

# Hierarchical Implicit Surface Joint Limits for Human Body Tracking

Lorna Herda, Raquel Urtasun and Pascal Fua

Computer Vision Lab

EPFL

CH-1015 Lausanne, Switzerland

[Pascal.Fua@epfl.ch](mailto:Pascal.Fua@epfl.ch)

<http://cvlab.epfl.ch/>

EPFL Technical Report ID: IC/2004/09

## 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 [10, 22, 21], the problem remains far from solved. This is in part because image data is typically noisy and in part because it is inherently ambiguous [26]. As shown in Fig. 1, several postures, some of which are anatomically impossible, can explain the data equally well. 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 by eliminating a many local minima.

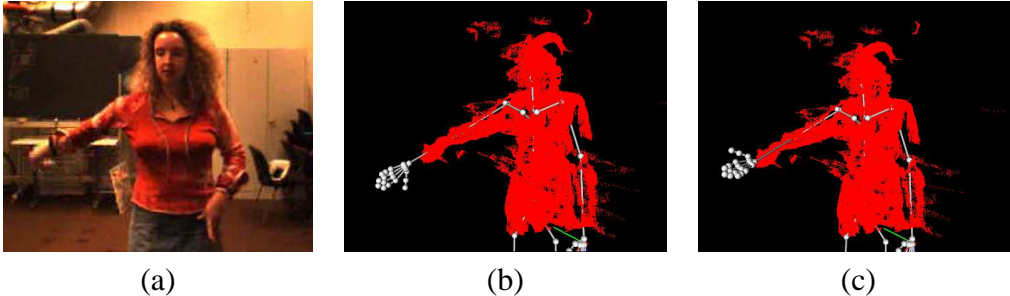


Figure 1: Motion capture from noisy stereo data. (a) One image from a stereo pair. (b,c) Two possible postures that account for the stereo data, which is depicted by the reprojections of triangulated 3-D points. These reprojected points appear in gray, or red if printed in color. Note the completely different shoulder and elbow twists that result in different hand orientations.

This is currently done in many existing vision systems [5, 6, 26, 29] 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 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 shoulder and elbow. 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 [15]. 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.

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 [24] 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 [14, 17] 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 [16] for the hip or [8, 20] 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 [21]. 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 [7, 2].

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 [32]. 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 [14, 17]. 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 [32]. 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, 22, 26, 29], 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 [4, 33]. The swing-twist representation, exponential map, and three-sphere embedding are all adequate to represent rotations and do not exhibit such flaws [11]. However, only quaternions are free of singularities [28]. As there is a good approximation of the 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 [23, 9].

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

For the shoulder and elbow coupled joint set, we will be using respectively quaternions and Euler angles to express their rotations. For the case of the shoulder joint, of all 3 DOF rotation representations, we opt for quaternions whose natural distance metric between rotations is well approximated by the Euclidean distance [18], thus supplying the most natural space in which to enforce 3 DOF joint-angle constraints by orthogonal projection onto the subspace of valid orientations [28]. Furthermore, quaternions are not subject to singularities such as the ‘‘Gimbal lock’’ of Euler angles or the mapping of  $2n\pi$  rotations to zero rotations of axis-angles. For the elbow joint, we have chosen to represent its 2 DOF rotation with two successive Euler angles, as this is the most compact representation for such a rotation (in terms of number of parameters), has no singularities in this configuration, and the rotation decomposition is unique, contrary to the 3 DOF case. As a result, it becomes easy to incorporate these sophisticated constraints into a motion tracking algorithm using standard constrained optimization techniques [1].

We will consider the set of possible joint orientations and positions in space as a path of referential frames in 3-D space [3]. 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.

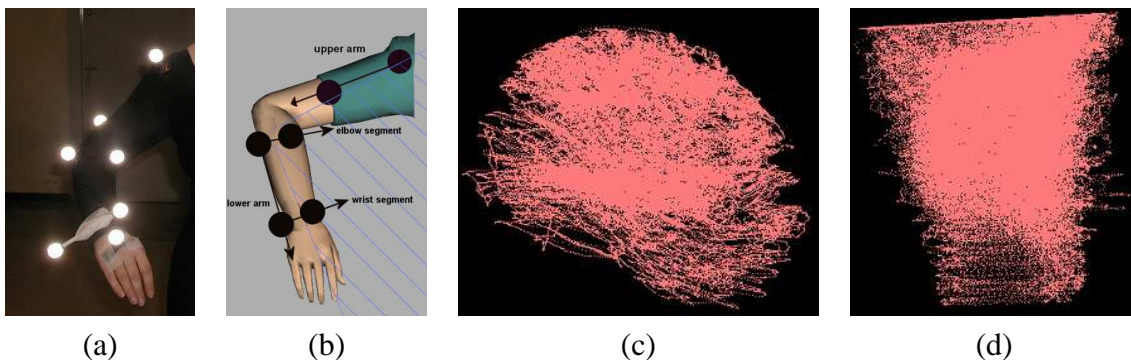


Figure 2: Marker positions and associated referentials. (a) Motion capture actor with markers. (b) Shoulder and elbow coordinate frame. (c) Quaternion shoulder data. (d) Euler angle elbow data.

Because we simultaneously measure swing and twist components, and because the quaternion formalism lets us express both within one rotation, this representation can capture the dependencies between swing and twist that will appear in our motion capture data.

### 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 2(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 as homogeneously, but also as densely as possible.

To acquire the data used in this paper, the motion capture actor was requested to place the upper arm at all possible elevations, and then to apply an incremental twist at the shoulder level. At each such position, the actor should then completely flex and extend the lower arm, as well as twist the forearm as far as possible in both directions. Once the entire reachable space has been so sampled, the Vicon system outputs the 3-D global positions of all the markers and labels them.

### 3.2 Motion representation

For each recorded position, we construct a rotating co-ordinate frame for the shoulder joint. As shown in Fig. 2(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.

