Staying Well Grounded in Markerless Motion Capture

Bodo Rosenhahn[1], Christian Schmaltz[2], Thomas Brox[3],
Joachim Weickert[2], and Hans-Peter Seidel[1]

[1] Max Planck Institute for Computer Science, Saarbrücken, Germany
rosenhahn@mpi-inf.mpg.de
[2] Mathematical Image Analysis, Saarland University, Germany
[3] Intelligent Systems, University of Dresden, Germany

Abstract. In order to overcome typical problems in markerless motion capture from video, such as ambiguities, noise, and occlusions, many techniques reduce the high dimensional search space by integration of prior information about the movement pattern or scene. In this work, we present an approach in which geometric prior information about the floor location is integrated in the pose tracking process. We penalize poses in which body parts intersect the ground plane by employing soft constraints in the pose estimation framework. Experiments with rigid objects and the HumanEVA-II benchmark show that tracking is remarkably stabilized.

1 Introduction

Markerless motion capture (MoCap) is an attractive field of research with applications in computer graphics (games, animation), sports science or medicine. Given an input image sequence of a moving person, the task is to determine its position and orientation, as well as the joint angles of the subject. Since there are no artificial markers as in marker based tracking systems, one usually assumes a 3D model of the person to be given. A popular representation is a surface model with known joint positions [8]. Multi-view image streams, i.e., images from synchronized and calibrated cameras, are very common to reduce ambiguities. One of the main problems in motion capture is the high dimensional search space, e.g. six pose parameters and at least 20 joint angles. For this reason it is beneficial to apply techniques that reduce the search space. Many techniques include a learning stage by using previously recorded MoCap data as prior information. These techniques comprise dimensionality reduction, density estimation, or regression techniques [1,14,5]. In [3] further prior knowledge about light sources and shadows is applied to exploit more information about the scene observed by the cameras and to introduce virtual *shadow cameras*. A physically motivated model for tracking the lower body has been proposed in [6]. Moreover, it is also possible to impose fixed joint angle limits, as suggested in [7].

The present work pursues the same basic idea of integrating prior information to stabilize tracking. The main contribution is to exploit prior information about the ground floor in the tracking process. The approach is motivated from the fact that many scenarios involve interaction of the tracked subject with the ground plane and indeed it is physically impossible that a body part (e.g. a foot) intersects the ground plane. The

G. Rigoll (Ed.): DAGM 2008, LNCS 5096, pp. 385–395, 2008.

present paper shows how to integrate such constraints in a markerless motion capture system. Moreover, it is verified to which extent such constraints improve the tracking performance. In particular, we show results on the recent HumanEVA-II [12] benchmark, which involves a quantitative error analysis.

2 The Motion Capture System

In this section we briefly introduce the markerless motion capture system we use in our experiments. It is a silhouette based system similar to [10]. However, in contrast to explicitly compute a silhouette using level-set functions we rely on the shape of the model in the image to separate the inner and outer regions. This system has been introduced in [11] and will be briefly summarized in the following subsections. The notation introduced here is also the basis for deriving the plane constraint later in Section 3.

2.1 Kinematic Chains

Our models of articulated objects, e.g. humans, are represented in terms of free-form surfaces with embedded kinematic chains. A kinematic chain is modeled as the consecutive evaluation of exponential functions, and twists ξ_i are used to model (known) joint locations [9]. In this work we denote a twist ξ as vector $\xi = (\omega_1, \omega_2, \omega_3, v_1, v_2, v_3)^T = (\omega, v)^T$ or as matrix $\hat{\xi}$ as done in [9]. It is further common to scale the twists ξ with respect to $\omega = (\omega_1, \omega_2, \omega_3)^T$, i.e. $\theta = \|\omega\|$, $\omega := \frac{\omega}{\theta}$ and $v := \frac{v}{\theta}$. We denote a scaled twist as $\theta\xi$ or $\theta\hat{\xi}$, respectively. The exponential function of a twist leads to a rigid body motion in space. The transformation of a mesh point of the surface model is given as the consecutive application of the local rigid body motions involved in the motion of a certain limb:

$$X_i' = \exp(\theta\hat{\xi})(\exp(\theta_1\hat{\xi}_1)\ldots\exp(\theta_n\hat{\xi}_n))X_i. \tag{1}$$

