

# Hierarchical Implicit Surface Joint Limits to Constrain Video-Based Motion Capture

Lorna Herda, Raquel Urtasun, and Pascal Fua\*

Computer Vision Laboratory  
EPFL  
CH-1015 Lausanne, Switzerland  
pascal.fua@epfl.ch

**Abstract.** To increase the reliability of existing human motion tracking algorithms, we propose a method for imposing limits on the underlying hierarchical joint structures in a way that is true to life. Unlike most existing approaches, we explicitly represent dependencies between the various degrees of freedom and derive these limits from actual experimental data.

To this end, we use quaternions to represent individual 3 DOF joint rotations and Euler angles for 2 DOF rotations, which we have experimentally sampled using an optical motion capture system. Each set of valid positions is bounded by an implicit surface and we handle hierarchical dependencies by representing the space of valid configurations for a child joint as a function of the position of its parent joint.

This representation provides us with a metric in the space of rotations that readily lets us determine whether a posture is valid or not. As a result, it becomes easy to incorporate these sophisticated constraints into a motion tracking algorithm, using standard constrained optimization techniques. We demonstrate this by showing that doing so dramatically improves performance of an existing system when attempting to track complex and ambiguous upper body motions from low quality stereo data.

## 1 Introduction

Even though many approaches to tracking and modeling people from video sequences have been and continue to be proposed [1,2,3], the problem remains far from solved. This in part because image data is typically noisy and in part because it is inherently ambiguous [4]. Introducing valid joint limits is therefore one important practical step towards restricting motion tracking algorithms to humanly feasible configurations, thereby reducing the search space they must explore and increasing their reliability.

This is currently done in many existing vision systems [5,6,4,7] but the limits are usually represented in an oversimplified manner that does not closely correspond to reality. The most popular approach is to express them in terms of

---

\* This work was supported in part by the Swiss National Science Foundation.

hard limits on the individual Euler angles used to parameterize joint rotations. This accounts neither for the dependencies between angular and axial rotations in ball-and-socket joints such as the shoulder joint nor those between separate joints such as the hip and knee. In other words, how much one can twist one’s arm depends on its position with respect to the shoulder. Similarly, one cannot bend one’s knee by the same amount for any configuration of the hip. An additional difficulty stems from the fact that experimental data on these joint limits is surprisingly sparse: medical text books typically give acceptable ranges in a couple of planes but never for the whole configuration space [8], which is what is really needed by an optimization algorithm searching that space.

In earlier work, we proposed a quaternion-based model approach to representing the dependencies between the three degrees of freedom of a ball-and-socket joint such as the shoulder [9]. It relies on measuring the joint motion range using optical motion capture, converting the recorded values to joint rotations encoded by a coherent quaternion field, and, finally, representing the subspace of valid orientations as an implicit surface. Here, we extend it so that it can also handle coupled joints, which we treat as parent and child joints. We represent the space of valid configurations for the child joint as a function of the position of the parent joint.

A major advantage of this quaternion representation is that it provides us with a rigorous distance measure between rotations, and thus supplies the most natural space in which to enforce joint-angle constraints by orthogonal projection onto the subspace of valid orientations [10]. Furthermore, it is not subject to singularities such as the “Gimbal lock” of Euler angles or mapping rotations of  $2\pi$  to zero rotations. As a result, it becomes easy to incorporate these sophisticated constraints into a motion tracking algorithm using standard constrained optimization techniques [11].

We chose the case of shoulder and elbow joints to validate our approach because the shoulder is widely regarded as the most complex joint in the body and because position of the arm constrains the elbow’s range of motion. We developed a motion capture protocol that relies on optical motion capture data to measure the range of possible motions of various subjects and build our implicit surface representation. We then used it to dramatically improve the performance of an existing system [12] when attempting to track complex and ambiguous upper body motions from low quality stereo data.

In short, the method we propose here advances the state-of-the-art because it provides a way to enforce joint limits on swing and twist of coupled joints while at the same time accounting for their dependencies. Such dependencies have already been described in the biomechanical literature [13,14] but using the corresponding models requires estimating a large number of parameters, which is impractical for most Computer Vision applications. Our contribution can therefore be understood as a way of boiling down these many hard-to-estimate parameters into our implicit surface representation, that can be both easily instantiated and used for video-based motion capture. Furthermore, the framework

we advocate is generic and could be incorporated into any motion-tracking approach that relies on minimizing an objective function.

In the remainder of the paper, we first briefly review the state of the art. We then introduce our approach to experimentally sampling the space of valid postures that the shoulder and elbow joints allow and to representing this space in terms of an implicit surface in Quaternion space. Finally, we demonstrate our method's effectiveness for tracking purposes.

## 2 Related Approaches

The need to measure joint limits arises most often in the field of physiotherapy and results in studies such as [15] for the hip or [16,8] for the shoulder. Many of these empirical results have subsequently been used in our community.