This conversion is achieved by first converting the computed frame 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 4-D space. The identification of the corresponding quaternion follows immediately from

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

up to the sign ambiguity between the two equivalent quaternions  $q$  or  $-q$ , which correspond to the same rotation [13]. To resolve this ambiguity, we will from here on always assume that a quaternion’s scalar component is positive. Such an assumption however causes a discontinuity in the 3-D space of so-mapped quaternions, as shown by Fig. 3. In the illustrated case, we are carrying out a single axis rotation, the corresponding quaternions with a positive scalar component moving from the centre of the 3-D sphere towards its surface, along the axis. When the surface of the sphere is reached, we are at a rotation of approximately  $\pi$ . If we rotate further than  $\pi$ , the equivalent quaternion with  $q_w > 0$  appears on the opposite pole. We therefore need to keep in mind this phenomenon when measuring the distance between two quaternions [27], as in reality the two rotations represented by Fig. 3 are close, but in 3-D space end up far

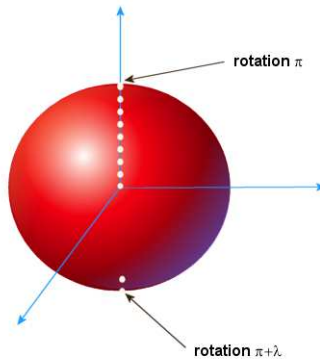


Figure 3: Discontinuity on the 3-sphere, for quaternions with a positive scalar component. When the rotations is equivalent to  $\pi$ , the corresponding quaternion is located on the surface of the sphere. As soon as the rotation exceeds  $\pi$ , the equivalent quaternion is situated at the opposite pole.

apart. In the case of joint rotations, however, we have positioned our local axes and defined our initial poses in such a manner as to never reach this discontinuity, all rotations involved being within  $] - \pi, +\pi[$ , but never including both ends of the interval. Special attention would need to be paid in a character animation context, when re-projecting an invalid rotation to the closest valid one, as in the case of a rotation exceeding  $\pi$  or  $-\pi$ , the rotation will get re-projected onto the wrong side of the unit quaternion space. The simplest means to avoid this is to limit the applicable range of motion to the  $[-\pi, +\pi]$  interval.

When converting our motion capture data in the manner described above, we obtain the volumetric data depicted by Fig. 2(c). For the elbow, we 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. 2(d), we represent the resulting data in terms of its two Euler angles. For 2 DOF rotations, two successive Euler angles are a perfectly acceptable representation [12], as they do not present a singularity in this configuration, and the decomposition of any rotation in two planes into a Euler angle is unique within the  $] - \pi, +\pi[$  interval.

## 4 Hierarchical Implicit Surface Representation of the Data

In order to capture the coupling between two joints in terms of range of motion, we propose a hierarchical scheme where for each set of similar postures of the parent joint, different joint limits are derived for the child joint. More precisely, joint limits, whether for parent or child joint, are represented by implicit surfaces. The hierarchical setup is based on a voxelisation of the parent joint range of motion, from which the child joint data sub-sets are then derived, to be in turn approximated by an implicit surface each.

Given the volumetric data of Fig.2(c,d), we approximate it as an implicit surface. This will provide us with a smooth and differentiable representation of the space of allowable rotation

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.

Implicit surfaces for shape reconstruction are extremely popular, and work well, under the condition that surface data is available, is sufficiently dense, and not too noisy. In our case, extraction of surface points through various methods proved unreliable, due to data undersampling for the postures that the motion capture actor deemed uncomfortable. Furthermore, our volumetric data is not smooth on the outside of the data cloud, and this added to the difficulty of attempting to derive surface points. For these reasons, we will approach the problem directly from its volumetric aspect.

## 4.1 Fitting an Implicit Surface

In order to get an approximate of the shape of the volumetric data, we voxelize our space and compute the point density of each voxel. This density corresponds to the number of points within each voxel, normalised with respect to voxel width. We then recursively 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. All voxels not satisfying this condition are discarded. Carrying out this voxelization for our shoulder and elbow data yields the results shown in Fig. 4(a,d), where the resulting voxel arrays already represent the shape. To obtain the implicit surface enclosing this shape, we propose to place an implicit surface primitive within each of the voxels. For this, we first define the primitives and implicit surface we use.

As in [30], given a set of spherical primitives of center  $S_i$  and thickness  $e_i$ , the implicit surface is defined as

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

where

$$F(P) = \sum_{i=1}^n f_i(P) \quad ,$$

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

where  $d = d(P, S_i)$  is the Euclidean distance,  $iso$  controls the distance of the surface to the primitives' surface, which is set by the thickness  $e_i$ , and  $k$  defines its blending properties. We additionally define a cut-off value at  $R_i = e_i + \frac{2}{k_i}$ , in order to ensure that the influence of each primitive is local, with respect to the total surface. All points beyond the radius of influence are discarded, and a spherical primitive so defined has a continuously decreasing function, as plotted in Fig. 5(a), for  $iso = 1.0$ ,  $k = 5.0$  and  $e = 1.0$ .

