

## A WEIGHTED FACTORIZATION APPROACH FOR ARTICULATED MOTION MODELLING

João K. Fayad\*, Alessio Del Bue<sup>†</sup> and Pedro M. Q. Aguiar<sup>†</sup>

\*School of Electronic Engineering and Computer Science, Queen Mary University of London  
Mile End Road, E1 4NS London, UK  
e-mail: jrkf@dcs.qmul.ac.uk,  
web page: <http://www.dcs.qmul.ac.uk/~jrkf>

<sup>†</sup>Institute for Systems and Robotics / Instituto Superior Técnico  
Av. Rovisco Pais, 1049-001 Lisboa, Portugal  
e-mails: adb@isr.ist.utl.pt, aguiar@isr.ist.utl.pt,  
web pages: <http://www.isr.ist.utl.pt/~adb>,  
<http://www.isr.ist.utl.pt/~aguiar>

**Keywords:** Human Body modelling, Biomechanics, Structure from Motion, Factorization.

**Abstract.** *Human body models and their analysis are nowadays used in a wide range of applications, spanning from medicine to security and surveillance. In this work we focus on the automatic creation of biomechanical human models for clinical and sports applications.*

*State of the art motion capture systems are nowadays able to measure the 3D coordinates of reflective markers placed above the skin with sufficient accuracy. Clinical gait analysis is currently facing problems like the repeatability of the measurements, the creation of subject-specific models, and the compensation for soft-tissue movement.*

*Recent developments on structure from motion approaches have allowed the efficient recovery of articulated bodies from a set of 2D images. In a similar mathematical framework, we present a set of computational tools able to create subject specific 3D models from motion capture data. Our aim is to obtain fully data-driven algorithms for the localization of the joints and their properties.*

*Specifically, our goal is to obtain a 3D model of the body parts, which is able to deal with the soft-tissue artifacts that are very common in this type of inference. The proposed approach has the distinguishing feature of being able to deal with marker occlusions that may occur during the extraction of the measurements from the motion capture system.*

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

## 1 INTRODUCTION

A human body model is a mathematical description of its anthropometry, physiology and topology [33]. For instance, models can be built to test the human physiological response on crashworthiness tests [13], or to predict bone remodelling behavior under different stress conditions [19]. In this work we focus on the development of models for full body motion.

A Motion Capture (MOCAP) system is a device able to recover a full description of the motion on a scene, in order to analyze or transfer it to a digital model of the object performing the motion [18]. MOCAP systems are subdivided in mechanical, magnetic, ultrasonic or optical systems, being the categorization given by the physical principle by which the motion is detected and captured. Although the setup may be distinct among the different types of systems, they all have in common the fact that they detect the spatial location of a set of feature points in order to describe the motion of the scene. Given all the different types of MOCAP systems, optical systems have emerged as the primary choice for most of the applications as this setup provides a precision of the order of 1 mm for the 3D position of the feature points [3, 7, 18]. The most usual setup for optical MOCAP systems is composed of calibrated infrared cameras (at least 2, but typically more than 6) arranged around the area to be captured, and passive (reflective) markers, with diameter between 9 to 25 mm, placed on the surface of the object [7]. The output of the system is the set of 3D coordinates of the tracked markers over time.

The output of MOCAP systems is usually applied to animate previously existent digital models. However this approach not only requires great effort in building the model, but also limits the accuracy and versatility of the analysis, as it is bounded to adjust the existent model to the current application. In opposition, we will focus on the problem of *creating a human model* based on data acquired by MOCAP systems, as this approach guarantees subject specific modeling.

### 1.1 Motivation

Nowadays MOCAP systems are used to capture human motion in a variety of fields such as medicine [7], sports [22, 16], computer vision [27], character animation [29], or recognition, security and surveillance [28]. In this work we will focus on biomechanical models of the human body *i.e.* on models that accurately describe the *motion* of the articulations, in order to predict the *forces* and *momenta* associated with it. This is in contrast with other applications such as character animation, where the focus is on the visual aspect of synthetic 3D shape [12] and its natural motion.

Biomechanics is the discipline that uses mechanical principles in order to study living organisms [22]. In sports, where the technique is a dominant factor, an analysis of the human motion can lead to an improvement of the performances, based on the information about articulations and muscles [23]. The information can also be used to evaluate the effect of a given motion in the muscles and articulations, in order to prevent injuries or generate preventive and rehabilitative therapies [22].

In medicine, gait analysis is the main application of motion analysis tools. The study of the alterations in normal gait patterns are very important on areas such as cerebral palsy or prosthetic limbs, orthoses and total joint replacements, providing information both for diagnosis and treatment options [7, 17, 32].

When performing biomechanical studies on the human body, building accurate human models is one of the key steps for achieving meaningful results. When a generic study about a given motion is being conducted, the model can be built based on anthropometric data. How-

ever, when applying biomechanics in clinical cases or sports, models must be subject specific in order to have accurate results. When we consider the universe of clinical patients that would benefit from this technology, the amount of resources dedicated just to build the custom models would be unbearable. Consequently, there is a strong need for finding methods to automatically create subject-specific reliable models.

One of the major source of error in MOCAP analysis are the *soft-tissue artifacts* [12, 10, 26]. In fact, although the relevant information about articulations used to build human models is given by the skeleton, the reflective markers used by MOCAP systems are placed above the skin. When examining a subject, there is an inherent relative motion between the soft-tissues surrounding the bone and the bone itself. These relative motions create artifacts on the data that degrade performances. This occurs especially when the non-rigid motion is directional (i.e. like the bending of a muscle). In such case, any inference algorithm solely based on rigidity assumption will produce a bias in the estimation.

Our aim is thus to tackle this issue and to introduce an algorithm which minimizes the effect of deformations from the observed 3D data. This will provide a reliable tool for clinical applications, which can aid in diagnosis and rehabilitation, and a method that can be applied to sports, in order to enhance performances and prevent injuries. Nevertheless, we must not forget that MOCAP systems and human models are used in a wider range of applications that will also benefit from these improvements.

## 1.2 Prior work

As stated before, systems used to capture human motion are generally optical MOCAP systems, based on markers placed over the skin of the subject performing the motion. Most of the existent clinical systems for gait analysis use markers placed at specific anatomic landmarks. *Regression* techniques are then used to fit this data to a *Conventional Gait Model* (sometimes called *Helen Hayes* model), which models the hip, knee and ankle articulations as joints with three degrees of freedom [7]. As stated above, a disadvantage of this setup is the fact that it uses a small number of feature points that have to be accurately placed over anatomic landmarks, which in some cases are not very well defined in patients with specific medical conditions [7]. Besides, evidence has been brought up that most common equations used for regression provide unsatisfying results [24].

Other approaches use a method called *anatomical calibration* [7], which relies on a previously existent model of the joints. The joint parameters are then computed by fitting the data from the MOCAP systems, acquired while performing *predetermined* actions, to the existing model. For instance, markers attached to a body segment belonging to an articulation modelled as a ball and socket joint, would have their coordinates lying on a sphere centered at the joint center. The joint centers are then computed by fitting the data to the ball and socket model using, for instance, least-squares optimization techniques [7]. The disadvantage of this method is that in some cases the performance of the test movements is impossible due to the medical conditions of the patients.

As mentioned previously, one of the major sources of error in these measurements are the soft-tissue artifacts, and so several attempts have been made to cope with this problem. Using optimization techniques to fit the motion capture data to a model, is one of them. For instance, attempts have been made to describe the real bone movement as a function of the observed soft-tissue movement, over all the range of motion [4, 25]. However, this approach suffers from an ambiguity in the definition of the true bone motion, as there is no consensus about a *gold-standard* for these measurements [7].

With the evolution of MOCAP systems, the number of feature points that can be tracked has increased since the very first systems were created. This allowed the layout of new methods such as *Point Cluster Techniques* (PCTs). PCTs rely on the fact that, when using a higher number of features than the minimum number required, there is a redundancy in information, which will make soft-tissue artifacts less relevant [5]. This technique was later extended to deal directly with soft-tissues artifacts by reformulating the problem as a chi-squared estimation. However, the functional form which model the relative displacement between bone and the skin has to be explicitly provided [4] and it is subject to the action performed by the patient.

