Noname manuscript No.
(will be inserted by the editor)

Optimal Metric Projections for Deformable and Articulated Structure-From-Motion

Marco Paladini · Alessio Del Bue · João Xavier · Lourdes Agapito · Marko Stošić · Marija Dodig

the date of receipt and acceptance should be inserted later

Abstract This paper describes novel algorithms for recovering the 3D shape and motion of deformable and articulated objects purely from uncalibrated 2D image measurements using a factorisation approach. Most approaches to deformable and articulated structure from motion require to upgrade an initial affine solution to Euclidean space by imposing metric constraints on the motion matrix. While in the case of rigid structure the metric upgrade step is simple since the constraints can be formulated as linear, deformability in the shape introduces non-linearities. In this paper we propose an alternating bilinear approach to solve for non-rigid 3D shape and motion, associated with a globally optimal projection step of the motion matrices onto the manifold of metric constraints. Our novel optimal projection step combines into a single optimisation the computation of the orthographic projection matrix and the configuration weights that give the closest motion matrix that satisfies the correct block structure with the additional constraint that the projection matrix is guaranteed to have orthonormal rows (*i.e.* its transpose lies on the Stiefel manifold). This constraint turns out to be non-convex. The key contribution of this work is to introduce an efficient convex relaxation for the non-convex projection step. Efficient in the sense that, for both the cases of deformable and articulated motion, the proposed relaxations turned out to be exact (*i.e.* tight) in all our numerical experiments. The convex relaxations are semi-definite (SDP) or second-order cone (SOCP) programs which can be readily tackled by popular solvers. An important advantage of these new algorithms is their ability to handle missing data which becomes crucial when dealing with real video sequences with self-occlusions. We show successful results of our algorithms on synthetic and real sequences of both deformable and articulated data. We also show comparative results with state of the art algorithms which reveal that our new methods outperform existing ones.

1 Introduction and Previous Work

The combined inference of the motion of a camera and the 3D geometry of an unconstrained scene viewed solely from a sequence of images is a longstanding challenge for the Computer Vision community. The fundamental assumption which has allowed robust solutions to the problem is that of scene rigidity. However, when dealing with image objects that vary their 3D shape, the Structure From Motion (SfM) problem becomes inherently ambiguous and non-linear. The seminal work of (Bregler *et al.*, 2000) was the first to deal with the case of deformable objects viewed by a single camera. Their key insight was to use a low-rank shape model to represent the deforming shape as a linear combination of k basis shapes which encode its main modes of deformation. This model not only provided an elegant extension of the rigid factorisation framework (Tomasi & Kanade, 1992) but has also opened up new computational and theoretical challenges in the field.

Marco Paladini E-mail: paladini@dcs.qmul.ac.uk
School of Electronic Engineering and Computer Science
Queen Mary University of London, UK.

Alessio Del Bue E-mail: alessio.delbue@iit.it
IIT - Istituto Italiano di Tecnologia, Genova, Italy

João Xavier E-mail: jxavier@isr.ist.utl.pt
ISR - Instituto Superior Técnico, Lisboa, Portugal.

Lourdes Agapito E-mail: lourdes@dcs.qmul.ac.uk
School of Electronic Engineering and Computer Science
Queen Mary University of London, UK.

Marko Stošić E-mail: mstosic@isr.ist.utl.pt
ISR - Instituto Superior Técnico, Lisboa, Portugal.

Marija Dodig E-mail: dodig@cii.fc.ul.pt
CELC, Universidade de Lisboa, Portugal.

Although this low-rank shape model has proved a successful representation, the Non-Rigid Structure from Motion (NRSfM) problem is inherently under-constrained. Most approaches formulate the problem as an optimisation problem where the objective function to minimise is the image reprojection error. Recent methods focus on overcoming the problems caused by ambiguities and degeneracies by proposing different optimisation schemes and the use of generic priors. Prior knowledge that the reconstructed shape does not vary much from frame to frame was used in (Aanæs & Kahl, 2002) while in (Del Bue *et al.*, 2006) the constraint imposed was that some of the points on the object are rigid. Both approaches use bundle adjustment to refine all the parameters of the model together. A coarse to fine shape model was introduced in (Bartoli *et al.*, 2008) where new deformation modes are added iteratively to capture as much of the variance left unexplained by previous modes as possible. Other authors (Torresani *et al.*, 2008) have also argued that simple linear subspace shape models are extremely sensitive to noise and missing data so statistical priors should be used to constrain the parameter space. Torresani *et al*. introduced priors as a Gaussian distribution on the deformation weights which represents an explicit assumption that these will be similar to each other for each pose. They then generalise the model to represent linear dynamics in the deformations. All these approaches impose orthonormality constraints on the rotation matrices through parameterisation.

One advantage of the linear subspace model is that it has allowed closed form solutions for the cases of both affine (Xiao *et al.*, 2006) and perspective (Xiao & Kanade, 2005; Hartley & Vidal, 2008) viewing conditions. In the affine case Xiao *et al*. proved that orthogonality constraints were insufficient to disambiguate rigid motion and deformations (Xiao *et al.*, 2006). They identified a new set of constraints on the shape bases which, when used in addition to the rotation constraints, provide a closed form solution to the problem of NRSfM. Later they extended the approach to the perspective case (Xiao & Kanade, 2005). Similarly, Wang and Wu propose a new camera model approximating a full perspective camera and enforcing basis constraints when estimating NRSfM (Wang & Wu, 2009). However, every solution employing basis constraints is known to be very sensitive to noise (Brand, 2005; Torresani *et al.*, 2008) and to the selection of the basis constraints. Brand describes a modified version of this algorithm using weaker constraints on the basis and nonlinear optimisation which improves the solution (Brand, 2005). Interestingly, Akhter *et al*. have recently argued that the use of the basis constraints is not necessary to compute a valid solution for the NRSfM problem. An exact 3D reconstruction can be obtained by solving the problem with the appropriate structure when upgrading for the metric constraints (Akhter *et al.*, 2009). However, their theoretical insight is not followed by a closed-form solution and the authors revert to non-linear optimisation in order to find the correct solution. Recently Hartley and Vidal have proposed a new closed form linear solution for the perspective camera case (Hartley & Vidal, 2008). This algorithm requires the initial estimation of a multifocal tensor, for which a linear method exists. The tensor is then factorised into the projection matrices and then simple linear algebraic techniques are used to enforce constraints on the projection matrices and estimate explicitly the corrective transformation. Although the entire approach is linear, the authors report that the initial tensor estimation and factorisation is very sensitive to noise. Moreover, none of the closed form solutions proposed so far can deal with missing data which becomes crucial when dealing with real video sequences.

Recently, a set of new approaches have departed from the low-rank linear shape model. Rabaud and Belongie assume that only small neighbourhoods of shapes are well modelled with a linear subspace (Rabaud & Belongie, 2008). They then adopt a manifold learning framework tailored to the NRSfM problem to constrain the degrees of freedom of the deforming object. A dual formulation of NRSfM has been proposed by Akhter *et al*. who describe the evolving 3D structure of a non-rigid body in trajectory space as a linear combination of basis trajectories (Akhter *et al.*, 2008). The obvious advantage of using trajectory rather than shape space is that there is no need to estimate an object dependant basis. Instead the trajectory bases are object independent and only the coefficients need to be computed. The authors use the Discrete Cosine Transform, therefore low frequency bases model smooth deformations while higher frequency bases model more complex deformations. Quadratic models for NRSfM have been proposed by Fayad *et al*. to describe more accurately deformations which involve strong bending motions, stretching or twists. The increased descriptive power of this model is paid with increased complexity and non-linearities in the parameter space (Fayad *et al.*, 2009).

Articulated motion has also been recently formulated using a structure from motion approach (Tresadern & Reid, 2005; Yan & Pollefeys, 2008) modelling the articulated motion space as a set of intersecting motion subspaces — the intersection of two motion subspaces implies the existence of a link between the parts. Articulation constraints can then be imposed during factorisation to recover the location of joints and axes. While Yan and Pollefeys only compute the location of joints and axes on the image plane and do not perform a 3D reconstruction, Tresadern and Reid go further and compute the metric upgrade, but only recover a linear approximation of the correcting transformation (Tresadern & Reid, 2005). Both approaches require full data and therefore cannot deal with missing tracks, a situation that commonly occurs for instance when tracking humans.

1.1 Related Work and Contributions

In this paper we present a new unified approach to perform the metric upgrade in the cases of articulated and deformable structure viewed by an orthographic camera in the presence of missing data.

In the non-rigid case our approach is most closely related to Torresani *et al*.'s and Wang *et al*.'s trilinear schemes (Torresani *et al.*, 2001; Wang *et al.*, 2008). Both approaches use an identical alternating least squares framework to estimate the configuration weights, basis shapes and orthographic camera matrices, solving iteratively for each of the unknowns leaving the others fixed. The only difference between these two approaches is in the way that the orthographic camera matrices are updated and the metric constraints imposed – the other two steps in the alternation are identical.

While Torresani *et al*. enforce the exact metric constraints through an exponential map parametrisation of the rotation matrices, the update of the camera matrix is only an approximation — the camera matrix cannot be updated in closed form and instead they perform a single Gauss-Newton step. Alternatively, in their Rotation Constrained Powerfactorization algorithm (RCPF) Wang *et al*. first update the orthographic camera matrix via least squares and an additional step is incorporated to project it onto the Stiefel manifold via its SVD decomposition. This simple projector is in fact almost identical to the one proposed by (Marques & Costeira, 2008) for the case of rigid structure. Finally, in order to deal with missing data the above trilinear approaches (Torresani *et al.*, 2001; Wang *et al.*, 2008) resort to using only the available image tracks in their alternating scheme.

Similarly to Torresani *et al*. and Wang *et al*. we also propose an iterative alternating scheme to solve the non-rigid structure from motion problem. However, our optimisation scheme is bilinear, alternating between the estimation of the motion and the shape matrices, with an additional projection step of the motion matrices onto the manifold of metric constraints. At the expense of solving a more complex optimisation problem, our efficient convex relaxation provides an optimal minimiser to solve simultaneously for the orthographic camera matrix and configuration weights that give a motion matrix that satisfies the appropriate block structure while also ensuring that the orthographic camera matrix satisfies the constraint of having orthonormal rows (its transpose lies on the Stiefel manifold[1]). Here and throughout the paper, the optimal projection of a matrix onto a given set of matrices, denotes the closest point on that set from the given matrix with respect to the Frobenius norm. Extensive tests carried out on motion capture sequences with ground truth 3D data, reported in Section 5, show that adding a projection step (Wang *et al*.'s or ours) improves greatly the results obtained in the case of missing data with respect to other methods. However, even better improvements are achieved when using our bilinear algorithm associated with the proposed metric projection instead of Wang *et al*.'s trilinear scheme and simpler projector (Wang *et al.*, 2008)

In order to deal with missing data, our algorithm performs an outer iterative loop in which, at each step of the iteration, we run our non-rigid factorisation algorithm and we use the new estimates of the rotations, translations, basis shapes and coefficients to provide a new estimate of the missing data. Our experimental tests shown in Section 5 reveal that dealing with incomplete tracks using this outer loop allows to cope with much higher percentages of missing data than the trilinear approaches (Torresani *et al.*, 2001; Wang *et al.*, 2008) that only use the available data.

In summary, we see three substantial contributions in our approach. First, in contrast to their trilinear schemes, our optimisation scheme is bilinear, alternating between the estimation of the motion and the shape matrices. Secondly, our novel optimal projection step combines into a single optimisation the computation of the camera matrix and the configuration weights that give the closest motion matrix that lies on the non-rigid *motion manifold* with the additional constraint that the camera matrix is guaranteed to have orthonormal rows (*i.e.* its transpose lies on the Stiefel manifold). Finally, our experiments reveal that dealing with missing data using an iterative outer loop to re-estimate the missing entries greatly improves the results with missing data.

This notion of *motion manifolds* was recently introduced in the case of rigid shapes by (Marques & Costeira, 2009). Notably, constraining the motion matrices to lie on the exact motion manifold leads to robust solutions for the problem of estimating rigid 3D structure in the case of high ratios of missing data or degenerate configurations. Our work extends and generalises Marques and Costeira's to the case of deformable and articulated shapes therefore we provide a general framework which allows us to deal with high ratios of missing data and different types of shape. In particular, we impose that the camera matrix must have orthonormal rows, therefore its transpose lies on the $\mathtt{V}_{2,3}$ Stiefel manifold.

This constraint is non-convex, but in the case of deformable structure we show that an efficient convex relaxation can be obtained which results in the constraint set being defined only by a set of linear matrix inequalities (LMI). Therefore we relax the problem of imposing the camera matrices metric constraints into a Semi-Definite-Program which can be solved with popular solvers such as SeDuMi. In the case of articulated structure, we also propose an efficient convex relaxation which in most cases consists of a semi-definite program(SDP) and of a second order cone program (SOCP)

[1] The Stiefel manifold $\mathtt{V}_{k,m}$ may be viewed as the collection of all $m \times k$ matrices whose columns form an orthonormal set. More precisely, the (real) Stiefel manifold $\mathtt{V}_{k,m}$ is the collection of all ordered sets of k orthonormal vectors in Euclidean space $\mathbb{R}^m$.

in the remaining cases. While we do have a theoretical proof of the tightness of the convex relaxations for certain special cases (Dodig *et al.*, 2009), we do not yet have a proof for every case. However, all the aforementioned convex relaxations turned out to be exact in the totality of our numerical simulations.

The result is an algorithm where the recovered motion matrices have the exact structure and the exact orthogonality constraints imposed. One of the main advantages of our approach is that it can be extended naturally to deal with missing data in a similar way to (Marques & Costeira, 2009). An earlier version of our work appeared in (Paladini *et al.*, 2009). There are two important new contributions in this paper:

– We have proposed a new efficient convex relaxation for the articulated case, while in our previous work we used an exhaustive search over the cost function constrained to the unit circle. This results in a unified approach to solve the metric projection problem in the deformable and articulated cases using convex optimisation techniques. This new efficient convex relaxation is shown in Appendix B.
– We propose an alternative optimisation algorithm for the deformable case which performs 130 times faster than our original convex relaxation solution. In Section 3.2 we present a new iterative Newton-like optimisation algorithm on the Stiefel manifold which constrains the solution to lie on the correct manifold. Although we lose the optimality given by the convex solution in all our experiments with ground truth data the algorithm converged to the same global minimum.

As a final observation we should stress that, while most NRSfM algorithms proposed to date need to rely on the use of priors to solve for the 3D shape and the camera motion (Bartoli *et al.*, 2008; Torresani *et al.*, 2008) avoiding ambiguities, our new algorithms can obtain reliable solutions without having to impose priors such as smoothness on the camera motion or the deformations.

2 Factorisation for Structure from Motion

Consider the set of 2D image trajectories obtained when the points lying on the surface of a 3D object are viewed by a moving camera. Defining the non-homogeneous coordinates of a point j in frame i as the vector $\mathbf{w}_{ij} = (u_{ij} v_{ij})^\top$ we may write the measurement matrix $\mathtt{W}$ that gathers the coordinates of all the points in all the views as:

$$\mathtt{W} = \begin{bmatrix} \mathbf{w}_{11} & \dots & \mathbf{w}_{1p} \\ \vdots & \ddots & \vdots \\ \mathbf{w}_{f1} & \dots & \mathbf{w}_{fp} \end{bmatrix} = \begin{bmatrix} \mathtt{W}_1 \\ \vdots \\ \mathtt{W}_f \end{bmatrix} \tag{1}$$

where f is the number of frames and p the number of points.

The measurement matrix can be factorised into the product of two low-rank matrices as $\mathtt{W} = \mathtt{M}_{2f\times r} \ \mathtt{S}_{r\times p}$, where $\mathtt{M}$ and $\mathtt{S}$ correspond to the motion and shape subspaces respectively. As a result, the rank of $\mathtt{W}$ is constrained to be $\mathrm{rank}\{\mathtt{W}\} \leq r$ where $r \ll \min\{2f, p\}$. The rank of these subspaces is dictated by the properties of the camera projection and the nature of the shape of the object being observed (rigid, deformable, articulated, etc.). This rank constraint forms the basis of the factorisation method for the estimation of 3D structure and motion.

Matrices $\mathtt{M}$ and $\mathtt{S}$ can be expressed as $\mathtt{M} = \left[\mathtt{M}_1^\top \cdots \mathtt{M}_f^\top\right]^\top$ and $\mathtt{S} = [\mathbf{S}_1 \cdots \mathbf{S}_p]$ where $\mathtt{M}_i$ is the $2 \times r$ camera matrix that projects the 3D shape onto the image frame i and $\mathbf{S}_j$ encodes the 3D coordinates of point j.