We place such a primitive in each voxel of our data voxelization, adjusting the primitive's parameters to that its radius of influence is half the width of the voxel, as in Fig. 5(b). This



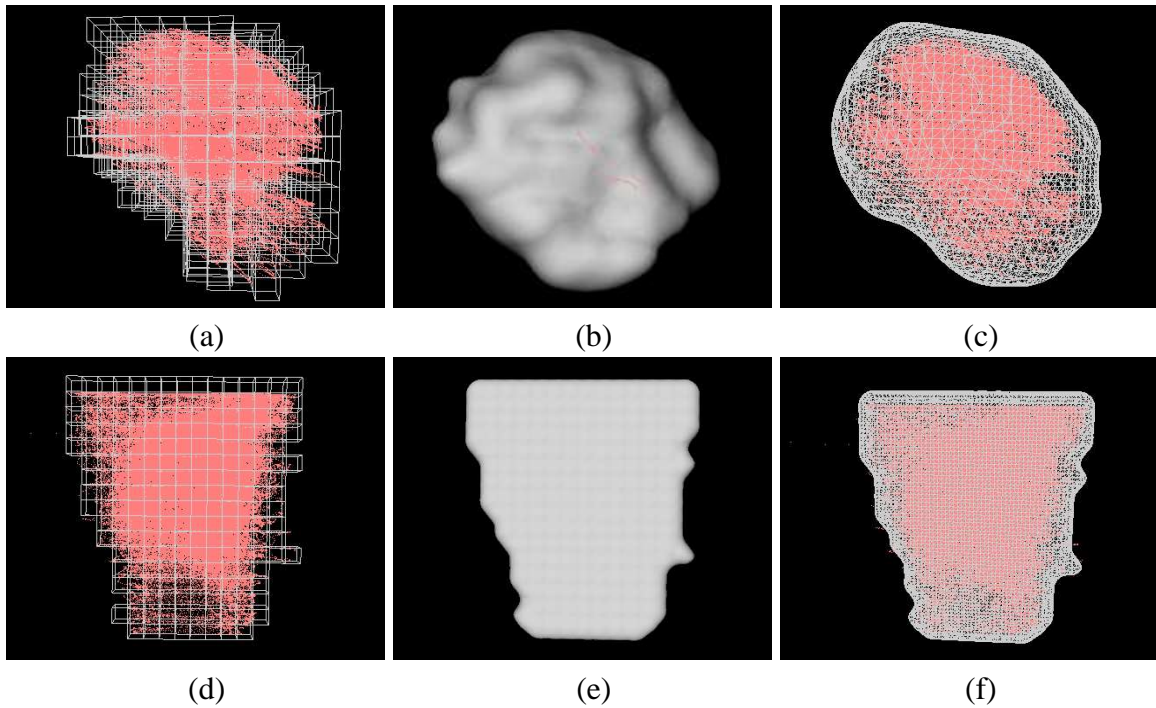


Figure 4: Joint limits for the shoulder and elbow joints. (a) voxelization of the shoulder joint quaternions. (b) extracted implicit surface. (c) wire-frame shoulder implicit surface and data. (d) voxelization of the elbow joint Euler angles. (e) extracted fat implicit surface. (f) wire-frame elbow implicit surface and data.

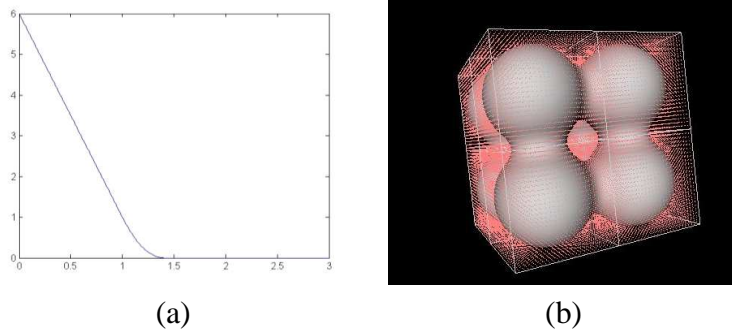


Figure 5: (a) Local influence of a spherical primitive. (b) Filling a voxelization with spherical primitives.

yields the implicit surfaces depicted by Fig. 4(b,e), where  $iso = 7.0$  and stiffness  $k = 20.0$ , these values having been determined experimentally. To see how closely our envelope fits our data, we display the implicit surface in wire-frame, in Fig. 4(c,f). 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 for the 3-D data of the shoulder joint.

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. 6, 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

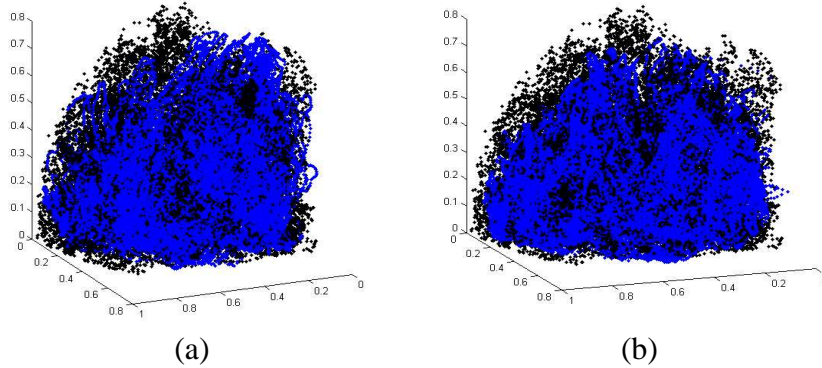


Figure 6: Comparing subjects against each other. In black, the data for the female reference subject we used to compute the field function  $F$  of Eq. 2. In gray, the data corresponding to a second female subject (a) and to a male subject (b). We computed the average distance in terms of closest points 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.

closest-point distance between the points of the three data-sets, as well as the corresponding standard deviation. The computed values highlight the similarity between the measures for the three subjects over the entire range of motion.

## 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. Having measured simultaneously the shoulder and elbow rotations, we could consider the coupled postures as 5-D vectors comprising the vectorial components of each rotation. However, the notion of implicit surface and especially the distance metric are not transposable to a dimension higher than 3-D, as we observed when attempting to generalise our method to 5-D. This being the case, we will instead work with hierarchical 3-D spaces. Our method is based on the observation that for each set of rotations of the shoulder joint, there is a defined set of acceptable rotations for the elbow joint. We take advantage of the voxel structure to obtain these data sets. Each voxel of the parent shoulder joint defines a local cluster of similar joint positions, which we will refer to as *keyframe voxels*. As to each measured shoulder joint rotation is associated an elbow joint position, we immediately obtain the sub-set of elbow rotations corresponding to this keyframe voxel. As shown in Fig. 7(a), for each keyframe voxel, 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.