For abbreviation, we note a pose configuration by the $(6+n)$-D vector $\chi = (\xi, \theta_1, \ldots, \theta_n) = (\xi, \Theta)$ consisting of the 6 degrees of freedom for the rigid body motion ξ and the nD vector Θ comprising the joint angles. In the MoCap-setup, the vector χ is unknown and has to be determined from the image data.

2.2 Region Based Pose Tracking

Given an articulated surface model and an image stream, we are interested in fitting the mesh to the image data by computing the position, orientation, and joint angle configuration in such a way that the surface projection optimally covers the region of the person in the images. Many existing approaches [8] rely on previously extracted silhouettes, for instance by background subtraction, and establish correspondences between contour points and points on the model surface. This involves a contour extraction and a matching of the projected surface with the contour. The work in [11] avoids explicit computations of contours and contour matching but directly adapts the pose parameters χ such that the projections of the surface optimally split all images into homogeneously distributed object and background regions.

Energy model. Similar to a segmentation technique [10], the model states the conditions for an optimal partitioning of the image domain Ω. This can be expressed as minimization of the energy function

$$E(\chi) = -\int_{\Omega} \big(P(\chi,q)\log p_1 + (1-P(\chi,q))\log p_2\big)dq \ , \tag{2}$$

where the function $P : \mathbb{R}^{6+n} \times \Omega \ni (\chi,q) \mapsto \{0,1\}$ is 1 if and only if the surface of the 3-D model with pose χ projects to the point q in the image plane. P splits the image domain into two parts, the object and background region. In both regions the homogeneity of the feature distributions is to be maximized. These distributions are modeled by probability density functions (pdf) p_1 and p_2. In order to model the image features by pdfs we use the color distributions in the CIELAB color space.

Minimization. To minimize $E(\chi)$, a gradient descent can be applied. It results in the desired pose that minimizes $E(\chi)$ locally. In order to approximate the gradient of $E(\chi)$, 2D-3D point correspondences (q_i,x_i) are established by projecting silhouette points x_i of the surface with current pose χ to the image plane where they yield points q_i. Each image point q_i obtained in this way which seems to belong to the object region – i.e. those points for which $p_1(q_i)$ is greater than $p_2(q_i)$ – will be moved in outward normal direction to a new point q_i'. Points where $p_1(q_i) < p_2(q_i)$ holds will be shifted into the opposite direction to q_i', respectively. In order to compute the normal direction ∇P, we use Sobel operators. The length $l := \|q_i - q_i'\|$ of the shift vector is set to a constant. More details are given in [11].

The 2D-3D point correspondences (q_i',x_i) obtained this way are used in the point based pose tracking algorithm to get a new pose, as explained in the following section. This forms one optimization step, which is iterated until the pose changes induced by the force vectors will start to mutually cancel each other. We stop iterating when the average pose change after up to three iterations is smaller than a given threshold.

2.3 Pose Estimation from Point Correspondences

Assuming a given set of corresponding 2D points (3D rays) and 3D points, a 3D point-line based pose estimation algorithm for kinematic chains is applied to minimize the spatial distance between both contours: each 2D point is reconstructed to a 3D Plücker line $L_i = (n_i,m_i)$, see Section 3.1. 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

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

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. This mapping is denoted by π. The resulting nonlinear equation system can be linearized in the unknown twist parameters by using the first two elements of the sum representation of the exponential function:

$$\exp(\theta\hat{\xi}) = \sum_{i=0}^{\infty} \frac{(\theta\hat{\xi})^i}{i} \approx (I + \theta\hat{\xi}). \tag{4}$$

This approximation is used in (3) and leads to the linear equation system

$$((I+\theta\hat{\xi})X_i)_\pi \times n_i - m_i = 0. \tag{5}$$

Gathering a sufficient amount of point correspondences and appending the single equation systems leads to an overdetermined linear system of equations in the unknown pose parameters $\theta\hat{\xi}$. The least squares solution is used for reconstruction of the rigid body motion using Rodrigues' formula [9]. Then the model points are transformed and a new linear system is built and solved until convergence. The final pose is given as the consecutive evaluation of all rigid body motions during iteration.