Recovering the structure of general objects based on information about their motion is a subject of interest in the field of computer vision, originating the family of algorithms designated *structure from motion* (SfM). These algorithms were first developed aiming to recover 3D shapes from a set of 2D images with multiple views of the scene. Factorization methods for SfM are computational methods that use the unique rank properties of a measurement matrix to factorize it in a *motion* and *shape* factors. The first factorization methods for SfM were able to recover the shape and motion of a rigid object moving in a scene [34, 30]. Later, the approach was extended to deal with objects moving independently [11] and deformable shapes [9, 8, 36, 14]. Until recently this set of techniques was upgraded to deal with articulated structures as the one appearing in the context of human motion modelling [35, 38].

Although marker-based MOCAP systems are the most popular systems in clinical research, alternative approaches have also been considered. Among them, there are methods such as stereo radiography, bone pins, external fixation devices, and single or double plane fluoroscopy. However these methods are either invasive, or they limit motion range, or they require the exposure of the subject to radiation [7]. Models based on magnetic resonance imaging (MRI) have also been used [31, 6]. Although they can provide a detailed image of bones and muscles, this technique only works within small volumes and so their application is limited.

### 1.3 Objective and proposed approach

The objective of this work is to develop an algorithm to automatically compute biomechanical models of the human body using the data provided by MOCAP systems. We seek an algorithm that can be independent of the given MOCAP system's setup, only requiring a relatively high number of markers on each body segment. We also assume that the segmentation of the data is known (*i.e.* to which body segment belongs a given set of points). The algorithm we propose is particularly tuned to deal with the case of soft-tissue artifacts corrupting the measured data. This effect could produce a consistent bias both for the 3D structure computation and the motion estimation of the articulated model.

Our novel contribution proposed in Section 3 is a PCT-like approach. Since in the context of human motion modelling we are normally dealing with non-rigid objects, our approach explicitly penalize the 3D trajectories that are strongly deforming. The idea is to first compute a coarse rigid approximation based on least-squares optimization. Then, we switch to a weighted inference that automatically decrease the weights associated to the measured 3D data which present higher degrees of deformation. This weighted inference has the property to achieve a 3D rigid shape which is less affected by the bias given by the deforming motion. Moreover, we do not assume any specific prior over the property of the deformation as in [4]. Finally, we combine these results with an extension of the method for articulated SfM to 3D data [15], to provide a final 3D articulated model of the human skeleton.

We perform experiments using both synthetic and real data. Model validation is done solely by using synthetic data. Validating these models with real data requires the knowledge of the

real locations of joint centers and bone motions, which are not trivial to obtain since it requires specific equipment [12]. Therefore, validating with real data will be left as future work. However, we apply the algorithm to real data with the purpose of illustrating real applications of these algorithms, from which a first qualitative evaluation of performance can be done.

## 2 FACTORIZATION METHODS FOR STRUCTURE FROM MOTION

Factorization methods for structure from motion are a family of image based algorithms that model moving objects as a product of two factors: *motion* and *shape*. The shape components define the 3D geometric properties of the object; the motion components define the time-varying parameters of the motion (*e.g.* rotations and translations of the rigid body) that the object performs in a metric space. Factorization methods were pioneered by Tomasi and Kanade in [34], where they successfully recovered the motion of the camera motion and the geometry (*i.e.*, the shape) of a rigid object, based on a stream of 2D images. In [34], a number of feature points were selected and tracked across a stream of images, providing the input for the factorization algorithm. In our case, instead of 2D images, we work with the 3D coordinates provided by the MOCAP system, thus we start by re-formulating the factorization method to accommodate this scenario.

We assume a set of  $P$  3D points being tracked over  $F$  frames by a MOCAP system. The factorization methods rely on the key fact that the 3D trajectories of points belonging to the same body share *global properties*. The 3D trajectories provided by the MOCAP system are arranged in a  $3F \times P$  measurement matrix  $W$ , given by

$$W = \begin{bmatrix} \mathbf{w}_{11} & \dots & \mathbf{w}_{1P} \\ \vdots & \ddots & \vdots \\ \mathbf{w}_{F1} & \dots & \mathbf{w}_{FP} \end{bmatrix}, \quad (1)$$

where  $\mathbf{w}_{ij}$  are the 3D coordinates of point  $j$  at frame  $i$ .

### 2.1 Single rigid body

When all the feature points lie on a single rigid body, their coordinates  $\mathbf{w}_{ij}$  can be written as

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

where  $\mathbf{s}_j$  is a 3-vector that has the 3D coordinates of point  $j$ , describing the shape, on a local coordinate system;  $\mathbf{R}_i$  and  $\mathbf{t}_i$  are, respectively, the  $3 \times 3$  rotation matrix and 3-vector of the translation parameters that describe  $\mathbf{s}_j$  on a global coordinate system (see the illustration in Figure 1). Stacking these equalities for all the  $F$  frames and  $P$  points, results in the matrix equation

$$W = \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_F \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{R}_2 \\ \vdots \\ \mathbf{R}_F \end{bmatrix} \begin{bmatrix} x_1 & x_2 & \dots & x_P \\ y_1 & y_2 & \dots & y_P \\ z_1 & z_2 & \dots & z_P \end{bmatrix} + \begin{bmatrix} \mathbf{T}_1 \\ \mathbf{T}_2 \\ \vdots \\ \mathbf{T}_F \end{bmatrix} = \mathbf{MS} + \mathbf{T}, \quad (3)$$

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.

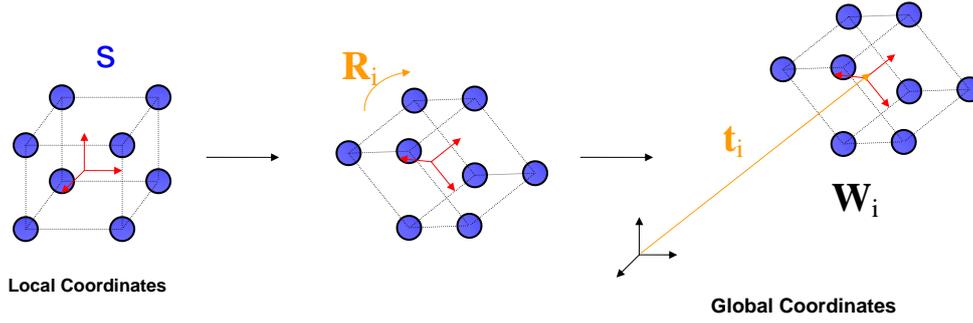


Figure 1: Graphical representation of the physical meaning of the motion and shape factors. The shape matrix  $S$  contains the 3D coordinates of the (blue) points that define the body, on the local (red) coordinate system. The rotation matrix  $R_i$  and translation vector  $t_i$  represent the coordinate transformations that describe  $S$  on the global (black) coordinate system, resulting in  $W_i$ .

The translational component of the motion,  $t_i$ , is given by the coordinates of the centroid of the point cloud at each frame  $W_i$ . Thus, it can be easily eliminated by registering the point cloud at each frame to the origin *i.e.* at each frame we subtract to the coordinates of every point the mean of the point cloud coordinates. In this scenario, it frequently occurs that, instead of  $W$ , we consider a registered form of this matrix called  $\bar{W}$  such that

$$\bar{W} = W - T = MS. \quad (4)$$

Let us consider the model defined in Eq. (3). Since  $M$  is a  $3F \times 3$  matrix, and usually  $F \gg 3$ , by the properties of the *rank* of a matrix,  $rank(M) \leq 3$ . On the other hand, since  $S$  is a  $3 \times P$  matrix, and usually  $P \gg 3$ , we also know that  $rank(S) \leq 3$ . Although  $T$  is a  $3F \times P$  matrix we know that all its columns are equal. Thus, as it only has one linearly independent column,  $rank(T) = 1$ . From these considerations on the *rank* of  $M$ ,  $S$  and  $T$  and according to Eq. (3), it becomes clear that  $rank(W) \leq 4$ . For the model defined by Eq. (4), since we have no translations, only  $M$  and  $S$  contribute to the *rank* of  $\bar{W}$  and we get  $rank(\bar{W}) \leq 3$ . However, the *rank* properties are only valid in the ideal case with no noise. When performing real experiments there will be noise affecting the coordinates, which will increase the rank of  $\bar{W}$ . Noise can originate for instance from the MOCAP system's uncertainty in the position of the tracked feature points or non-rigidity of the tracked object.