As shown in Fig. 7(b), to refine this representation and ensure a smoother transition between elbow joint limits from one keyframe voxel to the next, we can compute intermediate keyframe surfaces by morphing between neighboring ones.

We have chosen to implement an interpolation scheme that morphs between unions of

spheres, and we will designate by A be the source object and by B the target object. We use the distance function between a primitive  $a$  of shape A and a primitive  $b$  of shape B defined by [31] as follows:

$$d(a, b) = [(x_a - x_b)^2 + (y_a - y_b)^2 + (z_a - z_b)^2] + (e_a - e_b) \quad (3)$$

where  $(x_a, y_a, z_a)$  is the centre and  $e_a$  the thickness of primitive  $a$  and  $(x_b, y_b, z_b)$  and  $e_b$  the corresponding parameters of primitive  $b$ .

Starting from the shape with lowest primitive cardinality, we perform an injective matching of its primitives with those of the other shape, such a matching being carried out between primitives that are closest in terms of the distance notion of eq.(3). After this matching, the shape that has the larger number of primitives is now left with some unmatched ones. These we simply match to the closest primitive of the other shape, therefore yielding a one-to-many match between the shapes. Once this matching has been established, we just need to interpolate between the centres and radii of the matched primitives, over the chosen number of interpolation steps.

In Fig. 8, we show the effect of applying hierarchical shoulder and elbow joint limits to a tennis serve motion where rotation on both joints where voluntarily set to invalid rotations, to demonstrate the effect of joint limits.

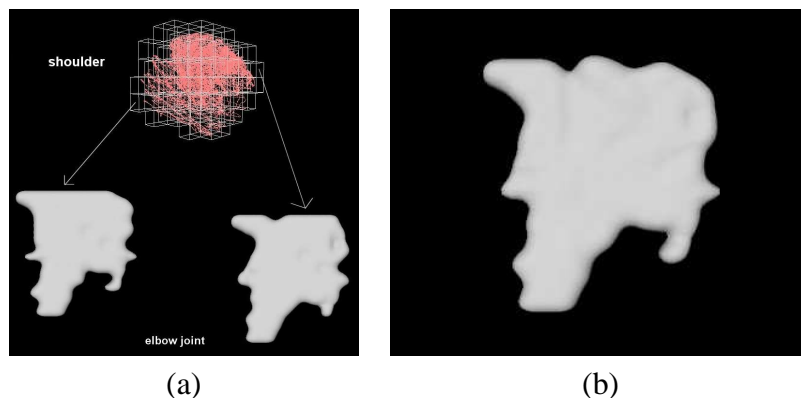


Figure 7: Hierarchical joint limits. (a) Two keyframe voxels and the corresponding keyframe surfaces. (b) Example of an intermediate keyframe surface obtained midway through morphing one keyframe surface into the other.

## 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 [25] 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

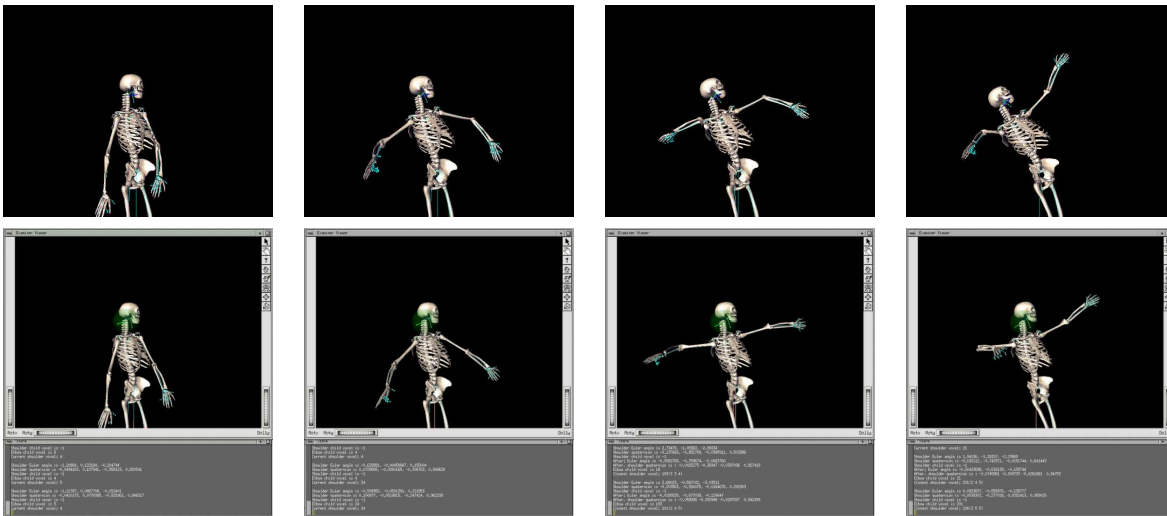


Figure 8: Applying hierarchical joint limits to a tennis serve keyframe sequence. In the top row, we show the frame of the sequence with invalid rotations both at the shoulder and elbow level. In the bottom row, the invalid rotations are corrected using the coupled implicit surface joint limits. The corresponding mpeg movies are available on the web at <http://cvlab.epfl.ch/research/body/limits/limits.html#Animation> .

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.

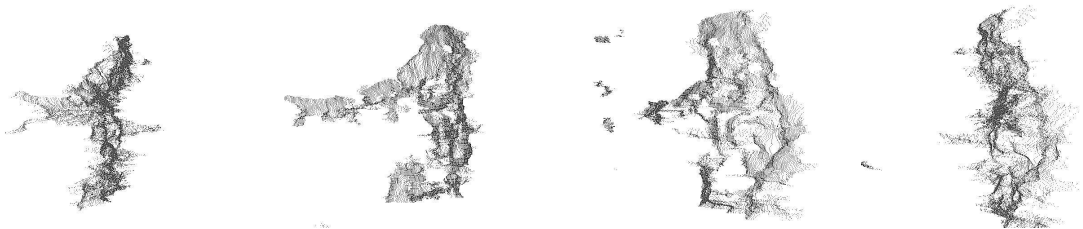


Figure 9: Stereo data for a subject standing in the capture volume, rotated from a left-side view to a right-side view.