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

$$(\exp(\theta\hat{\xi})\exp(\theta_1\hat{\xi}_1)\ldots\exp(\theta_j\hat{\xi}_j)X_i)_\pi \times n_i - m_i = 0 \tag{6}$$

which is linearized in the same way as the rigid body motion itself. It leads to three linear equations with the six unknown twist parameters and j unknown joint angles.

3 Integrating Ground Plane Constraints

The key idea of the present work is now to extend the previous system of equations with additional equations that reflect the ground plane constraint. These equations regularize the system and reduce the space of possible solutions to a desired subspace. This section explains how to formalize the geometric constraint and how to integrate it in the tracking system as soft constraint.

3.1 Lines and Planes

Although we note the necessary equations here in matrix calculus, we point out that other representations, e.g. Clifford or geometric algebras [13], provide a much more compact and unifying way to represent entities, their interactions and transformations. Due to space limits we avoid to introduce a possible higher-order algebraic model and stick to matrices.

In this work, we use an implicit representation of a 3D line in its Plücker form [4]. A Plücker line $L=(n,m)$ is given by a normalized vector n (the direction of the line) and a vector m, called the moment, which is defined by $m := X' \times n$ for a given point X' on the line. Collinearity of a point X to a line L can be expressed by

$$X \in L \quad \Leftrightarrow \quad X \times n - m = 0, \tag{7}$$

and the distance of a point X to the line L can easily be computed by $\|X \times n - m\|$, see [10] for details.

For planes we use an implicit representation as Hessian form. A plane $P=(D,d)$ is given by a normalized vector D (the normal of the plane) and the distance of the plane to the origin d. A point X is on the plane iff

$$X \in P \quad \Leftrightarrow \quad X \cdot D + d = 0. \tag{8}$$

The resulting scalar value also provides a signed distance from the point to the plane.

For a given line $L = (n, m)$ and plane $P = (D, d)$ the intersection point of these entities is given as

$$X \quad = \quad L \vee P \quad = \quad \frac{D \times m - nd}{D \cdot n}. \tag{9}$$

For a more detailed analysis of the constraints, we refer to [13].

3.2 Soft-Constraints for Penalizing Floor Intersections

As the described tracking procedure so far does not integrate any prior information about the scene, the surface mesh is able to take all possible configurations and positions. However, it is physically not possible that extremities intersect with the ground plane during, e.g., walking or jogging. For penalizing such configurations, we use the previously introduced representations of lines in Plücker coordinates and planes in Hessian from. We first determine all points which are below the ground floor. These can be found by testing the sign of Equation (8) for all points on the surface mesh. Each detected point that is below the ground plane is then mapped to its closest point on the plane by following the steps of Figure 1, left: We assume a given 3D-plane $P = (D, d)$ and a point X below the ground plane. From the point X and normal D we generate a Plücker line $L = (n, m) = (D, X \times D)$ and compute the intersection of L and P. Since $D = n$ and $D \cdot n = 1$, Equation (9) reduces to

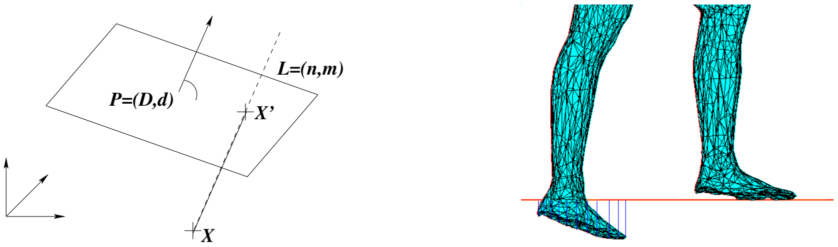

Fig. 1. Left: Projection of a 3D point onto a 3D plane. Right: Force vectors (marked in blue) shift the right foot towards the ground plane.