Consider the singular value decomposition (SVD) of the registered matrix  $\bar{W}$ , defined by

$$\bar{W} = U_{3F \times 3F} \Sigma_{3F \times P} V_{P \times P}^T, \quad (5)$$

where  $U$  and  $V$  are orthogonal matrices, and  $\Sigma$  is a diagonal matrix whose entries are the singular values  $\sigma_i$  of  $\bar{W}$ . Singular values are by definition non-negative ( $\sigma_i \geq 0$ ) and non-increasingly ordered in  $\Sigma$ . Also there are as many positive singular values in a matrix as its rank *i.e.* if  $r = rank(\bar{W})$ ,  $\sigma_i > 0, \forall i \leq r$ .

The truncated SVD is a version of the decomposition that constraints the result to a *rank*  $-k$  matrix. This is done by setting to zero all but the first  $k$  singular values. Consequently we can use only the first  $k$  columns of  $U$  and  $V$  to compute the transformation. Specifically, the *rank*  $-k$  truncated SVD is given by

$$\hat{W} = U_k \Sigma_k V_k^T, \quad (6)$$

where  $U_k$  is a  $3F \times k$  matrix with only the first  $k$  columns of  $U$ ,  $V_k$  a  $P \times k$  matrix with only the first  $k$  columns of  $V$ , and  $\Sigma_k$   $k \times k$  the diagonal matrix with the first  $k$  diagonal entries of  $\Sigma$ . The matrix  $\hat{W}$  in Eq. 6 is the best  $rank - k$  approximation of  $\bar{W}$  in the Frobenius norm sense. Since the ideal (noise-free) value for  $rank(\bar{W})$  is 3, we use a  $rank - 3$  truncated SVD as a first global optimal fit to the measurements.

The  $rank - 3$  truncated SVD is not only useful in noise reduction but it can also be used the starting point for the factorization algorithm. In fact, considering the expected dimensions of  $M$  and  $S$  we can compute a first estimate of these as

$$\hat{M} = U_3 \Sigma_3^{1/2}; \quad (7)$$

$$\hat{S} = \Sigma_3^{1/2} V_3^T. \quad (8)$$

Naturally, there exists an ambiguity in this solution, since any  $3 \times 3$  invertible matrix  $A$  will satisfy the equality

$$\hat{M} \hat{S} = \hat{M} A A^{-1} \hat{S}. \quad (9)$$

Being  $A$  an invertible matrix, it can be shown that it has an QR factorization, *i.e.*, that it can be factorized as  $A = QR$ , where  $R$  is a  $3 \times 3$  orthogonal matrix, and  $Q$  is a  $3 \times 3$  upper triangular matrix. This implies that

$$A A^{-1} = Q R R^{-1} Q^{-1} = Q Q^{-1}, \quad (10)$$

with the ambiguity being now expressed in terms of the matrix product of an upper triangular matrix and its inverse.

To resolve the ambiguity, we remark that the initial factorization proposed in Eqs. (7) and (8) does not guarantee that  $\tilde{M}$  is in fact a collection of  $F$   $3 \times 3$  rotation matrices. Thus the ambiguity stated in Eq. (10) is solved by finding the matrix  $Q$  that will map each  $3 \times 3$  matrix  $\hat{M}_i$  to a rotation (orthogonal) matrix  $R_i$ . This can be computed by imposing orthogonality constraints on  $\hat{M}_i Q$ , which is done by solving the following set of linear equations, for all the  $F$  frames:

$$\mathbf{m}_{ik}^T H \mathbf{m}_{ik} = 1, \quad (11)$$

$$\mathbf{m}_{ik}^T H \mathbf{m}_{il} = 0, \quad l \neq k, \quad (12)$$

with  $k, l = 1, 2, 3$ ,  $\mathbf{m}_{ik}$  and  $\mathbf{m}_{il}$  are respectively the  $k$ -th and  $l$ -th row of matrix  $\hat{M}_i$ , and  $H = Q Q^T$  is a symmetric matrix (since  $Q$  is upper triangular).  $Q$  is recovered from  $H$  by using Cholesky decomposition. We thus update the factorization in Eqs. (7) and (8) to:

$$M = \hat{M} Q; \quad (13)$$

$$S = Q^{-1} \hat{S}. \quad (14)$$

While some algorithms solve this ambiguity in a frame-by-frame analysis, by using all the data available in  $\tilde{W}$  to compute  $Q$ , we are actually taking in consideration all the frames to compute  $S$ . In this way we can find the factors  $M$  and  $S$  that are more consistent with the whole motion. Finally, we can reconstruct  $\hat{W}$  as the product of the two factors estimated by Eqs. (13) and (14).

## 2.2 Multiple rigid bodies

When the scene is composed of  $N$  rigid objects, moving independently, the considerations above are valid for each object. The model is thus simply expanded, accommodating the independent objects, with  $S$  showing a blockwise diagonal structure:

$$\bar{W} = \begin{bmatrix} M_1 & M_2 & \dots & M_N \end{bmatrix} \begin{bmatrix} S_1 & & & \\ & S_2 & & \\ & & \ddots & \\ & & & S_N \end{bmatrix}. \quad (15)$$

In this case, we have  $\text{rank}(\bar{W}) \leq 3N$ , when translations for each bodies are not considered. If we consider translations, each model will increase its  $\text{rank}$  by 1, leading to  $\text{rank}(W) \leq 4N$ .

## 2.3 Articulated objects

As just seen, when  $N$  rigid objects are moving independently, the rank of the measurement matrix  $\bar{W}$  grows proportionally to  $N$ . However if the rigid objects are linked by joints, their motions are not independent, and there is a loss in the degrees of freedom of the system. This constraint on the movement manifests itself in the measurement matrix  $W$  as a decrease in rank, see *e.g.*, [35, 37]. We now detail this scenario for systems described by two rigid bodies, linked by either an universal joint or a hinge joint. The constraints below are easily extended to the cases of a linked chain of rigid bodies.

### 2.3.1 Universal joint

By universal joint we mean a joint for which each of the two bodies is at a fixed distance from the joint center, being the relative position of the bodies constrained. Naturally, their rotations remaining independent. In mechanics literature, this kind of joint is also termed *spherical joint*. A scheme of this joint is presented in Figure 2.

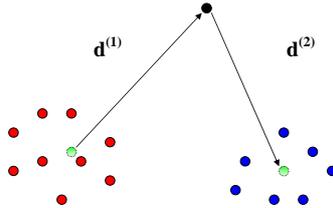


Figure 2: Scheme of a universal joint. The first body is represented by the red points, while the second body is represented by the blue points. The joint centre is shown as a black point. The 3-vector  $\mathbf{d}^{(1)}$  stands for the 3D coordinates of the joint center, in the local coordinate system of the first body. The 3-vector  $\mathbf{d}^{(2)}$  stands for the 3D coordinates of the joint center, in the local coordinate system of the second body.

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

The joint center can thus be seen as a point that belongs to both bodies. In other words, its position can be described using either the motion equations for the first or the second body. With these considerations, a geometrical analysis of the joint structure reveals that we can factorize the measurement matrix  $W$  as

$$W = [ W^{(1)} \mid W^{(2)} ] = [ R^{(1)} \quad R^{(2)} \quad \mathbf{t}^{(1)} ] \begin{bmatrix} S^{(1)} & D^{(2)} \\ 0_{3 \times P_1} & S^{(2)} + D^{(2)} \\ \mathbf{1}_{P_1}^T & \mathbf{1}_{P_2}^T \end{bmatrix}, \quad (16)$$

where  $W^{(1)}$  and  $W^{(2)}$  are, respectively, the measurement matrices for the first and second body;  $D^{(2)} = \mathbf{d}^{(2)} \mathbf{1}_{P_2}^T$ ,  $\mathbf{1}_{P_1}$  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  $P_1$  and  $P_2$  are the number of points belonging the first and second bodies, respectively; and  $0_{3 \times P_1}$  is a  $3 \times P_1$  zero matrix. In this case, we have that  $\text{rank}(W) \leq 7$ . Naturally, in order to separate  $W$  into  $W^{(1)}$  and  $W^{(2)}$ , we must assume the body segmentation is known. Details on how to factorize  $W$  can be found in Ref. [15].