The stereo data is depicted by Fig. 9. 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 Unconstrained Least Squares

To derive the posture of the body model from the stereo data, we apply the Levenberg-Marquardt least-squares optimiser. As discussed earlier, the body model is represented by an articulated structure to which volumetric primitives are attached. Let  $\Theta = (\Theta_1, \dots, \Theta_m)$  correspond to the vector of joint angle values defining the current posture of the model. Given  $n$  3-D data points  $1 \leq x_i \leq n$ , let  $D(x_i, \Theta)$  be the distance to be minimized, from the data points to the skin surface defined by the sum of the field functions of the primitive(s) minus the iso-value of the surface.

In the absence of constraints, fitting the model to  $n$  data points  $x_i$  simply amounts to minimizing:

$$\sum_{i=1}^n D(x_i, \Theta)^2 \quad (4)$$

with respect to  $\Theta$ . The expression of the derivative of  $D(x_i, \Theta)$  with respect to a parameter  $\Theta_j$  is given by [25]:

$$\frac{\partial D(x_i, \Theta)}{\partial \Theta_j} = 2 \cdot x_i^T \cdot Q_{\Theta}^T \cdot \left[ \frac{\partial Q_{\Theta}}{\partial \Theta_j} \right] \cdot x_i$$

where  $Q_{\Theta}$  defines the position, orientation and size of the primitive(s) the current observation is attached to, for state vector  $\Theta$ .

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

$$\Delta \Theta_0 = J_D^+ [D(x_1, \Theta), \dots, D(x_n, \Theta)]^t$$

to find the value of  $\Theta$  that minimizes  $D(x_i, \Theta)$ .

## 5.2 Constrained Least Squares

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 [1]: 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 rotation 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.

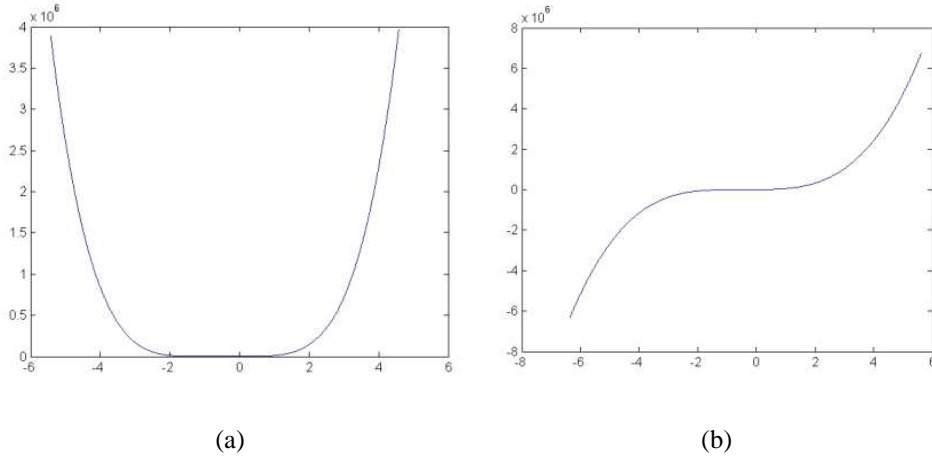


Figure 10: Objective function associated to a joint-limit constraint. (a) Value of the objective function value along a line drawn through the middle of an implicit surface with 16 primitives. (b) Gradient along the same line.

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

$$\Delta\Theta_1 = J_C^+ C(\Theta) + (I - J_C^+ J_C) \Delta\Theta_0 ,$$

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 [1].

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 evaluate the derivatives using the corresponding implicit surface representation. This allows us to express a lower priority constraint using the corresponding field function.

In practice, for each constraint, the algorithm minimizes

$$c(\Theta) = \begin{cases} (F(\theta) - iso)^2 & \text{if } F(\theta) < iso \\ 0 & \text{elsewhere} \end{cases} ,$$

whose behavior is depicted by Fig. 10. This is natural given that the points for which  $F(\Theta) = iso$  correspond to the largest allowable rotations.  $c(\Theta)$  is smooth and convex, thereby guaranteeing that joint limit constraints will be satisfied at every iteration. It is also algebraic and its derivatives can be computed by differentiating the  $f_i$  polynomials of Eq. 2.

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

### 5.3 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 11, 12, and 13, 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 identical error values, each state vector equally explaining the data, and each such state representing a local minima of the error function. 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 11, Figs. 12 and 13. 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, of necessity, still pictures that appear in the printed version of the paper.