$$X' = D \times m - nd. \tag{10}$$

Now we can establish point-point correspondences between all points X^i below the ground plane and their mapping X_i' onto the ground plane. The correspondences are used in the pose estimation procedure to enforce the point X^i being close to the plane P: we define X^i as a set of 3D model points, which correspond to a set of points X_i' on the spatial ground plane. We express incidence in terms of

$$\forall i : X^i - X_i' = 0. \tag{11}$$

Since X^i belongs to a kinematic chain, X^i can be expressed as

$$X^i = \exp(\theta\hat{\xi}) \prod_{j \in \mathscr{J}(x_i)} \exp(\theta_j \hat{\xi}_j) X_i \tag{12}$$

and we can generate a set of equations forcing the transformed point X^i to stay close to X_i':

$$\exp(\theta\hat{\xi}) \prod_{j \in \mathscr{J}(x_i)} \exp(\theta_j \hat{\xi}_j) X_i - X_i' = 0 \tag{13}$$

Some exemplary force vectors are shown in Figure 1 on the right. Note that the vectors X_i' are treated as constant vectors, so that the only unknowns are the pose and kinematic chain coefficients. Also note that the unknowns are the same as for Equation (3), the unknown pose parameters. Only the point-line constraint is replaced by a point-point constraint. It expresses the involved geometric properties which are to be fulfilled during the interaction of the subject with the ground plane. As mentioned before, to generate the correspondences we run a simple test on all surface points if a point is below the ground plane or above. For all points below the ground plane we apply Equations (10) - (13) to generate equations which enforce the points below the ground plane to stay on (or above) the ground plane. The matrix of gathered linear equations is attached to the one presented in Section 2.3. Furthermore, its effect is to regularize the equations. The structure of the generated linear system is $A\chi = b$, with two parts generated from the linearized equations (5) and (13). Since the additional equations act only as soft constraints we add a strong weighting factor to Equation (13) in order to avoid any severe violations of the plane constraint. Infinite weights would turn the soft constraints into hard constraints.

4 Experiments

In the first experiment we considered a simple rigid object consisting of Duplo bricks and a controlled movement in a lab environment. The object was captured in a stereo set-up and the sequence consists of a simple ground plane movement. Since the object is just moving on the ground-plane and the ground plane is given by $y = 0$, negative y-values indicate an intersection with the given ground plane. Figure 2 shows on the right some example frames of the stereo cameras. As can be seen, it is possible to track the object, though parts are occluded by other bricks. The left diagram shows y-axis coordinates of two control points on the model ground plane. The expected ground truth is a simple constant function at 0. The red curves show the outcome without the additional constraints from Section 3 and the black ones show the result with the additional constraints from Section 3. The red curves deviate up to 4mm in space. This is already pretty accurate but the black curves show an even better and smoother tracking with a deviation of up to just 2*mm*. In the second experiment we took a sequence of the HumanEVA-II benchmark [12]. Here a surface model, calibrated image sequences, and background images are provided. Everyone is invited to experiment with the data and upload tracking results (as 3D marker positions) to a server at Brown University.

As the sequences are captured in parallel with a marker based tracking system (here a Vicon system is used), an automated script evaluates the accuracy of the tracking in terms of relative errors in *mm*. Figure 3 shows the four provided camera views and the surface mesh in its initial pose of the subject *S*4, which we used in our experiments. Figure 4 shows some example poses of our tracking system. The top two rows show results without the proposed soft constraints and the two bottom rows depict the outcome with the additional constraints. White ellipses mark some tracking deviations which