### 2.1 Biomedical Considerations

When we refer to the shoulder joint, we actually mean the gleno-humeral joint, which is the last joint in the shoulder complex hierarchy. It is widely accepted that modeling it as a ball-and-socket joint, which allows motion in three orthogonal planes, approximates its motion characteristics well enough for visual tracking purposes [3]. This approximation has been validated by a substantial body of biomechanical research that has shown that, because of large-bone-to-skin displacements, no clavicular or scapular motions can be recovered using external markers [17,18].

However, the dependency between arm twist and arm orientation, or swing, is a direct consequence of the complex joint geometry of the shoulder complex [19]. Coupling between elbow and shoulder is not only due to anatomical reasons, but also to the physical presence of the rest of the body, namely the thorax and the head, that limit the amount of elbow flexion for certain shoulder rotations. As to elbow twist, the dependency is anatomical and the available range of motion is directly linked to shoulder orientation [20]. It is those intra- and inter-joint dependencies that make the shoulder and elbow complex ideal to validate our approach. Furthermore, similar constraints exist for the hip and knee joints and our proposed approach should be easy to transpose.

Of course, the interdependence of these joint limits has been known for a long time and sophisticated models have been proposed to account for them, such as those reported in [13,14]. However, the former involves estimating over fifty elastic and viscous parameters, which may be required for precise biomedical modeling, but is impractical for Computer Vision applications and the latter focuses in motions in the sagittal plane as opposed to fully 3-D dimensional movements.

It is worth noting that inter-subject variance has been shown to be extremely small at the shoulder joint level [20]. The online documentation for the Humanoid Animation Working Group confirms that the difference in range of motion of women over men is minimal at the shoulder joint level, and small for the elbow

joint. The experimental data we present in Section 3 confirms this. Thus, it is acceptable to generalize results obtained on the basis of measurements carried out on a very small number of subjects, as we have done in our case, where data collection was carried out on three subjects, two females and one male.

## 2.2 Angular Constraints and Body Tracking

The simplest approach to modeling articulated skeletons is to introduce joint hierarchies formed by independent 1-Degree-Of-Freedom (DOF) joints, often described in terms of Euler angles with joint limits formulated as minimal and maximal values. This formalism has been widely used [5,6,2,4,7], even though it does not account for the coupling of the intra- or inter-joint limits and, as a result, does not properly account for the 3-D accessibility space of real joints.

Furthermore, Euler angles suffer from an additional weakness known as ‘‘Gimbal lock’’. This refers to the loss of one rotational degree of freedom that occurs when a series of rotations at 90 degrees is performed, resulting in the alignment of the axes [21,22]. The swing-twist representation, exponential map, and three-sphere embedding are all adequate to represent rotations and do not exhibit such flaws[23]. However, only quaternions are free of singularities [10]. Furthermore, because there is a natural distance between rotations in quaternion space, it is also the most obvious space for enforcing joint-angle constraints by orthogonal projection onto the subspace of valid orientations. These properties have, of course, been recognized and exploited in our field for many years [24, 25].

The joint limits representation we propose can therefore be understood as a way of encoding the workspace of the human upper arm positions using a formalism that could be applied to any individual joint, or set of coupled joints, in the human body model.

## 3 Measuring and Representing Shoulder and Elbow Motion

We will consider the set of possible joint orientations and positions in space as a path of referential frames in 3-D space [26]. In practice, we represent rotations by the sub-space of unit quaternions  $S^3$  forming a unit sphere in 4-dimensional space. Any rotation can be associated to a unit quaternion but we need to keep in mind that the unitary condition needs to be ensured at all times. A rotation of  $\theta$  radians around the unit axis  $v$  is described by the quaternion:

$$q = [q_x, q_y, q_z, q_w]^T = [\sin(\frac{1}{2}\theta)v, \cos(\frac{1}{2}\theta)]^T$$

Since we are dealing with unit quaternions, the fourth quaternion component  $q_w$  is a dependent variable and can be deduced, up to a sign, from the first three. Given data collected using optical markers, we obtain a cloud of 3-D points by keeping the spatial or  $(q_x, q_y, q_z)$  coordinates of the quaternion. In other

words, these three numbers serve as the coordinates of quaternions expressed as projections on three conventional Cartesian axes.

Because we simultaneously measure swing and twist components, and because the quaternion formalism lets us express both within one rotation, this representation captures the dependencies between swing and twist. In this manner, we will have generated joint limits on the basis of motion capture. We will then be able to make use of these joint limits as constraints for tracking and pose estimation, by eliminating all invalid configurations.

### 3.1 Motion Measurement

We captured shoulder and elbow motion using the Vicon<sup>TM</sup> System, with a set of strategically-placed markers on the upper arm as shown in Figure 1(a). An additional marker is placed at neck level to serve as a fixed reference.