## 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



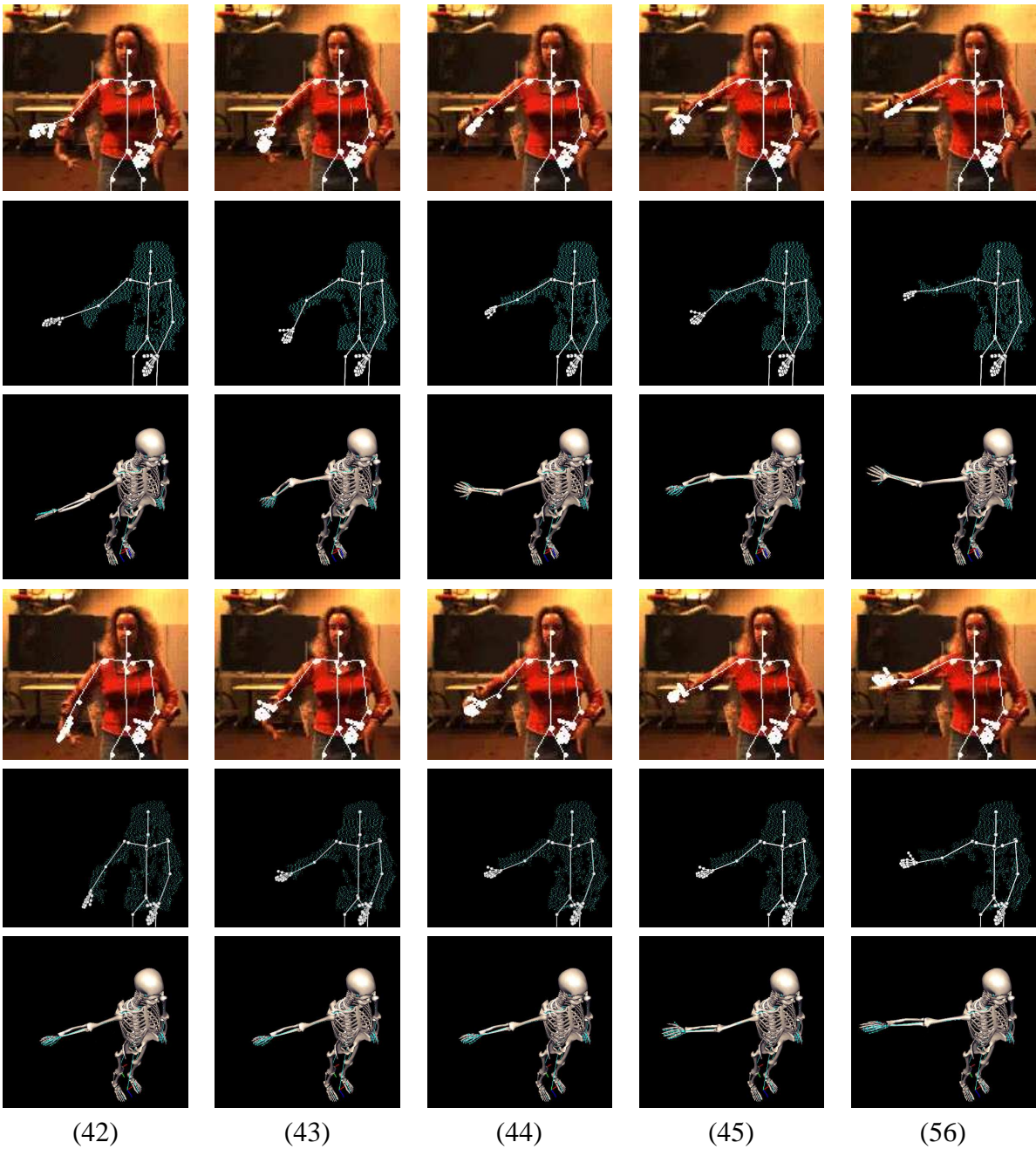


Figure 11: Top rows: Unconstrained tracking. Bottom rows: Tracking with joint limits enforced. Up until the first frame shown here, the arm is tracked correctly in both cases. However, at frame 42, an elbow flexion occurs. In the unconstrained case, this is accounted for by backward bending of the elbow 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.

be improved further: currently, when 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.