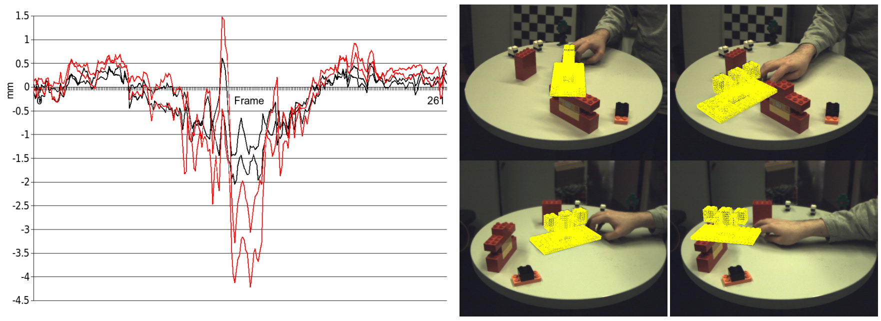

Fig. 2. Lego marble sequence with a planar movement. Left: y-axis coordinates of two control points on the ground plane. Red: without additional constraints to stay on the ground plane, black: with additional constraints. The black curves reveal smoother results where the object stays closer to the ground plane. Right: Some frames of the stereo sequence.

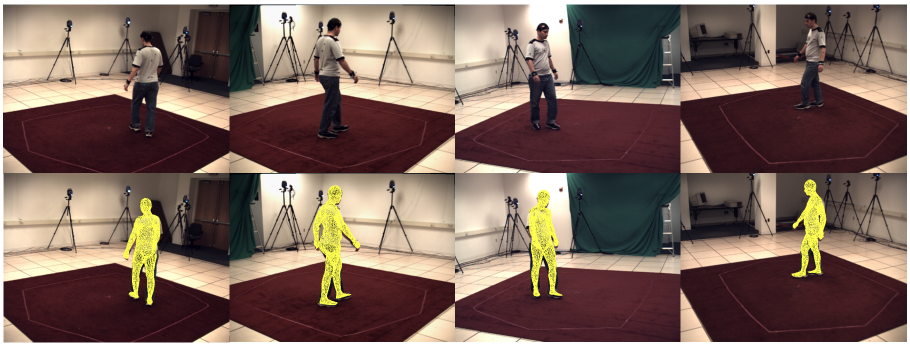

Fig. 3. Sample frame and pose of the HumanEVA-II database

are obviously smaller in the result where the ground plane constraint has been applied. Figure 5 shows an overlay of some poses with constraints (green) and without (purple). Without the additional constraints the feet can cross the ground plane and leg crossings occur as a consequence of the wrong configuration. The constraint clearly avoids such problems.