2.1 Rigid Shape

In the case of a rigid object viewed by an orthographic camera, if we assume the measurements in $\mathtt{W}$ are registered to the image centroid, the camera motion matrices $\mathtt{M}_i$ and the 3D points $\mathbf{S}_j$ can be expressed as: $\mathtt{M}_i = \begin{bmatrix} r_{i1} & r_{i2} & r_{i3} \\ r_{i4} & r_{i5} & r_{i6} \end{bmatrix} = \mathtt{R}_i$ and $\mathbf{S}_j = \begin{bmatrix} X_j Y_j Z_j \end{bmatrix}^\top$ where $\mathtt{R}_i$ is a 2×3 matrix whose transpose lies on the Stiefel manifold (i.e. a 3×2 Stiefel matrix), since $\mathtt{R}_i$ contains the first two rows of a rotation matrix (i.e. $\mathtt{R}_i\mathtt{R}_i^\top = \mathtt{I}_{2\times 2}$) and $\mathbf{S}_j$ is a 3-vector containing the metric coordinates of the 3D point. Therefore the rank of the measurement matrix is $r \leq 3$. The rigid *motion manifold* corresponds to the manifold of matrices with pairwise orthogonal rows.

2.2 Deformable Shape Model

In the case of deformable objects the observed 3D points change as a function of time. In this paper we use the low-rank shape model defined in (Bregler *et al.*, 2000) in which the 3D points deform as a linear combination of a fixed set of k rigid shape bases according to time varying coefficients. In this way, $\mathtt{S}_i = \sum_{d=1}^{k} l_{id}\mathtt{B}_d$ where the matrix $\mathtt{S}_i = [\mathbf{S}_{i1}, \cdots \mathbf{S}_{ip}]$ is the 3D shape of the object at frame i, the $3 \times p$ matrices $\mathtt{B}_d$ are the shape bases and l_{id} are the coefficient weights. If we assume an orthographic projection model the coordinates of the 2D image points observed at each frame i are then given by:

$$\mathtt{W}_i = \mathtt{R}_i \left(\sum_{d=1}^{k} l_{id}\mathtt{B}_d \right) + \mathtt{T}_i \tag{2}$$

where the matrix $\mathtt{R}_i$ is 2×3 with orthonormal rows, such that $\mathtt{R}_i^\top$ is a *Stiefel matrix* and the $2 \times p$ matrix $\mathtt{T}_i$ aligns the image coordinates to the image centroid. The aligning matrix $\mathtt{T}_i$ is such that $\mathtt{T}_i = \mathbf{t}_i \mathbf{1}_p^\top$ where the 2-vector $\mathbf{t}_i$ is the 2D image centroid and $\mathbf{1}_p$ a vector of ones. When the image coordinates are registered to the centroid of the object and we consider all the frames in the sequence, we may write the measurement matrix as:

$$\mathtt{W} = \begin{bmatrix} l_{11}\mathtt{R}_1 & \dots & l_{1k}\mathtt{R}_1 \\ \vdots & \ddots & \vdots \\ l_{f1}\mathtt{R}_f & \dots & l_{fk}\mathtt{R}_f \end{bmatrix} \begin{bmatrix} \mathtt{B}_1 \\ \vdots \\ \mathtt{B}_k \end{bmatrix} = \begin{bmatrix} \mathtt{M}_1 \\ \vdots \\ \mathtt{M}_f \end{bmatrix} \begin{bmatrix} \mathtt{B}_1 \\ \vdots \\ \mathtt{B}_k \end{bmatrix} = \mathtt{MS} \tag{3}$$

Since $\mathtt{M}$ is a $2f \times 3k$ matrix and $\mathtt{S}$ is a $3k \times p$ matrix in the case of deformable structure the rank of $\mathtt{W}$ is constrained to be at most $3k$. The motion matrices now have the form $\mathtt{M}_i = [\mathtt{M}_{i1} \dots \mathtt{M}_{ik}] = [l_{i1}\mathtt{R}_i \dots l_{ik}\mathtt{R}_i]$. Therefore, in the deformable *motion manifold* the motion matrices have a distinct repetitive structure and every 2×3 $\mathtt{M}_{ik}$ sub-block is composed of the transpose of a *Stiefel matrix* multiplied by a scalar.

2.3 Articulated Shape Model

In the case of articulated structure, the relative motions of the segments that form an articulated body are dependent and this results in a drop in the dimensionality of the measurement matrix $\mathtt{W} = \left[\, \mathtt{W}^{(1)} \middle| \mathtt{W}^{(2)} \,\right]$ that contains the 2D image points of the two segments. In the case of a *universal joint* the two shapes share a common translation (i.e. the distance between the centres of mass of the shapes is constant) while in the case of a *hinge joint* the shapes also share a common rotation axis (Tresadern & Reid, 2005; Yan & Pollefeys, 2008). Naturally, this approach requires that an initial segmentation stage has taken place to assign the trajectories in $\mathtt{W}$ to the respective shapes for which a solution was recently provided in (Yan & Pollefeys, 2008).

In a *universal joint* (Tresadern & Reid, 2005) the distance between the centres of the two shapes is constrained to be constant (for instance, the head and the torso of a human body) but with independent rotation components. At each frame the shapes connected by a joint satisfy:

$$\mathbf{t}^{(1)} + \mathtt{R}^{(1)}\mathbf{d}^{(1)} = \mathbf{t}^{(2)} + \mathtt{R}^{(2)}\mathbf{d}^{(2)} \tag{4}$$

where $\mathbf{t}^{(1)}$ and $\mathbf{t}^{(2)}$ are the 2D image centroid of the two objects, $\mathtt{R}^{(1)}$ and $\mathtt{R}^{(2)}$ the 2×3 orthographic camera matrices and $\mathbf{d}^{(1)}$ and $\mathbf{d}^{(2)}$ the 3D displacement vectors of each shape from the joint. The relation in equation (4) gives the reduced dimensionality in the motion and shape subspaces. Thus, the shape matrix $\mathtt{S}$ can be written as:

$$\mathtt{S} = \begin{bmatrix} \mathtt{S}^{(1)} & \mathbf{d}^{(1)} \\ \mathtt{0} & \mathtt{S}^{(2)} - \mathbf{d}^{(2)} \\ \mathbf{1} & \mathbf{1} \end{bmatrix} \tag{5}$$

where $\mathtt{S}$ is a full rank-7 matrix. The motion for a frame i has to be accordingly arranged to satisfy equation (4) as:

$$\mathtt{M}_i = \left[\, \mathtt{R}_i^{(1)} \; \mathtt{R}_i^{(2)} \; \mathbf{t}_i^{(1)} \,\right]. \tag{6}$$

In the case of a *hinge joint*, if we assume the image coordinates to be registered to the centroid of each segment, then the motion matrices $\mathtt{M}_i$ that lie on the articulated *motion manifold* can be written as:

$$\mathtt{M}_i = \left[\, \mathbf{u}_i \; \mathtt{A}_i \; \mathtt{B}_i \,\right] \tag{7}$$

where $\mathbf{u}$ is the common rotation axis for both objects, $\mathtt{A}_i$ and $\mathtt{B}_i$ are 2×2 matrices such that $\left[\, \mathbf{u}_i \mathtt{A}_i \,\right]$ and $\left[\, \mathbf{u}_i \mathtt{B}_i \,\right]$ are the 2×3 camera matrices (with orthonormal rows) associated with the first and second shape respectively. The metric constraints in the case of a hinge can therefore be expressed as:

$$\begin{aligned} [\mathbf{u}_i \; \mathtt{A}_i] \begin{bmatrix} \mathbf{u}_i^\top \\ \mathtt{A}_i^\top \end{bmatrix} &= \mathtt{I}_{2\times 2} \\ [\mathbf{u}_i \; \mathtt{B}_i] \begin{bmatrix} \mathbf{u}_i^\top \\ \mathtt{B}_i^\top \end{bmatrix} &= \mathtt{I}_{2\times 2} \end{aligned} \tag{8}$$

where, without loss of generality, we have implicitly assumed that the axis of rotation is aligned with the x-axis of the first object. Thus we can write $\mathtt{S}$ as:

$$\mathtt{S} = \begin{bmatrix} x_1^{(1)} & \cdots & x_{p_1}^{(1)} & x_1^{(2)} & \cdots & x_{p_2}^{(2)} \\ y_1^{(1)} & \cdots & y_{p_1}^{(1)} & 0 & \cdots & 0 \\ z_1^{(1)} & \cdots & z_{p_1}^{(1)} & 0 & \cdots & 0 \\ 0 & \cdots & 0 & y_1^{(2)} & \cdots & y_{p_2}^{(2)} \\ 0 & \cdots & 0 & z_1^{(2)} & \cdots & z_{p_2}^{(2)} \end{bmatrix} \tag{9}$$

where now $\mathtt{S}$ is a $5 \times p$ matrix and $p = p_1 + p_2$ (we assume the shapes have been registered to the respective object centroids). Therefore, in the case of a hinge joint the rank of the measurement matrix is at most 5.

3 Metric Upgrade

The classic approach in factorisation is to exploit the rank constraint to factorise the measurement matrix into an initial affine solution with a motion matrix $\tilde{\mathtt{M}}$ and a shape matrix $\tilde{\mathtt{S}}$ by truncating the SVD of $\mathtt{W}$ to the rank r specific to the problem. However, this factorisation is not unique since any invertible $r \times r$ matrix $\mathtt{Q}$ can be inserted, leading to the alternative factorisation: $\mathtt{W} = (\tilde{\mathtt{M}}\mathtt{Q})(\mathtt{Q}^{-1}\tilde{\mathtt{S}})$. The problem is to find the transformation matrix $\mathtt{Q}$ that removes the affine ambiguity, upgrading the reconstruction to metric and constraining the motion matrices to lie on the appropriate *motion manifold*.

While in the rigid case the matrix $\mathtt{Q}$ can be explicitly computed linearly by imposing orthonormality constraints on the rows of the motion matrix (Tomasi & Kanade, 1992), in the non-rigid and articulated cases the metric constraints on the motion matrices are non-linear. Although some closed-form solutions have been recently proposed (Xiao & Kanade, 2005; Xiao *et al.*, 2006; Hartley & Vidal, 2008) these algorithms perform poorly in the presence of noise and cannot cope with missing data. Iterative solutions provide a viable alternative in the presence of noise and missing data and this procedure will be adopted in our proposed algorithm. The factorisation of $\mathtt{W}$ is solved with an alternating least-squares problem where at each step t the motion $\mathtt{M}^{(t)}$ and shape $\mathtt{S}^{(t)}$ matrices are optimised separately keeping the other one fixed as shown in Algorithm 1. This strategy is not uncommon in optimisation problems for SfM (Buchanan & Fitzgibbon, 2005) however it is important to notice is that, differently from previous optimisation schemes, we use a projection step which computes a solution that satisfies the metric constraints exactly. The metric constraints consist of two parts: imposing the correct block structure to the motion matrix and constraining the transpose of the orthographic camera matrices to lie on the Stiefel manifold. In our approach, we impose both constraints simultaneously projecting the motion matrix optimally onto the appropriate motion manifold. As already noticed by (Marques & Costeira, 2008) for the rigid case, these projections not only provide camera matrices which exactly comply with the projection model but also are generally robust to missing and degenerate data.

Algorithm 1 Iterative metric upgrade via alternation for deformable and articulated shape. At each step of the iteration, the motion matrix estimated via least squares is projected onto the motion manifold.

```
Require: An initial estimate $\mathtt{M}^{(0)}$.
Ensure: A factorisation of $\mathtt{W}$ that satisfies the given metric constraints.
1: Project each frame of $\mathtt{M}^{(t)}$ onto the *motion manifold* of the motion matrices (See Section 3.1 for the deformable case and Section 3.3 for the articulated case).
2: Estimate $\mathtt{S}^{(t)}$ from the projected $\mathtt{M}^{(t)}$ as: $\mathtt{S}^{(t)} = \mathtt{M}^{(t)\dagger}\mathtt{W}$ (where the symbol $^\dagger$ indicates the MoorePenrose pseudo-inverse.
3: Estimate $\mathtt{M}^{(t+1)}$ such that: $\mathtt{M}^{(t+1)} = \mathtt{W}\mathtt{S}^{(t)\dagger}$.
4: Repeat until convergence.
```

Crucially, Step 1 represents the real and novel contribution of this algorithm: an optimisation method which computes the projection of the affine motion components onto the *motion manifold* in which the exact metric constraints are satisfied. Although this problem is non-convex we propose efficient convex relaxations (in the sense that the relaxations turned out to be exact, in our numerical simulations) that transform the problems into semi-definite (SDP) or second-order cone (SOCP) programs. Steps 2 and 3 alternate the estimation of $\mathtt{M}^{(t)}$and $\mathtt{S}^{(t)}$ assuming the other one known.

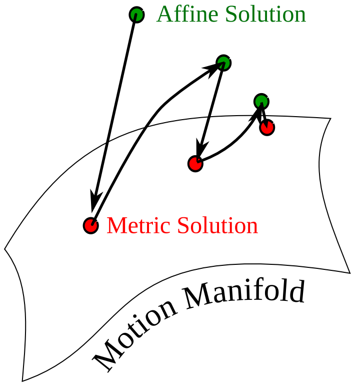


Fig. 1 Iterative scheme: at each step of the iteration, the motion matrix computed via least squares is projected onto the motion manifold of metric constraints. The process is iterated until convergence

Previous approaches have also used iterative methods to perform the metric upgrade in the case of non-rigid structure including the trilinear alternating least-squares methods described in (Torresani *et al.*, 2001) and in (Wang *et al.*, 2008). However, even though Torresani *et al.*'s method imposes exact metric constraints on the camera matrices by parametrisation, the update of the camera matrix relies on the assumption that the current estimate differs from the next one only by small rotations. Moreover, the recovery of camera matrices is not optimal. In our case we have an optimal solution to the projection step, which re-estimates the camera matrices and the coefficients to obtain the closest matrix that satisfies the metric constraints. The metric projection step can be visualised in Figure 1. Also Wang *et al.* (Wang *et al.*, 2008) adopt a trilinear approach where the constraints on the orthographic camera matrices at each frame are imposed using a projection. Their projector is in fact equivalent to the one developed in parallel by (Marques & Costeira, 2008) for rigid shape in the scaled orthographic case. The projection is computed as: $\mathtt{M}_i \mapsto \mathtt{R}_i = \alpha \mathtt{U}\mathtt{V}^\top$ where α is given by the mean of the two singular values $\frac{\sigma_1(\mathtt{M}_i) + \sigma_2(\mathtt{M}_i)}{2}$ obtained from the SVD of $\mathtt{M}_i$ (i.e. $\mathtt{M}_i = \mathtt{U}\mathtt{D}\mathtt{V}^\top$). In order to extend such procedure to non-rigid shapes, we first need to define the *motion manifold* for the deformable and articulated cases and to provide the computational tools to project the motion matrices exactly from affine to metric space.

While other papers have chosen to use priors on the shape to constrain the solution to the optimisation problem and obtain the metric upgrade (Bartoli *et al.*, 2008; Torresani *et al.*, 2008; Del Bue, 2008), in this paper we provide a metric upgrade step that solves an unconstrained least-squares problem and optimally projects the solution onto the *motion manifold* (i.e, computes the closest matrix in the motion manifold with respect to the Frobenius norm). In such regard, we postulate that reliable solutions to the NRSfM problem can be obtained without the use of prior information about the motion of the object or the smoothness of its deformations. In the case of articulated structure, we solve globally for both the motion components related to the bodies and the joint axis with a similar procedure. We now give details on how these projections are computed and the theoretical insights for the *motion manifold* of deformable and articulated shapes.

