

# Articulated three-dimensional human modelling from motion capture systems

João K. Fayad, Alessio Del Bue, Pedro M. Q. Aguiar  
Instituto Superior Técnico  
Av. Rovisco Pais 1 – 1049-001 Lisboa – Portugal

## Abstract

*Human 3D models and motion analysis are nowadays used in a wide range of applications, spanning from medicine to security and surveillance. In this work, we will focus on the creation of biomechanical models for clinical and sports application of these methods. State of the art motion capture systems are capable of measuring with sufficient accuracy the 3D coordinates of reflective markers placed above the skin. The main issues clinical gait analysis is currently facing are the repeatability of the measurements and the compensation of soft-tissue motion.*

*Here we present a general framework to automatically recover joint parameters modelling the human articulations, from the 3D coordinates of a point cloud provided by motion capture systems. Additionally, we describe an approach capable of recovering a more accurate rigid body description of non-rigid bodies, and able to deal with the problem of marker occlusion. We then propose a new quadratic model to explain soft-tissue artifacts formed as a natural extension of the existing rigid body models. The parameters of this model are computed using an algorithm for large scale non-linear optimisation.*

*Finally we use synthetic data to assess the performance of the algorithms and to compare the results with ground truth data. Qualitative analysis of real data sequences over different motion capture databases are also presented.*

## 1. Introduction

Computational models of human articulations are fundamental to perform an accurate analysis of the mechanical motion of a human body. Applications of these models span various fields, ranging from engineering to life sciences, where the analysis of human motion is crucial to accurate clinical analysis and credible animations of skeleton models [1, 6, 12, 14]. In this work we will focus on the task of building biomechanical models of the human body from motion capture (MOCAP) systems.

MOCAP systems are devices able to recover a 3D description of the motion of a shape (see [8] for a complete review of the state of the art). The most usual setup for optical MOCAP systems is composed by calibrated infrared cameras (at least

two, but typically more than six) arranged around the area to be captured, while passive (reflective) markers, with diameter between 9 to 25 mm, are placed on the surface of the object [1]. The output of these systems are the 3D coordinates of the markers over time. State of the art MOCAP systems are able to acquire the 3D trajectories of the markers with an error of the order of 1mm.

One of the main problems of current existing methods for joint parameter estimation based on MOCAP systems is their *limited repeatability* [1]. This is a strong setback on the applications of these methods as accuracy and repeatability is of extreme importance in clinical analysis. Another major source of error in these analysis are the *soft-tissue artifacts* [4, 3, 13]. The relevant information about articulations used to build human models is given by the skeleton. At this scale, bones can be considered to be rigid. However the reflective markers used by MOCAP systems are placed above the skin. Thus, when examining a subject, there is an inherent relative motion between the soft-tissues surrounding bone and bone itself. These relative motions create data artifacts that degrade performances.

A key step in joint parameter estimation is the *model calibration* [1]. To obtain clinically relevant data, the model must be subject specific. Most of current methods use some kind of regression technique to fit the models to the subjects, being bound to the quality of the regression. On the other hand, a great number of these methods require markers to be placed at specific anatomical landmarks. Such positioning is affected by human errors and it has the further problem of not always being possible to locate those landmarks. Thus, methods that do not rely on specific landmarks locations would be preferred.

In this work we propose methods to semi-automatically compute human skeleton models while dealing with the current limitations of the existent methods. Using recent developments in Structure from Motion (SfM) algorithms that allow retrieval of articulated structures from a set of 2D images [19, 17], we propose a method to extract joint parameters based on the 3D coordinates provided by the MOCAP systems. This approach does not depend on the process by which the data was acquired. All it assumes is to know to each body segment the 3D coordinates belong to. Inspired by previous works on the field of computer graphics [16, 15], we present a new quadratic model for non-rigid bodies. By modelling non-rigid bodies we will be able to separate the rigid contribution (skeleton) from the non-

rigid one (soft-tissue) and provide more accurate estimates for the joint models.

In Section 2 we provide the rigid body formulation for rigid bodies moving independently and coupled by universal and hinge joints. In Section 3 we discuss a weighted factorization approach to deal with non-rigidity and occlusion. Section 4 describes our quadratic model for non-rigid bodies, while in Section 4.3 we present the Bundle Adjustment method used to estimate its parameters. In Section 5 we present a performance analysis of our algorithms applied to synthetic data. We also apply them to real data for qualitative assessment. Finally, we conclude in Section 6.

## 2. Rigid Body Factorization

Factorization methods for SfM are a family of image based algorithms that model moving objects as a product of two factors: *motion* and *shape*. Our factorization approach assumes a set of  $P$  3D points being tracked over  $F$  frames by a MOCAP system. This method relies on the key fact that 3D trajectories of points belonging to the same body share the same global properties.

We can thus define for a rigid body the coordinates of marker  $j$  at frame  $i$  as:

$$\mathbf{w}_{ij} = \left[ \begin{array}{c|c} \mathbf{r}_{1i}^T & t_{xi} \\ \mathbf{r}_{2i}^T & t_{yi} \\ \mathbf{r}_{3i}^T & t_{zi} \end{array} \right] \begin{pmatrix} x_j \\ y_j \\ z_j \\ 1 \end{pmatrix} = [ \mathbf{R}_i \mid \mathbf{t}_i ] \begin{bmatrix} \mathbf{s}_j \\ 1 \end{bmatrix}, \quad (1)$$

where  $\mathbf{R}_i$  is a  $3 \times 3$  rotation matrix,  $\mathbf{t}_i$  is a 3-vector representing the translation and  $\mathbf{s}_j$  is a 3-vector that has the global shape parameters for point  $j$ . Stacking these eqs. for all the  $F$  frames and  $P$  points, we may form a single  $3F \times P$  measurement matrix  $\tilde{\mathbf{W}}$  such that:

$$\tilde{\mathbf{W}} = \begin{bmatrix} \tilde{\mathbf{w}}_1 \\ \tilde{\mathbf{w}}_2 \\ \vdots \\ \tilde{\mathbf{w}}_F \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{R}_2 \\ \vdots \\ \mathbf{R}_F \end{bmatrix} \begin{bmatrix} x_1 & x_2 & \cdots & x_P \\ y_1 & y_2 & \cdots & y_P \\ z_1 & z_2 & \cdots & z_P \end{bmatrix} + \begin{bmatrix} \mathbf{T}_1 \\ \mathbf{T}_2 \\ \vdots \\ \mathbf{T}_F \end{bmatrix} = \mathbf{M}\mathbf{S} + \mathbf{T}, \quad (2)$$

where  $\mathbf{T}_i = \mathbf{t}_i \mathbf{1}_P^T$ , with  $\mathbf{1}_P^T$  being a  $P$ -vector with all entries equal to 1. The translational component  $\mathbf{t}_i$  can be computed as the coordinates of the centroid of the point cloud at each frame  $\tilde{\mathbf{w}}_i$ . Thus it can be easily eliminated by registering, at each frame, the point cloud to the origin. In this scenario it frequently occurs that instead of  $\tilde{\mathbf{W}}$  we consider a registered version  $\tilde{\tilde{\mathbf{W}}}$  defined as:

$$\tilde{\tilde{\mathbf{W}}} = \tilde{\mathbf{W}} - \mathbf{T} = \mathbf{M}\mathbf{S}. \quad (3)$$

### 2.1. Independent Body Motion

Considering the model defined in eq. 2 for a single body, it follows that  $\text{rank}(\tilde{\mathbf{W}}) \leq 4$  [10]. On the other hand if we defined the system without the translational component as in

eq. (3) we have that  $\text{rank}(\tilde{\tilde{\mathbf{W}}}) \leq 3$ . However, when performing real experiments, there will always be noise involved which will increase the rank of  $\tilde{\tilde{\mathbf{W}}}$ . Noise can originate for instance from the MOCAP system's uncertainty in the position of the tracked feature points or in possible non-rigidity of the tracked objects.