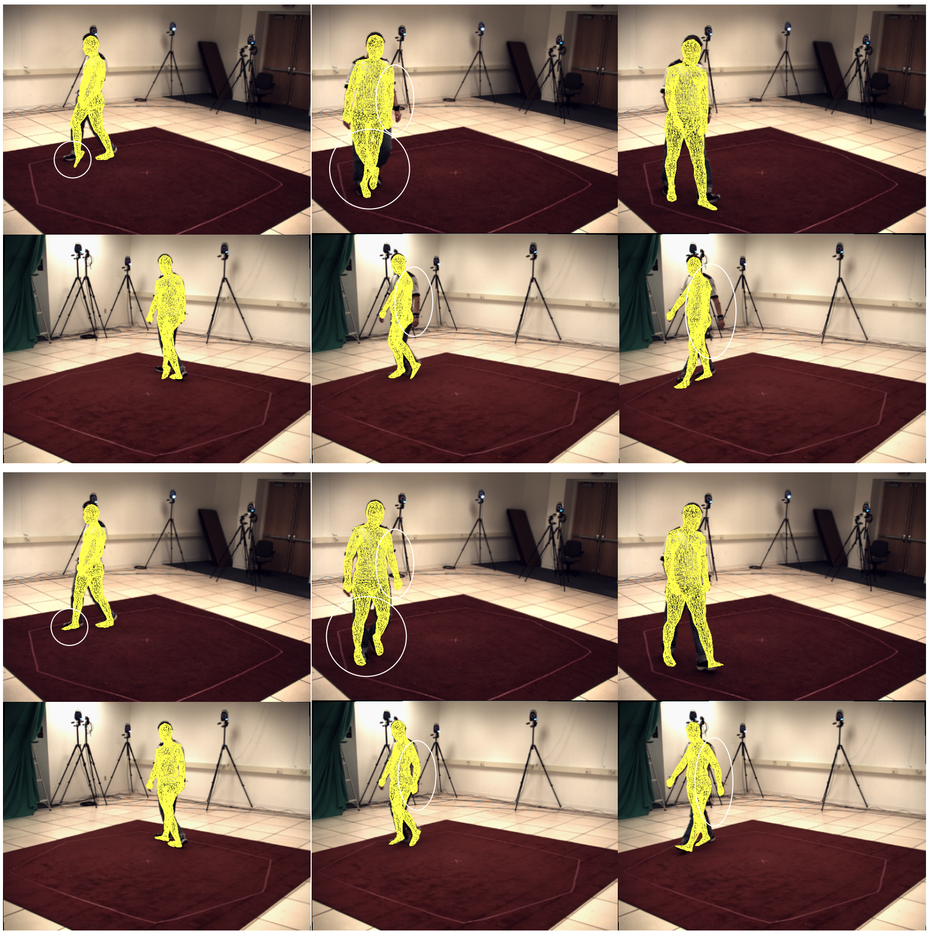

Fig. 4. Sample poses of our tracking system. Top: Pose results without the proposed soft constraints. Bottom: Pose results with the additional constraints.

The HumanEVA-II benchmark allows for a quantitative comparison. Figure 6 shows the deviations of our tracking system to the tracked markers. It depicts the unconstrained results in red and the constrained results in black. Table 1 compares the errors (automatically evaluated [12]). Overall the tracking has been improved remarkably using the additional ground plane constraint. In [2] a similar walking pattern on the HumanEVA-I database is analyzed. Since HumanEva-I does not provide a surface mesh, it was not possible for us to compare our outcome directly with the results reported in [2]. Therefore, the comparison in Table 1 should be interpreted deliberately. However, an overall improvement is clearly visible.

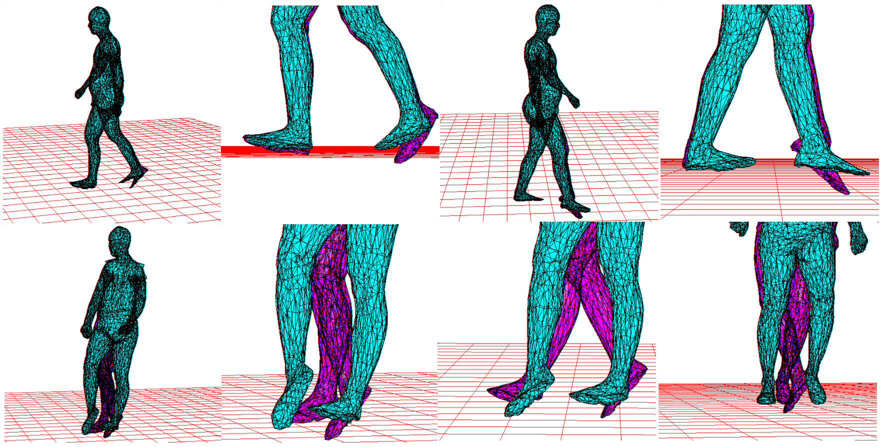

Fig. 5. Pose results in a simulation environment. Green: with constraints, purple: without.

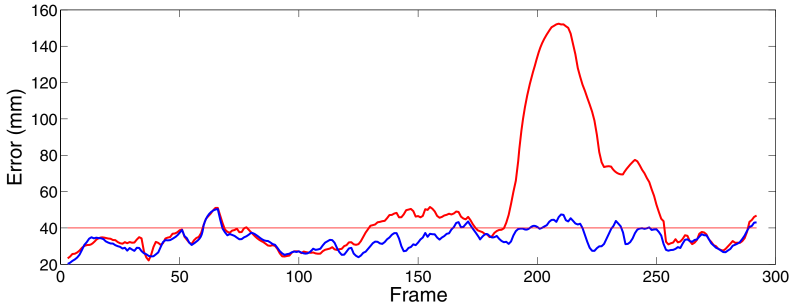

Fig. 6. Deviations of our tracking results with a marker based outcome. Red: without constraints, blue: with constraints.