3.1 Metric Projection: Deformable Case

The projection is carried out on each $2 \times 3k$ sub-matrix $\mathtt{M}_i$ as defined in Section 2 and it corresponds to solving the following minimisation problem at each frame:

$$\min_{\mathtt{R}_i, l_{i1} \ldots l_{ik}} ||\mathtt{M}_i - [l_{i1}\mathtt{R}_i|\ldots|l_{ik}\mathtt{R}_i]||_F^2 \tag{10}$$

with the added constraint that $\mathtt{R}_i$ be a 2×3 matrix with orthonormal rows (i.e. $\mathtt{R}_i\mathtt{R}_i^\top = \mathtt{I}_{2\times2}$). This is equivalent to minimising separately all the 2×3 blocks of $\mathtt{M}_i$ giving:

$$\min_{\mathtt{R}_i} \sum_{d=1}^{k} \min_{l_{i1}\ldots l_{ik}} ||\mathtt{M}_{id} - l_{id}\mathtt{R}_i||_F^2 \tag{11}$$

which is equivalent to:

$$\min_{\mathtt{R}_i, l_{i1}\ldots l_{ik}} \sum_{d=1}^{k} ||\mathtt{M}_{id}||_F^2 + l_{id}^2 \, ||\mathtt{R}_i||_F^2 - 2 l_{id} \operatorname{Tr}[\mathtt{M}_{id}^\top \mathtt{R}_i]. \tag{12}$$

We can then reformulate the problem by computing the minimum first for l_d (i.e. solving for the zeros of the derivative of eq. (11)) given $\mathtt{R}$. This resolves in computing the minimum of the quadratic function in l_d given by $f(l_d) = a\, l_d^2 - 2\, b\, l_d + c$. Such minimum is found in $l_d = b/a$ giving in our case that:

$$l_{id} = \frac{\operatorname{Tr}[\mathtt{M}_{id}^\top \mathtt{R}_i]}{||\mathtt{R}_i||_F^2} = \frac{1}{2} \operatorname{Tr}[\mathtt{M}_{id}^\top \mathtt{R}_i]. \tag{13}$$

Putting this value back in eq. (11) and following with the simplification, the minimisation can be written as:

$$\begin{array}{ll} \min_{\mathtt{R}_i} & \mathbf{r}_i^\top \left[-\sum_{d=1}^{k} \mathbf{m}_{id}\mathbf{m}_{id}^\top \right] \mathbf{r}_i \\ \text{such that} & \mathtt{R}_i\mathtt{R}_i^\top = \mathtt{I}_{2\times2} \end{array} \tag{14}$$

where $\mathbf{r}_i = \operatorname{vec}(\mathtt{R}_i^\top)$ and $\mathbf{m}_{id} = \operatorname{vec}(\mathtt{M}_{id}^\top)$. Therefore, this quadratic minimisation problem presents a non-convex constraint given by $\mathtt{R}_i$. In Appendix A we show that it is possible to derive an efficient convex relaxation of the constraint set. This set is defined only by linear matrix inequalities (LMI). Therefore the optimisation problem is a Semi-Definite Program (SDP) which can be solved using SeDuMi (Sturm, 1999). Further details, including a proof of the relaxation can be found in (Dodig *et al.*, 2009).

The computed *Stiefel matrix* $\mathtt{R}_i^\top$ is then used to recover the weights l_{id}, obtaining a full non-rigid motion matrix that satisfies the metric constraints. This allows us to solve iteratively for the motion and shape as described in Algorithm 1. This optimal metric projection step was first introduced in (Paladini *et al.*, 2009). The disadvantage of this approach is that the computational complexity of solving a quadratic minimisation problem for each frame in the sequence is too onerous. Each minimisation takes about 2 seconds using SeDuMi toolbox (on a Athlon X2 processor running at 2.6GHz), therefore a sequence of 120 frames would take around 4 minutes to process. While this computation time is not unreasonable for a batch process, in Section 3.2 we present a new algorithm based on a Newton optimisation method on the Stiefel manifold to speed up the computation by a factor of around 130. First we describe the initialisation to the minimisation.

3.1.1 Initialisation for the deformable case

Algorithm 1 requires an initial estimate of the motion matrix $\mathtt{M}_i$ at each frame. This in turn requires initial estimates for the camera matrices $\bar{\mathtt{R}}_i$ and the configuration weights $\bar{l}_{id}$. The rigid motion $\bar{\mathtt{R}}_i$ and the first basis shape $\bar{\mathtt{S}}_1$ are estimated from a rank 3 rigid factorisation of the measurement matrix. The second component of the shape bases is estimated from the residual

$$\mathtt{W}_r = \mathtt{W} - \bar{\mathtt{M}}\bar{\mathtt{S}}_1 \tag{15}$$

A new rank 3 factorisation is performed on $\mathtt{W}_r$ and the new configuration weights l_{i2} can be estimated solving for $l_{i2}\bar{\mathtt{R}}_i = \mathtt{M}_{i2}$ keeping the rotations fixed. This can be solved in a simple way by taking advantage of the orthonormality of $\mathtt{R}$:

$$\begin{aligned} \operatorname{vec}(\mathtt{R}_i) l_{ij} &= \operatorname{vec}(\mathtt{M}_{ij}) \\ \operatorname{vec}(\mathtt{R}_i)^\top \operatorname{vec}(\mathtt{R}_i) l_{ij} &= \operatorname{vec}(\mathtt{R}_i)^\top \operatorname{vec}(\mathtt{M}_{ij}) \\ ||\mathtt{R}||_F^2 l_{ij} &= \operatorname{vec}(\mathtt{R}_i)^\top \operatorname{vec}(\mathtt{M}_{ij}) \\ 2 l_{ij} &= \operatorname{vec}(\mathtt{R}_i)^\top \operatorname{vec}(\mathtt{M}_{ij}) \end{aligned}$$

This process is repeated to obtain all k deformation modes. The first rigid factorisation needs full data to give a solution, so we use Marques and Costeira's rigid factorisation algorithm (Marques & Costeira, 2009) if missing data are present.

3.2 Newton method on the Stiefel manifold

The approach described in the previous section will provide an optimal projection onto the *motion manifold* of deformable structure. The first observation we made is that the motion matrix for one frame is not unrelated to the next one. For most common image sequences the motion of the camera is smooth, thus each motion matrix $\mathtt{M}_i$ will not vary much from frame to frame. Therefore, it is not unrealistic to assume that the camera pose at frame i is a good initialisation for an iterative algorithm which tries to compute the pose in the next frame $i+1$. This *warm-start* strategy is not explicitly designed for standard solvers for convex optimisation problems ((Sturm, 1999)). Instead, we have adopted a Newton-like iterative optimisation algorithm based on the work of (Edelman *et al.*, 1999). We perform iterative optimisation directly on the Stiefel manifold which, for the case of smoothly varying camera poses, will converge locally to the minimum. Of course we lose the optimality of the convex relaxation algorithm. However, empirically we found that in all our experiments with ground truth data both algorithms converged to the same minimum.

We now provide additional details on how to compute the Newton step update for the *motion manifold* of deforming shapes. To adhere to the notation in (Edelman *et al.*, 1999) we define the problem as that of minimising a function $\mathtt{F}(\mathtt{Y})$, where $\mathtt{Y}$ is constrained to the set of matrices such that $\mathtt{Y}^\top\mathtt{Y} = \mathtt{I}$ i.e. it is a *Stiefel matrix*. The current estimate of the Stiefel matrix is updated in the Newton direction $\boldsymbol{\Delta}$ using the geodesic formula for a unit step $t = 1$

$$\mathtt{Y}(t) = \mathtt{YM}(t) + \mathtt{QN}(t) \tag{16}$$

where $\mathtt{QR}$ is the compact QR-decomposition of $(\mathtt{I}-\mathtt{YY}^\top)\boldsymbol{\Delta}$, with the Newton direction $\boldsymbol{\Delta}$ given by

$$\boldsymbol{\Delta} = -\text{Hessian}^{-1}(\mathtt{F}_Y - \mathtt{YF}_Y^\top\mathtt{Y}) \tag{17}$$

(where $\mathtt{F}_Y$ is the first derivative with respect to $\mathtt{Y}$) and, finally, the matrices $\mathtt{M}(t)$ and $\mathtt{N}(t)$ are given by the matrix exponential

$$\begin{pmatrix} \mathtt{M}(t) \\ \mathtt{N}(t) \end{pmatrix} = \exp t \begin{pmatrix} \mathtt{A} & -\mathtt{R}^\top \\ \mathtt{R} & 0 \end{pmatrix} \begin{pmatrix} \mathtt{I}_p \\ 0 \end{pmatrix} \tag{18}$$

with $\mathtt{A} = \mathtt{Y}^\top\boldsymbol{\Delta}$.

We apply the iterative Newton method (more theoretical insights can be found in (Edelman *et al.*, 1999)) to the cost function given by equation (14), using the solution to the previous frame as an initialisation. Evidently, the first frame has to be solved with the previously proposed convex relaxation. In our experiments this new solution provided a remarkable speedup, solving the whole factorisation problem about 130 times faster than the original method, without losing optimality as observed in the experimental trials. Notice that in this case the assumption that the camera pose varies smoothly is just an initialisation strategy and not a prior term in our minimisation. Our smoothness assumption does not add an explicit penalty term to the cost function to penalise strong deformations or camera motions as other authors do (Bartoli *et al.*, 2008; Torresani *et al.*, 2008).

3.3 Metric Projection: Articulated Case

Projection onto the *motion manifold* of the universal joint can be simply solved by performing two separate rigid factorisations for each of the parts of the articulated object followed by an estimation of the joint location as presented in (Tresadern & Reid, 2005). The hinge joint is far more interesting given the non-linear relations between the motion subspaces. Here the problem is to find the closest matrix that satisfies the metric constraints given a rotation axis between two objects. Following eq. (6) the projection problem for the hinge *motion manifold* can be written at each frame as the following minimisation:

$$\min_{\mathbf{u},\mathtt{A},\mathtt{B}} J(\mathbf{u},\mathtt{A},\mathtt{B}) = \|\mathbf{u}-\mathbf{x}\|^2 + \|\mathtt{A}-\mathtt{Y}\|_F^2 + \|\mathtt{B}-\mathtt{Z}\|_F^2\,, \tag{19}$$

subject to the constraints defined in eq. (8). Here $\mathbf{x}$, $\mathtt{Y}$ and $\mathtt{Z}$ are obtained directly from the affine motion matrix $\tilde{\mathtt{M}}_i = [\mathbf{x}|\mathtt{Y}|\mathtt{Z}]$, recovered through SVD. Equation (19) can be reformulated (Paladini *et al.*, 2009) as the minimisation of $J(\mathbf{u},\mathtt{A},\mathtt{B})$ only as a function of the common axis $\mathbf{u}$ such that:

$$\min_{\mathbf{u},\mathtt{A},\mathtt{B}} J(\mathbf{u},\mathtt{A},\mathtt{B}) = \min_{\mathbf{u}} J(\mathbf{u}). \tag{20}$$

This is possible as we will show that, once the optimal $\mathbf{u}$ is estimated, it is straightforward to obtain $\mathtt{A}$ and $\mathtt{B}$ in closed form. The equivalent cost function $J(\mathbf{u})$ can be written as:

$$\min_{\mathbf{u}} J(\mathbf{u}) = \min_{\mathbf{u}} \left\{\|\mathbf{u}-\mathbf{x}\|^2 + \phi_Y(\mathbf{u}) + \phi_Z(\mathbf{u})\right\}. \tag{21}$$

Thus now we will show how to transform the minimisation of $\|\mathtt{A}-\mathtt{Y}\|_F^2$ into the minimisation of $\phi_Y(\mathbf{u})$ (the same reasoning can be replicated for $\phi_Z(\mathbf{u})$). First, we use the polar decomposition to change variables as $\mathtt{A} = \mathtt{PQ}$ where $\mathtt{P} \succeq 0$ (i.e. $\mathtt{P}$ is a semidefinite matrix) and $\mathtt{Q}$ is orthogonal (both $\mathtt{P}$ and $\mathtt{Q}$ are 2×2). Moreover, given the metric constraints in eq. (8), it follows that $\mathtt{P}^2 = \mathtt{I} - \mathbf{uu}^\top$. Thus, the matrix $\mathtt{I} - \mathbf{uu}^\top$ must be positive definite, restricting the vector $\mathbf{u}$ to be inside the unitary circle. Then, for a chosen $\mathbf{u}$ we can write $\phi_Y(\mathbf{u})$ as:

$$\begin{aligned}\phi_Y(\mathbf{u}) &= \min_{\mathtt{QQ}^\top=\mathtt{I}} \left\|(\mathtt{I}-\mathbf{uu}^\top)^{1/2}\mathtt{Q}-\mathtt{Y}\right\|_F^2 \\ &= \min_{\mathtt{QQ}^\top=\mathtt{I}} \left\{\left\|(\mathtt{I}-\mathbf{uu}^\top)^{1/2}\right\|_F^2 + \|\mathtt{Y}\|_F^2 \right. \\ &\quad \left. - 2\,\mathrm{Tr}\left(\mathtt{Y}^\top\left(\mathtt{I}-\mathbf{uu}^\top\right)^{1/2}\mathtt{Q}\right)\right\}.\end{aligned}$$

Minimising this cost function over the orthogonal matrix $\mathtt{Q}$ equals to maximising the trace in the previous expression.

Using the property:

$$\max_{\mathtt{QQ}^\top=\mathtt{I}} \{\mathrm{Tr}\,(\mathtt{XQ})\} = \sigma_1(\mathtt{X}) + \sigma_2(\mathtt{X}) + \cdots + \sigma_n(\mathtt{X}) = \|\mathtt{X}\|_{\mathrm{N}} \tag{22}$$

where $\|\mathtt{X}\|_{\mathrm{N}}$ denotes the *nuclear norm* of $\mathtt{X}$ (i.e. the sum of its singular values), we can write that:

$$\phi_Y(\mathbf{u}) = 2 - \|\mathbf{u}\|^2 + \|\mathtt{Y}\|_F^2 - 2\left\|\left(\mathtt{I} - \mathbf{u}\mathbf{u}^\top\right)^{1/2}\mathtt{Y}\right\|_{\mathrm{N}} \quad (23)$$

The same reasoning can be replicated for $\phi_Z(\mathbf{u})$ giving the final optimisation problem to be solved as:

$$\begin{array}{ll} \min & -\|\mathbf{u}\|^2 - 2\mathbf{u}^\top\mathbf{x} - 2\left\|\left(\mathtt{I} - \mathbf{u}\mathbf{u}^\top\right)^{1/2}\mathtt{Y}\right\|_{\mathrm{N}} \\ \|\mathbf{u}\| \leq 1 & \qquad -2\left\|\left(\mathtt{I} - \mathbf{u}\mathbf{u}^\top\right)^{1/2}\mathtt{Z}\right\|_{\mathrm{N}} \end{array} \quad (24)$$

Once the optimal $\mathbf{u}^*$ is found we substitute back in order to recover the solution for $\mathtt{A}$ (and similarly for $\mathtt{B}$). First we obtain $\mathtt{Q}$ from the SVD of $\mathtt{Y}^\top(\mathtt{I} - \mathbf{u}^*\mathbf{u}^{*\top})^{1/2} \mapsto \mathtt{UDV}^\top$ leading to $\mathtt{Q} = \mathtt{VU}^\top$. The matrix $\mathtt{P}$ is simply given knowing that $\mathtt{P}^2 = \mathtt{I} - \mathbf{u}^*\mathbf{u}^{*\top}$. This will result in the matrix that exactly satisfies the metric structure of a hinge joint. The optimisation of the cost function in eq. (24) is not trivial since the cost function is non-convex and non-smooth. However the domain in which the function resides is very constrained (i.e. the unitary circle) and the value of eq. (24) for an arbitrary $\mathbf{u}$ can be computed efficiently without the need of calculating the nuclear norm at each sample. The optimisation can be then solved with a simple exhaustive search algorithm in which the function samples can be computed in a small amount of time (details on this computation can be found in (Paladini *et al.*, 2009)).

3.3.1 Convex relaxation for the articulated case

Although the cost function in equation (24) is non-convex, in Appendix B we propose an efficient convex relaxation. Differently from the deformable case, the reformulation leads to two cases. As shown in Appendix B, in one case the problem becomes a semi-definite program (SDP) and in the other a second order cone program (SOCP) both of which can be efficiently solved with standard convex optimisation tools (Sturm, 1999). In all of our numerical experiments we found that the proposed convex relaxations were exact, thereby solving indeed (24). Compared to the full search method presented in (Paladini *et al.*, 2009), this convex optimisation speeds up the computation by a factor of around ten. A second advantage is that we avoid the problem of the accuracy of the solution depending on the density of the interval grid in the parameter space as in the full-search algorithm. The full details of the proposed convex relaxation can be found in Appendix B.

3.3.2 Initialisation for the articulated case

We first consider the two bodies separately and then perform a rigid factorisation for each shape. Given this factorisation, we can then obtain an initial closed form solution for the metric upgrade in the case of a hinge using the linear approximation of (Tresadern & Reid, 2005).

4 Reconstruction with Missing Data

Incomplete image tracks are a common occurrence in SfM tasks and several algorithms have been proposed in order to cope with the missing data problem within the factorisation framework (Buchanan & Fitzgibbon, 2005). Our new factorisation approach presented in the previous section can be modified to account for missing entries in $\mathtt{W}$. The strength of our approach lies in the fact that the *motion manifold* constrains the estimated motion of the missing 2D image points since we only allow trajectories that satisfy the metric constraints exactly.

Instead of using only the known image tracks to solve for the camera matrices, basis shapes and deformation coefficients as the trilinear least-squares approaches do (Torresani *et al.*, 2001; Wang *et al.*, 2008), we opt for an iterative scheme. At each step of the iteration we re-compute the missing entries in the measurement matrix $\mathtt{W}$ using the current estimates of the motion and shape matrices that have been projected onto the correct *motion manifold*. In our experimental validation, reported in Section 5, we have found that dealing with missing data using the iterative scheme described here allows to deal with higher percentages of missing data than using only the available data as Wang *et al*. do in their RCPF approach (Wang *et al.*, 2008). The steps of this method are summarised in Algorithm 2.