Let us consider the  $\text{rank} - 3$  truncated SVD  $\tilde{\tilde{\mathbf{W}}} = \mathbf{U}_3 \Sigma_3 \mathbf{V}_3^T$ . This decomposition is not only useful for noise reduction, as we apply an ideal rank constraint to the data matrix, but it can also be used as the starting point for the factorization algorithm. Considering the expected dimensions of  $\mathbf{M}$  and  $\mathbf{S}$  we can compute a first estimation as:

$$\hat{\mathbf{M}} = \mathbf{U}_3 \Sigma_3^{1/2}; \quad (4)$$

$$\hat{\mathbf{S}} = \Sigma_3^{1/2} \mathbf{V}_3^T. \quad (5)$$

However there exists an ambiguity in this factorization as  $\tilde{\tilde{\mathbf{W}}} = \hat{\mathbf{M}}\hat{\mathbf{S}} = \hat{\mathbf{M}}\mathbf{Q}\mathbf{Q}^{-1}\hat{\mathbf{S}} = \mathbf{M}\mathbf{S}$ . Since this factorization does not guarantee that  $\hat{\mathbf{M}}$  is in fact a collection of  $F$  rotation matrices, the ambiguity is solved by finding the matrix  $\mathbf{Q}$  that will transform each  $3 \times 3$  matrix  $\hat{\mathbf{M}}_i$  in a rotation matrix  $\mathbf{R}_i$ . This can be achieved by imposing the following orthogonality constraints:

$$\mathbf{m}_{ik}^T \mathbf{H} \mathbf{m}_{ik} = 1, \quad (6)$$

$$\mathbf{m}_{ik}^T \mathbf{H} \mathbf{m}_{il} = 0, \quad l \neq k, \quad (7)$$

where  $k, l = 1, 2, 3$ ,  $\mathbf{m}_{ik}$  and  $\mathbf{m}_{il}$  are respectively the  $k$ -th and  $l$ -th row of matrix  $\hat{\mathbf{M}}_i$ , and  $\mathbf{H} = \mathbf{Q}\mathbf{Q}^T$  is symmetric  $3 \times 3$  matrix.  $\mathbf{Q}$  can thus be recovered from  $\mathbf{H}$  using Cholesky decomposition. We then correct the factorization eqs. (4) and (5) as:

$$\mathbf{M} = \hat{\mathbf{M}}\mathbf{Q} \quad (8)$$

$$\mathbf{S} = \mathbf{Q}^{-1} \hat{\mathbf{S}}. \quad (9)$$

Notice that the solution of this system of equations makes use of the whole information stored in the motion matrices. In such way we obtain a transformation which is given globally for all the frames in the dataset.

### 2.2. Universal Joint

In a universal joint, each of the two bodies is at a fixed distance to the joint centre. In this scenario, the relative position of the bodies is constrained but their rotations remain independent. A scheme of this joint is presented in Figure 1.

Let  $\mathbf{d}^{(1)} = [u, v, w]^T$  be the 3D coordinates of the joint centre in the local referential of the first body;  $-\mathbf{d}^{(2)} = [u', v', w']^T$  be the 3D coordinates of the joint centre in the local referential of the second body;  $\mathbf{R}^{(1)}$  and  $\mathbf{R}^{(2)}$  the  $3F \times 3$  matrices corresponding to a collection of  $3 \times 3$  global rotation matrices over  $F$  frames for the first and second body respectively;  $\mathbf{t}^{(1)}$  and  $\mathbf{t}^{(2)}$  the  $3F$ -vectors corresponding respectively to the first and second body global translation vectors.

The joint centre can thus be seen as a point that belongs to both bodies. In other words, its position can be described using

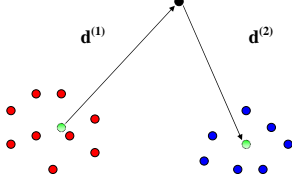


Figure 1. Scheme of a universal joint. The first body is represented by red points and the second by blue points. The joint centre is represented by a black point. The 3-vectors  $\mathbf{d}^{(1)}$  and  $-\mathbf{d}^{(2)}$  are respectively the 3D coordinates of the joint centre in the local referential of the first and second body.

the motion eqs. for the first and second body. With these considerations, a geometrical analysis of the joint structure reveals that:

$$\mathbf{R}^{(1)}\mathbf{d}^{(1)} + \mathbf{t}^{(1)} = -\mathbf{R}^{(2)}\mathbf{d}^{(2)} + \mathbf{t}^{(2)}. \quad (10)$$

Thus we can state that both 4D subspaces have a 1D intersection. The result of this consideration is that the measurement matrix of the universal joint must have  $rank(\mathbf{W}) \leq 7$ , one dimension less when comparing to the case of two independent where  $rank(\mathbf{W}) \leq 8$ . We are now able to factorize the measurement matrix as:

$$\begin{aligned} \mathbf{W} &= \left[ \mathbf{W}^{(1)} \mid \mathbf{W}^{(2)} \right] \\ &= \left[ \mathbf{R}^{(1)} \quad \mathbf{R}^{(2)} \quad \mathbf{t}^{(1)} \right] \begin{bmatrix} \mathbf{S}^{(1)} & \mathbf{D}^{(1)} \\ \mathbf{S}^{(2)} + \mathbf{D}^{(2)} \\ \mathbf{1}_{P_1}^T & \mathbf{1}_{P_2}^T \end{bmatrix}. \end{aligned} \quad (11)$$

where  $\mathbf{W}^{(1)}$  and  $\mathbf{W}^{(2)}$  are respectively the measurement matrices for the first and second body. The matrices  $\mathbf{D}^{(1)}$  and  $\mathbf{D}^{(2)}$  are given such that  $\mathbf{D}^{(1)} = \mathbf{d}^{(1)} \mathbf{1}_{P_2}^T$  and  $\mathbf{D}^{(2)} = \mathbf{d}^{(2)} \mathbf{1}_{P_2}^T$ . The components  $\mathbf{1}_{P_1}$  a  $P_1$ -vector with all entries equal to 1 and  $\mathbf{1}_{P_2}$  a  $P_2$ -vector with all entries equal to 1, where  $P_1$  is the number of points belonging to the first body, and  $P_2$  the number of points belonging to the second body. Notice that in order to separate  $\mathbf{W}^{(1)}$  from  $\mathbf{W}^{(2)}$ , we must assume the body segmentation to be known.

