Markerless Motion Capture of Man-Machine Interaction

Bodo Rosenhahn[1]
[1]MPI Computer Science
Stuhlsatzenhausweg 85
66271 Saarbrücken, Germany
rosenhahn@mpi-inf.mpg.de

Christian Schmaltz[2]
[2]Mathematical Image Analysis
Saarland University
66041 Saarbrücken, Germany

Thomas Brox[3]
[3]Intelligent Systems
University of Dresden
01062 Dresden, Germany

Joachim Weickert[2]

Daniel Cremers[4]
[4]Computer Vision
University of Bonn
53117 Bonn, Germany

Hans-Peter Seidel[1]

Abstract

This work deals with modeling and markerless tracking of athletes interacting with sports gear. In contrast to classical markerless tracking, the interaction with sports gear comes along with joint movement restrictions due to additional constraints: while humans can generally use all their joints, interaction with the equipment imposes a coupling between certain joints. A cyclist who performs a cycling pattern is one example: The feet are supposed to stay on the pedals, which are again restricted to move along a circular trajectory in 3D-space. In this paper, we present a markerless motion capture system that takes the lower-dimensional pose manifold into account by modeling the motion restrictions via soft constraints during pose optimization. Experiments with two different models, a cyclist and a snowboarder, demonstrate the applicability of the method. Moreover, we present motion capture results for challenging outdoor scenes including shadows and strong illumination changes.

1. Introduction

Markerless Motion Capture (MoCap) is an active field of research in computer vision and graphics [15] with applications in animation (games, avatars), medicine or sports science. In contrast to commonly used marker based approaches, the analysis for markerless methods is based on sensor (usually image) data without special preparation of the subject. The goal is to determine the position and orientation as well as the joint angles of a human body from image data. In most approaches, the body parts are modeled as so-called kinematic chains, which are unconstrained in the sense that each body part has only one parent joint (and possibly multiple children) [2, 9, 15, 12, 5, 8]. In robotics, this is called an open chain or serial link system. In an open chain system, each joint position is equally likely. However, in MoCap it is often of interest to incorporate constraints on the pose configuration. Such constraints can either be learned from training data [19, 20, 22], or they can be modeled explicitly, for instance, by imposing fixed joint angle limits [21, 13] or by modeling the physics of body motion [3]. In [1] prior knowledge about light sources and shadows is applied to exploit more information about the scene observed by the cameras.

Figure 1. Examples of an athlete interacting with sports equipment. **Top:** cycling on a bike. **Bottom:** snowboarding (in the summer).

In the present work we are interested in tracking athletes who interact with sports equipment, for instance a snowboarder, who has his two feet fixed to his snowboard. This results in joint restrictions, which are known as so-called closed chain systems or constrained kinematic chains. In robotics and controller design, such systems are well known and subject to intensive investigations in or-

der to find analytic interdependencies of the involved joints [16, 11, 14, 6, 23]. In principle, closed chain systems come along with limited degrees of freedom. A theoretic work dealing with reduced equations of robotic systems using Lie groups can be found in [17]. Figure 1 shows two examples we will use for our experiments, an athlete riding a bike and jumping with a snowboard. Both examples are highly interesting for sports science [4, 7] and challenging from a mathematical point of view, since in the first case it is necessary to model the kinematics of the pedals, and the second case requires to model that the feet are rigidly connected to the snowboard.

The contribution of this paper is to propose a formalism which allows to express such geometric constraints in a markerless human pose estimation process. Both examples from Figure 1 can be modeled using conventional kinematic chains with additional constraint equations, which express the motion restrictions during pose estimation. The constraints are thereby modeled in terms of invariances of points on the body surface rather than analytical expressions on the joint interdependencies, as usually done in controller design. Due to this numerical approach, our framework can handle much more complex systems and is less involved. An alternative would be the automatic, implicit modeling of the interdependencies by learning the subspace of allowed configurations. However, this requires training data stemming from a marker-based system, which is not required in our approach.

In experiments we demonstrate the applicability of our concept by tracking a person in two highly challenging setups, one in a lab with cluttered background and one in an outdoor scene with strong illumination changes and dynamic motion patterns, such as jumps and turns.

The paper is organized as follows: Section 2 recalls mathematic foundations, pose estimation and region based tracking. Section 3 deals with constricted kinematic chains, followed from the experiments in Section 4. Finally, Section 5 concludes the paper with a summary.

2. Twists, kinematic chains and pose estimation

In this section we recall mathematic foundations needed for modeling kinematic chains. These are essential in order to understand the constraint equations developed later in Section 3.2. We further introduce the pose estimation and region based 2D-3D tracking procedure used for the experiments.