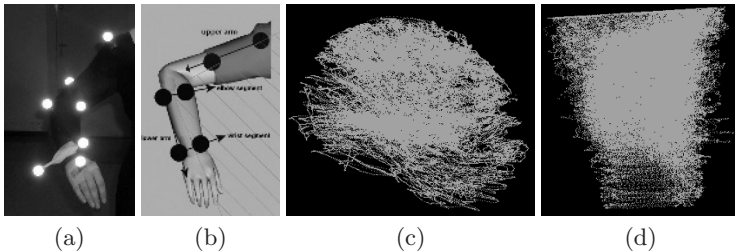
If we wish our joint limits to be as precise as possible, and to reflect the range of motion as closely as possible, we need to pay attention to sampling the space of attainable postures not only homogeneously, but also densely. In general, such a capture sequence lasts several minutes, so that we may obtain such a data set.

### 3.2 Motion Representation

To compute our quaternion field, each joint orientation is first converted to a  $3 \times 3$  matrix  $M$ , where, using Euler’s theorem,  $M$  may be expressed in terms of its lone real eigenvector  $\hat{\mathbf{n}}$  and the angle of rotation  $\theta$  about that axis. This in turn may be expressed as a point in quaternion space, or, equivalently, a point on a three-sphere  $S^3$  embedded in a Euclidean 4D space. The identification of the corresponding quaternion follows immediately from

$$q(\theta, \hat{\mathbf{n}}) = \left( \cos \frac{\theta}{2}, \hat{\mathbf{n}} \sin \frac{\theta}{2} \right) \tag{1}$$

up to the sign ambiguity between the two equivalent quaternions  $q$  or  $-q$ , which correspond to the exact same rotation matrix  $M$ . When referring to quaternion data, we will from here on always assume that its scalar component is positive.



**Fig. 1.** Marker positions and associated referentials. (a) Motion capture actor with markers. (b) Shoulder and elbow coordinate frame. (c) Quaternion shoulder data. (d) Elbow coordinate frame.

For each recorded position, we construct a rotating co-ordinate frame for the shoulder joint. As shown in Fig. 1(b), the first axis of the frame corresponds to the line defined by the shoulder and upper arm markers. The second axis is the normal to the triangle whose vertices are the upper arm, elbow and forearm markers. The corresponding plane represents axial rotation and the third axis is taken to orthogonal to the other two. The orientation of each frame is then converted into a quaternion, yielding the volumetric data depicted by Fig. 1(c). For the elbow, we also use the configuration depicted by Fig. 1(b) and transform all marker positions from the global referential to the local shoulder joint referential. Since the elbow has only two degrees of freedom, in Fig. 1(d), we represent the resulting data in terms of its two Euler angles.

## 4 Hierarchical Implicit Surface Representation of the Data

Given the volumetric data of Fig.1(c,e), we approximate it as an implicit surface. This will provide us with a smooth and differentiable representation of the space of allowable rotations and its associated metric, which we will use in Section 5 to enforce the corresponding constraints in a very simple manner. This is important because, having been produced by people instead of robots, this data is very noisy. In particular, the regions of lower point density often correspond to motion boundaries and therefore to uncomfortable positions.

### 4.1 Fitting an Implicit Surface

As in [27], given a set of spherical primitives of center  $S_i$  and thickness  $e_i$ , we define an implicit surface as

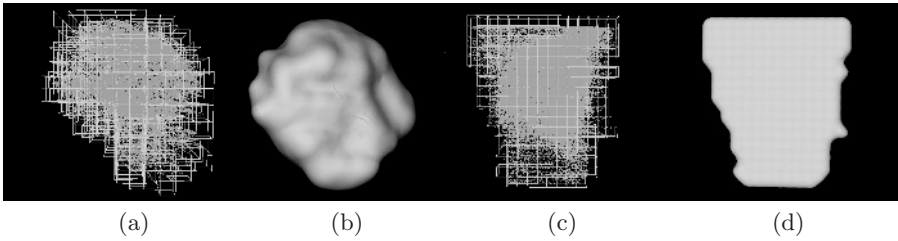
$$S = \{P \in \mathbb{R}^3 | F(P) = iso\} \quad (2)$$

where

$$F(P) = \sum_{i=1}^n f_i(P) \text{ and } f_i(P) = \begin{cases} -kd(P, S_i) + k_i e_i + 1 & \text{if } d(P, S_i) \in [0, e_i] \\ \frac{1}{4} [k(d(P, S_i) - e_i) - 2]^2 & \text{if } d(P, S_i) \in [e_i, R_i] \\ 0 & \text{elsewhere} \end{cases} ,$$

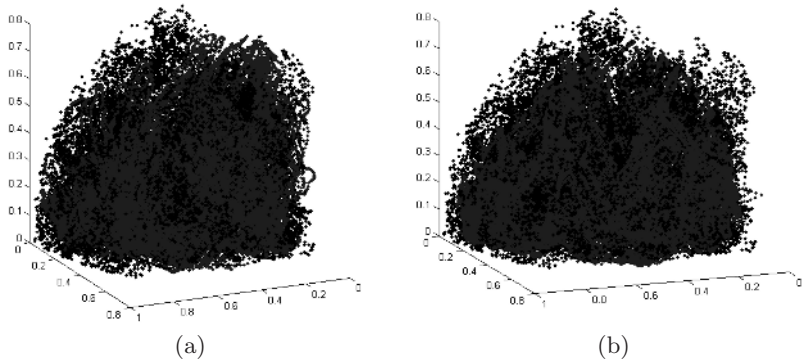
where  $d$  is the Euclidean distance,  $R_i = e_i + \frac{2}{k_i}$  is the radius of influence,  $iso$  controls the distance of the surface to the primitives' centers and  $k$  its blending properties.