To recover the structure of the joint one needs to find  $\mathbf{d}^{(1)}$  and  $\mathbf{d}^{(2)}$ . From eq. (10) we can write:

$$\left[ \mathbf{R}^{(1)}, \mathbf{R}^{(2)}, \mathbf{t}^{(2)} - \mathbf{t}^{(1)} \right] \begin{bmatrix} \mathbf{d}^{(1)} \\ \mathbf{d}^{(2)} \\ -1 \end{bmatrix} = 0. \quad (12)$$

Therefore the joint parameters  $\mathbf{d}^{(1)}$  and  $\mathbf{d}^{(2)}$  are easily computed once we have found the motion parameters  $\mathbf{R}^{(1)}$ ,  $\mathbf{R}^{(2)}$ ,  $\mathbf{t}^{(1)}$  and  $\mathbf{t}^{(2)}$ . As the universal joint constraint is described by a dependency between the translational components, registering the measurement matrix will result in a full-rank-6 system defined by:

$$\tilde{\mathbf{W}} = \left[ \mathbf{R}^{(1)} \mid \mathbf{R}^{(2)} \right] \begin{bmatrix} \mathbf{S}^{(1)} & 0 \\ 0 & \mathbf{S}^{(2)} \end{bmatrix}, \quad (13)$$

where  $\mathbf{S}^{(1)}$  is a  $3 \times P_1$  global shape matrix for the first body and  $\mathbf{S}^{(2)}$  is a  $3 \times P_2$  global shape matrix for the second body. The initial step in the factorization is again done by performing an SVD and by keeping the first six components:

$$\tilde{\mathbf{W}} = \mathbf{U}_6 \Sigma_6^{1/2} \Sigma_6^{1/2} \mathbf{V}_6^T = \left[ \mathbf{U}^{(1)} \mid \mathbf{U}^{(2)} \right] \left[ \mathbf{V}^{(1)} \mid \mathbf{V}^{(2)} \right]. \quad (14)$$

However  $\left[ \mathbf{V}^{(1)} \mid \mathbf{V}^{(2)} \right]$  is a dense matrix while the structure matrix defined in eq. (13) is block diagonal. If we define an operator  $N_l(\cdot)$  that returns the left null-space of its argument, we can define a  $6 \times 6$  transformation matrix  $\mathbf{T}_U$  such that:

$$\mathbf{T}_U = \begin{bmatrix} N_l(\mathbf{V}^{(2)}) \\ N_l(\mathbf{V}^{(1)}) \end{bmatrix}. \quad (15)$$

We can now recover  $\mathbf{S}$  by pre-multiplying it with  $\mathbf{T}_U$  giving:

$$\begin{aligned} \mathbf{S} &= \begin{bmatrix} N_l(\mathbf{V}^{(2)}) \\ N_l(\mathbf{V}^{(1)}) \end{bmatrix} \left[ \mathbf{V}^{(1)} \mid \mathbf{V}^{(2)} \right] \\ &= \begin{bmatrix} \mathbf{S}^{(1)} & 0 \\ 0 & \mathbf{S}^{(2)} \end{bmatrix}. \end{aligned} \quad (16)$$

As we must keep the original data unaltered, we have to post-multiply  $\left[ \mathbf{U}^{(1)} \mid \mathbf{U}^{(2)} \right]$  with  $\mathbf{T}_U^{-1}$ :

$$\mathbf{M} = \left[ \mathbf{U}^{(1)} \mid \mathbf{U}^{(2)} \right] \begin{bmatrix} N_l(\mathbf{V}^{(2)}) \\ N_l(\mathbf{V}^{(1)}) \end{bmatrix}^{-1} = \left[ \mathbf{M}^{(1)} \mid \mathbf{M}^{(2)} \right], \quad (17)$$

where  $\mathbf{M}^{(1)}$  and  $\mathbf{M}^{(2)}$  are respectively the motion matrix for the first and second body. Note that the ambiguity seen in Section 2.1 is still present for each body, being no guarantee that  $\mathbf{M}^{(1)}$  or  $\mathbf{M}^{(2)}$  are a collection of  $3 \times 3$  rotation matrices. Due to the specific configuration of  $\mathbf{S}$  seen in eq. (13), there is no linear method to impose the orthogonality constraints to  $\mathbf{M}$  while assuring that structure for  $\mathbf{S}$ . We have chosen to separate  $\mathbf{W}^{(1)}$  and  $\mathbf{W}^{(2)}$  treating them individually in the same way it was done for the independent rigid body in Section 2. Even though it is a suboptimal solution, as we are not using all the available data to solve the ambiguity, it is good approximation and it uses a simple linear form. Thus we apply in each case the transformation matrix  $\mathbf{Q}$  as used before on eqs. (8) and (9).

### 2.3. Hinge Joint

In a hinge joint, two bodies can rotate around an axis such that the distance to that rotation axis is constant. Therefore their rotation matrices  $\mathbf{R}^{(1)}$  and  $\mathbf{R}^{(2)}$  are not completely independent. A scheme of the hinge joint is presented in Figure 2.

In this kind of joint, any vector belonging to any of the two bodies that is parallel to the joint axis must remain so throughout the movement. Let us choose an appropriate local referential, without loss of generality, where the axis of rotation of the

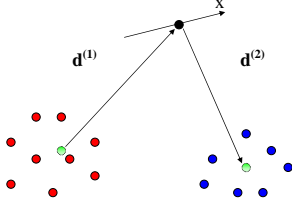


Figure 2. Scheme of a hinge joint. The first body is represented by red points and the second one by blue points. The joint centre is represented by a black point. The  $x$ -axis is the rotation axis. The 3-vectors  $\mathbf{d}^{(1)}$  and  $-\mathbf{d}^{(2)}$  are respectively the 3D coordinates of the joint centre in the local referential of the first and second body.

joint is coincident with the  $x$ -axis. Thus, in order to comply with the hinge joint constraint, the first column of  $\mathbf{R}^{(1)}$  must be equal to the first column of  $\mathbf{R}^{(2)}$ . We can now define the rotation matrices as  $\mathbf{R}^{(1)} = [\mathbf{c}_1 \ \mathbf{c}_2 \ \mathbf{c}_3]$  and  $\mathbf{R}^{(2)} = [\mathbf{c}_1 \ \mathbf{c}_4 \ \mathbf{c}_5]$ . As all the points belonging to the rotation axis must fulfil both movement conditions, this results in a 2D intersection of the original 4D subspaces. Thus, when considering translations,  $\tilde{\mathbf{w}}$  is given by:

$$\tilde{\mathbf{w}} = [\mathbf{c}_1 \ \mathbf{c}_2 \ \mathbf{c}_3 \ \mathbf{c}_4 \ \mathbf{c}_5 \ \mathbf{t}^{(1)}] \begin{bmatrix} x_1^{(1)} \cdots x_{P_1}^{(1)} & x_1^{(2)} \cdots x_{P_2}^{(2)} \\ y_1^{(1)} \cdots y_{P_1}^{(1)} & y_1^{(2)} \cdots y_{P_2}^{(2)} \\ z_1^{(1)} \cdots z_{P_1}^{(1)} & z_1^{(2)} \cdots z_{P_2}^{(2)} \\ \mathbf{1}_{P_1}^T & \mathbf{1}_{P_2}^T \end{bmatrix}, \quad (18)$$

obtaining  $\text{rank}(\tilde{\mathbf{w}}) = 6$ . Consequently, registering each body to the origin will result in  $\text{rank}(\tilde{\mathbf{w}}) = 5$ .

Once again we use the truncated SVD of  $\tilde{\mathbf{w}}$  as the first step on the parameter estimation. Constraining  $\tilde{\mathbf{w}}$  to a  $\text{rank} - 5$  matrix eq. (14) become:

$$\tilde{\mathbf{w}} = [\mathbf{U}^{(1)} \mid \mathbf{U}^{(2)}] [\mathbf{V}^{(1)} \mid \mathbf{V}^{(2)}]. \quad (19)$$

Where  $[\mathbf{U}^{(1)} \mid \mathbf{U}^{(2)}]$  is a  $3F \times 5$  matrix, and  $[\mathbf{V}^{(1)} \mid \mathbf{V}^{(2)}]$  is a  $5 \times (P_1 + P_2)$  dense matrix. Similarly as before we have to enforce the specific structure as in eq. (18). Let  $\mathbf{T}_H$  be a transformation matrix such that:

$$\mathbf{T}_H = \begin{bmatrix} \mathbf{b}^T \\ N_l(\mathbf{v}^{(2)}) \\ N_l(\mathbf{v}^{(1)}) \end{bmatrix}, \quad (20)$$

where  $\mathbf{b}^T = [1 \ 0 \ 0 \ 0 \ 0]$ . By pre-multiplying  $[\mathbf{V}^{(1)} \mid \mathbf{V}^{(2)}]$  with  $\mathbf{T}_H$  we leave the first row intact and we zero-out some entries in order to get the structure presented in eq. (18). As observed in Section 2.1, in this approach arises an ambiguity, which will be solved in the same way defined for the universal joint.