2.1. Twists

A rigid body motion of a 3D point x can be expressed in homogeneous coordinates as

$$\begin{aligned} X' = (x',1)^T &= \boldsymbol{M}X = \boldsymbol{M}(x,1)^T \\ &= \begin{pmatrix} \boldsymbol{R} & t \\ 0_{3\times 1} & 1 \end{pmatrix} \begin{pmatrix} x \\ 1 \end{pmatrix}. \end{aligned} \quad (1)$$

The matrix $\boldsymbol{R}$ is a rotation matrix, $\boldsymbol{R} \in SO(3)$ and t is a translation vector. The set of all matrices of type $\boldsymbol{M}$ is called the Lie Group $SE(3)$. To every Lie group there exists an associated Lie algebra, whose underlying vector space is the tangent space of the Lie group, evaluated at its origin [10]. The Lie algebra associated with $SE(3)$ is $se(3) := \{(v,\omega)|v \in \mathbb{R}^3, \omega \in so(3)\}$, with $so(3) := \{\boldsymbol{A} \in \mathbb{R}^{3\times 3}|\boldsymbol{A} = -\boldsymbol{A}^T\}$. Elements of $so(3)$ and $se(3)$ can be written as vectors $\omega = (\omega_1,\omega_2,\omega_3)^T$, $\xi = (\omega_1,\omega_2,\omega_3,v_1,v_2,v_3)^T$ or matrices

$$\hat{\omega} = \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix}, \quad \hat{\xi} = \begin{pmatrix} \hat{\omega} & \boldsymbol{v} \\ 0_{3\times 1} & 0 \end{pmatrix}. \quad (2)$$

It is further common to scale the twists ξ with respect to ω, i.e. $\theta = \|\omega\|$, $\omega := \frac{\omega}{\theta}$ and $\boldsymbol{v} := \frac{\boldsymbol{v}}{\theta}$. We denote a scaled twist as $\theta\xi$. To reconstruct a group action $\boldsymbol{M} \in SE(3)$ from a given twist, the exponential function $\boldsymbol{M} = \exp(\theta\hat{\xi}) = \sum_{k=0}^{\infty} \frac{(\theta\hat{\xi})^k}{k!}$ must be computed. This can be done efficiently by applying the Rodriguez formula [16]. Note, that for varying θ, the one-parametric Lie-subgroup $M_\theta = \exp(\theta\hat{\xi})$ yields a screw motion around an axis in space. A degenerate screw (without pitch component) will be used to model joints [2].

2.2. Kinematic chains

A kinematic chain is modeled as the consecutive evaluation of exponential functions of twists ξ_i [2]. A point at an end effector, additionally transformed by a rigid body motion is given as

$$X_i' = \exp(\theta\hat{\xi})(\exp(\theta_1\hat{\xi_1})\dots\exp(\theta_n\hat{\xi_n}))X_i. \quad (3)$$

In the remainder of this paper we will note a pose configuration by the $(6+n)$D vector

$$\chi = (\xi,\theta_1,\dots,\theta_n) = (\xi,\Theta), \quad (4)$$

consisting of the 6 degrees of freedom for the rigid body motion ξ and the joint angle vector Θ. In our setup, the vector χ is unknown and has to be determined from the image data. In this work, the angles Θ are further constricted and, therefore, depend on each other.

2.3. Pose estimation

A 3D point-line based pose estimation algorithm for kinematic chains is applied to minimize the spatial distance between given 2D image and 3D model point correspondences: therefore, each image point is reconstructed to a 3D line. The line is modeled as a 3D Plücker line $L_i = (n_i, m_i)$, see [16]. For pose estimation the reconstructed Plücker lines are combined with the twist 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

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

Since $\exp(\theta\hat{\xi})X_i$ is a 4D vector, the function π denotes the projection of the homogeneous 4D vector to a 3D vector by neglecting the homogeneous component.

For the case of kinematic chains, we exploit the property that joints are expressed as special twists 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

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

To minimize for all correspondences in a least squares sense, we optimize

$$\underset{\chi}{\operatorname{argmin}} \sum_i \left\| \left(\exp(\theta\hat{\xi}) \prod_{j\in\mathcal{J}(x_i)} \exp(\theta_j\hat{\xi}_j) \begin{pmatrix} x_i \\ 1 \end{pmatrix} \right)_\pi \times n_i - m_i \right\|_2^2. \tag{7}$$

The function $\mathcal{J}(x_i)$ denotes the ordered set of joints that affect the point x_i. Linearization of the exponential function by approximating $\exp(\theta\hat{\xi})$ with $\exp(\theta\hat{\xi}) \approx I + \theta\hat{\xi}$ leads to three linear equations with $6+n$ unknowns, the six pose parameters and n joint angles. Collecting enough correspondences yields an over-determined linear system of equations and allows to solve for these unknowns in the least squares sense. Then the Rodrigues formula is applied to reconstruct the group action and the process is iterated for the transformed points until convergence.