The properties of implicit surfaces and their field functions being the same in 2 and 3-D, we apply the same fitting procedure to the 2-D data for the elbow joint as the 3-D data for the shoulder joint. We begin by voxelizing our space and assigning a point density to each voxel, in very much the same way the Marching Cubes algorithm does for the vertices of its voxels[28]. We then sub-divide the voxels until each voxel has a point density higher than a given threshold, which can be, for example, the density of the data around the center of mass.



**Fig. 2.** Joint limits for the shoulder and elbow joints. (a) Voxelization of the shoulder joint quaternions. (b) Extracted implicit surface. (c) Voxelization of the elbow joint Euler angles. (c) Extracted flat implicit surface.

As shown in Fig. 2(a,c), the resulting voxel arrays already represents the shape. We then simply place a spherical primitive at the center of each voxel and take its radius of influence to be half the width of the voxel, yielding the implicit surfaces depicted by Fig. 2(b,d), where  $iso = 7.0$  and the stiffness  $k = 20.0$ .

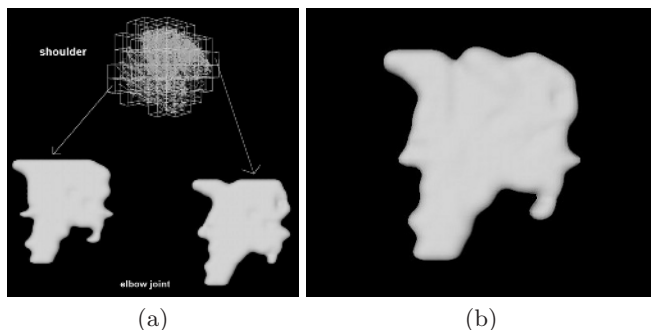


**Fig. 3.** Comparing subjects against each other. In black, the data for the female reference subject we used to compute the field function  $F$  of Eq. 2. In gray, the data corresponding to a second female subject (a) and to a male subject (b). We computed the average distance in terms of closest points between each cloud set, as well as the standard deviation. For (a), this yields an average distance of 0.0403 and a standard deviation of 0.0500. For (b), we obtain an average distance of 0.0314 and a standard deviation of 0.0432.

To illustrate the relative insensitivity of these measurements across subjects, we have gathered motion data for two additional people, one of each sex. In Fig. 3, we overlay the sets of quaternions for each additional person on those corresponding to the reference subject. Visual inspection in 3-D shows that they superpose well. This is confirmed by computing the average closest-point distance between the points of the three data sets, as well as the corresponding standard deviation.

## 4.2 Representing Dependencies

The method described above treats the data for the shoulder and the elbow independently, which does not account for known anatomical dependencies. To remedy this, we now take advantage of the voxel structure as follows. Each voxel for the parent joint, the shoulder in our case, defines a local cluster of similar joint positions, which we will refer to as *keyframe voxels*. As shown in Fig. 4(a), for each one, we compute the implicit *keyframe surface* corresponding to the subset of child joint rotations that have been observed for those positions of the parent joint. This follows directly from the fact that we simultaneously measure shoulder *and* elbow rotations. As shown in Fig. 4(b), to refine this representation and ensure a smoother transition between child joint limits from one keyframe voxel to the next, we can compute intermediate keyframe surfaces by morphing between neighboring ones [29].

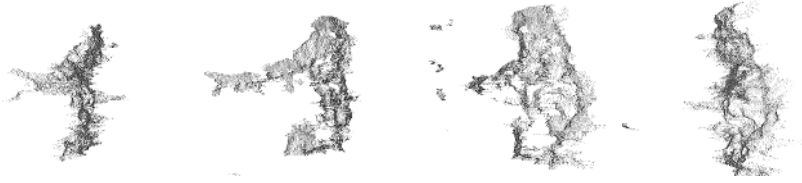


**Fig. 4.** Hierarchical joint limits. (a) Two keyframe voxels and the corresponding keyframe surfaces. (b) Intermediate keyframe surface obtained midway through morphing one keyframe surface into the other.

## 5 Enforcing Constraints during Tracking

To validate our approach to enforcing joint limits, we show that it dramatically increases the performance of an earlier system [12] that fits body models to stereo-data acquired using synchronized video cameras. It relies on attaching implicit surfaces, also known as soft objects, to an articulated skeleton to represent body shape. The field function of the primitives however differs from the one used for defining our joint limits in the sense that its density field is exponential, which increase the robustness of the system in the presence of erroneous data points. The skin is taken to be a level set of the sum of these fields. Defining the body model surface in this manner yields an algebraic distance function from 3-D points to the model that is differentiable. We can therefore formulate the problem of fitting our model to the stereo data in each frame as one of minimizing the sum of the squares of the distances of the model to the cloud of points produced by the stereo.