In this case, the joint centre can lie anywhere on the axis of rotation. Still it must obey the motion eqs. for both bodies.

Combining eq. (10) with the properties of  $\mathbf{R}^{(1)}$  and  $\mathbf{R}^{(2)}$  for the hinge joint, the null-space problem defined by eq. (12) can now be stated for this case as:

$$[\mathbf{c}_1 \ \mathbf{c}_2 \ \mathbf{c}_3 \ \mathbf{c}_4 \ \mathbf{c}_5 \ \mathbf{t}^{(2)} - \mathbf{t}^{(1)}] \begin{bmatrix} u + u' \\ v \\ w \\ v' \\ w' \\ -1 \end{bmatrix} = 0. \quad (21)$$

As the rotation axis is parallel to the  $x$ -axis, we can now represent it by a parametric eq. of a line  $l(\alpha)$  defined as:

$$l(\alpha) = [\mathbf{c}_1 \ \mathbf{c}_2 \ \mathbf{c}_3] [\alpha, v, w] + \mathbf{t}^{(1)} \quad \forall \alpha \in \mathbb{R}. \quad (22)$$

### 3. Quasi-rigid Objects and Weighted Factorization

When using an SVD to estimate the motion and shape parameters, the resulting shape will be the one that minimises the error in a least-squares sense over all the frames. Nonetheless this might not be the best representation of the rigid component of the non-rigid shape. Factorizing with the previous algorithms can be seen as averaging the shape throughout the frames, resulting in an attenuation of the deformations. Inspired by [9], we present an approach that uses a weighted SVD in order to penalise the contribution of the points which deform most. By doing so we will attenuate the deformations contribution and we will attain a more accurate rigid representation of the body.

If we define  $\tilde{\mathbf{w}}^{(r)}$  as the best rigid description of a non-rigid body, a measure of the non-rigidity of a given point on the matrix  $\tilde{\mathbf{w}}$  can be given by how distant its trajectory is from that best rigid description. With that in mind, let us rearrange the registered data matrix  $\tilde{\mathbf{w}}$  as:

$$\tilde{\mathbf{w}} = \begin{bmatrix} \tilde{\mathbf{w}}_{11}^T & \tilde{\mathbf{w}}_{21}^T & \cdots & \tilde{\mathbf{w}}_{P_1}^T \\ \tilde{\mathbf{w}}_{21}^T & \tilde{\mathbf{w}}_{22}^T & \cdots & \tilde{\mathbf{w}}_{P_2}^T \\ \vdots & \vdots & & \vdots \\ \tilde{\mathbf{w}}_{F1}^T & \tilde{\mathbf{w}}_{F2}^T & \cdots & \tilde{\mathbf{w}}_{F3}^T \end{bmatrix}, \quad (23)$$

and let us consider a similar rearrangement of  $\tilde{\mathbf{w}}^{(r)}$ . The distance vector between a point  $j$  in the reconstructed and best description can thus be given by:

$$\mathbf{E}_j = \tilde{\mathbf{w}}_j^{(r)} - \tilde{\mathbf{w}}_j, \quad (24)$$

with  $\mathbf{E}_j$  an  $F \times 3$ . Thus,  $\|\mathbf{E}_j\|$  will represent a distance between the points, being higher for highly deformable points. A weight matrix that satisfies the conditions stated earlier can thus be defined as:

$$\mathbf{C}_j = \text{cov}(\mathbf{E}_j)^{-1}, \quad (25)$$

as deformable points are bound to originate higher covariance values. Still,  $\tilde{\mathbf{w}}_j^{(r)}$  and thus  $\mathbf{C}_j$  are unknown in this problem. Indeed, this does not allow us to solve the optimization in closed

form. Therefore, we choose to solve the weighted factorization by considering the following cost function:

$$\arg \min_{\mathbf{M}_i, \mathbf{s}_j} \sum_{i=1}^F \sum_{j=1}^P (\tilde{\mathbf{w}}_{ij} - \mathbf{M}_i \mathbf{s}_j)^T \mathbf{C}_j (\tilde{\mathbf{w}}_{ij} - \mathbf{M}_i \mathbf{s}_j), \quad (26)$$

where we can iteratively solve for the components  $\mathbf{M}_i$ ,  $\mathbf{s}_j$  and  $\mathbf{C}_j$  with three different least square estimations. In order to achieve this, we have to express  $\mathbf{M}_i$  and  $\mathbf{s}_j$  as a system by rearranging eq. (26). The  $\mathbf{C}_j$  can be then obtained simply by applying eq. (24) and (25) where we approximate  $\tilde{\mathbf{w}}_j^{(r)}$  with the current rigid estimates  $\mathbf{M}_i$  and  $\mathbf{s}_j$ . In such way the algorithm keep penalising points which are showing a non-rigid behavior (i.e. points which not fit the rigid model).

If we keep fixed  $\mathbf{M}$  and with some matrix manipulations, we have that the solution for  $\mathbf{s}_j$  is given by:

$$\mathbf{s}_j = \left( \sum_{i=1}^F \mathbf{M}_i^T \mathbf{C}_j \mathbf{M}_i \right)^{-1} \sum_{i=1}^F \mathbf{M}_i \tilde{\mathbf{w}}_{ij}. \quad (27)$$

This eq. computes  $\mathbf{S}$  performing simple matrix products and the inversion of a small  $3 \times 3$  matrix.

On the other hand if we assume  $\mathbf{S}$  to be known, a similar linear solution can be found. We must first rearrange  $\mathbf{M}_i$  into a 9-vector  $\mathbf{m}_i$  defined by:

$$\mathbf{m}_i = \begin{bmatrix} \mathbf{r}_{1i} \\ \mathbf{r}_{2i} \\ \mathbf{r}_{3i} \end{bmatrix}, \quad (28)$$

and then rearrange  $\mathbf{s}_j$  into a  $3 \times 9$  block diagonal matrix  $\mathbf{S}_j$  such that:

$$\mathbf{S}_j = \begin{bmatrix} \mathbf{s}_j^T & & \\ & \mathbf{s}_j^T & \\ & & \mathbf{s}_j^T \end{bmatrix}. \quad (29)$$

Based on eq. (26) we can now compute each vector  $\mathbf{m}_i$  as

$$\mathbf{m}_i = \left( \sum_{j=1}^P \mathbf{S}_j^T \mathbf{C}_j \mathbf{S}_j \right)^{-1} \sum_{j=1}^P \mathbf{S}_j \tilde{\mathbf{w}}_{ij}. \quad (30)$$

Each 9-vector  $\mathbf{m}_i$  can then be rearranged again into a  $3 \times 3$  matrix  $\mathbf{M}_i$ . However, there is again no guarantee that  $\mathbf{M}_i$  will comply orthogonality. Instead of solving for a common transformation matrix  $\mathbf{Q}$  as done previously, we project each  $\mathbf{M}_i$  to its closest rotation matrix. This projection have been proved experimentally to deliver better estimates in case of missing data. This projection can be done optimally [11] by decomposing each matrix  $\mathbf{M}_i$  using a SVD ( $\mathbf{M}_i \stackrel{SVD}{=} \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T$ ) and imposing  $\mathbf{\Sigma} = \mathbf{I}_{3 \times 3}$ . If we denote the projection by  $\hat{\mathbf{M}}_i$ , we have that  $\hat{\mathbf{M}}_i = \mathbf{U} \mathbf{V}^T$ . Equations (27) and (30) form an iterative method for the computation of  $\mathbf{M}$  and  $\mathbf{S}$ . The last component required is an initial estimate of  $\tilde{\mathbf{w}}_j^{(r)}$  to compute the weight matrix  $\mathbf{C}_j$ . The most straightforward way is to initialise using the aforementioned rigid body factorization defined in Section 2.