This pose estimation procedure requires known correspondences between 2D points (3D lines) and 3D points. To generate a set of correspondences for a given surface mesh and calibrated images, image region statistics are analyzed, as explained in the next section.

2.4. Region based pose tracking

Many contour-based pose estimation algorithms expect an explicit contour to establish correspondences between contour points and points on the model surface. This involves matching the projected surface and the previously computed contour. In [18], an approach has been suggested that avoids explicit computations of contours and contour matching. It rather adapts the pose parameters for the projections of the surface to optimally split all images into homogeneously distributed object and background regions. This approach is briefly reviewed.

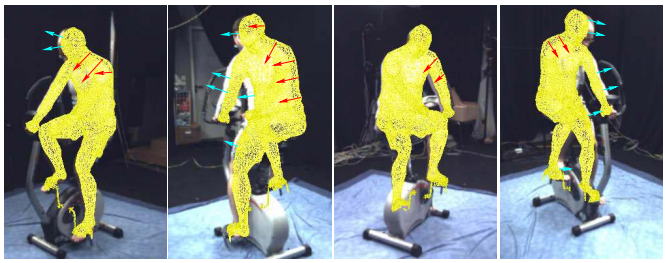

Figure 2. Example forces acting on the contour of the projected surface mesh. Vectors in cyan point outwards, those in red point inwards. For visualization, the force vectors have been enlarged.

2.4.1 Energy model

The partitioning of the image domain Ω can be expressed as minimization of the energy function

$$E(\chi) = -\int_\Omega \big(P(\chi,q)\log p_1 + (1-P(\chi,q))\log p_2\big)dq\,, \tag{8}$$

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 can be regarded as the projection of the model surface with the current pose χ and splits the image domain into the object region and its background. Hence, the energy function enforces adapting the pose parameters in a way that the foreground and background regions are as homogeneous as possible. Homogenity is modelled by the probability density functions p_1 and p_2 estimated in the two regions by using the color distributions in the CIELab color space.

2.4.2 Minimization

For minimization of the above, highly nonlinear energy, an iterative procedure approximating a gradient descent is applied. The fact that there is no analytic expression of how $P(\chi)$ depends on χ rules out an exact gradient descent. The gradient is approximated by creating 2D-3D point correspondences (q_i, x_i). To this end, surface points x_i are projected to the image plane using the current pose χ. The points q_i that generate the silhouette in the image are collected for computing a pose update. Those points q_i that better fit to the object region – i.e. those points for which $p_1(q_i)$ is greater than $p_2(q_i)$ – are shifted in outward normal direction to a new point q_i' (cyan in Figure 2). Points where $p_1(q_i) < p_2(q_i)$ are shifted into the opposite direction to q_i', respectively (red in Figure 2). Figure 2 visualizes some example forces acting on a pose for the bike scene.

The 2D-3D point correspondences (q_i', x_i) obtained this way are used in the point based pose estimation algorithm explained in Section 2.3 to get a new pose. Computation of shift vectors and pose estimation are iterated until the force vectors mutually cancel each other and the pose does not change any longer. We stop iterating when the average pose change after up to three iterations is smaller than a small ϵ.

3. Constricted kinematic chains

The focus of this paper is on tracking closed chain systems, which have reduced degrees of freedom due to interdependencies between the involved joints. In order to determine the degrees of freedom of a closed chain manipulator, Grueblers formula can be applied [16]. Let N be the number of links in the mechanism, g the number of joints, and f_i the degrees of freedom for the ith joint. The number of degrees of freedom of the mechanism is

$$F = 6N - \sum_{i=1}^{g}(6 - f_i) = 6(N - g) + \sum_{i=1}^{g} f_i. \quad (9)$$

For planar motions, the scalar 6 needs to be replaced with 3.

The key idea in the present work is to use open chain models, as the one reviewed in the previous section, and to add constraint equations in the pose optimization procedure to enforce their configuration as a constricted kinematic chain. These further constraints will automatically result in equations of rank $g - F$, with the degrees of freedom of the mechanism as the remaining unknowns. An alternative would be the analytic derivation of the joint restrictions. However, the next subsection demonstrates by means of an example that this approach very quickly gets very complex and is in fact impracticable for complex systems.

3.1. Analytic derivation of joint restrictions

Let us consider a simple pedal model for the 2D case, see Figure 3. In this toy example we compute the redundancies explicitly, to show their increasing complexity (of higher order trigonometric functions) and to justify that this approach is impracticable for more complex systems, hard to generalize to other models, and very hard to optimize within the pose estimation context. Section 3.2 will later show that the numerical modeling of the restrictions as soft constraint is much more simple and practicable.