**Fig. 5.** Stereo data for a subject standing in the capture volume, rotated from a left-side view to a right-side view.

The stereo data is depicted by Fig. 5. It was acquired using a Digiclops<sup>tm</sup> operating at a  $640 \times 480$  resolution and a 14Hz framerate. It is very noisy, lacks depth, and gives no information on the side or the back of the subject. As a result, in the absence of constraints, there are many sets of motion parameters that fit the data almost as well, most of which correspond to anatomically impossible postures.

In this Section, we will show that enforcing the constraints using the formalism allows to eliminate these impossible postures very effectively and results in much more robust tracking.

## 5.1 Constrained Least Squares

In the absence of constraints, the posture in each frame is obtained by least-squares minimization of the algebraic distance of the stereo data points to the skin surface, defined as a level set of the field function of Eq. 2. Enforcing hierarchical constraints can be effectively achieved using well known task-priority strategies. Here we use a damped least-squares method that can handle potentially conflicting constraints [11]: When a high-priority constraint is violated, the algorithm projects the invalid posture onto the closest valid one, which requires computing the pseudo-inverse of its Jacobian matrix with respect to state variables, which in our case are the rotational values of the model's joints. When a lower-priority constraint is violated, the algorithm reprojects the Jacobians into the null-space of the higher level constraints so that enforcing the lower-order constraint does not perturb the higher level one.

More specifically, the body is modeled as an articulated structure to which we attach a number of volumetric primitives. This lets us define a field function  $D(x, \Theta)$  that represents the algebraic distance of a 3-D point  $x$  to the skin surface given the vector of joint angles values  $\Theta = [\Theta_1, \dots, \Theta_m]$ . In the absence of constraints, fitting the model to a set of  $n$  3-D points  $x_{i, 1 \leq i \leq n}$  simply amounts to minimizing

$$\sum_{i=1}^n D(x_i, \Theta)^2 \quad ,$$

with respect to  $\Theta$ . Given the Jacobian matrix  $J_D = \left( \frac{\partial D(x_i, \Theta)}{\partial \Theta_j} \right)_{1 \leq i \leq n, 1 \leq j \leq m}$  and its pseudo-inverse  $J_D^+$ , this involves iteratively adding to  $\Theta$  increments proportional to

$$\Delta\theta_0 = J_D^+[D(x_1, \theta), \dots, D(x_n, \theta)]^t .$$

Let us assume we are given a vector of constraints  $C$  with Jacobian matrix  $J_C$ . The problem becomes minimizing  $D$  subject to  $C(\theta) = 0.0$ . This can be done very much in the same way as before, except that the increments are now proportional to

$$J_C^+ C(\theta) + (I - J_C^+ J_C) \Delta\theta_0 ,$$

where  $(I - J_C^+ J_C)$  is the projector into the null space of  $C$ . This extends naturally to additional constraints with higher levels of priority, but additional care must be taken when constructing the projectors [11].

In short, all that is needed to enforce the constraints, is the ability to compute their Jacobian with respect to state variables. The implicit surface formulation of Section 4 lets us do this very simply:

1. For the parent joint, determine whether its rotation is valid by evaluating the function  $F$  of Eq. 2 and its derivatives with respect to joint angles if not. In other words, the higher priority constraint can be expressed as  $\max(0, iso - F(\theta))$  or, equivalently, treated as an inequality constraint.
2. For the child joint, determine to which voxel its parent rotation belongs, load the corresponding child joint limits, and verify its validity and, in case of an invalid rotation, evaluate the derivatives using the corresponding implicit surface representation. This allows us to express a lower priority constraint using the corresponding field function.