### 2.3.2 Hinge joint

In a hinge joint, two bodies can rotate around an axis in such way that the distance to that rotation axis remains constant. Therefore, their rotation matrices  $R^{(1)}$  and  $R^{(2)}$  are not completely independent. A scheme of the hinge joint is presented in Figure 3.

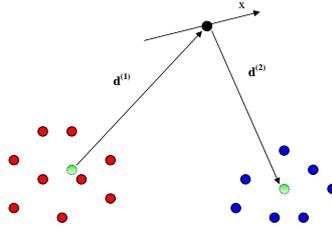


Figure 3: Scheme of a hinge joint. The first body is represented by the red points, while the second body is represented by the blue points. The joint center is shown as a black point. The  $x$ -axis represents the rotation axis. The 3-vector  $\mathbf{d}^{(1)}$  stands for the 3D coordinates of the joint center, in the local coordinate system of the first body. The 3-vector  $\mathbf{d}^{(2)}$  stands for the 3D coordinates of the joint center, in the local coordinate system of the second body.

We use the same notation as for the universal joint.

By analyzing the geometry of the joint, we see that every vector belonging to any of the two bodies, that is parallel to the joint axis, must remain so throughout the movement. Without loss of generality, let us choose an appropriate local coordinate system, where the axis of rotation of the joint is coincident with the  $x$ -axis. To comply with the joint constraints, the first column of  $R^{(1)}$  must be equal to the first column of  $R^{(2)}$ . We can now define the rotation matrices as  $R^{(1)} = [\mathbf{c}_1 \ \mathbf{c}_2 \ \mathbf{c}_3]$  and  $R^{(2)} = [\mathbf{c}_1 \ \mathbf{c}_4 \ \mathbf{c}_5]$ . In this case, the measurement matrix  $W$  is given by

$$W = [ \mathbf{c}_1 \quad \mathbf{c}_2 \quad \mathbf{c}_3 \quad \mathbf{c}_4 \quad \mathbf{c}_5 \quad \mathbf{t}^{(1)} ] \begin{bmatrix} x_1^{(1)} & \cdots & x_{P_1}^{(1)} & x_1^{(2)} & \cdots & x_{P_2}^{(2)} \\ y_1^{(1)} & \cdots & y_{P_1}^{(1)} & 0 & \cdots & 0 \\ z_1^{(1)} & \cdots & z_{P_1}^{(1)} & 0 & \cdots & 0 \\ 0 & \cdots & 0 & y_1^{(2)} & \cdots & y_{P_2}^{(2)} \\ 0 & \cdots & 0 & z_1^{(2)} & \cdots & z_{P_2}^{(2)} \\ & & \mathbf{1}_{P_1}^T & & & \mathbf{1}_{P_2}^T \end{bmatrix}, \quad (17)$$

Again, details about the factorization of  $\bar{W}$  can be found in Ref. [15].

### 3 QUASI-RIGID OBJECTS AND WEIGHTED FACTORIZATION

The algorithm described in Section 2.1 solves the problem in closed-form when the observed body is exactly rigid. When dealing with non-rigid bodies they can still be used as a coarse rigid approximation of the data. As mentioned in Section 1, the meaningful information about how the human body articulates is given by modelling the skeleton, which, at this level of analysis, can be considered rigid. Still, MOCAP systems work with markers placed above the skin, and they are affected by the relative motions between soft-tissue and bone that happen while the subject is moving. Dealing with non-rigid bodies is thus one of the main challenges when developing algorithms for this purpose.

When using an SVD to estimate the motion and shape parameters, the resulting shape will be the one that minimizes the error in a least-squares sense over all the frames. Nonetheless this might not be the best representation of the rigid shape since the estimation could be biased towards the deformation estimation. Inspired by previous approaches [1, 20, 2], what we present here is an approach that uses a weighted estimation in order to penalize the contribution of the points which deform most. By doing so we will attenuate the contribution of the deformations, obtaining a more accurate rigid representation of the body.

#### 3.1 The weighted factorization algorithm

Let us assume that the best global rigid shape is given, and that its registered 3D coordinates over time are described by a matrix  $\bar{W}^{(r)}$ . Let the matrix  $\bar{W}$  represent the data matrix resulting from tracking the quasi-rigid body with a MOCAP system, also registered. A measure of the non-rigidity of a given point on the matrix  $\bar{W}$  can be given by how distant its trajectory is from the best rigid description given by  $\bar{W}^{(r)}$ . Based on this idea, we will rearrange the data matrix defined in Eq. (1) as:

$$\tilde{W} = \begin{bmatrix} \bar{w}_{11}^T & \bar{w}_{12}^T & \dots & \bar{w}_{1P}^T \\ \bar{w}_{21}^T & \bar{w}_{22}^T & \dots & \bar{w}_{2P}^T \\ \vdots & \vdots & & \vdots \\ \bar{w}_{F1}^T & \bar{w}_{F2}^T & \dots & \bar{w}_{FP}^T \end{bmatrix}, \quad (18)$$

where  $\tilde{W}$  is an  $F \times 3P$  matrix, with  $P$  is the number of feature points tracked over  $F$  frames by the MOCAP system. The matrix corresponding to the best global rigid shape  $\bar{W}^{(r)}$  will be arranged similarly as in Eq. (18). The 3D trajectories of a generic point  $j$  are thus described in the  $F \times 3$  matrix  $\tilde{W}_j$  defined by:

$$\tilde{W}_j = \begin{bmatrix} \bar{w}_{1j}^T \\ \bar{w}_{2j}^T \\ \vdots \\ \bar{w}_{Fj}^T \end{bmatrix}, \quad (19)$$

with  $j = 1, \dots, P$ . We define  $\tilde{W}_j^{(r)}$  as the 3D trajectories of the same generic point  $j$  in the best global rigid shape description. We can now define a non-rigid error matrix  $E_j$  as:

$$E_j = \tilde{W}_j^{(r)} - \tilde{W}_j, \quad (20)$$

with  $E_j$  being a  $F \times 3$  matrix. As described above, this matrix indicates how distant the rigid and non-rigid trajectories of a generic point  $j$  are. Thus, for deformable points,  $\|E_j\|$  will be

higher than for rigid points. A weight matrix that assigns higher weights to rigid points and lower weights to deformable points can now be defined as:

$$C_j = \text{cov}(E_j)^{-1}, \quad (21)$$

as deformable points are bound to originate higher covariance values. Given this weight matrix, a better rigid description of the deformable body can be found by solving the least-squares problem given by:

$$\arg \min_{M_i, S_j} \sum_{i=1}^F \sum_{j=1}^P (\bar{\mathbf{w}}_{ij}^{(r)} - M_i \mathbf{s}_j)^T C_j (\bar{\mathbf{w}}_{ij}^{(r)} - M_i \mathbf{s}_j). \quad (22)$$

What we propose here is a two-step iterative algorithm that, from an initial estimation of the data for the best rigid shape, will compute a better global rigid shape description based on Eq. (22). If we have an estimation for  $\bar{\mathbf{w}}^{(r)}$  the weight matrix  $C_j$  can be computed. However, neither  $\bar{\mathbf{w}}^{(r)}$  nor  $C_j$  are known. Thus, the registered measurement matrix  $\tilde{\mathbf{W}}$  will be used as an estimation of  $\bar{\mathbf{w}}^{(r)}$ . With this, we intend to find the factors  $M$  and  $S$  that minimize the Frobenius distance of the measurement matrix to the weighted rigid body description. We will now rewrite Eq. (22) as:

$$\arg \min_{M_i, S_j} \sum_{i=1}^F \sum_{j=1}^P (\bar{\mathbf{w}}_{ij} - M_i \mathbf{s}_j)^T C_j (\bar{\mathbf{w}}_{ij} - M_i \mathbf{s}_j). \quad (23)$$

Still Eq. (23) is not trivial to solve as it is a bilinear minimisation of two matrices. However, if we first keep constant the values for  $M$ , we can find a solution for  $S$  by rearranging Eq. (23) as:

$$\mathbf{s}_j = \left( \sum_{i=1}^F M_i^T C_j M_i \right)^{-1} \sum_{i=1}^F M_i \tilde{\mathbf{w}}_{ij}. \quad (24)$$

This equation computes  $S$  using simple matrix products and the matrix inversion of a small matrix.