The simplified planar model of a leg on a pedal is visualized in Figure 3 and consists of 2 joints (black bullets). One joint is located at the origin $(0, 0)$ with angle θ_1 and the other one is at $j = (0, 3)$ with angle θ_2. The end effector (green bullet) is given at $p = (0, 5)$. Furthermore, we want to restrict the movements of the end effector to stay on the dashed circle (with center $C = (0, 4)$ and radius 1), so that

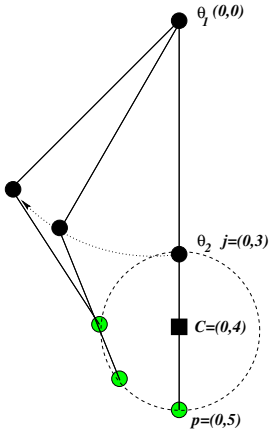

Figure 3. 2D example for a simplified leg model performing a cycling pattern.

the end effector only performs a circular trajectory, similar to a foot on a pedal during cycling.

The position p' of the end effector in terms of θ_1 and θ_2 can be written as

$$p' = \begin{pmatrix} \cos(\theta_1) & -\sin(\theta_1) \\ \sin(\theta_1) & \cos(\theta_1) \end{pmatrix} \left(\begin{pmatrix} \cos(\theta_2) & -\sin(\theta_2) \\ \sin(\theta_2) & \cos(\theta_2) \end{pmatrix} (p - j) + j \right) \quad (10)$$

$$= \begin{pmatrix} -2\sin(\theta_2 + \theta_1) - 3\sin(\theta_1) \\ 2\cos(\theta_2 + \theta_1) - 3\cos(\theta_1) \end{pmatrix}. \quad (11)$$

The requirement for p' to stay on the dashed circle can be expressed as

$$(p'(1) - C(1))^2 + (p'(2) - C(2))^2 = 1. \quad (12)$$

Solving this equation in θ_2 yields two solutions (dependent on a sign) from which one is selected as
$\theta_2 = -\theta_1 + atan(-(18\sin(\theta_1)\cos(\theta_1) - 6(-6 + 15\cos(\theta_1) - 9\cos^2(\theta_1))^{(1/2)}\cos(\theta_1) - 21\sin(\theta_1) + 8(-6 + 15\cos(\theta_1) - 9\cos^2(\theta_1))^{(1/2)})/(24\cos(\theta_1) - 25)$.

Resubstitution in Equation (11) leads to

$$p' = \begin{pmatrix} p_1' \\ p_2' \end{pmatrix} \quad (13)$$

with
$p_1' = -(36\sin(\theta_1)\cos(\theta_1) + 12(-6 + 15\cos(\theta_1) - 9\cos^2(\theta_1))^{(1/2)}\cos(\theta_1) - 33\sin(\theta_1) - 16(-6 + 15\cos(\theta_1) - 9\cos^2(\theta_1))^{(1/2)})/(24\cos(\theta_1) - 25)$ and
$p_2' = -(-36\cos^2(\theta_1) - 15\cos(\theta_1) + 12\sin(\theta_1)(-6 + 15\cos(\theta_1) - 9\cos^2(\theta_1))^{(1/2)} + 56)/(24\cos(\theta_1) - 25)$.
So, instead of expressing the end effector with two exponential functions as proposed in Equation (3):

$$p' = (\exp(\theta_1\hat{\xi}_1)\exp(\theta_2\hat{\xi}_2))p, \quad (14)$$

with ξ_i being respective 2×2 matrices, a higher order non-linear term is the price to model the inherent restrictions. For more complex kinematic chains (e.g. containing 8 or more involved joints), this method is not suitable and requires a detailed analysis of each model in advance to the pose computations. Hence, we propose to model the kinematic chains as open chain systems and to add the involved restrictions as additional constraints acting on 3D points rather than the joint angles. This is described in the next section.

3.2. Soft-constraints for constricted kinematic chains

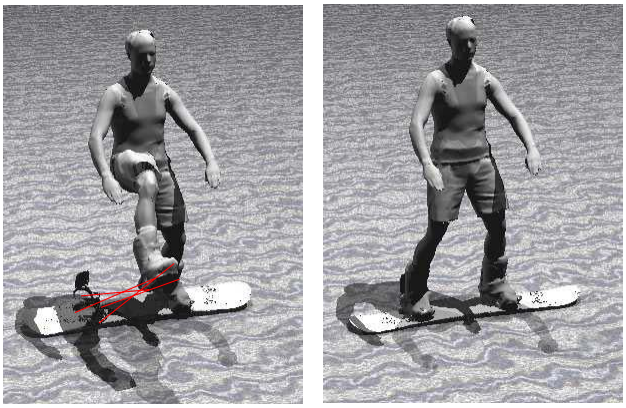

Figure 4. The used snowboard model: One foot is rigidly connected to the snowboard and the other foot is allowed to move freely in space. Extra forces (marked in red) enforce the second foot to stay on the snowboard.