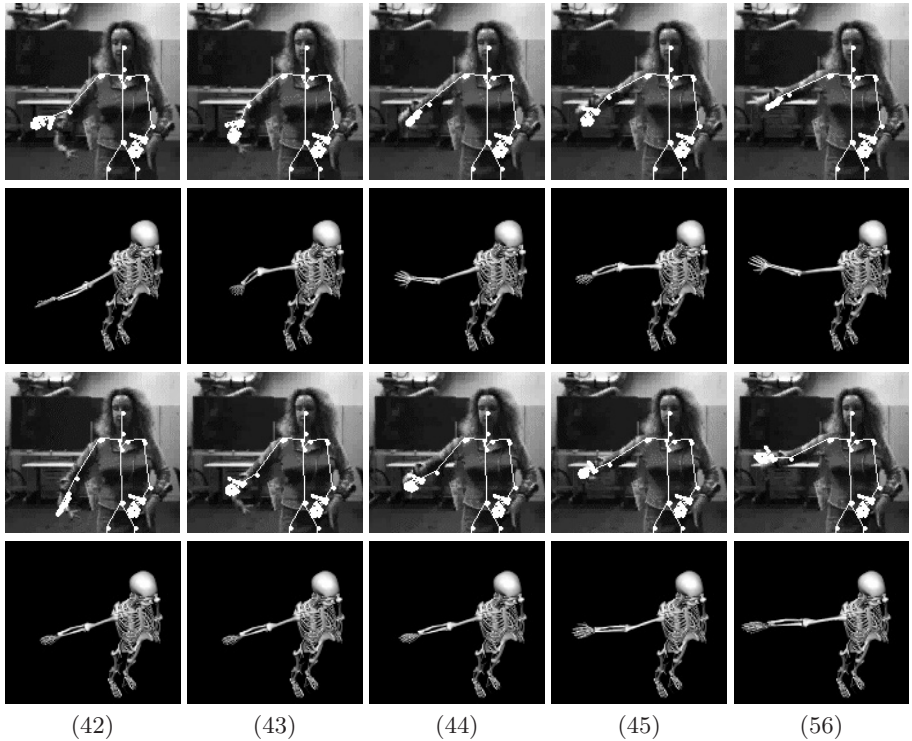
This results in an algorithm that fits the model to data, while enforcing the joint angles constraints at a minimal additional computational cost.

## 5.2 Tracking Results

We applied unconstrained and constrained tracking to several 100-frame long sequences, which corresponds to a little over 7 seconds at 14 Hz. In each sequence, the subject moves and rotates her right arm and elbow. In Figs 6, 7, and 8, we reproject the recovered 3-D skeleton onto one of the images. We also depict the skeleton as seen from a slightly different view to show whether or not the recovered position is feasible or not.

The unconstrained tracker performs adequately in many cases, but here we focus on the places where it failed, typically by producing the solution that matches the data but is not humanly possible. Among other things, this can be caused by the sparsity of the data or by the fact that multiple state vectors can yield limb positions that are almost indistinguishable given the poor quality of the data. We show that enforcing hierarchical joint limits on the shoulder and elbow joints during tracking allows our system to overcome these problems.

The interested reader can download mpeg movies for Figs. 7 and 8 from our <http://cvlab.epfl.ch/research/body/limits/limits.html> web site. They include the complete sequences along with depictions of the fit of the model to the 3-D data that are easier to interpret than the still pictures that appear in the printed version of the paper.



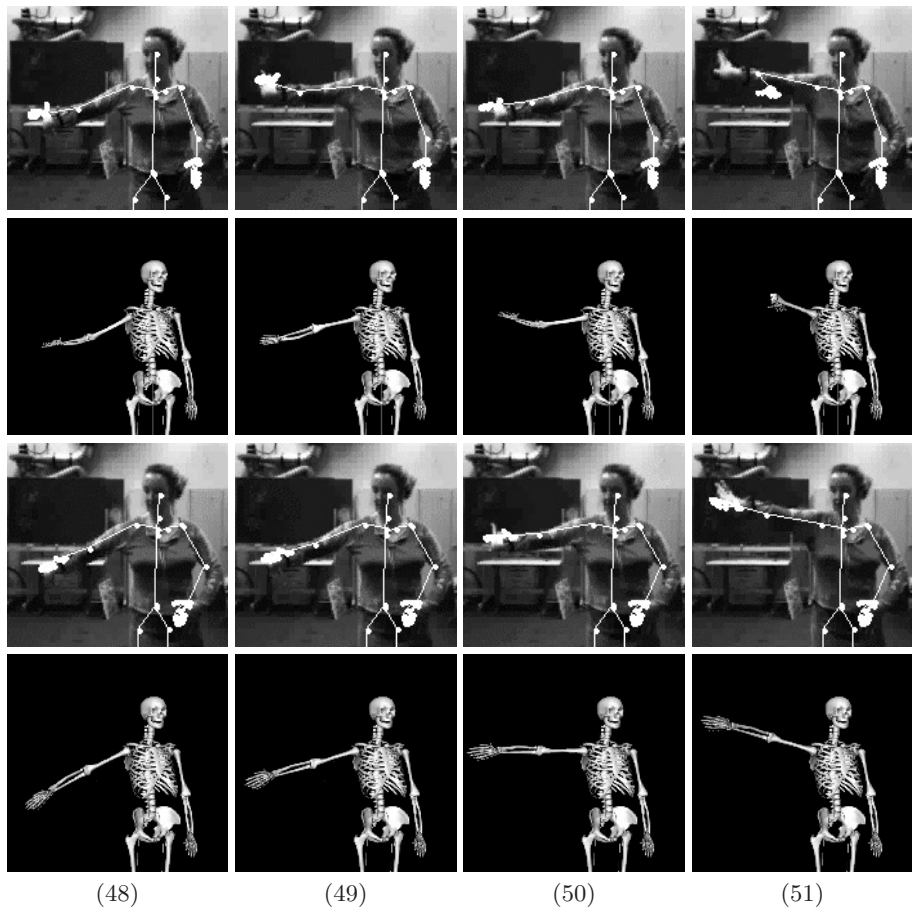
**Fig. 6.** Top rows: Unconstrained tracking. Bottom rows: Tracking with joint limits enforced. Up until the first frame shown here, the arm is tracked correctly in both cases. However, at frame 42, an elbow flexion occurs. In the unconstrained case, this is accounted for by backward bending of the elbow joint, which results in the correct reprojection but the absolutely impossible position of frame 56. By contrast, when the constraints enforced, the reprojection is just as good but the position is now natural.