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

$$\mathbf{m}_i = \begin{bmatrix} \mathbf{r}_{1i} \\ \mathbf{r}_{2i} \\ \mathbf{r}_{3i} \end{bmatrix}, \quad (25)$$

and also rearrange  $\mathbf{s}_j$  into a  $3 \times 9$  block diagonal matrix  $S_j$  defined by:

$$S_j = \begin{bmatrix} \mathbf{s}_j^T & & \\ & \mathbf{s}_j^T & \\ & & \mathbf{s}_j^T \end{bmatrix}. \quad (26)$$

Based on Eq. (23) we can now compute each vector  $\mathbf{m}_i$  as:

$$\mathbf{m}_i = \left( \sum_{j=1}^P S_j^T C_j S_j \right)^{-1} \sum_{j=1}^P S_j \tilde{\mathbf{w}}_{ij}. \quad (27)$$

Each 9-vector  $\mathbf{m}_i$  can now be rearranged into a  $3 \times 3$  matrix  $M_i$ . However there is again no guarantee that  $M_i$  will be a rotation matrix. We chose to project each known affine matrix  $M_i$

into its closest rotation matrix. This can be done optimally [21] by decomposing each matrix  $M_i$  using an SVD ( $M_i \stackrel{SVD}{=} U \Sigma V^T$ ) and imposing  $\Sigma = I_{3 \times 3}$ , where  $I_{3 \times 3}$  is the identity matrix. If we denote the projection by  $\hat{M}_i$ , it can be defined as:

$$\hat{M}_i = U V^T. \quad (28)$$

Eqs. (24) and (27) form an iterative method for the computation of  $M$  and  $S$  as the output of one step is the input of the other one. All we need now is an initial estimate of  $W$  to compute the weight matrix  $C_j$ , an initial estimate of  $M$  and  $S$ , and a stoppage criterion.

For the initial estimations of  $W$ ,  $M$  and  $S$  we will use the aforementioned rigid body factorization defined in Section 2.1. For the stopping criterion, we have chosen to use the convergence of the Frobenius norm of the global error matrix  $E$  defined by:

$$\|E\| = \|[E_1 \ E_2 \ \dots \ E_P]\|. \quad (29)$$

Finally the algorithm can be summarized into the following steps:

1. Initialization: Compute the *rank* – 3 approximation of  $W$  and factorize into  $M$  and  $S$  using the method described in Section 2.1.
2. With the current estimations of  $M$  and  $S$  compute the weight matrices  $C_j$  using Eq. (21).
3. Using the current estimation of  $M$ , compute  $S$  by using Eq. (24).
4. Based on the current estimation of  $S$  from Step 3, compute  $M$  by using Eq. (27).
5. Apply the orthogonality constraints to  $M_i$  as defined in Eq. (28).
6. Repeat Steps 2 to 5 until convergence of the Frobenius norm of  $E$  is reached.

### 3.2 Weighted factorization and translation

In Section 3.1 we defined a weighted algorithm to estimate a better rigid representation of the global shape, based on the registered data. Still, if the deformations are strongly directional (*e.g.* muscular contraction) the rigid translation may be biased towards the deformation direction. Here we present a new version of the weighted factorization algorithm summarized at the end of Section 3.1, that incorporates an estimation for the translation. Based on Eq. (2) we can update Eq. (25) to:

$$\mathbf{m}_i = \begin{bmatrix} \mathbf{r}_{1i} \\ t_{xi} \\ \mathbf{r}_{2i} \\ t_{yi} \\ \mathbf{r}_{3i} \\ t_{zi} \end{bmatrix}, \quad (30)$$

where  $\mathbf{m}_i$  is now a 12-vector. The matrix  $S_j$  can be accordingly updated as:

$$S_j = \begin{bmatrix} \mathbf{s}_j^T & 1 & & \\ & \mathbf{s}_j^T & 1 & \\ & & \mathbf{s}_j^T & 1 \end{bmatrix}, \quad (31)$$

where  $S_j$  is now a  $3 \times 12$  block diagonal matrix. Now we can rewrite Eq. (27) to use the unregistered data matrix  $W$ :

$$\mathbf{m}_i = \left( \sum_{j=1}^P S_j^T C_j S_j \right)^{-1} \sum_{j=1}^P S_j \mathbf{w}_{ij}, \quad (32)$$

where  $S_j$  is defined by Eq. (31) and  $\mathbf{m}_i$  defined by Eq. (30). The estimation of the global shape parameters is still done by Eq. (24) using the registered data matrix  $\tilde{W}$ . However, since we have defined a new way to compute the translation, this registration is made by subtracting to every 3D point coordinates at each frame, not the mean of the point cloud, but the new translational component  $\mathbf{t}_i = [t_{xi} \ t_{yi} \ t_{zi}]^T$  extracted from  $\mathbf{m}_i$  as given in Eq. (32). Summarizing, the weighted algorithm to compute a better representation for the global shape parameters and estimate the rotations and translations of the motion can be described by the following steps:

1. Initialization: Compute the initial estimations for  $M$ ,  $S$  and  $\mathbf{t}$  using the rigid body factorization method described in Section 2.1.
2. Use the current estimates of  $\mathbf{t}_i$  for  $i = 1 \cdots F$  to register the data matrix  $W$ .
3. With the current estimate of  $M$  and the registered matrix  $\hat{W}$ , compute a new estimation for  $S$  using Eq. (24).
4. Based on the estimate of  $S$  from Step 3, compute a new  $M$  and  $\mathbf{t}_i$  based on Eq. (32).
5. Apply the orthogonality constraints to  $M_i$  as defined in Eq. (28).
6. Repeat Steps 2 to 5 until convergence of the Frobenius norm of  $E$  is achieved.

### 3.3 Weighted factorization with occlusion

When using MOCAP systems one of the problems that might occur is the *occlusion* of the markers (this problem is also named as the *missing data problem*). This can happen for several reasons, such as self-occlusion, markers detaching from the object, etc. When this happens, there is no information on all the points over all the frames, causing  $W$  to have fewer 3D point coordinates.

However, the algorithm presented in Section 3.1 and Section 3.2 can be easily modified to handle occlusions, as long as it is assumed to be know exactly which points are missing on a given frame. Let us define a  $F \times P$  binary matrix  $Z$  such that:

$$z_{ij} = \begin{cases} 1, & \text{if } \mathbf{w}_{ij} \text{ is available.} \\ 0, & \text{if } \mathbf{w}_{ij} \text{ is occluded.} \end{cases} \quad (33)$$

Now all we need to do is use  $Z$  to set to zero the contributions of the missing data. This can be done by updating equations (32) and (24) as:

$$\mathbf{m}_i = \left( \sum_{j=1}^P z_{ij} S_j^T C_j S_j \right)^{-1} \sum_{j=1}^P z_{ij} S_j \mathbf{w}_{ij}; \quad (34)$$

$$\mathbf{s}_j = \left( \sum_{i=1}^F z_{ij} M_i^T C_j M_i \right)^{-1} \sum_{i=1}^F z_{ij} M_i \tilde{\mathbf{w}}_{ij}. \quad (35)$$

When  $w_{ij}$  is missing,  $E_j$  can still be computed. However, entries corresponding to missing data will not be used on the computations, making  $C_j$  independent of missing data. With this approach, we simply disregard the contribution of a missing point to the estimation of the factors, using only the information that is available. The resulting algorithm has the same outline as the one described at the end of Section 3.2, but with equations (32) and (24) being replaced respectively by (34) and (35).

