merged (the lower legs). **Rightmost:** Splitting suggested by the algorithm for a splitting threshold α in the interval $[0.18, 0.4]$.

3.3 Automatic Object Splitting

In order to perform an automatic splitting of kinematic chains, we assume that those parts with similar appearance should be in the same component M_i. Thus, we start by setting each M_i to a single segment of the kinematic chain. For the human model shown in Figure 2, this results in 14 different components, i.e., head, torso, three for each arm (upper, lower, and hand) and three for each leg (upper, lower, foot) (see the left image in Figure 3). Then, pdfs are estimated for every component.

Next, we search the two pdfs with minimal distance. However, there are numerous distances for probability density functions defined in the literature. We tested several of those distances, e.g., minimizing the sum of the squared differences, or the Kullback-Leibler difference [4], and found the Jensen-Shannon divergence [16] to give the best results for our problem.

Given two pdfs p and q, the Jensen-Shannon divergence, which is a smoothed and symmetrized version of the Kullback-Leibler divergence, is defined as

$$JSD(p,q) := \frac{J(p,M) + J(q,M)}{2}, \tag{5}$$

where $M = \frac{p+q}{2}$and where J is the Kullback-Leibler divergence

$$J(p,q) := \sum_i p(i) \log \frac{p(i)}{q(i)}. \tag{6}$$

The components M_a and M_b whose pdfs a and b have the smallest distance $JSD(a,b)$ are merged to a new component. This merging step is repeated until the Jensen-Shannon divergence of all pairs of regions is bigger than a splitting threshold α. For the example

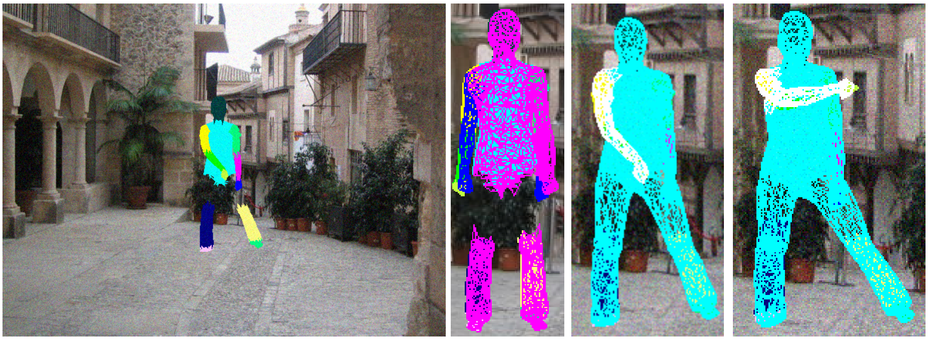

Fig. 3. Simulation of a human that moves an arm in front of its body. Here, every part of the object model was assigned a unique color and projected onto an image of a street in Palma de Mallorca. The only exception are the upper legs, which are transparent to simulate parts on the object that cannot be distinguished from the background. **From left to right**: (a) Frame 20 of the input sequence. (b) Splitting used in the initial pose in the first frame (magnified). The two components are shown in blue and magenta, respectively. (c), (d) Pose estimation results for frame 20 and 30 (magnified).

shown in Figure 2, this results in the three components also shown in that figure. Furthermore, we show an image which encodes the similarities between the different parts of the model (see left image in Figure 3). The brighter the dot, the large the divergence.

It is also possible to include the background as an additional part M_0 in the initial splitting. Every part of the model that is merged with M_0 is considered as similar to the background, and is therefore not assigned to any model part.

An interesting result of the proposed automatic splitting is that it does not include the upper arms into the same region as the lower arms. This differs from a manual splitting we have tested previously. Since both the torso and the upper parts of the arms are partly orange and partly have the color of the skin, the splitting computed automatically is to be preferred.

4 Experiments

We have tested the proposed algorithm in one synthetic environment and two real-world scenes. Figure 3 shows a synthetic image in which every joint of a human model – except the upper legs, which have intentionally been omitted – was drawn in a different color onto a cluttered background. Additionally, uncorrelated Gaussian noise with standard-deviation 15 was added after projecting to prevent the sequence from being too easy. In this monocular scene, the model moves one arm in front of its body. Consequently, we used two components M_i: One with the moving arm – shown in dark blue in the second image in Figure 3 – and the other with the remainder of the body except the upper legs, which are shown in magenta in that image. The model has 30 degrees of freedom.

Despite the above-mentioned challenges (cluttered background, only one view available, upper legs indistinguishable from the background), all thirty frames are easily tracked with the proposed approach. This is because most of the body is clearly distinguishable from the background and the surrounding body parts. The region of the upper legs, on the other hand, are simply ignored by the tracking algorithm since the upper legs do not belong to any component M_i. Tracking results are shown in Figure 3.