Table 1. Comparison of the HumanEVA-II gait sequence with and without imposed ground plane constraints and the *RoAM* model from [2] (on a similar pattern in the HumanEVA-I dataset)

	Average Error (mm).	Std. Deviation	min error (mm)	max error (mm)
Without constraints	50.1	31.7	22.1	152.4
With constraints	33.8	5.9	20.2	50.4
RoAM model [2]	>60mm	-	-	-

5 Summary

In this work we have presented an approach in which geometric prior information about the floor the person is standing and moving on is integrated into a markerless pose tracking process. Therefore, we penalize movements in which extremities intersect the ground plane using soft constraints in the pose estimation framework. Experiments with rigid objects and the HumanEVA-II benchmark show that tracking is stabilized significantly and the additional soft constraints can be decisive for a successful tracking. For future experiments we are interested in tracking people in more complex environments, i.e., using a reconstructed background model. This would allow to integrate knowledge about persons and their interaction with the environment in the tracking process.

Acknowledgments

The research was funded by the German Research Foundation (DFG), the Max Planck Center VCC and the Cluster of Excellence on Multimodal Computing and Interaction, M^2CI.

References

1. Agarwal, A., Triggs, B.: Recovering 3D human pose from monocular images. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(1), 44–58 (2006)
2. Balan, A., Black, M.: An adaptive appearance model approach for model-based articulated object tracking. In: Conference of Computer Vision and Pattern Recognition (CVPR). IEEE Computer Society Press, Los Alamitos (2006)
3. Balan, A., Sigal, L., Black, M., Haussecker, H.: Shining a light on human pose: On shadows, shading and the estimation of pose and shape. In: Proc. International Conference on Computer Vision. IEEE Computer Society Press, Los Alamitos (2007)
4. Blaschke, W.: Kinematik und Quaternionen, Mathematische Monographien, vol. 4. Deutscher Verlag der Wissenschaften (1960)
5. Brox, T., Rosenhahn, B., Kersting, U., Cremers, D.: Nonparametric density estimation for human pose tracking. In: Franke, K., Müller, K.-R., Nickolay, B., Schäfer, R. (eds.) DAGM 2006. LNCS, vol. 4174, pp. 546–555. Springer, Berlin (2006)
6. Brubaker, M., Fleet, D.J., Hertzmann, A.: Physics-based person tracking using simplified lower-body dynamics. In: Conference of Computer Vision and Pattern Recognition (CVPR), Minnesota. IEEE Computer Society Press, Los Alamitos (2007)
7. Herda, L., Urtasun, R., Fua, P.: Implicit surface joint limits to constrain video-based motion capture. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3022, pp. 405–418. Springer, Heidelberg (2004)
8. Moeslund, T.B., Hilton, A., Krüger, V.: A survey of advances in vision-based human motion capture and analysis. Computer Vision and Image Understanding 104(2), 90–126 (2006)
9. Murray, R.M., Li, Z., Sastry, S.S.: Mathematical Introduction to Robotic Manipulation. CRC Press, Baton Rouge (1994)
10. 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)
11. 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)

12. Sigal, L., Black, M.: Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion. Technical Report CS-06-08, Brown University, USA (2006), `http://vision.cs.brown.edu/humaneva/`
13. Sommer, G.: Geometric Computing with Clifford Algebras. Springer, Heidelberg (2001)
14. Urtasun, R., Fleet, D.J., Fua, P.: 3D people tracking with Gaussian process dynamical models. In: Proc. International Conference on Computer Vision and Pattern Recognition, pp. 238–245. IEEE Computer Society Press, Los Alamitos (2006)