## 4 EXPERIMENTAL RESULTS

### 4.1 Synthetic Experiments

The weighted factorization algorithm is able to reduce noise and it estimates a more accurate rigid shape representation of a non-rigid body. To test its performance, we built a synthetic test representing a cubic object, containing 26 feature points and performing random motions on a scene. We tested the algorithm with different levels of *additive white Gaussian noise* (AWGN), and later performed a statistical analysis of the accuracy of the reconstruction. Finally, we applied the algorithm to real data obtained by a MOCAP system.

#### 4.1.1 Performance measurements

The cubic object used in the synthetic data is represented in Figure 4, with its feature points represented in red. Performances were measured based not only on the accuracy of the reconstruction of the shape matrix, but also on the error over the reconstructed trajectories given the measured data. Notice that the shape is always recovered up to a rotation matrix  $R_p$ , as

$$\tilde{W} = MS^{(gt)} = MR_P R_P^T S^{(gt)} = \tilde{M}S^{(rec)}, \quad (36)$$

where  $S^{(gt)}$  is the ground truth shape matrix and  $S^{(rec)}$  the reconstructed shape matrix. However, to compare the matrices one needs to register them to the same referential.  $R_p$  is computed by performing a Procrustes analysis on the data [21].

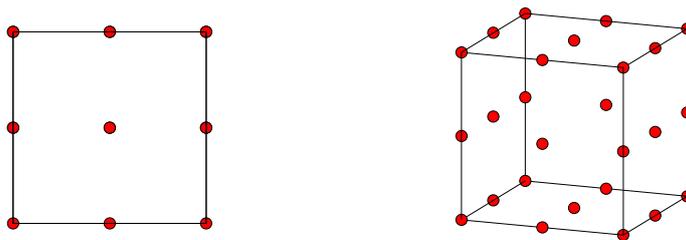


Figure 4: MATLAB plots of the cubic object used for the synthetic tests. The synthetic feature points are represented as red dots. The edges are shown to aid visualisation and they are not included in computations.

Another factor that prevents straight forward comparison between  $S^{(gt)}$  and  $S^{(rec)}$  is the scale of the objects. A solution is given by defining  $l_{gt}$  as the length of the edge of the ground truth cubic object. This value is a useful parameter to use with cubic objects as all the edges have equal length. On the other hand, the reconstructed object is not necessarily cubic. Thus we define  $l_r$  as the mean value of the length of all the edges of the reconstructed object. The scaling factor can now be defined as:

$$l_s = \frac{l_{gt}}{l_r}. \quad (37)$$

Let us also define a residual error matrix  $\mathbf{E}^{(S)}$  as:

$$\mathbf{E}^{(S)} = \mathbf{S}^{(gt)} - \mathbf{R}^{(P)} \mathbf{S}^{(rec)}. \quad (38)$$

We can now use equations (37) and (38) to define the measurement of the root means squared (RMS) error for the global shape matrix as:

$$\varepsilon^{(S)} = \sqrt{\frac{1}{P \times l_s} \sum_{j=1}^P \left\| \mathbf{E}_j^{(S)} \right\|^2}, \quad (39)$$

where  $\left\| \mathbf{E}_j^{(S)} \right\|$  represents the error on the global shape for point  $j$ . The factor  $\frac{1}{P}$  is a normalising factor for the number of points in the shape.

However, this error measurement is still depending on the shape size. In order to have system independent error measurement, some kind of normalisation must be done. For this purpose, we chose to normalize the RMS error by the norm of the ground truth shape matrix. We finally define the error measurement as:

$$\varepsilon_n^{(S)} = \frac{\varepsilon^{(S)}}{\left\| \mathbf{S}^{(gt)} \right\|}. \quad (40)$$

We defined similarly the error for the data matrix as:

$$\mathbf{E}^{(W)} = \mathbf{W}^{(gt)} - \mathbf{W}^{(rec)}, \quad (41)$$

with  $\mathbf{W}^{(gt)}$  the ground truth data matrix and  $\mathbf{W}^{(rec)}$  the reconstructed data matrix. The RMS error for the data matrix can thus be defined as:

$$\varepsilon^{(W)} = \sqrt{\frac{1}{P \times F} \sum_{i=1}^F \sum_{j=1}^P \left\| \mathbf{E}_{ij}^{(W)} \right\|^2}, \quad (42)$$

where  $\left\| \mathbf{E}_{ij}^{(W)} \right\|$  is the residual error for the feature point  $j$  at frame  $i$  in the global data matrix. Again, the RMS error is normalized by  $\left\| \mathbf{S}^{(gt)} \right\|$ :

$$\varepsilon_n^{(W)} = \frac{\varepsilon^{(W)}}{\left\| \mathbf{S}^{(gt)} \right\|}. \quad (43)$$

#### 4.1.2 Weighted factorization with additive Gaussian noise

Now that we have defined how to compute the error for the algorithm, we can carry out an analysis of its performance. Using the test battery we created, we tested the accuracy of retrieving the shape matrix and reconstructing the original motion, defined respectively by Eqs. (40) and (43), on 1000 completely random motions using 7 different levels of AWGN. The noise levels were defined based on the variance of the Gaussian distribution, and had the following values:  $\sigma^2 = 0, 0.01, 0.05, 0.1, 0.2, 0.4, \text{ and } 0.6$ . In order to avoid the algorithm being stuck on a local minima, we limited the iterations to 500 on every test. To compare the performance, we also tested the non-weighted factorization approach presented as in Section 2.1 to the same experiments. The box plot analysis of the error for the shape matrix reconstruction is represented in Figure 5. The mean error values versus the noise levels for the shape and data matrix reconstructions can be found in Table 1.

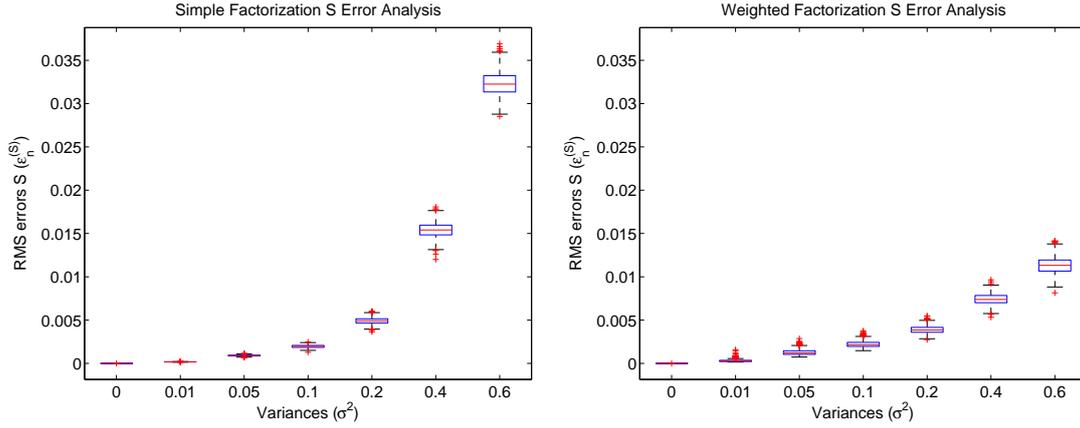


Figure 5: On the left, the box plot for the analysis of the shape matrix reconstruction using the simple factorization method. On the right the box plot for the analysis of the shape matrix reconstruction using the weighted factorization method.