Algorithm 2 Metric Projections algorithm in the presence of missing data.

```
Require: An initial estimate $\mathtt{W}^{(0)}$ of the missing data in $\mathtt{W}$.
Ensure: A factorisation of $\mathtt{W}$ that satisfies the given metric constraints.
1: Remove the 2D centroid $\mathtt{T}^{(t)}$ from $\mathtt{W}^{(t)}$, i.e. $\hat{\mathtt{W}}^{(t)} = \mathtt{W}^{(t)} - \mathtt{T}^{(t)}$.
2: Factorise $\hat{\mathtt{W}}^{(t)} = \mathtt{M}^{(t)}\mathtt{S}^{(t)}$ using Algorithm 1.
3: Estimate the missing data entries of $\mathtt{W}$ as $\mathtt{W}^{(t+1)} = \mathtt{M}^{(t)}\mathtt{S}^{(t)} + \mathtt{T}^{(t)}$
4: Repeat until convergence.
```

The algorithm requires an initial estimate of the missing entries in the measurement matrix $\mathtt{W}$. For this purpose, we have used the rigid factorisation algorithm of (Marques & Costeira, 2009) to obtain an initial rigid fit of the missing entries. In the case of articulated structure we apply the algorithm independently to each of the bodies. The iterations are stopped when the distance $||\mathtt{W}^{(t+1)} - \mathtt{W}^{(t)}||_F$ falls below a user-defined threshold, that is, when the new estimate does not modify the previous values much.

5 Experiments

First we show results for the recovery of deformable structure, followed by results for articulated structure. We evaluate the performance of our algorithms quantitatively on various motion capture sequences, for which ground truth was available, and we compare our results with some current

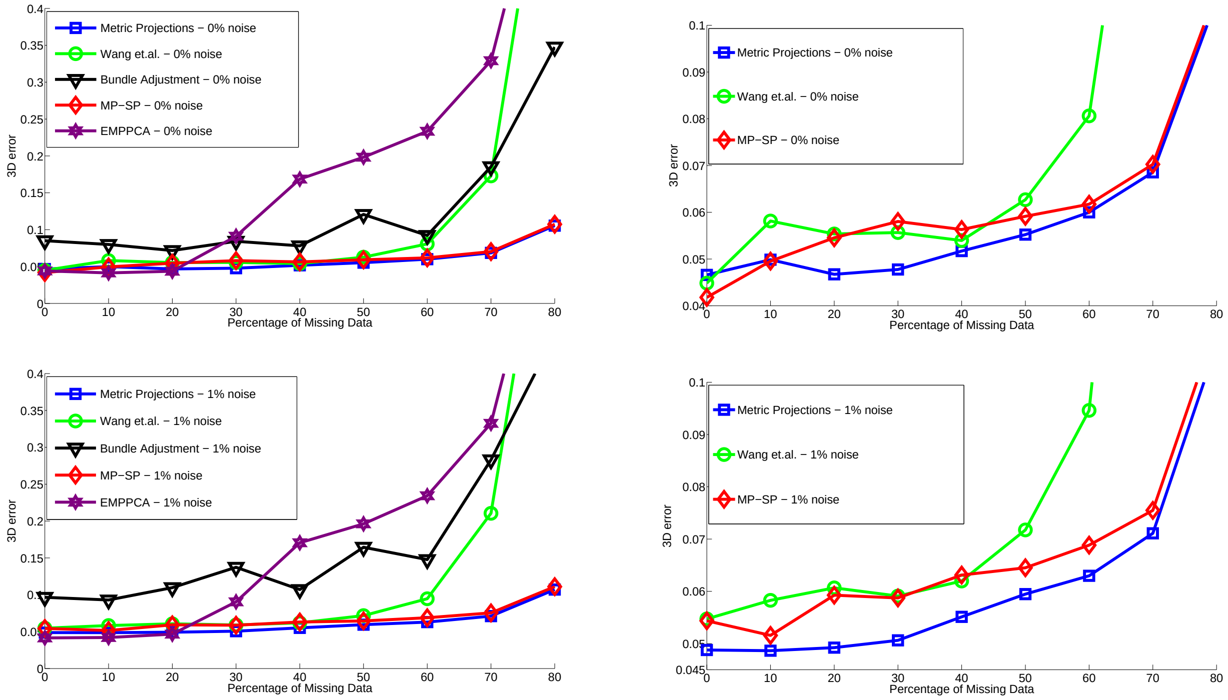


Fig. 2 Missing data tests on the *Face1* Motion Capture sequence. Plots show the average 3D error over 100 tests for increasing levels of randomly generated missing data. We compare the results obtained with: Metric Projections (MP), EMPPCA, Bundle Adjustment (BA), Rotation Constrained Powerfactorization (RCPF) and MP with a Simple Projector (MP-SP). The plots on the left column show the average 3D errors in the noise-less case (top) and with added Gaussian noise (bottom) of $\sigma = 1\%$. The plots on the right show a zoomed-in version of the three best performing algorithms (MP, RCPF and MP-SP). The performance of MP and MP-SP is similar although MP outperforms MP-SP.

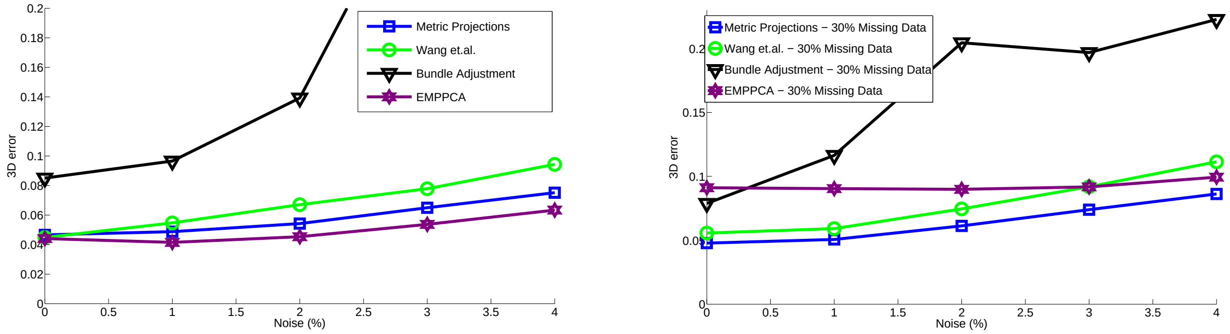
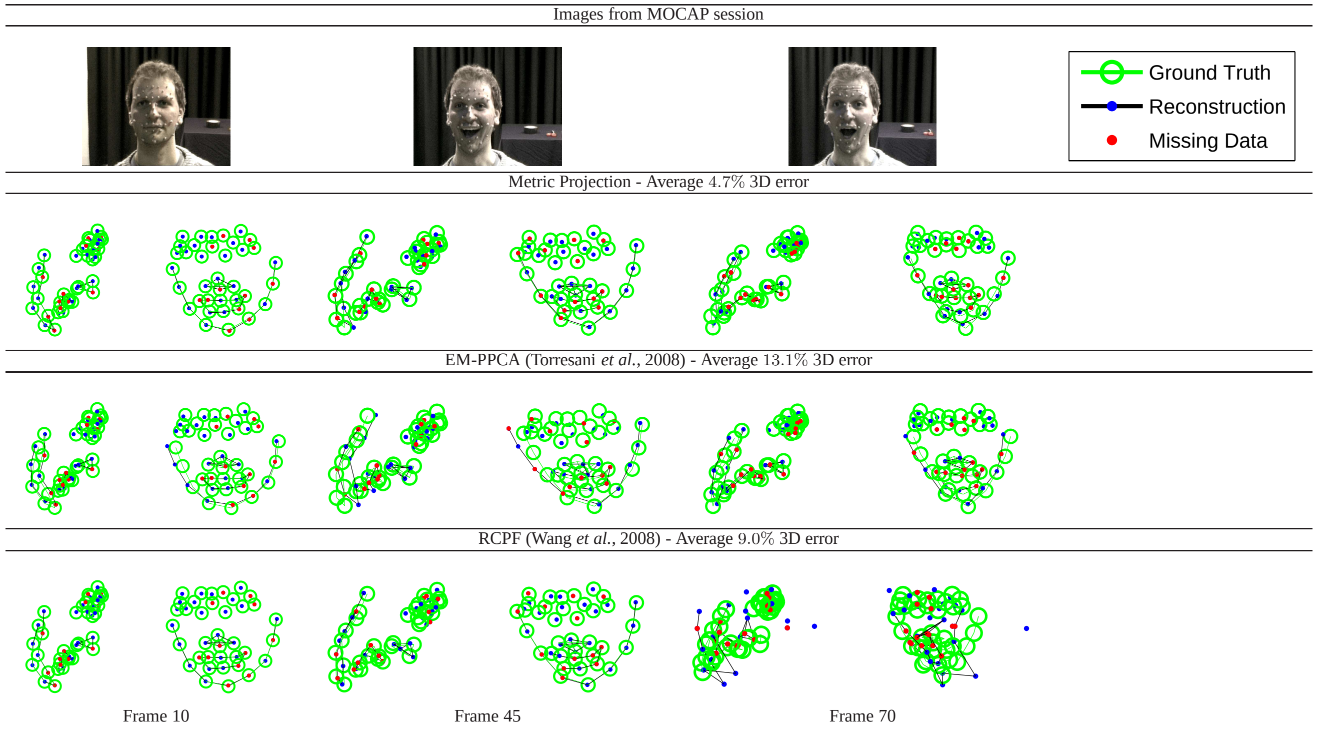


Fig. 3 Noise test for the *Face1* Motion Capture sequence in the cases of full data case (left) and 30% missing data (right). We show 3D errors versus percentage of added Gaussian noise. In the full data case (left), EMPPCA performs marginally better while in the missing data case (right) MP is the best performing algorithm.

state of the art NRSfM algorithms (Torresani *et al.*, 2008; Del Bue *et al.*, 2007; Wang *et al.*, 2008). In the case of the articulated Metric Projections (MP) algorithm we evaluated against (Tresadern & Reid, 2005). Notice that we do not compare with Yan and Pollefeys' approach (Yan & Pollefeys, 2008) since their proposed method does not perform a 3D metric reconstruction of the shape and joint axes – only the 2D projection of the axes in the image is computed. Finally we demonstrate our algorithms on real image sequences. We have made our code and sequences available for download on our website[2].

[2] `http://www.dcs.qmul.ac.uk/~lourdes/code.html`


Images from MOCAP session

Metric Projection - Average 4.7% 3D error

EM-PPCA (Torresani *et al.*, 2008) - Average 13.1% 3D error

RCPF (Wang *et al.*, 2008) - Average 9.0% 3D error

Fig. 4 3D reconstruction results for a single run of the the *Face1* motion capture sequence with 40% missing data. The points that were missing in each frame of the sequence are highlighted in red. Top row: Some frames of the original motion capture take (note that the images do not correspond exactly to the reconstructed frames shown below). Second, third and fourth rows: side and front views for some frames of the 3D reconstruction for our Metric Projection method, Torresani *et al.*'s EM-PPCA and Rotation Constrained Power Factorisation. We show ground truth (green circles) and reconstructed points (dots/ blue if visible red if not). The wire-frame lines are only shown for visualisation purposes.

5.1 Deformable Structure

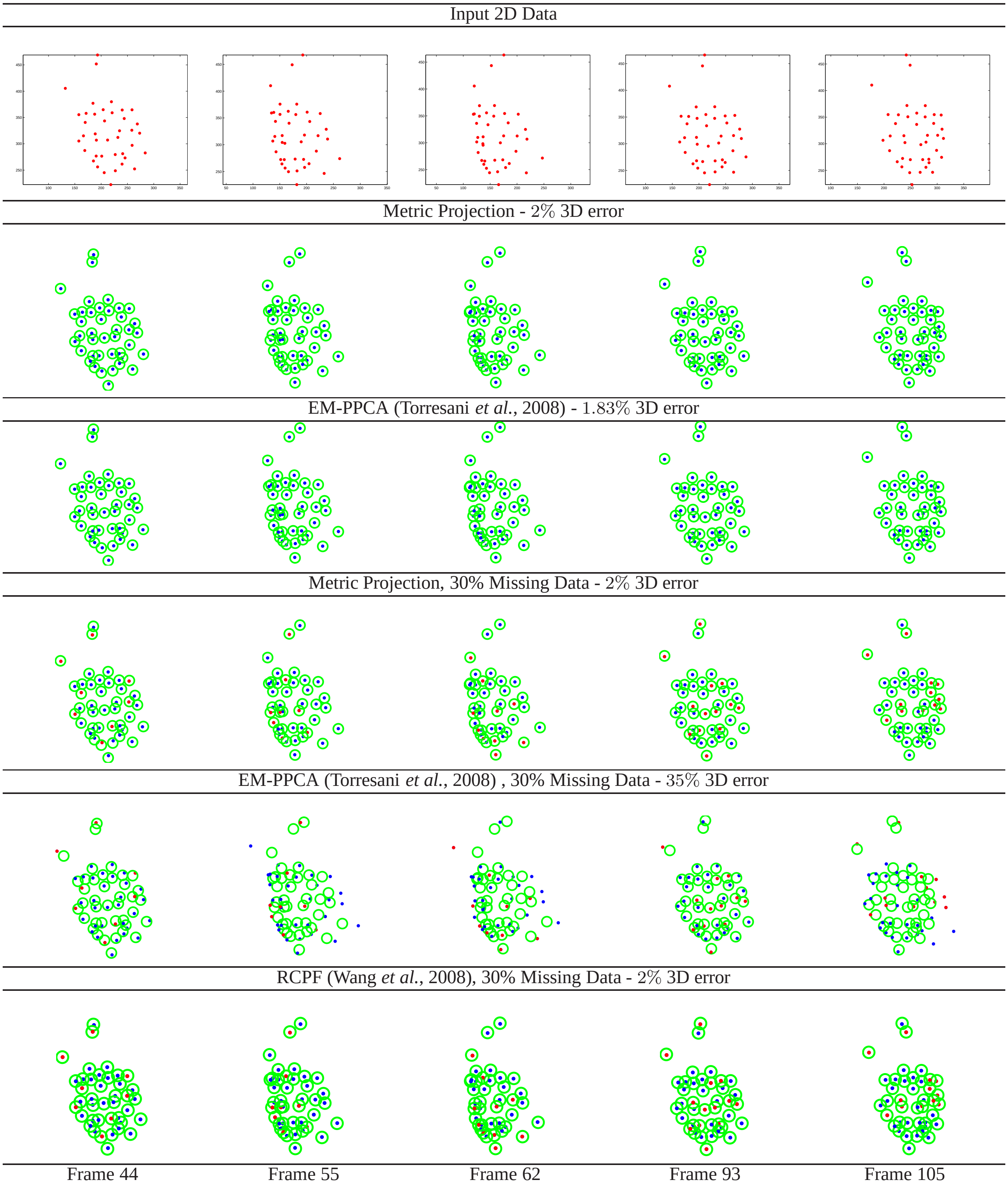
Synthetic Experiments – Motion capture data

In our synthetic experiments we used two different 3D motion capture sequences, both showing faces. The first sequence, *Face1*, was captured in our own laboratory using a VICON system tracking a subject wearing 37 markers on the face. The 3D points were then projected synthetically onto an image sequence 74 frames long using an orthographic camera model. The second sequence, *CMU* face sequence[3], is motion capture data made available by (Torresani *et al.*, 2008). The subject wore 40 markers tracked by a motion capture system and the orthographic projection is performed by simply discarding the third coordinate of each 3D point. Note that although the projection of the ground truth 3D data on the images is synthetic the deformations are realistic since they come from real motion capture sequences. The 2D image data is therefore not synthetic and it contains some noise due to the motion capture estimation errors.

Our proposed Metric Projection algorithm (MP) is tested against various state of the art algorithms: EMPPCA (Torresani *et al.*, 2008), which is currently perceived to be the state of the art/baseline algorithm and for which code has been made available online; Rotation Constrained Power Factorisation (RCPF) (Wang *et al.*, 2008), which is the most closely related approach to our new MP algorithm since it also performs a (rigid) projection of the camera matrices as we described in Section 1.1, and a Bundle Adjustment algorithm (BA) designed for NRSfM (Del Bue *et al.*, 2007) where the orthonormality constraint on the rotation matrices is imposed through parameterisation.