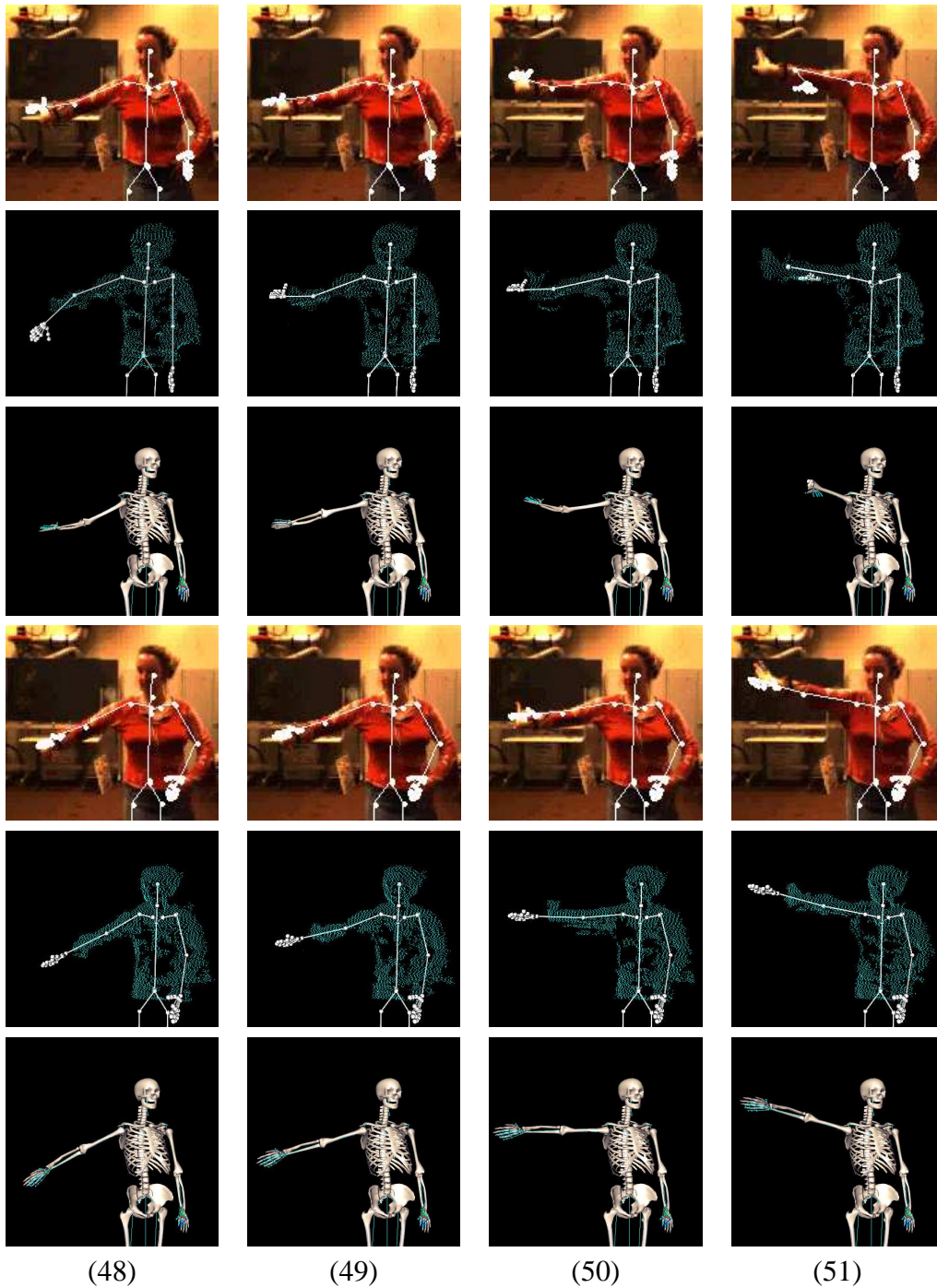


Figure 12: Top rows: Unconstrained tracking. Bottom rows: Tracking with joint limits enforced. Tracking without constraints results in excessive shoulder axial rotation at frame 50, followed by wildly invalid elbow extension on top of the incorrect shoulder twisting at frame 51. In this frame, there 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> .

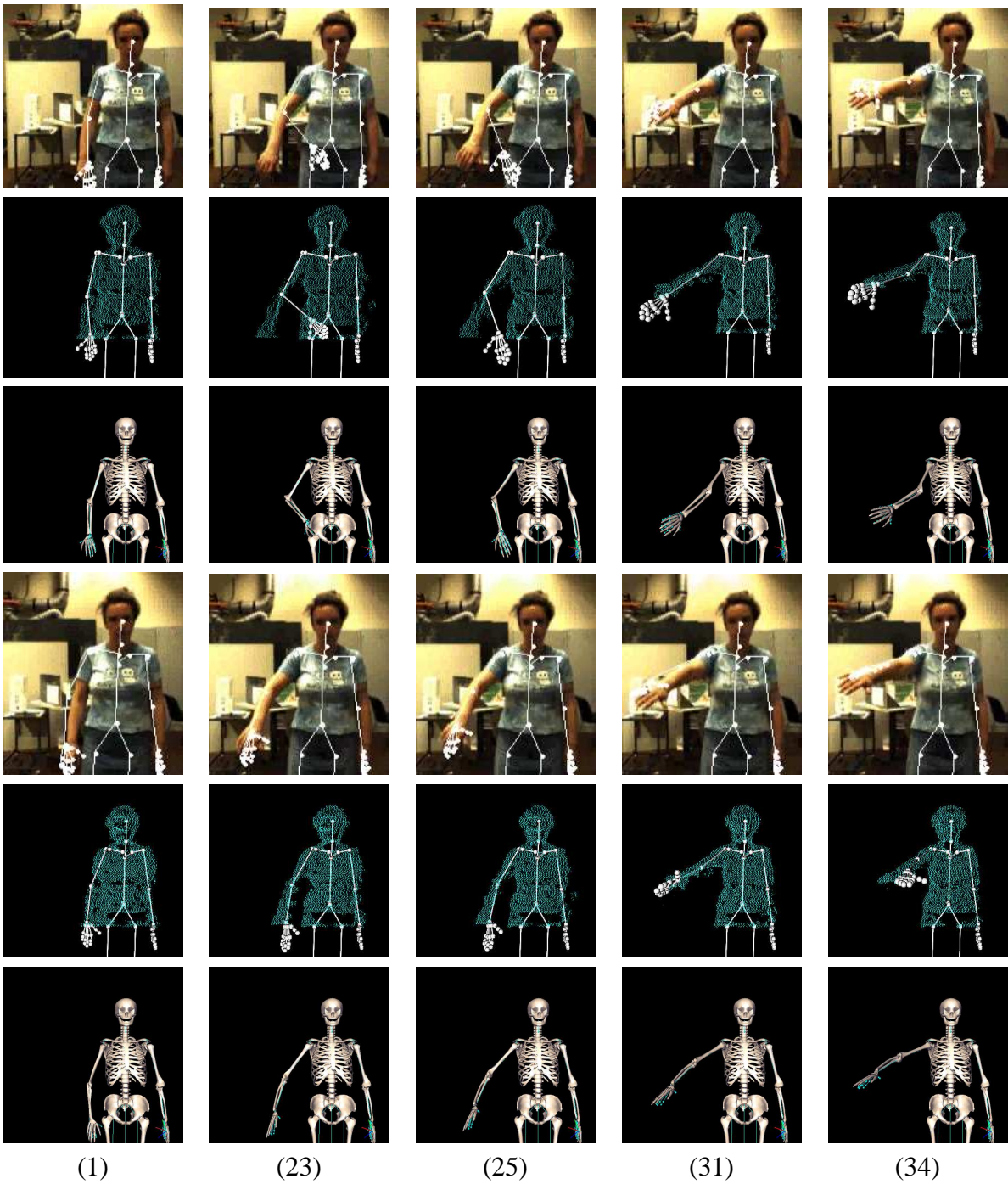


Figure 13: Top rows: Unconstrained tracking. Bottom rows: Tracking with joint limits enforced. In the absence of constraints, the shoulder axial rotation is wrong from frame 1 onwards. In frames 23 to 25, this results in the arm being erroneously “attracted” by the 3-D data corresponding to the hip. 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.epflch/research/body/limits/limits.html#Capture>.