As this factorization gives us an approximation of a rigid body motion, it can be a good initialization for this algorithm. Finally, we stop the algorithm iterations using the convergence of the Frobenius norm of the global error matrix  $\mathbf{E}$  defined by  $\|\mathbf{E}\| = \|[\mathbf{E}_1 \mathbf{E}_2 \dots \mathbf{E}_P]\|$ .

In cases where the deformation is strongly directional (e.g. muscular contraction. See Figure 3) the rigid translation may be biased towards the deformation direction. Thus we need to find a way to compute a weighted version of the translation component. Since translation is not part of the global shape parameters, we modify our previous eqs. when computing  $\mathbf{M}_i$ . Given eq. (1), we can rewrite eq. (28) as:

$$\mathbf{m}_i = \begin{bmatrix} \mathbf{r}_{1i} \\ \mathbf{t}_{xi} \\ \mathbf{r}_{2i} \\ \mathbf{t}_{yi} \\ \mathbf{r}_{3i} \\ \mathbf{t}_{zi} \end{bmatrix}, \quad (31)$$

where  $\mathbf{m}_i$  is now a 12-vector; and rewrite eq. (29) as:

$$\mathbf{S}_j = \begin{bmatrix} \mathbf{s}_j^T \mathbf{1} & & \\ & \mathbf{s}_j^T \mathbf{1} & \\ & & \mathbf{s}_j^T \mathbf{1} \end{bmatrix}, \quad (32)$$

where  $\mathbf{S}_j$  is now a  $3 \times 12$  block diagonal matrix. Now we can update eq. (30) using the unregistered  $\mathbf{W}$ :

$$\mathbf{m}_i = \left( \sum_{j=1}^P \mathbf{S}_j^T \mathbf{C}_j \mathbf{S}_j \right)^{-1} \sum_{j=1}^P \mathbf{S}_j \mathbf{w}_{ij}. \quad (33)$$

The estimation of the global shape parameters is still done by eq. (27) using the registered data matrix  $\tilde{\mathbf{w}}$ . However this registration must be done using the new translational component computed in eq. (33).

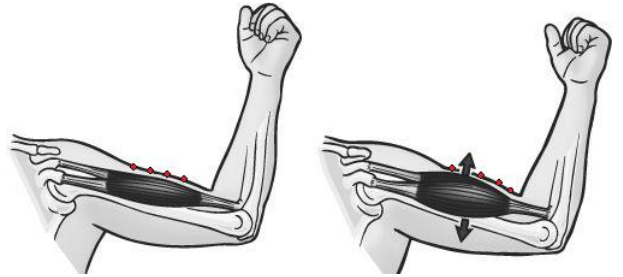


Figure 3. Example of a strongly directional deformation resulting from muscular contractions.

When using MOCAP systems one of the problems that might occur is the *occlusion* (also known as *missing data*) of some feature points i.e. in some of the frames there might not be any data available for some markers. This can happen due to the nature of the motion or other problems, such as the body moving out of the range of the MOCAP system, or the markers simply detaching from the subject. When occlusion happens

the measurement matrix will contain, in the respective frames, missing 3D point coordinates.

Even though some points may be occluded in a given frame, eq. (1) still holds true for the points that are not occluded, as do the rank constraints. If a given feature point  $j$  was occluded at frame  $i$ , then the 3D coordinates  $\mathbf{w}_{ij}$  will be missing. Since we do not have any information about missing markers we simply ignore its contribution to the computation using only the available information.

Let us define a  $F \times P$  binary matrix  $\mathbf{Z}$  such that:

$$z_{ij} = \begin{cases} 1, & \text{if } \mathbf{w}_{ij} \text{ is available.} \\ 0, & \text{if } \mathbf{w}_{ij} \text{ is occluded.} \end{cases} \quad (34)$$

Now all we need to do is use  $\mathbf{Z}$  to set to zero the contributions of the missing data. This can be done by updating eqs. (33) and (27) as:

$$\mathbf{m}_i = \left( \sum_{j=1}^P z_{ij} \mathbf{S}_j^T \mathbf{C}_j \mathbf{S}_j \right)^{-1} \sum_{j=1}^P z_{ij} \mathbf{S}_j \mathbf{w}_{ij}; \quad (35)$$

$$\mathbf{s}_j = \left( \sum_{i=1}^F z_{ij} \mathbf{M}_i^T \mathbf{C}_j \mathbf{M}_i \right)^{-1} \sum_{i=1}^F z_{ij} \mathbf{M}_i \tilde{\mathbf{w}}_{ij}. \quad (36)$$

Similarly, when  $\mathbf{w}_{ij}$  is missing,  $\mathbf{E}_j$  can still be computed. Clearly, there is a limit on the amount of data that can be occluded for the algorithm to work. However finding a theoretical limit is not trivial, being this usually done with experimental results. Finally, the Weighted Factorization algorithm dealing with translations and occlusion can be summarized as:

1. Initialization: Compute the *rank* - 3 approximation of  $\mathbf{W}$ . Compute  $t$  by registering the data and factorize into  $\mathbf{M}$  and  $\mathbf{S}$  using the method described in Section 2.1.
2. With the current estimations of  $\mathbf{M}$  and  $\mathbf{S}$  compute the weight matrices  $\mathbf{C}_j$  using eq. (25).
3. Using the current estimation of  $\mathbf{M}$ , compute  $\mathbf{S}$  by using eq. (36).
4. Based on the current estimation of  $\mathbf{S}$  from Step 2, compute  $\mathbf{M}$  and  $t$  by using eq. (35).
5. Project optimally  $\mathbf{M}_i$  to the closest orthogonal matrix.
6. Repeat Steps 2 to 5 until convergence of the Frobenius norm of  $\mathbf{E}$  is reached.

#### 4. Quadratic Model for Non-Rigid Bodies

Some approaches have tried to model deformations as a weighted sum of linear basis shapes [5, 2]. However deformations in the human body tend to have a quadratic nature, requiring a high number of linear basis shapes to accurately describe them. Inspired by previous works on the field of computer graphics [16, 15], we present a new quadratic model for non-rigid bodies using geometric constraints, built as an extension to the factorization-based rigid body model.

#### 4.1. Quadratic model formulation

Our model expands the rigid body formulation defined by eq. (2), to a formulation that uses linear, quadratic and crossed-terms of the previous rigid shape matrix, while keeping the same factorization-based approach. Let us define the new shape matrix as:

$$\mathbf{S} = \begin{bmatrix} x_1 & x_2 & \dots & x_P \\ y_1 & y_2 & \dots & y_P \\ z_1 & z_2 & \dots & z_P \\ \hline x_1^2 & x_2^2 & \dots & x_P^2 \\ y_1^2 & y_2^2 & \dots & y_P^2 \\ z_1^2 & z_2^2 & \dots & z_P^2 \\ \hline x_1 y_1 & x_2 y_2 & \dots & x_P y_P \\ y_1 z_1 & y_2 z_2 & \dots & y_P z_P \\ z_1 x_1 & z_2 x_2 & \dots & z_P x_P \end{bmatrix} = \begin{bmatrix} \mathbf{S}^{(\Gamma)} \\ \mathbf{S}^{(\Omega)} \\ \mathbf{S}^{(\Lambda)} \end{bmatrix}, \quad (37)$$