In the case of missing data we also report results with a modified version of our Algorithm 2. We are interested in assessing (in the case of missing data) the gain in performance achieved by using our bilinear scheme followed by our new optimal metric projector instead of Wang *et al.*'s trilinear scheme followed by their simpler projector of the camera matrices onto the motion manifold (Wang *et al.*, 2008). In order to do this we have designed a new algorithm that we call MP-SP: *Metric Projection with Simple Projection*. The idea is to use our outer loop to deal with the missing data and substitute Step 2 in Algorithm 2 with Wang *et al.*'s RCPF algorithm. In this way we can test an algorithm with the same initialisation, the same iterative outer loop to deal with missing data but using Wang *et al.*'s trilinear approach with the simpler projection step to perform factorisation. Note that this new scheme (MP-SP) is not Wang *et al.*'s RCPF algo-

[3] `http://www.cs.dartmouth.edu/~lorenzo/nrsfm.html`



Input 2D Data

Metric Projection - 2% 3D error

EM-PPCA (Torresani *et al.*, 2008) - 1.83% 3D error

Metric Projection, 30% Missing Data - 2% 3D error

EM-PPCA (Torresani *et al.*, 2008) , 30% Missing Data - 35% 3D error

RCPF (Wang *et al.*, 2008), 30% Missing Data - 2% 3D error

Frame 44 Frame 55 Frame 62 Frame 93 Frame 105

Fig. 5 3D reconstruction results for the "CMU" face motion capture sequence. First row shows the input image data. Second and third rows show the results with full data obtained with our Metric Projection algorithm and Torresani *et al.*'s EM-PPCA respectively. The 3D reconstruction results (blue dots) are compared with ground truth data (green circles). Fourth, fifth and sixth rows show comparative results for 30% missing data (missing data points are highlighted in red). Our MP algorithm can recover the 3D shape accurately even with a high percentage of missing data points, while Torresani *et al.*'s algorithm gives poor results. The RCPF method also obtains a good reconstruction (2% 3D error) in both cases of full and missing data.

rithm: the missing data is dealt with in a different way. Effectively, our Algorithm 2 (MP in the case of missing data) and the new MP-SP have exactly the same structure. They only differ in the factorisation algorithm used in Step 2: in the case of Algorithm 2 it is our MP algorithm for full data (Algorithm 1) while in the case of MP-SP it is Wang *et al.*'s RCPF algorithm.

To test the performance of the algorithms we computed the 3D error, which we defined as the Frobenius norm of the difference between the recovered 3D shape $\mathtt{S}$ and the ground truth 3D shape $\mathtt{S}_{GT}$. The error is normalised against the Frobenius norm of the ground truth shape $||\mathtt{S}-\mathtt{S}_{GT}||_F/||\mathtt{S}_{GT}||_F$. We subtract the centroid of each shape and align them with Procrustes analysis. In the noise tests zero mean additive Gaussian noise was applied with standard deviation $\sigma = \mathtt{n} \times \mathtt{s}/100$ where n is the noise percentage and s is defined as $\max(\mathtt{W})$ in pixels.

Initialisation: Each of the algorithms we tested requires a slightly different initialisation for the optimisation routine. This is dictated by the fact that each method starts the iterations from a different set of parameters. Therefore, evaluating each approach with exactly the same initialisation is not feasible. All the algorithms require an initial estimate of the camera matrices $\mathtt{R}_i$ and the mean shape. In order to make the initialisations as uniform as possible we have used the rigid factorization algorithm of (Marques & Costeira, 2009) to estimate them (except EMPPCA where we used the code provided by the authors). Here is a detailed description of the initialisation used for each algorithm.

- EMPPCA: requires initial estimates for the camera matrices $\mathtt{R}_i$, shape bases $\mathtt{B}_d$ and configuration weights l_{ij}. We used the initialisation provided by the authors in their implementation (Torresani *et al.*, 2008): (camera matrices and mean shape come from rigid factorisation (Tomasi & Kanade, 1992) while deformation basis and coefficients are estimated through iterative PCA of the shape residuals).
- BA: requires initial values for the same parameters as EMPPCA and was initialised in the same way, except (Marques & Costeira, 2009) was used as the rigid factorization algorithm.
- RCPF: needs an initialisation for the camera matrices $\mathtt{R}_i$ and shape bases $\mathtt{B}_d$. We used the initialisation proposed by the authors (Wang *et al.*, 2008): camera matrices and mean shape were estimated from rigid factorization (Marques & Costeira, 2009) and the shape bases $\mathtt{B}_d$ were initialised to small random values.
- MP and MP-SP: require initial values for the camera matrices $\mathtt{R}_i$, configuration weights l_{ij} and the missing data. Camera matrices and missing data were initialised from rigid factorization (Marques & Costeira, 2009) and the shape coefficients were were initialised through iterative PCA of the residuals of the measurement matrix $\mathtt{W}$ as explained in Section 3.1.1.

Note that only our algorithm, MP, uses the missing entries explicitly in the outer loop proposed in Algorithm 2, while EMPPCA, BA and RCPF only use the known data in the estimation.

Missing data and noise tests

In Figure 2 we compare the performance of our new algorithm MP with EMPPCA, RCPF, BA and MP-SP for the *Face1* sequence in the case of increasing levels of missing data ranging from 10% to 80%, generated by deleting entries from the measurement matrix randomly. For each level of missing data we averaged the results of 100 runs varying the missing data mask. Tests in which the 3D error was higher than 100% were considered as outliers and were not used to compute the average. In all experiments the number of basis shapes was fixed to $k = 5$.

The top row of Figure 2 shows the results in the noiseless case, while the bottom row shows the results in the more realistic case of 1% image noise. The plots in the left column show the 3D error of all the algorithms (MP, EMPPCA, RCPF, BA and MP-SP) while the plots on the right column show a zoomed-in version for the algorithms showing the best performance (MP, MP-SP and RCPF), which interestingly, enforce orthonormality constraints on the camera matrices through projection. The left plots in the noiseless (top) and 1% noise case (bottom) show that EMPPCA and BA are the worse performing algorithms in the presence of missing data. EMPPCA can cope with up to 20% missing data before the error starts to grow steadily. BA gives the highest 3D errors for low ratios of missing data but appears to show more resilience to higher ratios of missing data than EMPPCA. However, it also breaks down after 50% missing data.

It is important to record the number of reconstructions that ended up with a 3D error higher than 100% (those that we classified as outliers and did not enter the statistics). Our proposed methods MP and MP-SP did not have any outliers. In the noiseless experiments (Figure 2 (top)) the number of outliers for RCPF and EMPPCA were 60 and 1 respectively over the 800 trials (each method was run 100 times for 8 levels of missing data). In the experiments with 1% noise (Figure 2 (bottom)), RCPF had 59 outliers and EM-PPCA had 1. Most of the RCPF outliers were in the 80% case which is the highest level of occlusions in our tests.

The plots in the right column of Figure 2 show a zoomed-in view of the best performing algorithms. Our new MP algorithm achieves the smallest overall 3D errors both in the noiseless case (right-top) and more clearly in the 1% noise test (right-bottom). RCPF (Wang *et al.*, 2008) shows good performance until levels of around 50% missing data but the errors grow quickly after that. The second best performing algorithm is MP-SP which uses our outer loop to deal with missing data and RCPF internally to perform factorisation. Although its performance is comparable to MP, the 3D error curve for MP lies below – for instance in the 1% noise case (bottom-right)the 3D reconstructions obtained with MP are on average around 1% better than with MP-SP.

It is worth discussing three interesting facts revealed by the results of these tests for increasing levels of missing data. First, the top three performing algorithms (MP, MP-SP and RCPF) include a projection step of the camera matrices to deal with metric constraints. BA and EMPPCA, on the other hand, impose the orthonormality constraints through parameterisation (quaternions in the case of BA and exponential map in the case of EMPPCA). Secondly, while RCPF, MP-SP and MP show very similar performance for missing data ratios of up to 50%, for higher ratios MP-SP and MP greatly outperform RCPF. The only difference between MP-SP and RCPF is the way in which they deal with missing data: RCPF uses only the known 2D image tracks while MP-SP uses an outer loop to re-estimate the missing data at each step of the iteration. Note that they were both initialised in the same way as MP. Finally, the performance of MP is about 1% better than MP-SP. However, MP-SP runs around 25% faster (see Figure 6 for algorithm run-times). Therefore if run-time is an issue MP-SP could be used instead of MP without compromising performance too much but of course improved results would be achieved with MP.

In Figure 3 we show comparative noise tests for EMPPCA, BA, RCPF and MP in the case of full data (left) and 30% missing data (right). We show results for noise levels of up to 4% meaning that the value of the variance σ is up to 4% of the size of the object in the image. It is clear that BA, is the most vulnerable algorithm to noise in the image coordinates. Note also that EMPPCA, RCPF and MP perform very similarly with EMPPCA performing slightly better in the full data case and MP in the 30% missing data case. The results were averaged over 100 runs. None of these tests resulted in outliers.

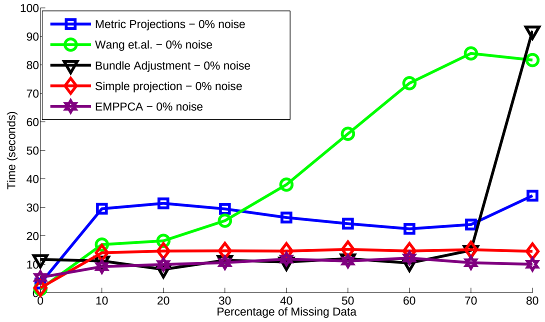


Fig. 6 Comparison of run-times (in seconds) averaged over 100 tests, versus percentage of missing data. Tests were performed using a 4-core Xeon processor running at 2.8GHz, with 24GB of RAM.

Figure 4 shows front and side views of the 3D reconstruction results for one of the runs of the *Face1* sequence with no noise and 40% missing data. The top row shows some frames of the motion capture session (which do not correspond to the reconstructed ones below), the second, third and fourth rows show ground truth values and 3D reconstruction results obtained with our method MP, EM-PPCA and RCPF respectively. Our reconstruction is closer to the ground truth shape. The average 3D reconstruction error over all the frames of this sequence was 4.7% with MP, 13.1% with EMPPCA and 9.0% with RCPF.

Figure 5 compares ground truth with the results obtained with MP, EMPPCA and RCPF for the *CMU* face sequence with full data and with 30% missing data. In the full data case all algorithms perform similarly. However, in the missing data case, our algorithm recovers the 3D shape correctly and outperforms Torresani *et al.*'s. The 3D errors against ground truth motion capture data were the same for RCPF and MP (2%), both for full data and 30% missing data, while for EMPPCA the 3D error is low (1.8%) in the full data case, but very high (35%) in the missing data case.

Figure 6 shows the mean run-times expressed in seconds, for the experiment in Figure 2, for EMPPCA, BA, RCPF and MP for different ratios of missing data. Tests were performed using a 4-core Xeon processor running at 2.8GHz, with 24GB of RAM. All implementations are in MATLAB. The fastest algorithms are BA and EMPPCA. However the code for BA and EMPPCA provided by the authors contains some parts of optimised MEX code. At the expense of losing some accuracy, as we saw in Figure 2, MP-SP runs around 30% faster than MP since the projection step is much more simple. Note that RCPF requires a large number of iterations in order to achieve convergence after 30% missing data. Therefore, adding the outer loop to RCPF to deal with missing data as we did in MP-SP improves the convergence in this case.

Synthetic Experiments – Structured occlusions

While it is important to conduct experiments with randomly generated missing data to control its percentage in the simulation, we also performed a test with a missing data mask where points are occluded in a structured way, as it would happen for instance due to self-occlusions.

In order to generate a more realistic missing data pattern we have computed surface normals from the sparse 3D motion capture data using the *taglut* algorithm[4].The computed angles between surface normal and camera viewing direction for all frames have been thresholded at 60 degrees, marking large angles as occluded. Although the knowledge of surface normals allows to simulate self-occlusions, the strong sparseness of the measured points does not permit

[4] `http://jmfavreau.info/?q=en/taglut`

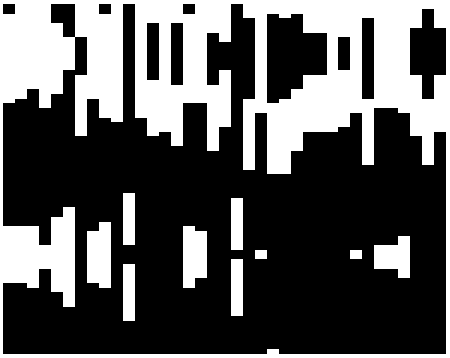


Fig. 7 Structured missing data mask used for the experiment described in Section 5.1. Each column is a point track, points in black are marked as visible, points in white are marked as occluded.

to simulate realistic self-occlusions. However, the resulting occlusion pattern is structured and not random as in the previous tests. The resulting occlusion mask is shown in Figure 7 – the amount of missing data resulting from this computation was 32%. The resulting visibility matrix captures well the structured disappearance of image features. We then ran our MP Algorithm 2 on the input 2D data, obtaining a 3D reconstruction error of 5.4%. A visual comparison of the reconstructed 3D against ground truth motion capture data is given in Figure 8. We also compare this result with other techniques, and show that MP outperforms other methods in this case. In particular, EMPPCA (Torresani *et al.*, 2008) obtains 8.6% 3D reconstruction error, and Wang *et al.*'s RCPF (Wang *et al.*, 2008) achieves 8.4% error.

Real Sequences

Cushion Sequence

In our first experiment we tested our algorithm on an image sequence of a cushion bending and stretching, in which 90 points were tracked manually. The results are shown in Figure 9. Our algorithm reconstructs successfully the 3D point cloud and its deformations. We used this data to generate a texture-mapped view of the reconstructed object. We also performed a quantitative evaluation by comparing the 3D reconstruction obtained with full data to those obtained with different percentages of missing data – generated by deleting randomly entries on the measurement matrix. The difference (computed in the same way as we compute the 3D error) between the 3D shape reconstructed with full data and the shapes obtained with 10%, 20% and 30% missing data are 3.8%, 5.7% and 5.9% respectively . We also measured the average image reprojection error which was 0.1 pixels with full data, and 1.1, 1.2 and 1.4 pixels for the 10%,20% and 30% missing data cases respectively.

In Figure 10 we show the 3D reconstruction results on the cushion sequence with 10% missing data generated randomly.

Franck Sequence

We also used the Franck sequence[5] taken from a video of a person engaged in conversation. We selected 700 frames from the 5000 frame sequence. An Active Appearance Model (AAM) was used to track 68 features on the face. Figure 11 shows three frames of the original images and a view of the resulting 3D reconstruction in the cases of complete 2D data (second row) and 20% missing data (third row). We also show the 3D reconstruction achieved with EMPPCA for the full data case as a baseline (fourth row). However, we could not show the results for EMPPCA for 20% missing data since already for that value, the errors were too high and the reconstruction was meaningless[6]. The last two rows (fifth and sixth) show the results achieved with the RCPF algorithm in the cases of full data and 20% missing data. The number of basis shapes was chosen to be 6 in this experiment. Our algorithm appears to achieve the best 3D reconstructions in this real sequence with and without missing data.

5.2 Articulated Structure

Synthetic sequence

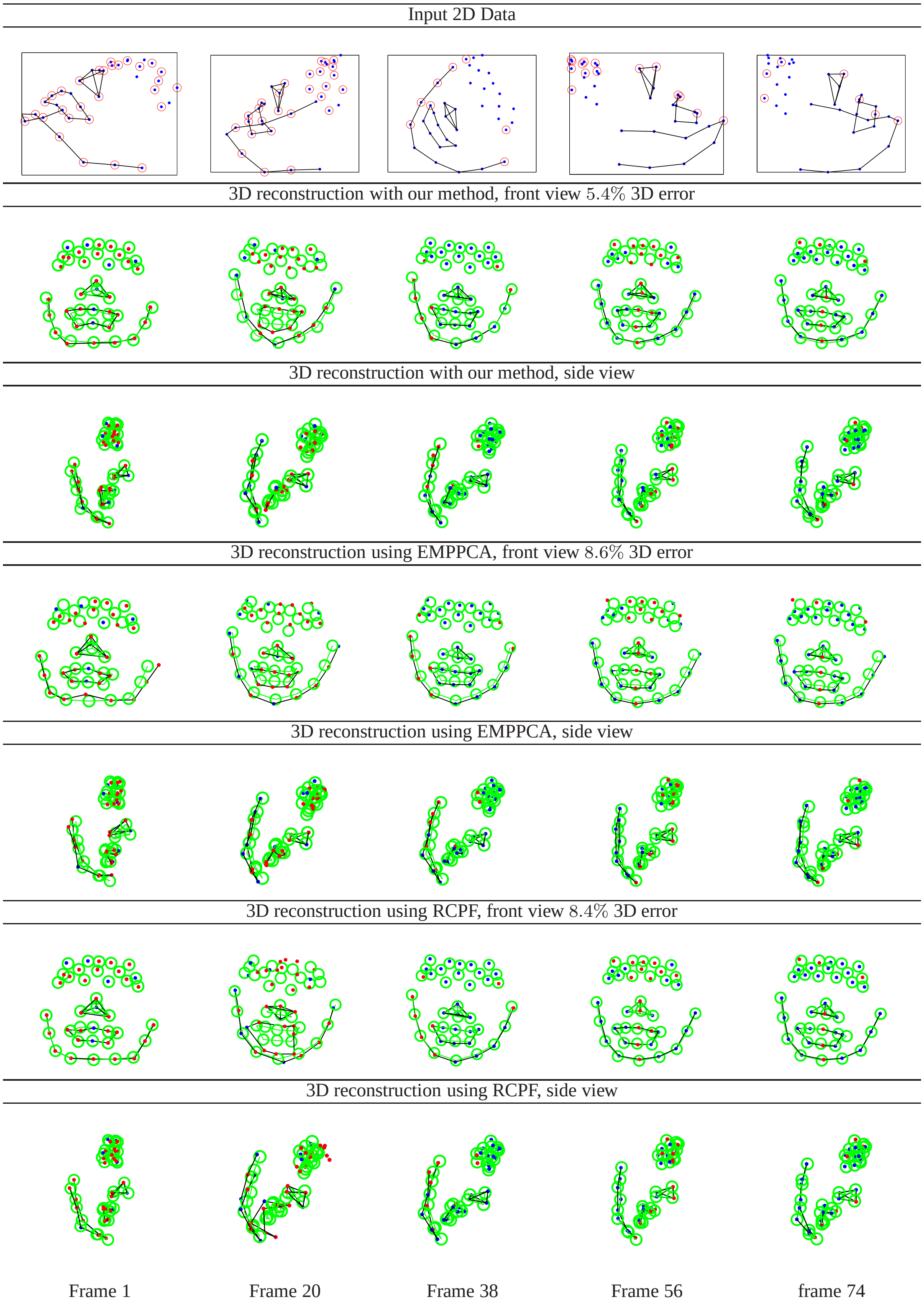
In the articulated case our synthetic data simulated two 3D boxes coupled by a hinge joint. The 3D ground truth is projected on the input images via orthographic projection. The sequence contained global rotation and translation as well opening and closing of the hinge. Each box contains 231 points, and the sequence is 63 frames long. We tested the algorithm in the case of full data for noise levels ranging from 0% to 4%. Figure 12 shows the absolute error in the recovered relative angle between the two boxes (averaged over all frames) and the 3D error of recovered 3D structure. The plots in Figure 12 show comparative results between the performance of (Tresadern & Reid, 2005) (TR) and our new approach (MP). Slightly superior results are achieved with our algorithm.