## References

- [1] P. Baerlocher and R. Boulic. Task priority formulations for the kinematic control of highly redundant articulated structures. In *International Conference on Intelligent Robots and Systems*, pages 323–329, Victoria, Canada, 1998.
- [2] H. Bao and P.Y. Willems. On the kinematic modelling and the parameter estimation of the human shoulder. *Journal of Biomechanics*, 32(9):943–950, 1999.
- [3] Jules Bloomenthal. Calculation of reference frames along a space curve. In Andrew Glassner, editor, *Graphics Gems*, pages 567–571. Academic Press, Cambridge, MA, 1990.
- [4] N. Bobick. Rotating objects using quaternions. *Game Developer*, 2, Issue 26, 1998.
- [5] Ch. Bregler and J. Malik. Tracking People with Twists and Exponential Maps. In *Conference on Computer Vision and Pattern Recognition*, Santa Barbara, CA, June 1998.
- [6] D. Demirdjian. Enforcing constraints for human body tracking. In *Workshop on Multi-Object Tracking*, 2003.
- [7] F.C.T. Van der Helm. A standardized protocol for motion recordings of the shoulder. In *Conference of the International Shoulder Group*, Maastricht, Netherlands, 1997.
- [8] A.E. Engin and S.T. Tümer. Three-dimensional kinematic modeling of the human shoulder complex. *Journal of Biomechanical Engineering*, 111:113–121, 1989.
- [9] O.D. Faugeras. *Three-Dimensional Computer Vision: a Geometric Viewpoint*. MIT Press, 1993.
- [10] D.M. Gavrila. The Visual Analysis of Human Movement: A Survey. *Computer Vision and Image Understanding*, 73(1), January 1999.
- [11] F.S. Grassia. Practical parameterization of rotations using the exponential map. *Journal of Graphics Tools*, 3(3):29–48, 1998.
- [12] Sebastian Grassia. A practical parameterization of 2 and 3 degree of freedom rotations. Technical Report CMU-CS-97-143, School of Computer Science, Carnegie Mellon University, Pittsburgh, USA, 1997.
- [13] A.J. Hanson. Constrained optimal framings of curves and surfaces using quaternion gauss maps. In *Visualization*, pages 375–382. IEEE Computer Society Press, 1998.
- [14] H. Hatze. A three-dimensional multivariate model of passive human joint torques and articular boundaries. *Clinical Biomechanics*, 12:128–135, 1997.
- [15] L. Herda, R. Urtasun, A.J. Hanson, and P. Fua. An automatic method for determining quaternion field boundaries for ball-and-socket joint limits. *International Journal of Robotics Research*, 22(6):419–436, 2003.
- [16] R. Johnston and G. Smidt. Measurement of hip joint motion during walking. *Journal of Bone and Joint Surgery*, 51(A):1083–1094, 1969.

- [17] T. Kodek and M. Munich. Identifying Shoulder and Elbow Passive Moments and Muscle Contributions. In *International Conference on Intelligent Robots and Systems*, 2002.
- [18] J. Lawton and R. Beard. Model independent approximate eigenaxis rotations via quaternion feedback. Technical report, Brigham Young University, Utah, USA, 2001.
- [19] W. Maurel. 3d modeling of the human upper limb including the biomechanics of joints, muscles and soft tissues, 1998.
- [20] C.G.M. Meskers, H.M. Vermeulen, J.H. de Groot, F.C.T. Van der Helm, and P.M. Rozing. 3d shoulder position measurements using a six-degree-of-freedom electromagnetic tracking device. *Clinical Biomechanics*, 13:280–292, 1998.
- [21] T.B. Moeslund. *Computer Vision-Based Motion Capture of Body Language*. PhD thesis, Aalborg University, Aalborg, Denmark, June 2003.
- [22] T.B. Moeslund and E. Granum. Pose estimation of a human arm using kinematic constraints. In *Scandinavian Conference on Image Analysis*, Bergen, Norway, 2001.
- [23] E. Pervin and J.A. Webb. Quaternions for computer vision and robotics. In *Conference on Computer Vision and Pattern Recognition*, pages 382–383, Washington, D.C., 1983.
- [24] R. Plänkers and P. Fua. Articulated Soft Objects for Video-based Body Modeling. In *International Conference on Computer Vision*, pages 394–401, Vancouver, Canada, July 2001.
- [25] R. Plänkers and P. Fua. Articulated Soft Objects for Multi-View Shape and Motion Capture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2003.
- [26] J. M. Rehg, D. D. Morris, and T. Kanade. Ambiguities in Visual Tracking of Articulated Objects using 2–D and 3–D Models. *International Journal of Robotics Research*, 22(6):393–418, 2003.
- [27] J. Schmidt and H. Niemann. Using Quaternions for Parametrizing 3–D Rotations in Unconstrained Nonlinear Optimization. In T. Ertl, B. Girod, G. Greiner, H. Niemann, and H.-P. Seidel, editors, *Vision, Modeling, and Visualization*, pages 399–406, Stuttgart, Germany, 2001. AKA/IOS Press, Berlin, Amsterdam.
- [28] K. Shoemake. Animating Rotation with Quaternion Curves. *Computer Graphics, SIGGRAPH Proceedings*, 19:245–254, 1985.
- [29] C. Sminchisescu and B. Triggs. Estimating articulated human motion with covariance scaled sampling. *International Journal of Robotics Research*, 2003.
- [30] N. Tsingos, E. Bittar, and M.P. Gascuel. Implicit surfaces for semi-automatic medical organs reconstruction. In *Computer Graphics International*, pages 3–15, Leeds, UK, 1995.
- [31] Ranjan V. and Fournier A. Shape transformations using union of spheres. Technical Report TR-95-30, Department of Computer Science, University of British Columbia, 1995.
- [32] X. Wang, M. Maurin, F. Mazet, N. De Castro Maia, K. Voinot, J.P. Verriest, and M. Fayet. Three-dimensional modelling of the motion range of axial rotation of the upper arm. *Journal of Biomechanics*, 31(10):899–908, 1998.
- [33] A. Watt and M. Watt. Advanced animation and rendering techniques, 1992.