where  $\mathbf{S}^{(\Gamma)}$  is the  $3 \times P$  linear shape matrix,  $\mathbf{S}^{(\Omega)}$  the  $3 \times P$  quadratic shape matrix and  $\mathbf{S}^{(\Lambda)}$  is the  $3 \times P$  cross-values shape matrix. Given this new structure of  $\mathbf{S}$ , we introduce the motion matrix  $\mathbf{M}_i$  defined by:

$$\mathbf{M}_i = \mathbf{R}_i \begin{bmatrix} \Gamma_i & \Omega_i & \Lambda_i \end{bmatrix}, \quad (38)$$

where  $\mathbf{R}_i$  is a  $3 \times 3$  rotation matrix, and  $\Gamma_i$  is a  $3 \times 3$  transformation matrix associated with linear deformations,  $\Omega_i$  is a  $3 \times 3$  transformation matrix associated with quadratic transformations and  $\Lambda$  is a  $3 \times 3$  transformation associated with cross-values deformations. Note that the deformations are defined in the local referential of the body. Stacking the equations for all the  $F$  frames, we can now define:

$$\tilde{\mathbf{W}} = \begin{bmatrix} \mathbf{R}_1 & & & \\ & \mathbf{R}_2 & & \\ & & \ddots & \\ & & & \mathbf{R}_F \end{bmatrix} \begin{bmatrix} \Gamma_1 \Omega_1 \Lambda_1 \\ \Gamma_2 \Omega_2 \Lambda_2 \\ \vdots \\ \Gamma_F \Omega_F \Lambda_F \end{bmatrix} \begin{bmatrix} \mathbf{S}^{(\Gamma)} \\ \mathbf{S}^{(\Omega)} \\ \mathbf{S}^{(\Lambda)} \end{bmatrix} = \mathbf{M} \mathbf{S}, \quad (39)$$

where  $\tilde{\mathbf{W}}_i$  is the data matrix containing the 3D coordinates of the feature points, registered to the origin of the global referential. Following similar considerations as done in Section 2, this rigid model is described by a  $\text{rank}(\tilde{\mathbf{W}}) \leq 9$ . Note that if  $\Gamma_i = I_{3 \times 3}$ ,  $\Omega_i = \mathbf{0}_{3 \times 3}$  and  $\Lambda_i = \mathbf{0}_{3 \times 3}$ , we will have the rigid body model with  $\text{rank}(\tilde{\mathbf{W}}) \leq 3$ . For the sake of notation simplicity, we will only consider one frame of the motion, and the  $i$  index will be dropped. However, results can be easily extended to a general case.

As a first observation on the model, every full-rank matrix can be expressed as an RQ decomposition. Thus, since deformations are defined in the local referential, a RQ decomposition of  $[\Gamma \Omega \Lambda]$  extract a  $3 \times 3$  rotation from  $\Gamma$ . This rotation matrix should be the identify matrix  $I_{3 \times 3}$ , otherwise we would have an ambiguity in the optimization of these components. This implies that  $\Gamma$  must be an upper triangular matrix.

In order to study the role of the quadratic deformation parameters, we applied different transformations to a synthetic cubic object (see Figure 4). Due to the particular symmetry of

the cube, results drawn from particular cases can be easily extended to more general cases. From the deformation observed

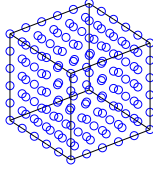


Figure 4. Representation of the cubic object used to test the quadratic model for non-rigid bodies.

some instances of  $M_i$  are not consistent with what is expected when modelling human body parts. For instance, deformation types that involve interpenetration of the shapes, or a twisting motion are not realistic. Examples of these deformations can be found in Figure 5. Instead, some cases of allowed deformations can be found in Figure 6. Based on these considerations,

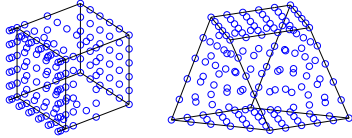


Figure 5. On the left, an example of the planar interpenetration caused by the diagonal entries of  $\Omega$ . On the right, the twisting deformations caused by some entries of  $\Lambda$ .

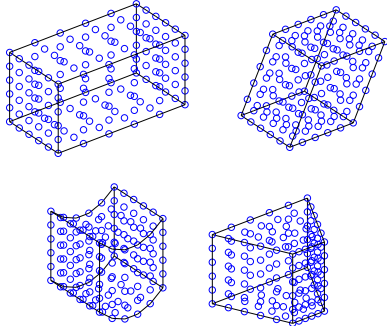


Figure 6. Allowed deformation for the quadratic model. On the top left corner, an example of the linear extension mode for the diagonal entries of  $\Gamma$ . On the top right corner, a pure shear deformation from the off-diagonal entries of  $\Gamma$ . On the lower left corner a bending/bulging deformation caused by the off-diagonal entries of  $\Omega$ . On the lower right corner, the lateral compression/extension mode caused by the allowed entries of  $\Lambda$ .

we defined the deformation matrices to be constrained as:

$$\Gamma = \begin{bmatrix} \Gamma_{11} & \Gamma_{12} & \Gamma_{13} \\ 0 & \Gamma_{22} & \Gamma_{23} \\ 0 & 0 & \Gamma_{33} \end{bmatrix}; \quad (40)$$

$$\Omega = \begin{bmatrix} 0 & \Omega_{12} & \Omega_{13} \\ \Omega_{21} & 0 & \Omega_{23} \\ \Omega_{31} & \Omega_{32} & 0 \end{bmatrix}; \quad (41)$$

$$\Lambda = \begin{bmatrix} \Lambda_{11} & 0 & \Lambda_{13} \\ \Lambda_{21} & \Lambda_{22} & 0 \\ 0 & \Lambda_{32} & \Lambda_{33} \end{bmatrix}. \quad (42)$$

## 4.2. Quadratic model bounds

The deformations allowed by this model have physical meaning only up to a certain degree. If every value is allowed on the deformation matrices, the model will allow unrealistic behaviours such as indefinite extension, or compression until the body collapses on a plane. In fact, if no constraints are used, the rigid component of the estimated shape factor will not resemble the shape observed on the motion (see Figure ??). Thus, we must create an upper and lower bound for the allowed values, but keeping in mind that the model must not be over-constrained, otherwise we will not be able to model the non-rigid bodies correctly. Based solely on empirical evaluation, we chose to bound every value of the deformation matrices to variations between  $pm0.5$  from the  $[\Gamma\Omega\Lambda]$  configuration of the rigid case.

As a further observation, biological soft-tissues are generally viewed as visco-hyperelastic, incompressible materials. Thus, imposing a volume conservation constraint is of the most importance. This can be done by forcing  $det(\Gamma) = 1$  [7].

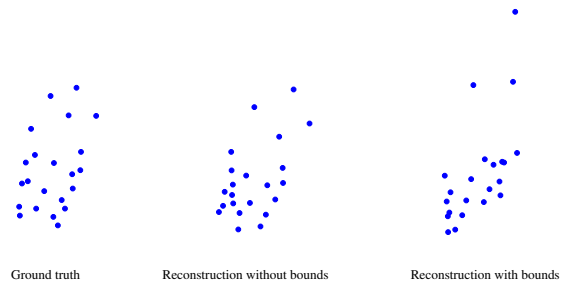


Figure 7. On the left, an example of the shape observed on the measurement matrix. On the middle, the rigid component of the shape matrix when using upper and lower bounds on the model. On the right, the rigid component of the shape matrix if no bounds on the deformation are used.

## 4.3. Non-linear optimization

In order to fit the quadratic model to our data, we need more powerful tools which can deal with the strong non-linearities of the model. Even an alternating approach as introduced with the weighted factorization algorithm would not suffice. The motion and shape matrices are intrinsically non-linear, thus a linear formulation is not possible at each step of the algorithm.