Real Sequence

We tested our algorithm on a sequence of 815 frames of two boxes linked by a hinge joint. The number of tracked points on the upper box was 21 and 47 on the lower box. Figure

[5] www-prima.inrialpes.fr/FGnet/data/01-TalkingFace/talking_face.html

[6] We have provided this result in our additional material



Input 2D Data

3D reconstruction with our method, front view 5.4% 3D error

3D reconstruction with our method, side view

3D reconstruction using EMPPCA, front view 8.6% 3D error

3D reconstruction using EMPPCA, side view

3D reconstruction using RCPF, front view 8.4% 3D error

3D reconstruction using RCPF, side view

Fig. 8 3D reconstruction results obtained for the *Face1* motion capture sequence with the structured missing data mask shown in Figure 7. Top row: 2D input data with missing data points highlighted with a red circle. Front and side views of the 3D reconstruction results (dots: blue if visible, red if not) are shown together with ground truth 3D data points (green circles) for three different algorithms: our MP algorithm (second and third rows), Torresani *et al.*'s EMPPCA (fourth and fifth rows), Wang *et al.*'s RCPF (sixth and seventh rows). The wire-frame lines are for visualisation purposes only.

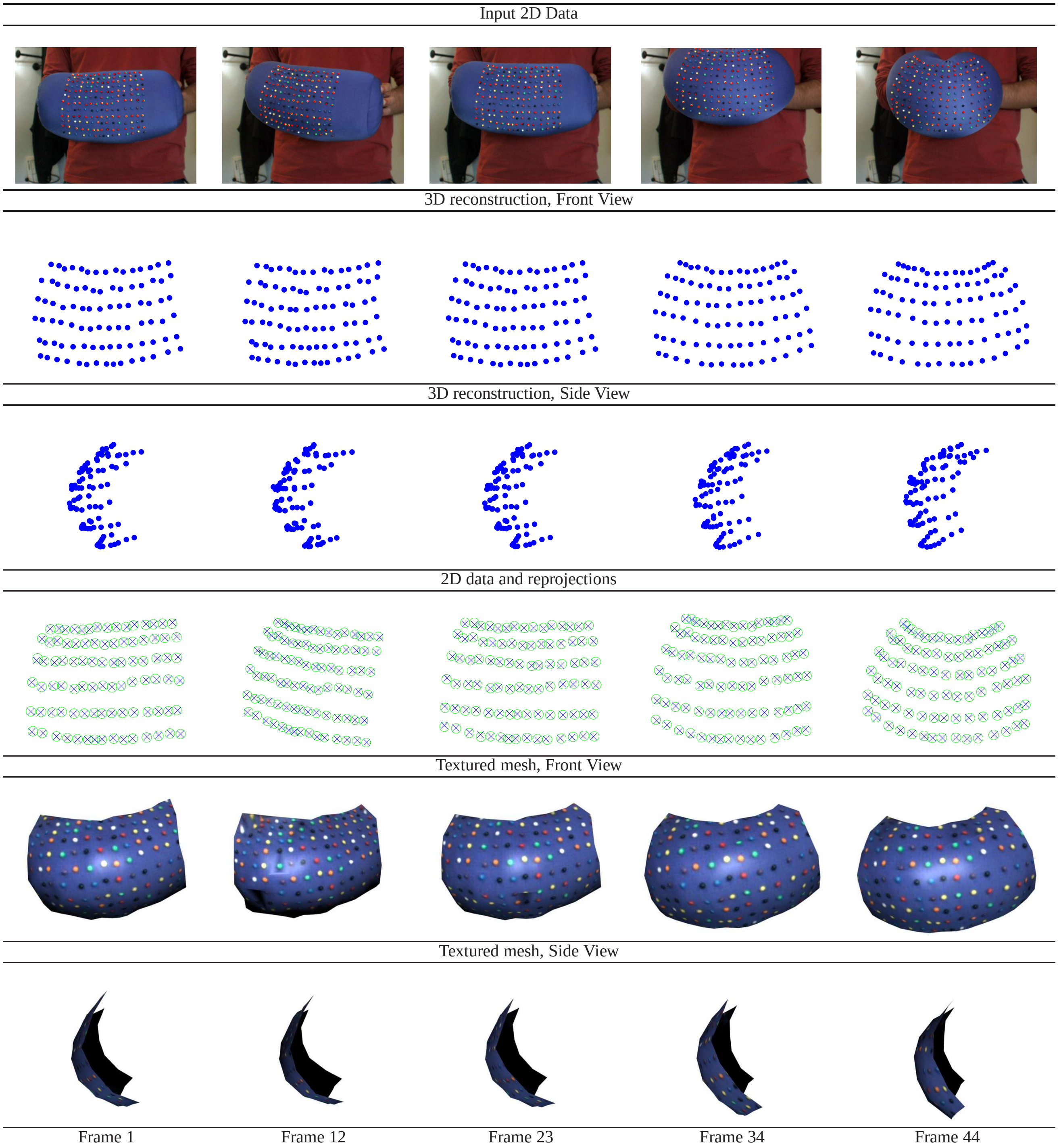

Input 2D Data
3D reconstruction, Front View
3D reconstruction, Side View
2D data and reprojections
Textured mesh, Front View
Textured mesh, Side View
Frame 1
Frame 12
Frame 23
Frame 34
Frame 44

Fig. 9 3D reconstruction results for the “cushion” real sequence. We show texture-mapped 3D reconstructions and use them to generate a virtual view of the object in 3D. First row: Input images and tracking data. Second and third rows: 3D reconstruction results with the proposed method. Fourth row: reprojection of reconstructed points (crosses) together with 2D input data (circles). Bottom rows: Texture-mapping rendered view of the 3D reconstruction.

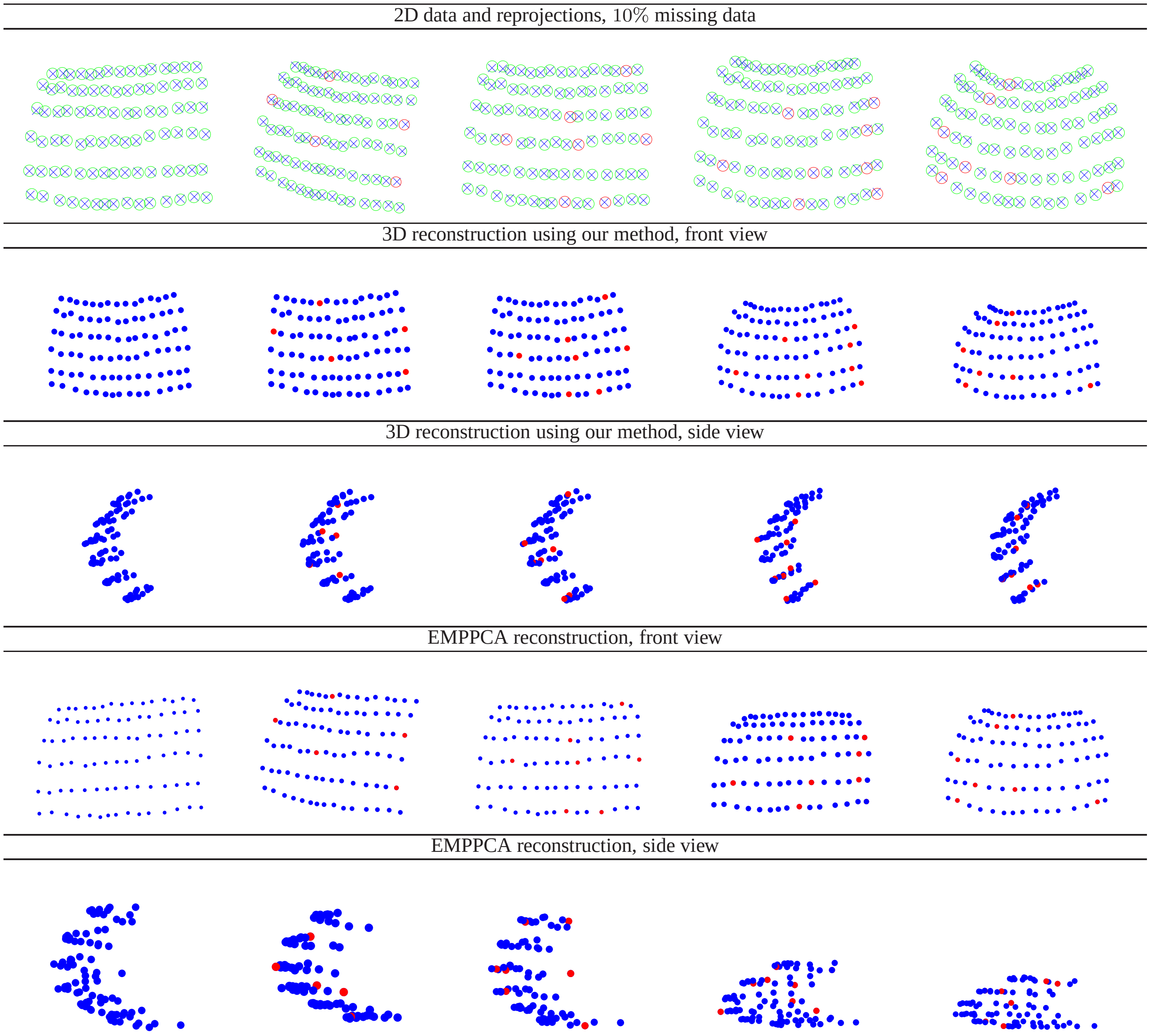


Fig. 10 Reconstruction results on the "cushion" real sequence with 10% missing data. Points were marked as not visible randomly. First row: Input 2D tracks (green circles) and reprojections calculated with our method (blue crosses). Missing 2D points (not used for reconstruction) are shown as red circles. Second and Third rows: 3D reconstruction with our method. Fourth and Fifth: 3D reconstruction using EMPPCA. note that although the frontal view matches the input data, the reconstruction suffers from bad depth estimation, visible in the side view.

13 shows two frames of the image sequence showing the tracked points and the recovered joint axis projected onto the images. The 3D reconstruction of the articulated structure together with the common hinge axis is also shown in Figure 13. In this case there was no missing data.

Finally we show results using a motion capture sequence of a person kicking a football. The motion capture system tracked 333 markers on the whole body. We selected the tracks on the leg, and projected the 3D coordinates on 2D images via orthographic projection. The viewing direction of the synthetic camera starts at the back of the leg and performs a random rotation around the body, resulting in the image sequence used for reconstruction. Some frames can be seen in Figure 15, first row. From the 2D images we can recover the rotation axis of the joint, and the 3D structure of the leg, as shown in Figure 15. The reconstructed 3D points and axis have been aligned to the MOCAP data to show the full body pose. Two closeup of the reconstruction and axis are shown. In Figure 14 we also show a comparison of the recovered rotation angle between our method and the linear method by Tresadern and Reid (Tresadern & Reid, 2005). We can see that although this sequence does not have ground truth information on the joint angle in the knee, we recover a smooth movement (purely from the data, without impos-

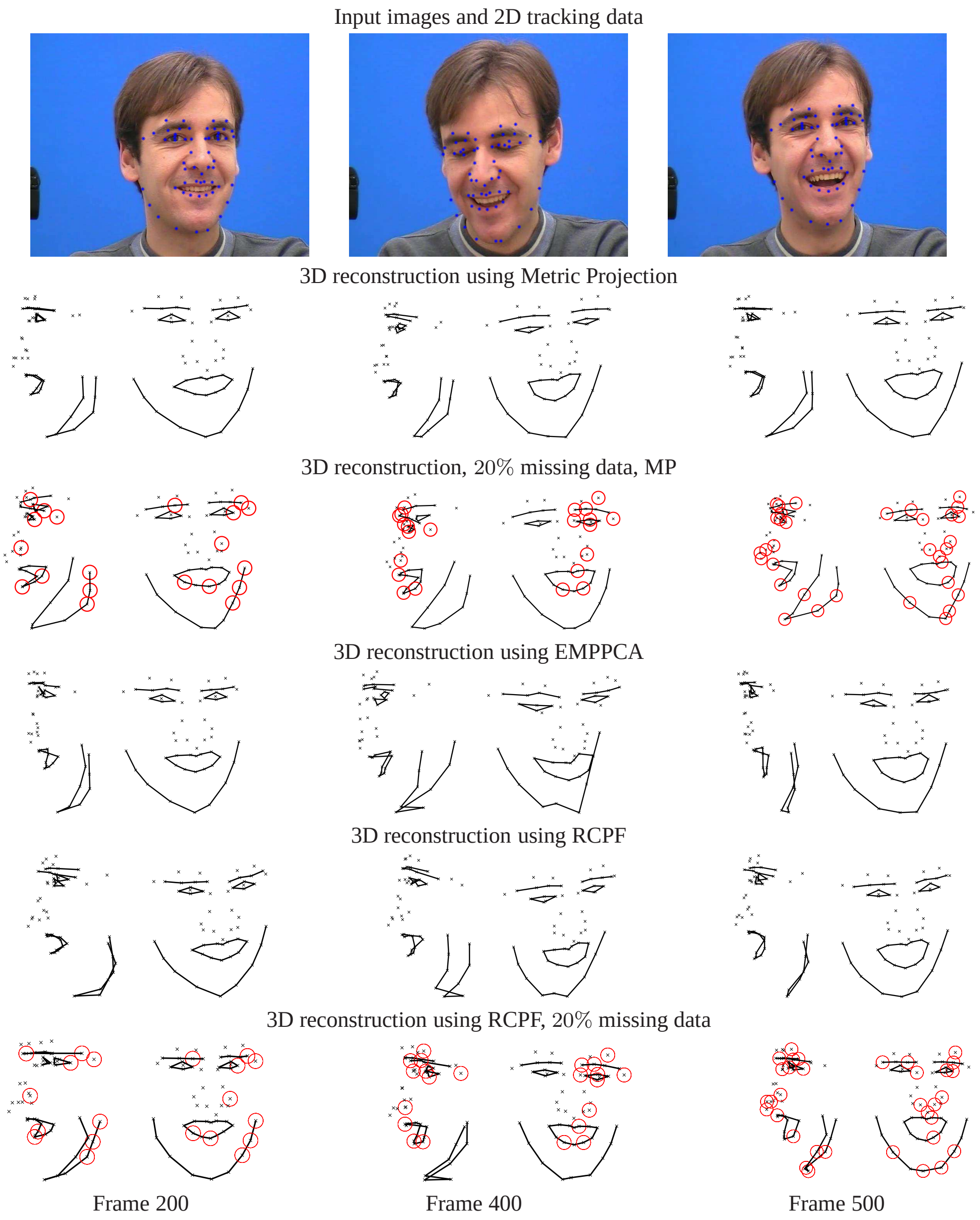

Input images and 2D tracking data
3D reconstruction using Metric Projection
3D reconstruction, 20% missing data, MP
3D reconstruction using EMPPCA
3D reconstruction using RCPF
3D reconstruction using RCPF, 20% missing data
Frame 200
Frame 400
Frame 500

Fig. 11 First row shows frames 200, 400 and 500 of the Franck sequence. We show front and side views of the 3D reconstructions in the case of full data and 20% missing data in the input tracks (randomly generated) achieved with our MP algorithm (second and third rows) EMPPCA (fourth row) and RCPF (fifth and sixth rows). Note that we do not show the reconstruction obtained for EMPPCA with missing data as it was of very poor quality. Missing points not visible in the corresponding frame are highlighted with a red circle.

ing smoothness constraints) while the linear solution obtains similar values with some discontinuities.