## 6 Conclusion

We have proposed an implicit surface based approach to representing joint limits that account for both intra- and inter-joint dependencies. We have developed a protocol for instantiating this representation from motion capture data and shown that it can be effectively used to improve the performance of a body-tracking algorithm.

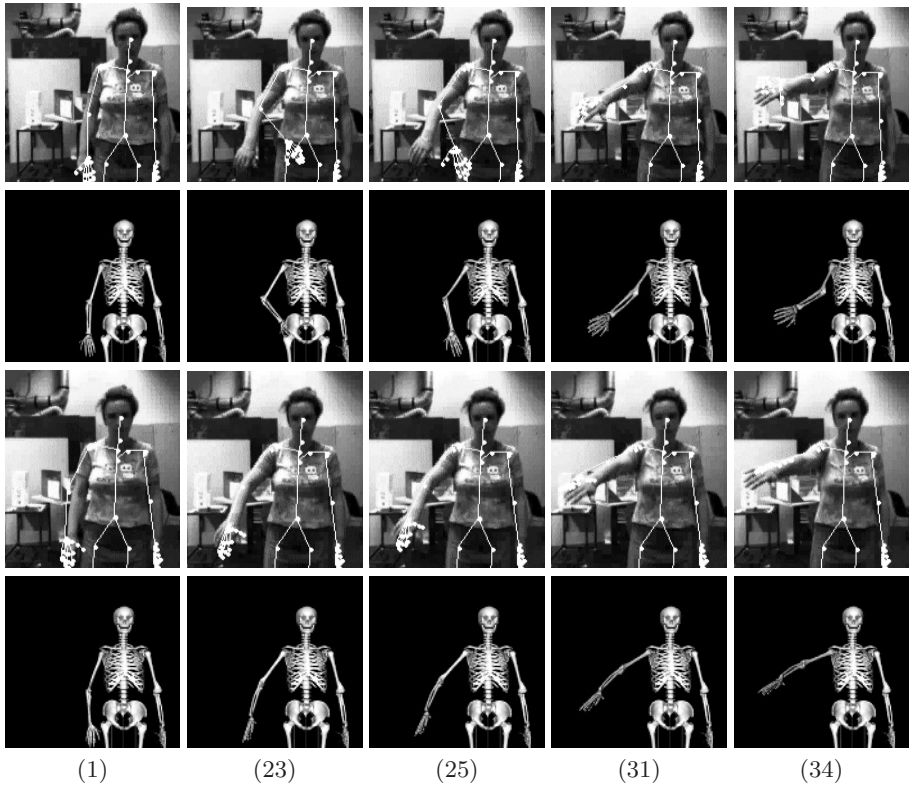
This effectiveness largely stems from the fact that our implicit surface representation allows us to quickly evaluate whether or not a constraint is violated and, if required, to enforce it using standard constrained optimization algorithms. We have demonstrated this in the specific case of the shoulder and elbow but the approach is generic and could be transposed to other joints, such as the hip and knee or the many coupled articulations in the hands and fingers.

The quality of the data we use to create our representation is key to its accuracy. The current acquisition process relies on optical motion capture. It is reasonably simple and fast, but could be improved further: Currently, when



**Fig. 7.** Top rows: Unconstrained tracking. Bottom rows: Tracking with joint limits enforced. Tracking without constraints results in excessive shoulder axial rotation at frame 50, followed by wildly invalid elbow extension on top of the incorrect shoulder twisting at frame 51. In this frame, there happens to be very little data for the forearm, which ends up being erroneously “attracted” by the data corresponding to the upper arm. As can be seen in the bottom rows, these problems go away when the constraints are enforced. The corresponding mpeg movies are available on the web at <http://cvlab.epfl.ch/research/body/limits/limits.html#Capture>.

sampling the range of motion of a joint, we have no immediate feed-back on whether we have effectively sampled the entire attainable space. To remedy this problem, we will consider designing an application that provides immediate visual feed-back directly during motion acquisition. This should prove very useful when extending the proposed technique to larger hierarchies of joints than the parent-and-child one considered in this paper.



**Fig. 8.** Top rows: Unconstrained tracking. Bottom rows: Tracking with joint limits enforced. In the absence of constraints, the shoulder axial rotation is wrong from frame 1 onwards. In frames 23 to 25, this results in the arm being erroneously “attracted” by the 3-D data corresponding to the hip. The tracker then recovers in frame 31, only to yield an invalid elbow flexion in frame 34. Once again, these problems disappear when the constraints are enforced. The corresponding mpeg movies are available on the web at <http://cvlab.epfl.ch/research/body/limits/limits.html#Capture>.