The challenges we created synthetically in the simulation can also appear in real world scenes, as shown in Figure 4. Again, we used one camera view and have to deal with a cluttered background. Also the appearance of the legs is close to that of the background region. As in the simulation, the right lower arm and the hand are completely inside the object region in some frames. Therefore, it is impossible to track this sequence with the basic algorithm explained in Section 2. In contrast, the tracking with multiple components works well.

In another scenario, we tested our algorithm using a sequence from the HumanEva-II database [13]. This database consists of several calibrated videos of humans performing different movements and provides background images and a surface model. These image sequences can be used for benchmarking since it is possible to automatically evaluate tracking results. This is done by using an online interface provided at Brown University which returns the tracking error in millimeter for every frame.

The automatic splitting computed with our algorithm when using a splitting threshold α between 0.12 and 0.31 is nearly identical to the splitting proposed for the sequence with the arm movement presented above. The only difference is that the head was assigned to a different component. The splitting, sample images, and tracking results are shown in Figure 5. A comparison of the proposed model to the basic one, in which the body consists of a single component, reveals that the left arm is tracked much more accurately due to the splitting introduced. This is also reflected by the results of the quantitative comparison. It is worth noting that the good quantitative results have been obtained without exploiting learned a-priori knowledge of probable motion patterns.

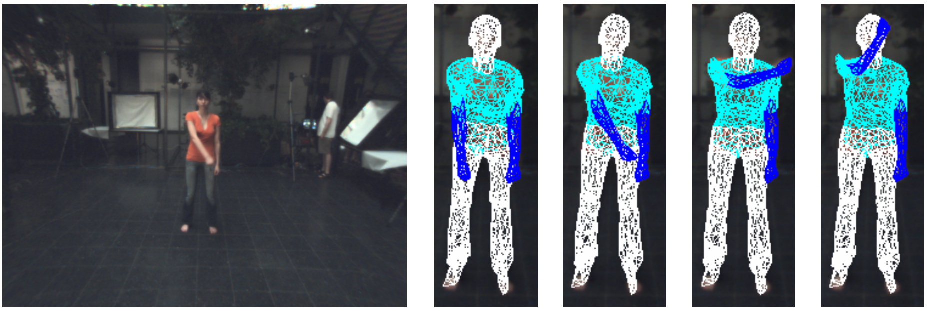

Fig. 4. Here, we tested our algorithm on a monocular real-world sequence of a woman that moves one of her arms in front of her body. The input image (left) was brightened for this paper (but not for pose tracking) in order to improve its visibility. See the left image in Figure 2 to get a feeling about the brightness of the unmodified images. **From left to right:** Input image of frame 38, and pose estimation results for frames 30,38,50, and 70 for the only view used (magnified). The different colors of the model indicate the internal regions used.

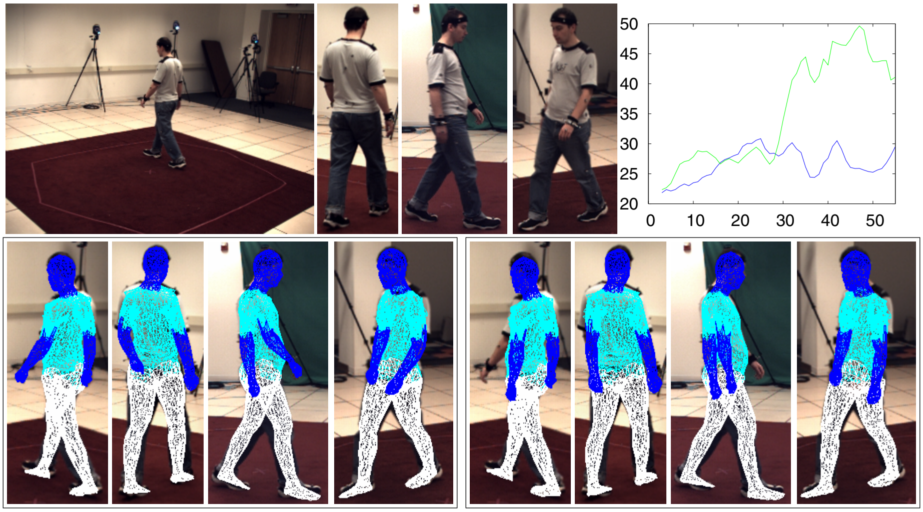

Fig. 5. This comparison shows tracking results with and without the improvements proposed when tracking a sequence from the HumanEva-II database [13]. **Top row, from left to right:** The four views available in frame 50 (three have been magnified) and the tracking error in millimeter plotted against the frame number with (blue) and without (green) using internal regions. **Bottom row:** Pose estimation result for frame 50, projected to the input images with (left) and without (right) using multiple regions. It can be seen that the left arm was only tracked correctly when using internal regions.