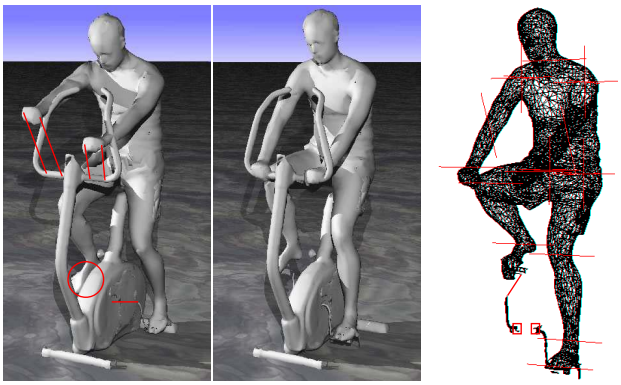

Figure 5. The used cycling model: One foot is rigidly connected to the pedal and the other foot is allowed to move freely in space. Forces (marked in red) constrain the second foot to stay on the second pedal (encircled) and the middle of the pedal axis on the virtual bike. The hands are further enforced to stay on the handle bar. The picture on the right shows the wireframe model, the joint axis (shown as red lines) and the pedals are part of the model. The red squares indicate the invariant positions of the pedal axis to enforce a cycling trajectory.

A snowboarder standing on a snowboard has to fulfill the property that both legs are rigidly connected on a 3D plane (the snowboard). To achieve modeling of a snowboarder standing on a snowboard, we firstly generate a model of an athlete and the snowboard, see Figure 4. The model is designed in a way that one of the legs is rigidly connected to the snowboard binding and the other one is allowed to move freely in space. Now we add forces (depicted with red lines), which enforce the second foot to stay on the snowboard binding.

We define P_i^b as a set of 3D points on the snowboard binding, which correspond to a set of points P_i^s on the right snowboard boot. Since the corresponding points belong to different parts of the kinematic chain (note, the snowboard is part of the left leg, whereas the points on the boot belong to the right leg), we express incidence in terms of

$$\forall i \quad : \quad P_i^s - P_i^b = 0. \tag{15}$$

Since P_i^s and P_i^b are parts of kinematic chains, they can be expressed as

$$P_i^s = \exp(\theta\hat{\xi}) \prod_{j \in \mathcal{J}(p_i^s)} \exp(\theta_j \hat{\xi}_j) p_i^s \tag{16}$$

$$P_i^b = \exp(\theta\hat{\xi}) \prod_{j \in \mathcal{J}(p_i^b)} \exp(\theta_j \hat{\xi}_j) p_i^b \tag{17}$$

and we can generate two sets of equations forcing the transformed point p_i^s to stay close to P_i^b and the transformed point p_i^b to stay close to P_i^s:

$$\exp(\theta\hat{\xi}) \prod_{j \in \mathcal{J}(p_i^s)} \exp(\theta_j \hat{\xi}_j) p_i^s - P_i^b = 0, \tag{18}$$

$$\exp(\theta\hat{\xi}) \prod_{j \in \mathcal{J}(p_i^b)} \exp(\theta_j \hat{\xi}_j) p_i^b - P_i^s = 0. \tag{19}$$

Note that P_i^b and P_i^s are treated as constant vectors, so that the only unknowns are the pose and kinematic chain coefficients. This can be done similarly with the kinematics of the bike, see Figure 5. Here, some more invariances can be exploited: The hands have to be on the handle of the bike, the feet on the pedals and the pedals have to spin around an axis. To achieve the circular trajectory of the feet on the pedals, the right foot is forced to stay on the pedals (as shown in Figure 5) and the pedal axis is forced to stay on an invariant (relative) position, as shown with the red squares in Figure 5, right. In this case we gain four additional constraints for the hands and three constraints for the feet. It results in seven point-point correspondences. Note: The pedals introduce one additional joint to the human model.

Note that the unknowns are the same as for Equation (7), the unknown pose parameters. Only the point-line constraints are replaced with simpler point-point constraints, which express the involved geometric invariances which

Figure 6. Seven synchronized color-VGA cameras are used for recording the snowboard sequences (one time-frame is shown).

occur during the interaction of the athlete with his sports equipment. The matrix of gathered linear equations is attached to the one presented in Section 2.3. Its effect is to regularize the equations and to constrain the solution to a desired subspace of possible joint configurations. The structure of the generated linear system is $A\chi = b$, with A and b containing three attached matrices/vectors generated from the linearized equations (6), (18) and (19).