Our choice has fallen to a set of non-linear optimisation techniques which are known with the name of Bundle Adjustment (BA) in the Photogrammetry and Computer Vision literature [18]. This approach joins efficient minimization using

Levenberg-Marquardt together with faster computation using the sparse properties of the Jacobian and Hessian matrices obtained from the cost function.

In order to define the cost function we observe that, a single 3D coordinates  $j$  at frame  $i$  can be expressed by our quadratic model as:

$$\tilde{\mathbf{w}}_{ij} = \mathbf{M}_i \mathbf{s}_j = \mathbf{R}_i [\Gamma_i \Omega_i \Lambda] \mathbf{s}_j; \quad (43)$$

while the residual error of reconstruction can be defined as:

$$\mathbf{e}_{ij} = \mathbf{w}_{ij} - \tilde{\mathbf{w}}_{ij}. \quad (44)$$

The best fit of the quadratic model to the MOCAP data is then found by minimizing the norm of the sum of residuals such that:

$$\arg \min_{\mathbf{R}_i, \Gamma_i, \Omega_i, \Lambda, \mathbf{s}_j} \sum_{i,j}^{F,P} \|\mathbf{e}_{ij}\|^2 = \arg \min_{\mathbf{R}_i, \Gamma_i, \Omega_i, \Lambda, \mathbf{s}_j} \sum_{i,j}^{F,P} \|\mathbf{w}_{ij} - \tilde{\mathbf{w}}_{ij}\|^2. \quad (45)$$

This non-linear cost function has two major disadvantages. Not only can the number of parameters to be estimated rise considerably with the number of frames, but also the combination of several parameters can create multiple local minima, resulting in difficult convergence to the global minimum. However, the aforementioned properties of the BA approach may help avoiding the local minima and, at the same time, speed-up the computation.

The problem of multiple local minima can as well be solved by providing a good initialization of the problem, close to the global minimum. In our approach we will initialize the algorithm based on the rigid body description provided by the weighted factorization approach described in Section 3.

## 5. Experimental Results

For the experimental validation of our algorithms, we built synthetic data specifically for that task. Afterwards, we applied our algorithms to real MOCAP output data for qualitative evaluation purposes as we do not know the ground truth data in these cases. This data was taken from the database of the Graphic Lab of Carnegie-Mellon University (<http://mocap.cs.cmu.edu/>) and also MOCAP data acquired at the Augmented Human Interaction Laboratory of the Department of Computer Science of Queen Mary University of London. In both cases the data was captured using a VICON commercial MOCAP system.

### 5.1. Weighted Factorization

To test the performance of the weighted factorization algorithm we built a synthetic test representing a cubic object, of side 2 units, containing 26 feature points. We tested the final algorithm with 7 different levels of *additive white Gaussian noise* (AWGN) (see Table 1) and different ratios of missing data. Performance was defined as the accuracy on the reconstruction of the shape matrix, measured as the RMS value

for the Frobenius norm between the real and correspondent reconstructed point, normalised by the norm of the ground truth shape matrix.

We then tested the algorithm with 100 random rigid body motions with the 7 levels of AWGN defined above, while using random generated Z matrices with different percentages of missing entries: 30%, 40%, 50% and 60%.

Table 1. Mean RMS Error for the Shape Matrix S with Different Levels of Noise and Occlusion.

$\sigma^2$	$\rho = 0.3$		$\rho = 0.4$	
	$\epsilon_n^{(S)}$	$N_Z$	$\epsilon_n^{(S)}$	$N_Z$
0	$5.04 \times 10^{-14}$	0	$4.62 \times 10^{-14}$	0
0.01	0.0398	0	0.0382	0
0.05	0.142	0	0.158	0
0.1	0.254	0	0.273	0
0.2	0.469	0	0.502	0
0.4	0.897	0	0.968	0
0.6	1.38	0	1.55	0
$\sigma^2$	$\rho = 0.5$		$\rho = 0.6$	
	$\epsilon_n^{(S)}$	$N_Z$	$\epsilon_n^{(S)}$	$N_Z$
0	$4.35 \times 10^{-14}$	3	0.350	53
0.01	0.0410	0	0.0784	34
0.05	0.176	0	0.218	41
0.1	0.301	1	0.381	45
0.2	0.554	0	0.646	53
0.4	1.05	1	1.24	48
0.6	1.67	4	1.89	47

From Table 1 we can see that our algorithm deals successfully with occlusion up to 50% of the data. Still, as it is possible that the algorithm will not converge even when only 30% of the data is missing. This can happen if, for instance, missing data is particularly strong on a given frame, making impossible to solve the system of eqs. for that case. When dealing with 60% of missing data approximately half of the tests will fail. Thus we will consider our algorithm to be limited to 50% missing data.

### 5.2. Universal Joint

To test the robustness of the universal joint factorization algorithm, we ran a test battery composed of 1000 different random universal joint motion with 7 different levels of AWGN defined above. Performance was measured based on the accuracy of the computation of the joint centre. On Table 2 we present the RMS error value ( $\epsilon^{(U)}$ ) for the joint centre reconstruction for each level of noise.

The universal joint factorization seems to be more affected by noise. Still, we are trying to estimate the 3D coordinates of a single point in space. Thus it is expected that for high levels of noise this estimation could be compromised.

### 5.3. Hinge Joint

For the hinge joint case we simulated a book-like scene, composed of two parallelepiped, with 90 feature points each, linked by a hinge joint. Again we tested the algorithm using 1000 completely random motions with the 7 levels of AWGN defined above. This synthetic hinge joint used is represented in

Table 2. Mean Error for the Universal Joint Case vs. Noise Level.

$\sigma^2$	$\varepsilon_n^{(U)}$ (%)
0	$7.47 \times 10^{-13}$
0.01	0.50
0.05	2.50
0.1	4.98
0.2	9.90
0.4	20.0
0.6	30.0

Figure 8. Performance was measured as the average absolute value of the angle between the reconstructed and ground truth axis overall frames. The RMS error for that value versus the noise level is represented on Table 3.

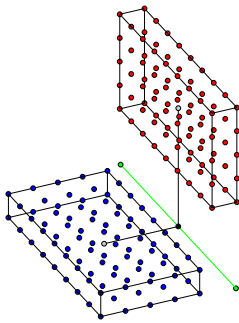


Figure 8. MATLAB plot of the synthetic setup of the hinge joint. The first body’s feature points are represented as blue dots and the second in red dots; the object centroids are represented in gray dots; the vectors  $\mathbf{d}^{(1)}$  and  $\mathbf{d}^{(2)}$  are represented as black lines; the joint centre is represented as a black dot and the joint axis is represented as a green line.

The direction of the reconstructed axis for the hinge joint is less sensitive to noise than the joint centre. This is explained by the fact that, when computing the axis of rotation, we already know the direction of the axis on the local referential, being the error only related to the inaccuracy on the rotation matrices.

Table 3. Mean error in the Hinge Joint case vs. Noise Level.

$\sigma^2$	$\varepsilon^{(H)}$ (degrees)
0	$5.18 \times 10^{-6}$
0.01	$7.1 \times 10^{-3}$
0.05	$3.2 \times 10^{-1}$
0.1	$6.1 \times 10^{-1}$
0.2	1.2
0.4	2.4
0.6	3.6

## 5.4. Real Data

Although there is no ground truth data to compare the reconstruction of the joints for real data sequences, these sequences can be used for a qualitative analysis of the algorithms. In Figure 9 we present the reconstruction of a universal joint from a human torso and head. In Figure 10 we present the algorithm applied to a jogging sequence, with the elbow and knee joints modelled as hinge joints, and the ankle joints modelled as universal joints. These results were obtained using a rigid body approximation for the body segments.