6 Conclusions

We have described a new bilinear alternating approach associated with a globally optimal projection step onto the manifold of metric constraints. At each step of the minimisation we project the motion matrices onto the correct deformable or articulated metric *motion manifolds* respectively. Although the constraints result in non-convex problems we introduced efficient convex relaxations in the form of semi-definite (SDP) or second-order cone (SOCP) programs. These relaxations revealed themselves to be exact in all our numerical experiments.

We have carried out experiments to compare the performance of our new Metric Projection algorithm with competing NRSfM methods. These have revealed that there are two main factors that make our Metric Projection (MP) algorithm more robust to missing data. The first strength is in the projector. It was first observed in (Marques & Costeira,

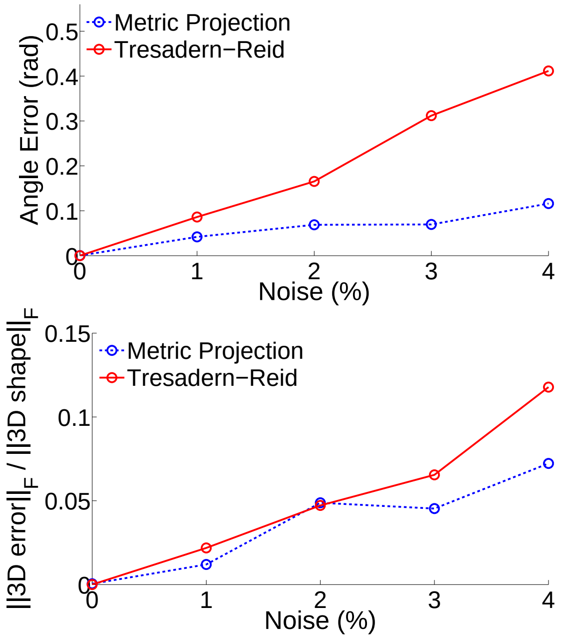


Fig. 12 Quantitative results on the synthetic articulated sequence. Top: Error on relative rotation angle between the two boxes in the synthetic experiment compared with Tresadern and Reid's linear approach. Bottom: 3D error of recovered structure. In both cases the Metric Projection method results more robust to noise and can recover rotation angles reliably.

2009), in the case of rigid SFM, that projecting the rotation matrices onto the Stiefel manifold allowed to cope with high percentages of missing data and degeneracies. Our experimental results show that, in the non-rigid case, the two algorithms that project the orthographic camera matrices onto the Stiefel manifold: our own MP and the simpler rotation constrained powerfactorization (RCPF) (Wang *et al.*, 2008) can cope with higher levels of missing data tracks than the two other baseline methods that do not (EMPPCA (Torresani *et al.*, 2001) and Bundle Adjustment (Del Bue *et al.*, 2007)). However, MP consistently outperforms RCPF (Wang *et al.*, 2008) for percentages of missing data above 50%.

This is due to the second strength of our MP algorithm: it simultaneously estimates the unknown entries of the measurement matrix $\mathtt{W}$, given the current estimates of the model parameters, within an iterative outer loop. Differently, RCPF, BA and EMPPCA estimate the model parameters using only the known data. This can have a very negative effect on the minimisation when few data are known. We also observed that, when included within our outer iterative loop to deal with missing data, the simple projector used by (Wang *et al.*, 2008) improved its performance significantly for percentages of missing data higher than 50%.

To conclude, imposing the metric constraints on the motion matrices provides reliable results without the need to impose additional smoothness priors on the camera pose or the deformations as most other NRSfM approaches to avoid ambiguous solutions. In the articulated case, we efficiently compute the joints given the non-linear constraints on the motion of the two bodies. In general, even though our methods were designed to solve SfM problems, the *motion manifolds* and the related optimal projections could be used for different tasks such as registration (where the shape $\mathtt{S}$ is known), image point matching and motion segmentation.

7 Acknowledgements

This research has received funding from the European Research Council under ERC Starting Grant agreement 204871-HUMANIS. The Portuguese research team was partially supported by FCT, under ISR/IST pluriannual funding (POSC program, FEDER), grant SIPM-PTDC/EEA-ACR/73749/2006, grant MODI-PTDC/EEA-ACR/72201/2006, the project ISFL-1-1431 and scholarship SFRH/BPD/26607/2006 . We gratefully acknowledge Manuel Marques and João Paulo Costeira for insightful discussions and for providing the code of their approach.

References

[Aanæs & Kahl, 2002]Aanæs, H., & Kahl, F. 2002. Estimation of Deformable Structure and Motion. *In: Workshop on Vision and Modelling of Dynamic Scenes, Copenhagen, Denmark*.

[Akhter *et al.*, 2008]Akhter, I., Sheikh, Y., Khan, S., & Kanade, T. 2008. Nonrigid Structure from Motion in Trajectory Space. *In: Neural Information Processing Systems*.

[Akhter *et al.*, 2009]Akhter, I., Sheikh, Y., & Khan, S. 2009. In Defense of Orthonormality Constraints for Nonrigid Structure from Motion. *In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, Miami, Florida*.

[Bartoli *et al.*, 2008]Bartoli, A., Gay-Bellile, V., Castellani, U., Peyras, J., Olsen, S., & Sayd, P. 2008. Coarse-to-Fine Low-Rank Structure-from-Motion. *In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, Alaska*.

[Brand, 2005]Brand, M. 2005. A Direct Method for 3D Factorization of Nonrigid Motion Observed in 2D. *In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Diego, California*.

[Bregler *et al.*, 2000]Bregler, C., Hertzmann, A., & Biermann, H. 2000. Recovering Non-Rigid 3D Shape from Image Streams. *In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, Hilton Head, South Carolina*.

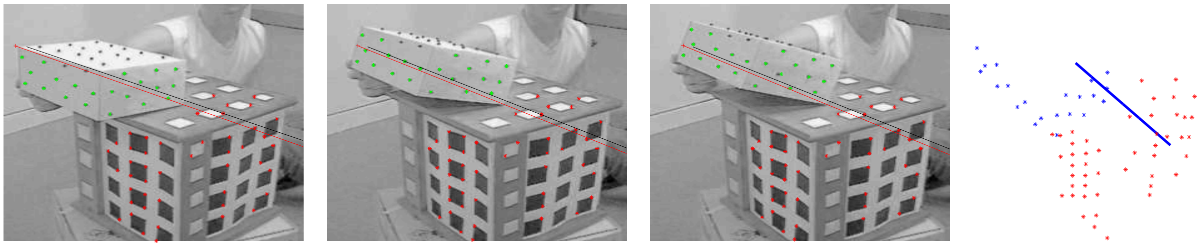


Fig. 13 Three images from the articulated sequence. The black line represents the hinge location computed with the linear algorithm by Tresadern and Reid, while the red line is the solution given by our method. The last figure shows the final 3D reconstruction and axis obtained using our approach.

[Buchanan & Fitzgibbon, 2005]Buchanan, A. M., & Fitzgibbon, A. 2005. Damped Newton algorithms for matrix factorization with missing data. *In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Diego, California*, vol. 2.

[Del Bue, 2008]Del Bue, A. 2008. A factorization approach to structure from motion with shape priors. *In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, Alaska*.

[Del Bue *et al.*, 2006]Del Bue, A., Lladó, X., & Agapito, L. 2006. Non-Rigid Metric Shape and Motion Recovery from Uncalibrated Images Using Priors. *In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, New York, NY*.

[Del Bue *et al.*, 2007]Del Bue, A., Smeraldi, F., & Agapito, L. 2007. Non-rigid structure from motion using ranklet–based tracking and non-linear optimization. *Image and Vision Computing*, **25**(3).

[Dodig *et al.*, 2009]Dodig, Marija, Stošić, Marko, & Xavier, João. 2009. *On minimizing a quadratic function on Stiefel manifolds*. Tech. rept. Instituto de Sistemas e Robotica. Available at http://users.isr.ist.utl.pt/˜jxavier/ctech.pdf.

[Edelman *et al.*, 1999]Edelman, Alan, Arias, T. A., & Smith, Steven T. 1999. The Geometry of Algorithms with Orthogonality Constraints. *SIAM Journal on Matrix Analysis and Applications*, **20**(2).

[Fayad *et al.*, 2009]Fayad, J., Del Bue, A., Agapito, L., & Aguiar, P. 2009. Non-rigid structure from motion using quadratic deformation models. *In: British Machine Vision Conference, London, UK*.

[Hartley & Vidal, 2008]Hartley, R., & Vidal, R. 2008. Perspective Nonrigid Shape and Motion Recovery. *In: Proc. European Conference on Computer Vision*.

[Marques & Costeira, 2008]Marques, Manuel, & Costeira, Joao. 2008. Optimal shape from motion estimation with missing and degenerate data. *Pages 1–6 of: WMVC '08: Proceedings of the 2008 IEEE Workshop on Motion and video Computing*. Washington, DC, USA: IEEE Computer Society.

[Marques & Costeira, 2009]Marques, Manuel, & Costeira, Joao. 2009. Estimating 3D shape from degenerate sequences with missing data. *Computer Vision and Image Understanding*, **113**(2).

[Paladini *et al.*, 2009]Paladini, M., Del Bue, A., Stošić, M., Dodig, M., Xavier, J., & Agapito, L. 2009. Factorization for non-rigid and articulated structure using metric projections. *In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, Miami, Florida*.

[Rabaud & Belongie, 2008]Rabaud, V., & Belongie, S. 2008. Re-thinking non-rigid structure from motion. *In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, Alaska*.

[Sturm, 1999]Sturm, J.F. 1999. Using SeDuMi 1.02, A Matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, **11**(1).

[Tomasi & Kanade, 1992]Tomasi, C., & Kanade, T. 1992. Shape and Motion from Image Streams under Orthography: A Factorization Approach. *International Journal of Computer Vision*, **9**(2).

[Torresani *et al.*, 2001]Torresani, L., Yang, D., Alexander, E., & Bregler, C. 2001. Tracking and Modeling Non-Rigid Objects with Rank Constraints. *In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii*.

[Torresani *et al.*, 2008]Torresani, L., Hertzmann, A., & Bregler., C. 2008. Non-Rigid Structure-From-Motion: Estimating Shape and Motion with Hierarchical Priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **30**(5).

[Tresadern & Reid, 2005]Tresadern, P., & Reid, I. 2005. Articulated Structure From Motion by Factorization. *In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Diego, California*, vol. 2.

[Wang & Wu, 2009]Wang, G., & Wu, Q.M. Jonathan. 2009. Quasi-perspective Projection Model: Theory and Application to Structure and Motion Factorization from Uncalibrated Image Sequences. *International Journal of Computer Vision*.

[Wang *et al.*, 2008]Wang, G., Tsui, H.T., & Wu, Q.M.J. 2008. Rotation constrained power factorization for structure from motion of nonrigid objects. *Pattern Recognition Letters*, **29**(1).

[Xiao & Kanade, 2005]Xiao, J., & Kanade, T. 2005. Uncalibrated Perspective Reconstruction of Deformable Structures. *In: Proc. 10th International Conference on Computer Vision, Beijing, China*.

[Xiao *et al.*, 2006]Xiao, J., Chai, J., & Kanade, T. 2006. A Closed-Form Solution to Non-Rigid Shape and Motion Recovery. *International Journal of Computer Vision*, **67**(2).

[Yan & Pollefeys, 2008]Yan, J., & Pollefeys, M. 2008. A Factorization-Based Approach for Articulated Non-rigid Shape, Motion and Kinematic Chain Recovery From Video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **30**(5).

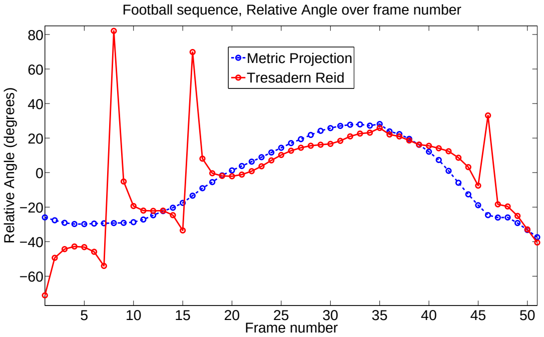


Fig. 14 Recovered rotation angle between two object: knee joint in the "football" sequence. Although this sequence does not have ground truth information on the joint angle in the knee, we recover a smooth movement (purely from the data, without imposing smoothness constraints) while the linear solution obtains similar values with some discontinuities

Appendix A: Convex relaxation – deformable case

For $\mathtt{E} \in \mathbb{R}^{6\times 6}$, our aim is to compute

$$\min_{\mathbf{q}=vec(\mathtt{Q})} \mathbf{q}^\top \mathtt{E}\mathbf{q}, \tag{25}$$

where $\mathtt{Q} \in \mathbb{R}^{3\times 2}$ runs through Stiefel matrices, i.e. $\mathtt{Q}^\top\mathtt{Q} = \mathtt{I}_{2\times 2}$. We rewrite (25) as

$$\min_{\mathbf{q}=vec(\mathtt{Q})} \mathrm{Tr}(\mathtt{E}\mathbf{q}\mathbf{q}^\top) = \min_{\mathtt{X}\in S} \mathrm{Tr}(\mathtt{EX}), \tag{26}$$

where S is the set of all real symmetric 6×6 matrices $\mathtt{X} = \begin{bmatrix} \mathtt{A} & \mathtt{B} \\ \mathtt{B}^\top & \mathtt{C} \end{bmatrix}$, with $\mathtt{A} \in \mathbb{R}^{3\times 3}$, satisfying

$$\mathtt{X} \succcurlyeq 0, \tag{27}$$

$$\mathrm{Tr}(\mathtt{A}) = \mathrm{Tr}(\mathtt{C}) = 1, \quad \mathrm{Tr}(\mathtt{B}) = 0, \tag{28}$$

$$\mathrm{rank}\,\mathtt{X} = 1. \tag{29}$$

This problem, has a non-convex constraint ($\mathrm{rank}\,\mathtt{X} = 1$). Since the cost function is linear we have

$$\min_{\mathtt{X}\in S} \mathrm{Tr}(\mathtt{EX}) = \min_{\mathtt{X}\in \mathrm{co}(S)} \mathrm{Tr}(\mathtt{EX}), \tag{30}$$

where $\mathrm{co}(S)$ is the convex hull of the set S. Here, we approximate the convex hull $\mathrm{co}(S)$ by the set of real symmetric 6×6 matrices $\mathtt{X}$ that satisfy

$$\mathtt{X} \succcurlyeq 0, \tag{31}$$

$$\mathrm{Tr}(\mathtt{A}) = \mathrm{Tr}(\mathtt{C}) = 1, \quad \mathrm{Tr}(\mathtt{B}) = 0, \tag{32}$$

$$\begin{bmatrix} \mathtt{I}_{3\times 3} - \mathtt{A} - \mathtt{C} & \mathbf{w} \\ \mathbf{w}^\top & 1 \end{bmatrix} \succcurlyeq 0, \tag{33}$$

with $\mathbf{w}$ given by

$$\mathbf{w} = \begin{bmatrix} b_{23} - b_{32} \\ b_{31} - b_{13} \\ b_{12} - b_{21} \end{bmatrix} \tag{34}$$

where $\mathtt{B} = [b_{ij}]$. Moreover, this set is defined only by linear matrix inequalities (LMI). Hence, we have that our problem (25) is relaxed into finding the minimum of a linear function ($\mathrm{Tr}(\mathtt{EX})$) on a convex set described by the LMIs (31)-(33). Thus, the optimisation problem in the right-hand side of (30) is a Semi-Definite Program (SDP). By using SeDuMi (Sturm, 1999), we quickly obtain the optimal matrix $\mathtt{X}$ for (30). In 100% of experiments that we ran, the optimal matrix $\mathtt{X}$ was always of rank 1. By factorising $\mathtt{X} = \mathbf{q}\mathbf{q}^\top$, we obtain the optimal *Stiefel matrix* as $\mathtt{Q} = vec^{-1}(\mathbf{q})$. For more details the reader can refer to (Dodig *et al.*, 2009).