The system is optimized in a least squares sense, similar to Equation (7). Since the additional equations act as soft constraints, it can still happen during minimization that the property of the system is broken due to ambiguities in the image data. These are reduced by adding a strong weighting factor to Equations (18) and (19). In our experiments, the pose of the kinematic chain system had a deviation of less then 5 mm to an explicitly modeled constrained kinematic chain. For many applications in tracking, this accuracy is fully sufficient.

4. Experiments

In this section we present experimental results of constrained kinematic chains we track in different set-ups. We present experiments with a cyclist and a snowboarder.

4.1. The cyclist

As mentioned in Section 3 our cyclist model consists of a human person and the involved pedals and pedal axes. The

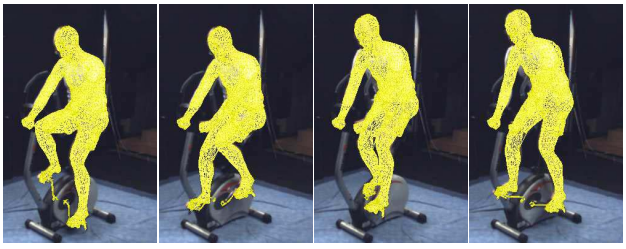

Figure 7. Pose results overlaid with one of the four used cameras. The pose is visualized by projecting the surface mesh onto the image plane (images are cropped). The subject is sitting and standing.

soft-constraints enforce the hands to stay on top of the handle bar and the feet on the pedals. Furthermore, the pedals are constrained to move along the trajectory of a circle by forcing the endpoints of the pedal axes to be on a constant position on the bike. In a lab environment we placed four

Figure 8. Simulation results of the cyclist in a virtual environment.

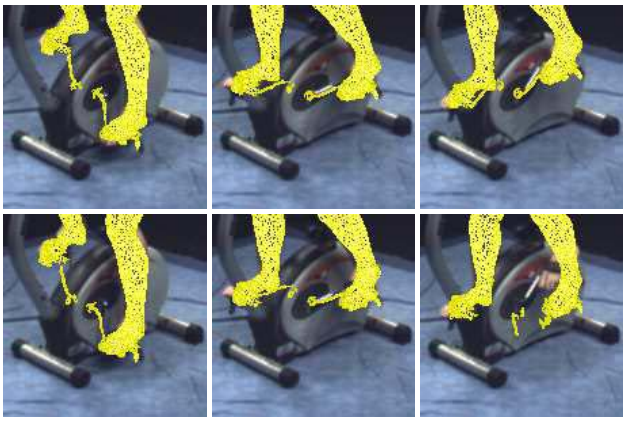

Figure 9. Close up of pedals and feet during tracking. Top: With soft-constraints, slight deviations can occur during tracking. Nonetheless, the cycling pattern is maintained. Bottom: Without soft-constraints, the first frames are still stable, but after a couple of frames, the tracking is instable and erroneous.

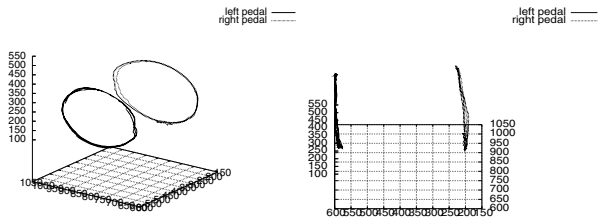

Figure 10. 3D-Trajectories of the feet during tracking from two perspective views. The circular path is clearly visible (units are in millimeter).

cameras around a fitness bike and captured a sequence with 100 frames per second (fps). Figure 7 shows some pose results by projecting the surface mesh onto the image planes (one of the four used cameras is shown). The bike is not

explicitly modeled in this set-up and though it causes problems during tracking, these ambiguities are overruled by the given geometric prior constraints and the visible parts of the legs. Therefore, tracking is successful. Figure 8 shows simulation results of the cyclist in a virtual environment from another view-point than the used cameras.

Figure 9 shows close ups of the pedals and feet during tracking, with (top) and without (bottom) the proposed soft-constraints. Obviously, without soft-constraints the tracking is unstable. The unsmoothed 3D-trajectories of the feet during (successful) tracking are shown in Figure 10 from two perspective views. The circular path is clearly visible.