Table 1: Mean error values for shape and measurement matrix reconstruction using normal and Weighted Factorization algorithms

$\sigma^2$	Simple Factorization		Weighted Factorization	
	$\varepsilon_n^{(S)}$ (%)	$\varepsilon_n^{(W)}$ (%)	$\varepsilon_n^{(S)}$ (%)	$\varepsilon_n^{(W)}$ (%)
0	$2.24 \times 10^{-13}$	$1.31 \times 10^{-4}$	$5.16 \times 10^{-14}$	$3.16 \times 10^{-15}$
0.01	0.017	0.00057	0.027	0.00040
0.05	0.090	0.028	0.120	0.020
0.1	0.197	0.056	0.217	0.031
0.2	0.491	0.112	0.387	0.076
0.4	1.54	0.229	0.738	0.159
0.6	3.23	0.333	1.13	0.232

In Figure 5 and on Table 1 is shown that while both algorithms, weighted and non-weighted, find the global solution for the shape matrix in the ideal case ( $\sigma^2 = 0$ ), their performances vary when the data is subject to noise. When the variance of the AWGN is small ( $\sigma^2 = 0.01$  and  $0.05$ ), the non-weighted algorithm performs better than our weighted approach. The number of outliers is also higher on the weighted algorithm, suggesting that in those cases the algorithm may be trapped in local minima. Still, the performance of our weighted algorithm matches the non-weighted algorithm approximately when  $\sigma^2 = 0.1$ , surpassing it for higher noise levels ( $\sigma^2 = 0.2, 0.4$  and  $0.6$ ), both in median error and in number of outliers.

The box plot analysis of the reconstructed data matrix is presented in Figure 6. In this case, for small levels of noise ( $\sigma^2 = 0, 0.01, 0.05$  and  $0.1$ ) the weighted and non-weighted algorithms have similar performances. Still, for higher noise levels ( $\sigma^2 = 0.2, 0.4$  and  $0.6$ ) the weighted algorithm has, again, a better performance.

The difference between the performances with noisy data is not as significant for the data matrix as it is for the shape matrix case. This can be explained by the fact that, when applying the orthogonality constraints to the motion matrix, the simple factorization algorithm finds a global solution for  $M$  (see equation 13). Differently, when data is noisy, it is likely that in several

frames  $M_i$  is not exactly a rotation matrix, thus allowing affine transformations of the shape matrix. On the other hand, the weighted factorization algorithm requires every matrix  $M_i$  to be a rotation matrix, at the expense of disregarding the continuity of the motion. Still, when higher levels of noise are used, this algorithm can not only provide a more accurate estimation of  $W$ , but also a more accurate estimation of  $S$ .

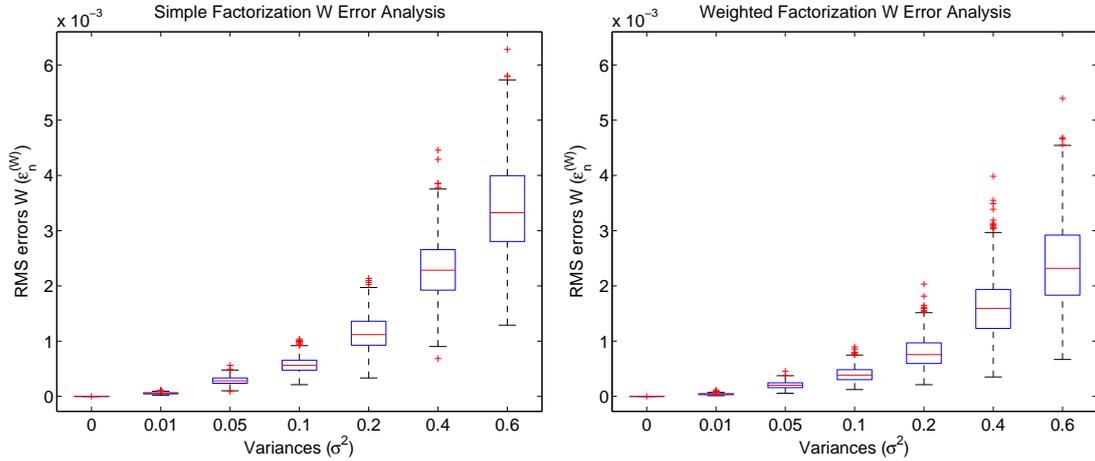


Figure 6: On the left, the box plot for the analysis of the data matrix reconstruction using the simple factorization method. On the right the box plot for the analysis of the data matrix reconstruction using the weighted factorization method.

## 4.2 Experiments with real data

We applied the algorithm to real data obtained by MOCAP. By using the weighted factorization algorithm presented in Section 3, combined with the extension of the articulated factorization algorithms presented in Section 2.3.1 and Section 2.3.2, we were able to compute estimates for the joint center and joint axis of human articulations. In Figure 7, we present a sequence of frames from the reconstruction of a jogging sequence. For the hinge joints, we represent the rotation axis in green; for the universal joints, we represent the joint center in red.

In Figure 8 we present a sequence of a human kicking a ball. Again, the joint axes for the hinge joints are represented in green, while the joint centers of the universal joints are represented in red.

Since there is no gold-standard method on determining the real axis and joint centers, we can only judge them by the graphical representation of the joints. Still, it is clear from Figures 7 and 8 that the reconstructed joint centers and axes are consistent with what is expected from those articulations. Both data sets are available from the Carnegie Mellon University MOCAP Database (<http://mocap.cs.cmu.edu/>).

The algorithm was applied to a sequence in which deformations due to muscle contractions are more clearly visible. The results of this sequence are presented in Figure 9.

As pointed out, it is only possible to make a qualitative evaluation of the results, since there is no ground truth data. Still, it is visible in Figure 9 that the recovered shape (represented by the red plus sign) do not present deformations when compared to the original sequence (represented by the blue circles). Furthermore, the joint axis reconstruction for the hinge joint used to model the elbow shows consistency with the analyzed motion.

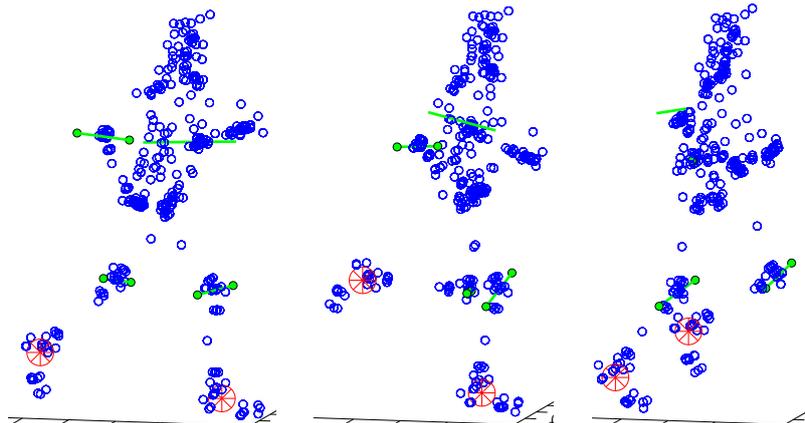


Figure 7: Multiple joint estimation for a jogging sequence. Knee and elbow were modelled as hinge joints. Their axis are represented in green. The ankle was modelled as an universal joint. Its center is represented in red.

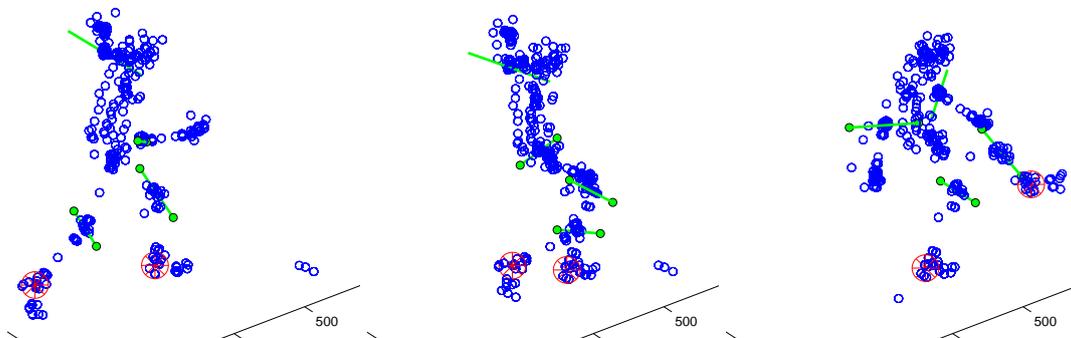


Figure 8: Multiple joint estimation of a subject kicking a ball. Color coding as in Figure 7.