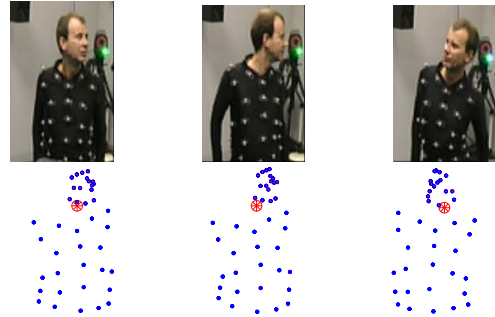


Figure 9. On top, sample frames of the real sequence. On the bottom, the corresponding frames of the reconstructed motion. The points of the torso are represented in blue circles, the points of the head as red circles and the joint centre represented as the large red circle with lines.

Even though in the neck there exists a much more complex set of articulations, in this case a universal joint model seems to be a good model. While it may not be detailed enough for clinical applications, in a gross analysis a universal joint model for the neck provides a good example of the applications of these algorithms, as it is clearly capable of determining what we would expect to be the joint centre.

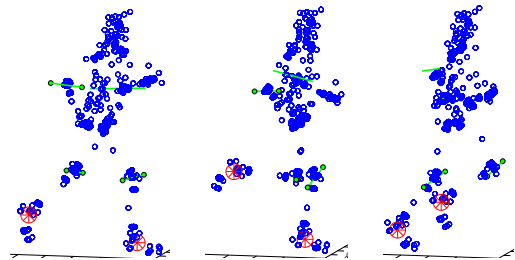


Figure 10. Multiple joint parameter estimation, during a jogging motion. Knee and elbow articulations were modelled as hinge joints; their joint axis are represented in green. The ankle was modelled as a universal joint represented in red.

In Figure 10 we present a clear example of the capacity of this methodology to provide full 3D human models based on the output from the MOCAP systems. Although there are no ground truth values to compare these reconstructions, in a qualitative analysis it is clear that the reconstructed joint cen-

tres and axes are consistent with what is expected from those articulations.

Finally, in order to qualitatively evaluate the performance of our quadratic model for non-rigid bodies, we reconstructed a motion of a flexing arm. This motion was acquired using 100 markers, thus providing a reasonable information about the soft-tissues motion. The reconstruction of this motion can be seen in Figure 11.

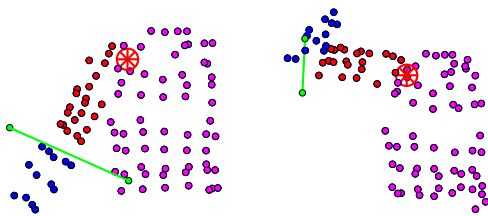


Figure 11. Multiple joint parameter estimation on a flexing arm. The elbow is modelled as a hinge joint and the shoulder as a universal joint. The rotation axis is presented in green while the joint centre of the universal joint is presented as a red circle with lines.

We can see in Figure 11 that both the rotation axis of the hinge joint and the joint centre of the universal joint are consistent with the motion of the arm. In these images it is also possible to notice soft-tissue motions that cannot be modelled when approximating to rigid bodies, such as the bicep contraction. Performance on the reconstruction of the data matrix  $W$ , for the BA optimization with the quadratic model for non-rigid bodies, was compared with the performance of the weighted factorization algorithm. Results show that when using the quadratic model for non-rigid bodies the error on the reconstruction was 1 to 2 orders of magnitude smaller, for each of the segments. This gives good indications on the accuracy of the joint reconstruction when compared to the rigid approximation approaches. Still, proper clinical validation must be done before any definitive conclusions can be taken.

## 6. Conclusion

In this article we presented a method to automatically create 3D articulated human body models, assuming motion segmentation as known, based on 3D MOCAP systems. Our algorithms, which are based on the factorization for SfM approaches, provide a more accurate rigid description of non-rigid bodies, and are able to deal up to 50% of occluded data. Additionally, we presented a new quadratic model for non-rigid bodies, which can describe soft-tissue deformations.

Future research lines include incorporating automatic motion segmentation based on the motion capture data, a thoroughly validation of these models and a refined study of the quadratic model for non-rigid bodies in order to fully understand its applicability in this scenario.

## Acknowledgments

The author would like to thank Prof. Pedro M. Q. Aguiar and Alessio Del Bue for the opportunity given to work on this subject. A special thanks will go to Alessio for all the support and motivation given almost on a daily basis, and for always believing in my abilities to fulfil this work.

## References

- [1] R. Baker. Gait analysis methods in rehabilitation. *Journal of Neuro-Engineering and Rehabilitation*, 3(1):4, 2006. 1
- [2] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3d shape from image streams. In *CVPR*, pages 690–696, June 2000. 6
- [3] A. Cappozzo, A. Cappello, U. Croce, and F. Pensalfini. Surface-marker cluster design criteria for 3-d bone movement reconstruction. *Biomedical Engineering, IEEE Transactions on*, 44(12):1165–1174, Dec. 1997. 1
- [4] B. Dariush. Human motion analysis for biomechanics and biomedicine. *Mach. Vision Appl.*, 14(4):202–205, 2003. 1
- [5] A. Del Bue, F. Smeraldi, and L. Agapito. Non-rigid structure from motion using ranklet-based tracking and non-linear optimization. *Image and Vision Computing*, 25(3):297–310, 2007. 6
- [6] A. V. Ferreira, B. G. Rosa, J. K. Fayad, and M. T. Silva. Análise de dinâmica directa do movimento de natação: Crawl. In *DSM2007*, pages 269–274(6), December 2007. In Portuguese. 1
- [7] Y. C. Fung. *Biomechanics: Mechanical Properties of Living Tissue*. Springer-Verlag, 1993. 7
- [8] L. Herda. *Using Biomechanical Constraints to Improve Video Based Motion Capture*. PhD thesis, École Polytechnique Fédérale de Lausanne, 2003. 1
- [9] M. Irani and P. Anandan. Factorization with uncertainty. In *ECCV*, pages 539–553, 2000. 4
- [10] T. Kanade and D. Morris. Factorization methods for structure from motion. *Philosophical Transactions: Mathematical, Physical and Engineering Sciences*, 356(1740):1153–1173, 1998. 2
- [11] K. Kanatani. *Statistical Optimization for Geometric Computation: Theory and Practice*. Elsevier Science Inc. New York, NY, USA, 1996. 5
- [12] D. Knudson. *Fundamentals of Biomechanics*. Springer Science, second edition, 2007. 1
- [13] S. M. Quantitative assessment of skin-bone movement at the knee. *The Knee*, 3:121–138(18), August 1996. 1
- [14] N. M.S. New advances in automatic gait recognition. *Information Security Technical Report*, 7:23–35(13), December 2002. 1
- [15] M. Müller, B. Heidelberger, M. Teschner, and M. Gross. Mesh-less deformations based on shape matching. *ACM Trans. Graph.*, 24(3):471–478, 2005. 1, 6
- [16] S. I. Park and J. K. Hodgins. Capturing and animating skin deformation in human motion. In *SIGGRAPH*, pages 881–889, New York, NY, USA, 2006. ACM. 1, 6
- [17] P. Tresadern and I. Reid. Articulated structure from motion by factorization. In *CVPR*, volume 2, pages 1110–1115, June 2005. 1
- [18] B. Triggs, P. McLauchlan, R. I. Hartley, and A. Fitzgibbon. Bundle adjustment – A modern synthesis. In W. Triggs, A. Zisserman, and R. Szeliski, editors, *Vision Algorithms: Theory and Practice*, LNCS, pages 298–375. Springer Verlag, 2000. 7
- [19] J. Yan and M. Pollefeys. A factorization-based approach to articulated motion recovery. In *CVPR*, volume 2, pages 815–821, June 2005. 1