4.2. The snowboarder

In this section, we will show results of a tracked snowboarder. Snowboarders have the property that both legs are rigidly connected to a plane (the snowboard). Tracking snowboarders and computing the forces acting on the joints requires an explicit modeling of this special situation. To achieve tracking of a snowboarder, we first generate a model of an athlete and the snowboard, see Figure 4. The model is designed as explained in Section 3. It is used for markerless motion capture in an outdoor environment. Seven VGA-color cameras are used to capture a multi-view image stream with 100 frames per second, see Figure 6. Due to summer during the experiments, the subject was standing on a table-cloth and jumping around. The strong sun lead to heavy shadows and the changing illumination made background subtraction methods unfeasible. The jumping and turning patterns further lead to motion blur during tracking.

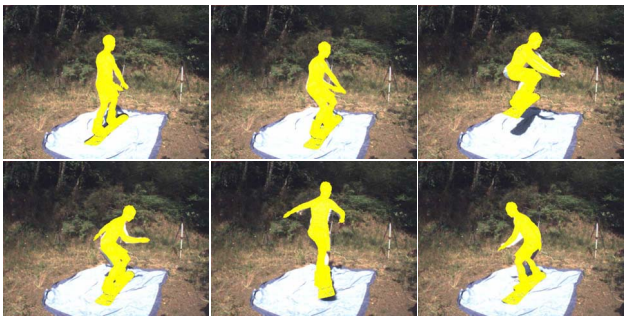

Figure 11. Pose results shown in one out of seven camera views. The pose is visualized by projecting the surface mesh onto the image plane.

Figure 11 shows some pose results in one camera. As can be seen, the algorithm is able to capture the movements, despite some inaccuracies in estimating the arm or hand positions.

Figure 12 demonstrates the impact of the proposed constraints. Without the constraints, the right boot is not properly placed on the snowboard and the tracking is more instable (see top, left picture and the red curve on the right).

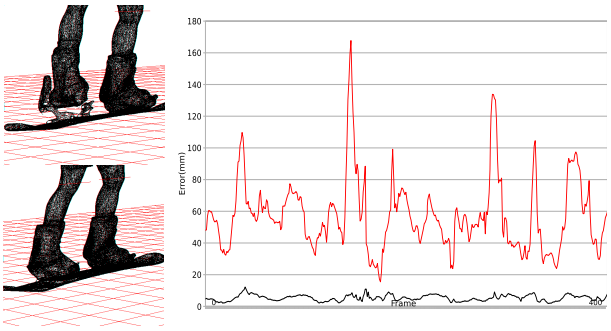

Figure 12. Impact of the proposed constraints: Left, top: without additional constraints, the boot is not in the snowboard binding. Left, bottom: with additional constraints, the boot is accurately fitted on the snowboard. Right: Distance between boot and binding without constraints (red) and with constraints (black).

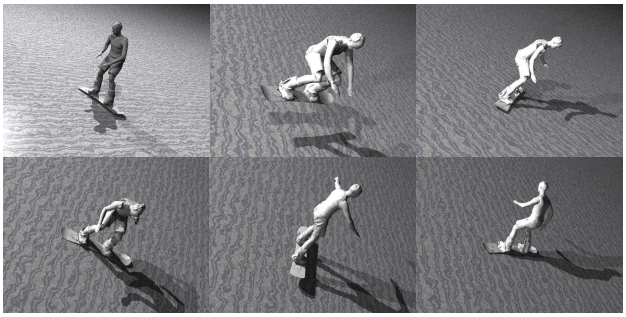

Figure 13. Simulation results in a virtual environment. For animation a slope and a time-dependent translation was added to synthesize an appealing action sequence.

With the additional constraints, the fitting is much more accurate (see bottom, left picture and the black curve). As error measure, the distance between the right boot and snowboard binding is used.

Figure 13 shows some tracking results in a virtual environment. Here a slope of 30 degrees is added and a translation during movement. The synthesized movement looks remarkably realistic and allows to generate highly dynamic action sequences, though the athlete was not really moving in the MoCap setup.

5. Summary

In this paper we presented an approach for markerless tracking of athletes interacting with sports equipment. We analyzed two examples, namely a cyclist and a snowboarder and showed that it is possible to restrict the degrees of freedom for the joint configurations on desired subspaces. The movement restrictions are driven by geometric prior knowledge on the sports gear and allow, e.g., to keep the feet fixed on a snowboard or to maintain a cycling trajectory with the feet. Instead of modeling the joint restrictions analytically, as is often done in robotics, or by using a learning stage, as

often done in human motion tracking, we propose to model these geometric restrictions via numerical constraints. This allows us to drive the solution towards the desired subspace without changing the basic, open chain pose estimation algorithm used for markerless tracking of unconstrained movements. The additional equations act as soft constraints and guarantee a robust tracking for both, smooth motion patterns (cycling) and patterns with fast contact (jumping). The presented approach is not limited to snowboarding or cycling. Extensions to many kinds of sporting activities including equipment, e.g. rowing, golfing or weight lifting, are straightforward within the proposed framework. Future work will concentrate on a more flexible model for joint restrictions, e.g. by optimizing an unknown but fixed distance or angle between the snowboard boots.