## 5 CONCLUSIONS AND FUTURE WORK

We presented algorithms to create articulated 3D human models from MOCAP data. These algorithms are based on an extension to 3D of the SfM algorithms originally developed for 2D images. We introduced a weighted factorization method, which penalizes large deformations, in order to retrieve a more accurate rigid approximation of non-rigid bodies. Our algorithm was validated through extensive testing with synthetic data and proved to be more accurate than the existent non-weighted factorization approaches. This method was further used to model articulated bodies and tested with real data of human motion, leading to promising results.

Our work provides an insight about the potential applications of SfM algorithms in the field of biomechanics. Although not thoroughly validated with clinical analysis, our weighted factorization approach gives promising indications towards its applicability. Naturally, there is room for improvement and research paths to follow, *e.g.*, the validation in a clinical scenario or the search for automatic segmentation, as a fully automatic method for building articulated models for the human body would require.

## AKNOWLEDGEMENTS

This work was partially supported by Fundacao para a Ciêncïa e a Tecnologia, through ISR/IST plurianual (including POSC, FEDER) funding and grant PTDC/EEA-ACR/72201/2006.

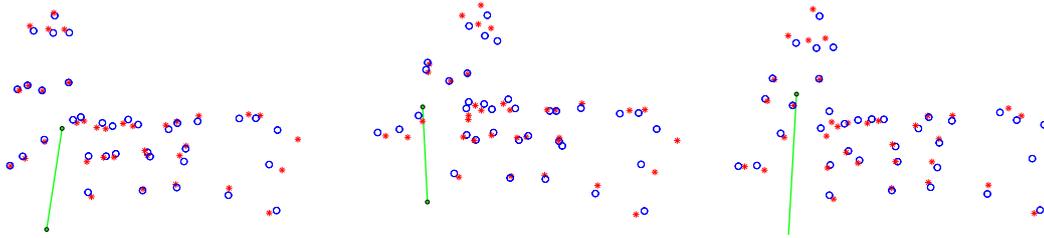


Figure 9: Upper and lower arm of a human subject. Blue circles show MOCAP data and red crosses show the reconstructed 3D shape. The hinge joint axis is shown in green.

## REFERENCES

- [1] P. Aguiar and J. Moura. Factorization as a rank 1 problem. *CVPR*, 1999.
- [2] P. Aguiar and J. Moura. Rank 1 weighted factorization for 3d structure recovery: Algorithms and performance analysis. *IEEE T-PAMI*, 25(9), 2003.
- [3] E. Alex, C. Bregler, and T. Andriacchi. Non-rigid modeling of body segments for improved skeletal motion estimation.
- [4] E. Alexander and T. Andriacchi. Correcting for deformation in skin-based marker systems. *Journal of Biomechanics*, (34), 2001.
- [5] T. Andriacchi et al. A point cluster method for *in vivo* motion analysis: Applied to a study of the knee kinematics. *Journal of Biomechanical Eng.*, 120(12), 1998.
- [6] D. Asakawa et al. Cine phase contrast magnetic resonance imaging as a tool for quantification of skeletal muscle motion. *Seminars on Musculoskeletal Radiology*, (7), 2003.
- [7] R. Baker. Gait analysis methods in rehabilitation. *Journal of NeuroEngineering and Rehabilitation*, 3(1):4, 2006.
- [8] M. Brand and R. Bhotika. Flexible flow for 3d nonrigid tracking and shape recovery. In *IEEE CVPR*, 2001.
- [9] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3d shape from image streams. In *IEEE CVPR*, 2000.
- [10] A. Cappozzo, A. Cappello, U. Croce, and F. Pensalfini. Surface-marker cluster design criteria for 3-d bone movement reconstruction. *IEEE T-BE*, 44(12), 1997.
- [11] J. Costeira and T. Kanade. A multibody factorization method for independently moving objects. *International Journal of Computer Vision*, 29(3):159 – 179, 1998.
- [12] B. Dariush. Human motion analysis for biomechanics and biomedicine. *Mach. Vision Appl.*, 14(4):202–205, 2003.
- [13] M. De Jager. *Mathematical Head-Neck Models for Acceleration Impacts*. PhD thesis, Universiteit Eindhoven, 2000.
- [14] A. Del Bue, F. Smeraldi, and L. Agapito. Non-rigid structure from motion using ranklet-based tracking and non-linear optimization. *IVC*, 25(3):297–310, March 2007.
- [15] J. Fayad, A. Del Bue, and P. Aguiar. Articulated motion analysis from motion capture data. In *Int. Symp. Computer Methods in Biomechanics and Biomedical Eng.*, 2008.
- [16] A. Ferreira, B. Rosa, J. Fayad, and M. Silva. Análise de dinâmica directa do movimento de natação: Crawl. In *Conf. Nac. Dinâmica de Sistemas Multicorpo*, 2007. In Portuguese.

- [17] J. R. Gage, P. Deluca, and T. Renshaw. Gait analysis: Principle and applications with emphasis on its use in cerebral palsy. *J Bone Joint Surg Am.*, (77):1607–1623, 1995.
- [18] L. Herda. *Using Biomechanical Constraints to Improve Video Based Motion Capture*. PhD thesis, École Polytechnique Fédérale de Lausanne, 2003.
- [19] R. Huiskes, H. Weinans, and D. M. Adaptive bone remodeling and biomechanical design considerations for noncemented total hip arthroplasty. *Journal of Biomechanics*, 1989.
- [20] M. Irani and P. Anandan. Factorization with uncertainty. In *Proc. 6th European Conference on Computer Vision, Dublin, Ireland*, pages 539–553, 2000.
- [21] K. Kanatani. *Statistical Optimization for Geometric Computation: Theory and Practice*. Elsevier Science Inc., New York, NY, USA, 1996.
- [22] D. Knudson. *Fundamentals of Biomechanics*. Springer Science, 2007.
- [23] A. D. Kuo. A least-squares estimation approach to improving the precision of inverse dynamics computations. *Journal of Biomechanical Engineering*, 120(1):148–159, 1998.
- [24] A. Leardini et al. Validation of a functional method for the estimation of hip joint centre location. *Journal of Biomechanics*, (32):99–103, 1999.
- [25] A. Leardini et al. Human movement analysis using stereophotogrammetry. part 3. soft-tissue artifact assessment and compensation. *Gait and Posture*, (21):212–225, 2005.
- [26] S. M. Quantitative assessment of skin-bone movement at the knee. *The Knee*, 3:121–138(18), August 1996.
- [27] T. Moeslund. A survey of computer vision-based human motion capture. *CVIU*, 81, 2001.
- [28] N. M.S. New advances in automatic gait recognition. *Information Security Technical Report*, 7:23–35(13), December 2002.
- [29] S. I. Park and J. K. Hodgins. Capturing and animating skin deformation in human motion. In *ACM SIGGRAPH*, 2006.
- [30] C. Poelman and T. Kanade. A paraperspective factorization method for shape and motion recovery. *Lecture Notes in Computer Science*, 800:97–110, 1994.
- [31] P. Rbmman and F. Sheehan. Precise 3d skeletal kinematics using fast phase contrast magnetic resonance imaging. *J. Magnetic Resonance Imaging*, (17), 2003.
- [32] C. Runge et al. Estimating net joint torques from kinesiological data using optimal linear system theory. *IEEE T-BE*, 42(12), 1995.
- [33] M. T. Silva. *Human Motion Analysis Using Multibody Dynamics and Optimization Tools*. PhD thesis, Instituto Superior Técnico, July 2003.
- [34] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization approach. *International Journal of Computer Vision*, 9(2):137–154, 1992.
- [35] P. Tresadern and I. Reid. Articulated structure from motion by factorization. In *IEEE CVPR*, 2005.
- [36] J. Xiao, J. Chai, and T. Kanade. A closed-form solution to non-rigid shape and motion recovery. In *ECCV*, May 2004.
- [37] J. Yan and M. Pollefeys. Articulated motion segmentation using ransac with priors. *ICCV Workshop on Dynamical Vision*, 2005.
- [38] J. Yan and M. Pollefeys. A factorization-based approach to articulated motion recovery. In *IEEE CVPR*, 2005.