## References

1. Gavrilu, D.: The Visual Analysis of Human Movement: A Survey. *Computer Vision and Image Understanding* **73** (1999)
2. Moeslund, T., Granum, E.: Pose estimation of a human arm using kinematic constraints. In: *Scandinavian Conference on Image Analysis*, Bergen, Norway (2001)
3. Moeslund, T.: *Computer Vision-Based Motion Capture of Body Language*. PhD thesis, Aalborg University, Aalborg, Denmark (2003)
4. Rehg, J.M., Morris, D.D., Kanade, T.: Ambiguities in Visual Tracking of Articulated Objects using 2-D and 3-D Models. *International Journal of Robotics Research* **22** (2003) 393–418
5. Bregler, C., Malik, J.: Tracking People with Twists and Exponential Maps. In: *Conference on Computer Vision and Pattern Recognition*, Santa Barbara, CA (1998)
6. Demirdjian, D.: Enforcing constraints for human body tracking. In: *Workshop on Multi-Object Tracking*. (2003)

7. Sminchisescu, C., Triggs, B.: Estimating articulated human motion with covariance scaled sampling. *International Journal of Robotics Research* (2003)
8. Engin, A., Tümer, S.: Three-dimensional kinematic modeling of the human shoulder complex. *Journal of Biomechanical Engineering* **111** (1989) 113–121
9. Herda, L., Urtasun, R., Hanson, A., Fua, P.: An automatic method for determining quaternion field boundaries for ball-and-socket joint limits. *International Journal of Robotics Research* **22** (2003) 419–436
10. Shoemake, K.: Animating Rotation with Quaternion Curves. *Computer Graphics, SIGGRAPH Proceedings* **19** (1985) 245–254
11. Baerlocher, P., Boulic, R.: An Inverse Kinematics Architecture for Enforcing an Arbitrary Number of Strict Priority Levels. *The Visual Computer* (2004)
12. Plänkers, R., Fua, P.: Articulated Soft Objects for Multi-View Shape and Motion Capture. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2003)
13. Hatze, H.: A three-dimensional multivariate model of passive human joint torques and articular boundaries. *Clinical Biomechanics* **12** (1997) 128–135
14. Kodek, T., Munich, M.: Identifying Shoulder and Elbow Passive Moments and Muscle Contributions. In: *International Conference on Intelligent Robots and Systems*. (2002)
15. Johnston, R., Smidt, G.: Measurement of hip joint motion during walking. *Journal of Bone and Joint Surgery* **51** (1969) 1083–1094
16. Meskers, C., Vermeulen, H., de Groot, J., der Helm, F.V., Rozing, P.: 3d shoulder position measurements using a six-degree-of-freedom electromagnetic tracking device. *Clinical Biomechanics* **13** (1998) 280–292
17. der Helm, F.V.: A standardized protocol for motion recordings of the shoulder. In: *Conference of the International Shoulder Group, Maastricht, Netherlands* (1997)
18. Bao, H., Willems, P.: On the kinematic modelling and the parameter estimation of the human shoulder. *Journal of Biomechanics* **32** (1999) 943–950
19. Maurel, W.: 3d modeling of the human upper limb including the biomechanics of joints, muscles and soft tissues (1998)
20. Wang, X., Maurin, M., Mazet, F., Maia, N.D.C., Voinot, K., Verriest, J., Fayet, M.: Three-dimensional modelling of the motion range of axial rotation of the upper arm. *Journal of Biomechanics* **31** (1998) 899–908
21. Bobick, N.: Rotating objects using quaternions. *Game Developer* **2**, Issue **26** (1998)
22. Watt, A., Watt, M.: *Advanced animation and rendering techniques* (1992)
23. Grassia, F.: Practical parameterization of rotations using the exponential map. *Journal of Graphics Tools* **3** (1998) 29–48
24. Pervin, E., Webb, J.: Quaternions for computer vision and robotics. In: *Conference on Computer Vision and Pattern Recognition, Washington, D.C.* (1983) 382–383
25. Faugeras, O.: *Three-Dimensional Computer Vision: a Geometric Viewpoint*. MIT Press (1993)
26. Bloomenthal, J.: Calculation of reference frames along a space curve. In Glassner, A., ed.: *Graphics Gems*. Academic Press, Cambridge, MA (1990) 567–571
27. Tsingos, N., Bittar, E., Gascuel, M.: Implicit surfaces for semi-automatic medical organs reconstruction. In: *Computer Graphics International, Leeds, UK* (1995) 3–15
28. Lorensen, W., Cline, H.: Marching Cubes: A High Resolution 3D Surface Construction Algorithm. In: *Computer Graphics, SIGGRAPH Proceedings*. Volume 21. (1987) 163–169
29. V., R., A., F.: Shape transformations using union of spheres. Technical Report TR-95-30, Department of Computer Science, University of British Columbia (1995)