5 Summary

In this paper we dealt with a silhouette-based pose tracking technique. In particular, we showed how typical ambiguities of silhouette-based methods caused by self-occlusion can be avoided by splitting the model into multiple components. We presented an energy minimization formulation of the problem, where the appearance of the separate components is modeled by probability density functions and the components interact in order to determine the optimum pose. Moreover, we presented a way to automatically split a given body model by means of the Jensen-Shannon divergence. The experimental evaluation revealed significantly improved results in synthetic as well as real world scenes.

References

1. Brox, T., Rosenhahn, B., Cremers, D., Seidel, H.-P.: Nonparametric density estimation with adaptive anisotropic kernels for human motion tracking. In: Elgammal, A., Rosenhahn, B., Klette, R. (eds.) Human Motion 2007. LNCS, vol. 4814, pp. 152–165. Springer, Heidelberg (2007)
2. Fossati, A., Dimitrijevic, M., Lepetit, V., Fua, P.: Bridging the gap between detection and tracking for 3D monocular video-based motion capture. In: Proc. 2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Minneapolis, MI, June 2007, pp. 1–8. IEEE Computer Society Press, Los Alamitos (2007)

3. Gavrila, D.M.: The visual analysis of human movement: a survey. Computer Vision and Image Understanding 73(1), 82–98 (1999)
4. Kullback, S., Leibler, R.A.: On information and sufficiency. Annals of Mathematical Statistics 22, 79–86 (1951)
5. Micilotta, A., Ong, E., Bowden, R.: Detection and tracking of humans by probabilistic body part assembly. In: Proceedings of the British Machine Vision Conference (BMVC 2005), Oxford, UK, September 2005, pp. 429–438 (2005)
6. Moeslund, T.B., Hilton, A., Krüger, V.: A survey of advances in vision-based human motion capture and analysis. International Journal of Computer Vision 104(2), 90–126 (2006)
7. Poppe, R.: Vision-based human motion analysis: An overview. Computer Vision and Image Understanding 108(1-2), 4–18 (2007)
8. Rosenhahn, B., Brox, T., Weickert, J.: Three-dimensional shape knowledge for joint image segmentation and pose tracking. International Journal of Computer Vision 73(3), 243–262 (2007)
9. Schmaltz, C., Rosenhahn, B., Brox, T., Cremers, D., Weickert, J., Wietzke, L., Sommer, G.: Region-based pose tracking. In: Martí, J., Benedí, J.M., Mendonça, A.M., Serrat, J. (eds.) IbPRIA 2007. LNCS, vol. 4478, pp. 56–63. Springer, Heidelberg (2007)
10. Schmaltz, C., Rosenhahn, B., Brox, T., Weickert, J., Cremers, D., Wietzke, L., Sommer, G.: Occlusion modeling by tracking multiple objects. In: Hambrecht, F., Schnörr, C., Jähne, B. (eds.) DAGM 2007. LNCS, vol. 4713, pp. 173–183. Springer, Heidelberg (2007)
11. Shreiner, D., Woo, M., Neider, J., Davis, T.: OpenGL programming guide, 5th edn. Addison-Wesley, Upper Saddle River (2006)
12. Sidenbladh, H., Black, M.J., Sigal, L.: Implicit probabilistic models of human motion for synthesis and tracking. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2350, pp. 784–800. Springer, Heidelberg (2002)
13. Sigal, L., Black, M.J.: Humaneva: Synchronized video and motion capture dataset for evaluation of articulated motion. Technical Report CS-06-08, Department of Computer Science, Brown University (September 2006)
14. Sigal, L., Sidharth, B., Roth, S., Black, M., Isard, M.: Tracking loose-limbed people. In: Proc. 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 2004, vol. 1, pp. 421–428. IEEE Computer Society Press, Los Alamitos (2004)
15. Urtasun, R., Fleet, D., Fua, P.: 3D people tracking with gaussian process dynamical models. In: Proc. 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, New York, September 2006, pp. 238–245. IEEE Computer Society Press, Los Alamitos (2006)
16. Wong, A.K.C., You, M.: Entropy and distance of random graphs with application of structural pattern recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 7(5), 599–609 (1985)
17. Zhang, J., Collins, R., Liu, Y.: Bayesian body localization using mixture of nonlinear shape models. In: Proc. Tenth International Conference on Computer Vision, Beijing, China, October 2005, pp. 725–732. IEEE Computer Society Press, Los Alamitos (2005)