Appendix B: Convex relaxation – Articulated Case

Problem statement

We consider the following optimisation problem which solves for the cost function as presented in eq. (24)

$$\begin{aligned} &\text{maximise } f(\mathbf{u}) \\ &\text{subject to } \|\mathbf{u}\| \le 1 \end{aligned} \tag{35}$$

where the variable to optimise is $\mathbf{u} \in \mathbb{R}^2$, the common joint axes for the two bodies. The objective function is

$$\begin{aligned} f(\mathbf{u}) = \|\mathbf{u}\|^2 + 2\mathbf{u}^\top\mathbf{x} + 2\left\|\left(\mathtt{I} - \mathbf{u}\mathbf{u}^\top\right)^{1/2}\mathtt{Y}\right\|_{\mathsf{N}} \\ + 2\left\|\left(\mathtt{I} - \mathbf{u}\mathbf{u}^\top\right)^{1/2}\mathtt{Z}\right\|_{\mathsf{N}} \end{aligned} \tag{36}$$

where the unknowns are the data triple

$$(\mathbf{x}, \mathtt{Y}, \mathtt{Z}) \in \mathbb{R}^2 \times \mathbb{R}^{2\times 2} \times \mathbb{R}^{2\times 2}.$$

Notice that for an $n \times n$ matrix $\mathtt{X}$, the symbol $\|\mathtt{X}\|_{\mathsf{N}} = \sigma_1(\mathtt{X}) + \cdots + \sigma_n(\mathtt{X})$ denotes its nuclear norm.

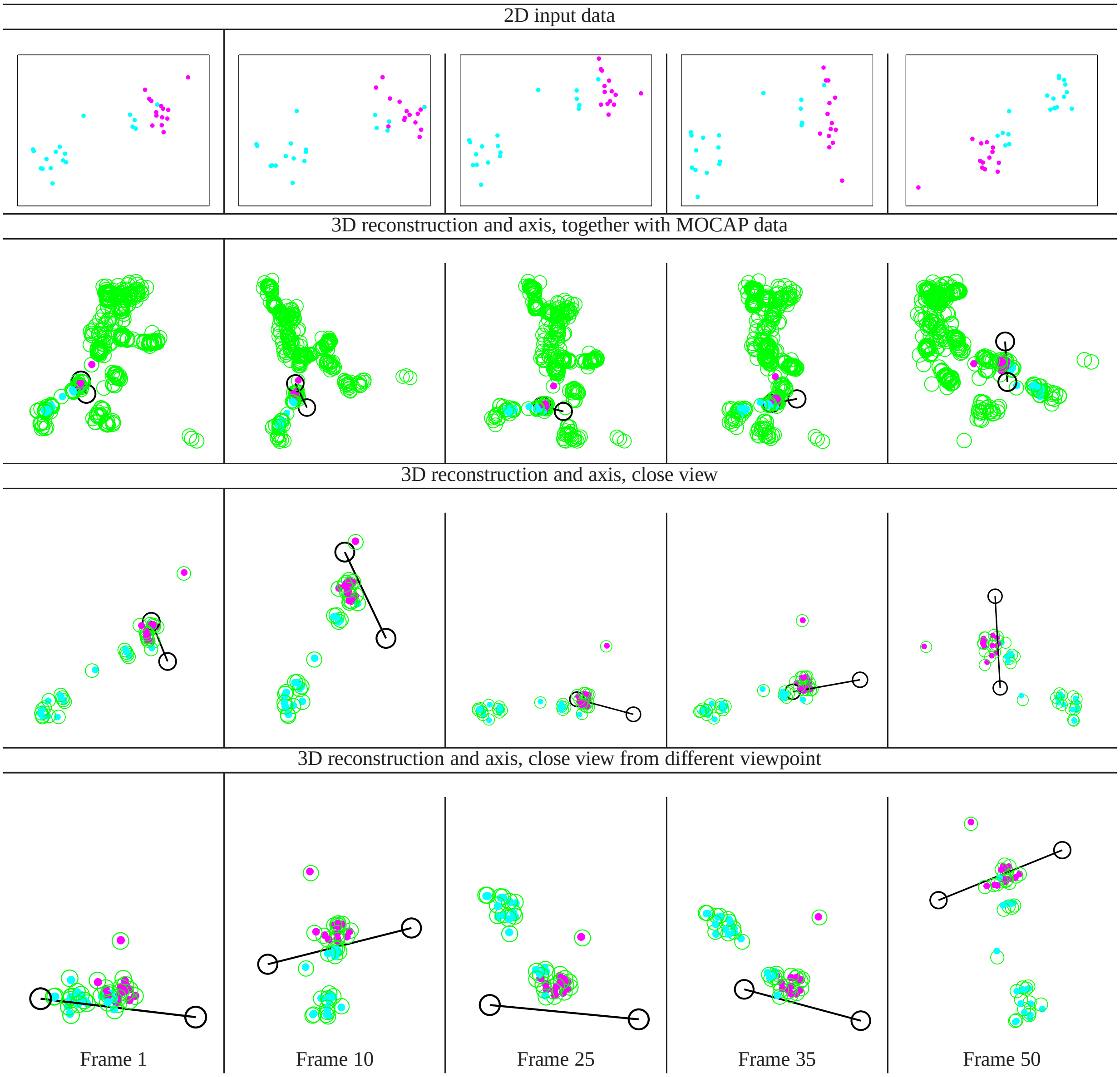


Fig. 15 Recovery of the knee joint in the “football” sequence. Top row: Input image points. Second row: 3D Reconstruction of the leg (magenta and cyan dots) and axis of rotation shown with the 3D ground truth motion capture sequence (green circles). Third row: Reconstructed 3D points (dots) with ground truth MOCAP data (green circles). Fourth row: 3D reconstruction imaged from a different angle.

Problem reformulation

We start by noting that (35) is equivalent to maximising

$$g(\mathbf{u}) = \|\mathbf{u}\|^2 + 2|\mathbf{u}^\top \mathbf{x}| + 2\left\|\left(\mathtt{I} - \mathbf{u}\mathbf{u}^\top\right)^{1/2}\mathtt{Y}\right\|_{\mathtt{N}} + \tag{37}$$

$$+2\left\|\left(\mathtt{I} - \mathbf{u}\mathbf{u}^\top\right)^{1/2}\mathtt{Z}\right\|_{\mathtt{N}}. \tag{38}$$

Note that $f(\mathbf{u}) \leq g(\mathbf{u})$ for all feasible $\mathbf{u}$. However, at a global maximiser of (35), say $\mathbf{u}^\star$, we must have $(\mathbf{u}^\star)^\top x \geq 0$. Thus, $(\mathbf{u}^\star)^\top x = |(\mathbf{u}^\star)^\top x|$ and $f(\mathbf{u}^\star) = g(\mathbf{u}^\star)$.

We rewrite $g(\mathbf{u})$ as

$$g(\mathbf{u}) = \|\mathbf{u}\|^2 + 2\sqrt{\mathbf{u}^\top\mathbf{x}\mathbf{x}^\top\mathbf{u}} + 2\left\|\left(\mathtt{I} - \mathbf{u}\mathbf{u}^\top\right)^{1/2}\mathtt{Y}\right\|_{\mathtt{N}} \tag{39}$$

$$+2\left\|\left(\mathtt{I} - \mathbf{u}\mathbf{u}^\top\right)^{1/2}\mathtt{Z}\right\|_{\mathtt{N}}. \tag{40}$$

Moreover, for a 2×2 matrix $\mathtt{X}$, there holds

$$\|\mathtt{X}\|_{\mathtt{N}} = \sqrt{\|\mathtt{X}\|^2 + 2|\det(\mathtt{X})|} \tag{41}$$

where $\|\mathtt{X}\| = \sqrt{\mathsf{tr}\left(\mathtt{XX}^\top\right)}$ denotes the Frobenius norm of $\mathtt{X}$. Using (41) in (40) gives

$$\begin{aligned} g(\mathbf{u}) = \|\mathbf{u}\|^2 + 2\sqrt{\mathbf{u}^\top\mathbf{x}\mathbf{x}^\top\mathbf{u}} \,+ \\ +2\sqrt{\|\mathtt{Y}\|^2 - \mathbf{u}^\top\mathtt{YY}^\top\mathbf{u} + 2|\det(\mathtt{Y})|\sqrt{1-\mathbf{u}^\top\mathbf{u}}} \,+ \\ +2\sqrt{\|\mathtt{Z}\|^2 - \mathbf{u}^\top\mathtt{ZZ}^\top\mathbf{u} + 2|\det(\mathtt{Z})|\sqrt{1-\mathbf{u}^\top\mathbf{u}}}. \end{aligned} \tag{42}$$

Now, we distinguish the following two cases which lead to two different optimisation strategies:

1. The matrices $\{\mathtt{I}_2, \mathtt{YY}^\top, \mathtt{ZZ}^\top\}$ are linearly independent
2. The matrices $\{\mathtt{I}_2, \mathtt{YY}^\top, \mathtt{ZZ}^\top\}$ are linearly dependent

Case 1 is the one that most frequently occurs in practice and it will be solved with a semi-definite program (SDP). In our experiments, we almost did not observe any occurrences of Case 2. In any case, we provide the solution to Case 2 by means of a 2^{nd} order cone program (SOCP).

Case 1: $\{\mathtt{I}_2, \mathtt{YY}^\top, \mathtt{ZZ}^\top\}$ *are linearly independent*

In this case, the matrices $\{\mathtt{I}_2, \mathtt{YY}^\top, \mathtt{ZZ}^\top\}$ form a basis for the three-dimensional vector space of 2×2 symmetric matrices. This means that there exists $\alpha, \beta, \gamma \in \mathbb{R}$ such that

$$\mathbf{x}\mathbf{x}^\top = \alpha\mathtt{I}_2 + \beta\mathtt{YY}^\top + \gamma\mathtt{ZZ}^\top. \tag{43}$$

Using (43) in (42) yields

$$\begin{aligned} g(\mathbf{u}) = \|\mathbf{u}\|^2 + 2\sqrt{\alpha\mathbf{u}^\top\mathbf{u} + \beta\mathbf{u}^\top\mathtt{YY}^\top\mathbf{u} + \gamma\mathbf{u}^\top\mathtt{ZZ}^\top\mathbf{u}} \,+ \\ +2\sqrt{\|\mathtt{Y}\|^2 - \mathbf{u}^\top\mathtt{YY}^\top\mathbf{u} + 2|\det(\mathtt{Y})|\sqrt{1-\mathbf{u}^\top\mathbf{u}}} \,+ \\ +2\sqrt{\|\mathtt{Z}\|^2 - \mathbf{u}^\top\mathtt{ZZ}^\top\mathbf{u} + 2|\det(\mathtt{Z})|\sqrt{1-\mathbf{u}^\top\mathbf{u}}} \end{aligned} \tag{44}$$

Our optimisation problem is

$$\begin{array}{ll} \text{maximise} & g(\mathbf{u}) \\ \text{subject to} & \|\mathbf{u}\| \leq 1 \end{array} \tag{45}$$

with $g(\mathbf{u})$ as in (44). In (45), the variable to optimise is $\mathbf{u} \in \mathbb{R}^2$. Problem (45) can be rewritten as

$$\begin{array}{ll} \text{maximise} & \phi(a,b,c) \\ \text{subject to} & (a,b,c) \in \mathcal{S} \\ & a \leq 1 \end{array} \tag{46}$$

where

$$\mathcal{S} := \{(a,b,c) \,:\, \exists_{\mathbf{u}} \,:\, a = \mathbf{u}^\top\mathbf{u},\ b = \mathbf{u}^\top\mathtt{YY}^\top\mathbf{u},\ c = \mathbf{u}^\top\mathtt{ZZ}^\top\mathbf{u}\},$$

and

$$\begin{aligned} \phi(a,b,c) := a + 2\sqrt{\alpha a + \beta b + \gamma c} \,+ \\ +2\sqrt{\|\mathtt{Y}\|^2 - b + 2|\det(\mathtt{Y})|\sqrt{1-a}} \,+ \\ +2\sqrt{\|\mathtt{Z}\|^2 - c + 2|\det(\mathtt{Z})|\sqrt{1-a}} \end{aligned}$$

is a concave function.

It is also given that we have the inclusion $\mathcal{S} \subset \mathcal{T}$ where

$$\begin{aligned} \mathcal{T} := \{(a,b,c) \,:\, \exists_{\mathtt{U}\succeq 0} \,:\, a = \mathsf{tr}(\mathtt{U}),\ b = \mathsf{tr}\left(\mathtt{YY}^\top\mathtt{U}\right), \\ c = \mathsf{tr}\left(\mathtt{ZZ}^\top\mathtt{U}\right)\} \end{aligned}$$

Using $\mathcal{T}$ instead of $\mathcal{S}$ in (46) gives the convex problem

$$\begin{array}{ll} \text{maximise} & \phi(a,b,c) \qquad . \\ \text{subject to} & a = \mathsf{tr}(\mathtt{U}) \\ & b = \mathsf{tr}\left(\mathtt{YY}^\top\mathtt{U}\right) \\ & c = \mathsf{tr}(\mathtt{ZZ}^\top\mathtt{U}) \\ & \mathtt{U} \succeq 0 \\ & a \leq 1 \end{array} \tag{47}$$

Let $\mathtt{U}^\star$ be a solution of (47) and let

$$\mathtt{U}^\star = \begin{bmatrix}\mathbf{u}_1 & \mathbf{u}_2\end{bmatrix} \begin{bmatrix}\lambda_1 & 0 \\ 0 & \lambda_2\end{bmatrix} \begin{bmatrix}\mathbf{u}_1^\top \\ \mathbf{u}_2^\top\end{bmatrix}$$

be an eigenvalue decomposition, where $\lambda_1 \geq \lambda_2$. A suboptimal solution for (35) is $\mathbf{u}^\star = \pm\sqrt{\lambda_1}\mathbf{u}_1$, where the sign is chosen such that $\mathbf{x}^\top\mathbf{u}^\star \geq 0$.

Case 2: $\{\mathtt{I}_2, \mathtt{YY}^\top, \mathtt{ZZ}^\top\}$ *are linearly dependent*

We assume that $\mathtt{ZZ}^\top$ can be written as a linear combination of $\mathtt{I}_2$ and $\mathtt{YY}^\top$, i.e.

$$\mathtt{ZZ}^\top = \alpha\mathtt{I}_2 + \beta\mathtt{YY}^\top,$$

for some $\alpha, \beta \in \mathbb{R}$. Our problem becomes

$$\begin{array}{ll} \text{maximise} & \phi(a,b,c) \\ \text{subject to} & (a,b,c) \in \mathcal{S} \\ & a \leq 1 \end{array} \tag{48}$$

where

$$\mathcal{S} := \left\{(a,b,c) \,:\, \exists_{\mathbf{u}} \,:\, a = \mathbf{u}^\top\mathbf{u}, b = \mathbf{u}^\top\mathtt{YY}^\top, c = \mathbf{u}^\top\mathbf{x}\mathbf{x}^\top\mathbf{u}\right\},$$

and

$$\begin{aligned} \phi(a,b,c) := a + 2\sqrt{c} + 2\sqrt{\|\mathtt{Y}\|^2 - b + 2|\det(\mathtt{Y})|\sqrt{1-a}} \,+ \\ +2\sqrt{\|\mathtt{Z}\|^2 - \alpha a - \beta b + 2|\det(\mathtt{Z})|\sqrt{1-a}} \end{aligned}$$

is a concave function. Similarly as the previous case, we have the following inclusion $\mathcal{S} \subset \mathcal{T}$ where

$$\begin{aligned} \mathcal{T} := \{(a,b,c) \,:\, \exists_{\mathtt{U}\succeq 0} \,:\, a = \mathsf{tr}(\mathtt{U}),\ b = \mathsf{tr}\left(\mathtt{YY}^\top\mathtt{U}\right), \\ c = \mathsf{tr}\left(\mathbf{x}\mathbf{x}^\top\mathtt{U}\right)\} \end{aligned}$$

Using $\mathcal{T}$ instead of $\mathcal{S}$ in (48) gives the convex problem

$$\begin{array}{ll} \text{maximise} & \phi(a,b,c) \qquad . \\ \text{subject to} & a = \mathsf{tr}(\mathtt{U}) \\ & b = \mathsf{tr}\left(\mathtt{YY}^\top\mathtt{U}\right) \\ & c = \mathsf{tr}(\mathbf{x}\mathbf{x}^\top\mathtt{U}) \\ & \mathtt{U} \succeq 0 \\ & a \leq 1 \end{array} \tag{49}$$

It can be shown that (49) can be rewritten as a SOCP. Let $\mathtt{U}^\star$ be a solution of (49). Let

$$\mathtt{U}^\star = \begin{bmatrix}\mathbf{u}_1 & \mathbf{u}_2\end{bmatrix} \begin{bmatrix}\lambda_1 & 0 \\ 0 & \lambda_2\end{bmatrix} \begin{bmatrix}\mathbf{u}_1^\top \\ \mathbf{u}_2^\top\end{bmatrix}$$

be an eigenvalue decomposition, where $\lambda_1 \geq \lambda_2$. A suboptimal solution for (35) is $\mathbf{u}^\star = \pm\sqrt{\lambda_1}\mathbf{u}_1$, where the sign is chosen such that $\mathbf{x}^\top \mathbf{u}^\star \geq 0$.