Acknowledgments

The authors would like to thank the reviewers and the area chair for their valuable comments and hints. 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] A. Balan, L. Sigal, M. Black, and H. Haussecker. Shining a light on human pose: On shadows, shading and the estimation of pose and shape. In *Proc. International Conference on Computer Vision*, 2007. 1

[2] 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. 1, 2

[3] M. Brubaker, D. J. Fleet, and A. Hertzmann. Physics-based person tracking using simplified lower-body dynamics. In *Conference of Computer Vision and Pattern Recognition (CVPR)*, Minnesota, 2007. IEEE Computer Society Press. 1

[4] G. Caldwell, J. Hagberg, S. McCole, and L. Li. Lower extremity joint moments during uphill cycling. *Journal of Applied Biomechanics*, (15):166–181, 1999. 2

[5] J. Carranza, C. Theobalt, M. A. Magnor, and H.-P. Seidel. Free-viewpoint video of human actors. In *Proc. SIGGRAPH 2003*, pages 569–577, 2003. 1

[6] H. Cheng and Y. Yiu. Dynamics and control of redundantly actuated parallel manipulators. *Trans. on Mechatronics*, 8(4):483–491, 2003. 2

[7] S. Delorme, S. Tavoularis, and M. Lamontagne. Kinematics of the ankle joint complex in snowboarding. *Journal of Applied Biomechanics*, 21(4):394–403, 2005. 2

[8] D. A. Forsyth, O. Arikan, L. Ikemoto, J. O'Brien, and D. Ramanan. Computational studies of human motion: part 1, tracking and motion synthesis. *Found. Trends. Comput. Graph. Vis.*, 1(2-3):77–254, 2005. 1

[9] P. Fua, R. Plänkers, and D. Thalmann. Tracking and modeling people in video sequences. *Computer Vision and Image Understanding*, 81(3):285–302, 2001. 1

[10] J. Gallier. *Geometric Methods and Applications For Computer Science and Engineering.* Springer-Verlag, New York Inc., 2001. 2

[11] X. Gao, D. Dawson, and Z. Qu. On the robust control of two manipulators holding a rigid object. *Journal of Intelligent and Robotic Systems*, 8:107–119, 1992. 2

[12] D. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73(1):82–92, 1999. 1

[13] 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, 2004. Springer. 1

[14] D. Kim, J. Kang, and K. Lee. Robust tracking control design for a 6 dof parallel manipulator. *Journal of Robotics Systems*, 17(10):527–547, 2000. 2

[15] T. B. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2):90–126, 2006. 1

[16] R. Murray, Z. Li, and S. Sastry. *Mathematical Introduction to Robotic Manipulation.* CRC Press, Baton Rouge, 1994. 2, 3, 4

[17] J. Ostrowski. Computing reduced equations for robotic systems with constraints and symmetries. *Trans. on Robotics and Automation*, 15(1):111–123, 1999. 2

[18] C. Schmaltz, B. Rosenhahn, T. Brox, D. Cremers, J. Weickert, L. Wietzke, and G. Sommer. Region-based pose tracking. In J. Martí, J. M. Benedí, A. M. Mendonça, and J. Serrat, editors, *Pattern Recognition and Image Analysis*, volume 4478 of *LNCS*, pages 56–63, Girona, Spain, June 2007. Springer. 3

[19] H. Sidenbladh, M. J. Black, and L. Sigal. Implicit probabilistic models of human motion for synthesis and tracking. In A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, editors, *Proc. European Conference on Computer Vision*, volume 2353 of *LNCS*, pages 784–800. Springer, 2002. 1

[20] C. Sminchisescu and A. Jepson. Generative modeling for continuous non-linearly embedded visual inference. In *Proc. International Conference on Machine Learning*, 2004. 1

[21] C. Sminchisescu and B. Triggs. Estimating articulated human motion with covariance scaled sampling. *International Journal of Robotics Research*, 22(6):371–391, 2003. 1

[22] R. Urtasun, D. J. Fleet, and P. Fua. 3D people tracking with Gaussian process dynamical models. In *Proc. International Conference on Computer Vision and Pattern Recognition*, pages 238–245. IEEE Computer Society Press, 2006. 1

[23] Y. Zweiri, L. Senevirante, and K. Althoefer. Modelling of closed-chain manipulators on an excavator vehicle. *Mathematical and Computer Modeling of Dynamical Systems*, 12(4):329–345, 2003. 2