Int J Comput Vis (2017) 122:371–387
DOI 10.1007/s11263-016-0972-8

Combining Local-Physical and Global-Statistical Models for Sequential Deformable Shape from Motion

Antonio Agudo[1] · **Francesc Moreno-Noguer**[1]

Received: 14 August 2015 / Accepted: 12 November 2016 / Published online: 5 December 2016

Abstract In this paper, we simultaneously estimate camera pose and non-rigid 3D shape from a monocular video, using a sequential solution that combines local and global representations. We model the object as an ensemble of particles, each ruled by the linear equation of the Newton's second law of motion. This dynamic model is incorporated into a bundle adjustment framework, in combination with simple regularization components that ensure temporal and spatial consistency. The resulting approach allows to sequentially estimate shape and camera poses, while progressively learning a global low-rank model of the shape that is fed back into the optimization scheme, introducing thus, global constraints. The overall combination of local (physical) and global (statistical) constraints yields a solution that is both efficient and robust to several artifacts such as noisy and missing data or sudden camera motions, without requiring any training data at all. Validation is done in a variety of real application domains, including articulated and non-rigid motion, both for continuous and discontinuous shapes. Our on-line methodology yields significantly more accurate reconstructions than competing sequential approaches, being even comparable to the more computationally demanding batch methods.

Keywords Sequential non-rigid structure from motion · Particle dynamics · Bundle adjustment · Low-rank models

Communicated by K. Ikeuchi.

✉ Antonio Agudo
aagudo@iri.upc.edu

Francesc Moreno-Noguer
fmoreno@iri.upc.edu

[1] Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Llorens Artigas 4-6, 08028 Barcelona, Spain

1 Introduction

Reconstructing deformable 3D shapes from single images or monocular videos is an active area of research in computer vision, with applications in very different domains. Medical imaging is maybe one of the most motivating examples where these techniques shall be deemed to be applied, for instance to obtain full 3D reconstructions of the organs in non-invasive surgery (Maier-Hein et al. 2014). For instance, a sequential estimation is mandatory in endoscopy to achieve an interaction between the estimated 3D model and the doctor in real time. Augmented reality and the entertainment industry are other fields that could greatly benefit from such techniques, by e.g. filming a person with standard cameras and capturing his/her motion or the deformation of the clothes (Koh et al. 2014). Unfortunately, recovering non-rigid shape from a single viewpoint is a severely under-constrained problem, in which many different 3D configurations can have very similar image projections. The problem becomes even more challenging if the camera is allowed to move, and on top of the ambiguities induced by the deformation itself, we also need to consider those introduced by the camera motion. This is the scenario addressed by Non-Rigid Structure *f*rom Motion (NRS*f*M) approaches, and which we contemplate in this paper. In short, the goal of the NRS*f*M is to simultaneously recover the camera motion and 3D shape of a deformable object from monocular images. Solving the inherent large number of parameters and ambiguous solutions of this problem requires introducing prior knowledge about the camera trajectory and scene deformation.

The most standard approach to solve these ambiguities is using statistical priors to approximate the global deformable structure as a linear combination of low-rank bases of shapes (Brand 2001; Bregler et al. 2000; Moreno-Noguer et al. 2011; Torresani et al. 2008), by means of a linear combination of

3D point trajectories (Akhter et al. 2008; Park et al. 2010; Valmadre and Lucey 2012), or even using a shape-trajectory combination (Gotardo and Martínez 2011b). This is typically used with additional smoothness constraints that further disambiguate the problem (Bartoli et al. 2008; Garg et al. 2013; Paladini et al. 2009). Yet, while low-rank methods can effectively encode global deformations, they cannot, in general, handle non-linear motion patterns and strong local deformations. Piecewise strategies (Chhatkuli et al. 2014; Russell et al. 2011; Taylor et al. 2010; Varol et al. 2009) allow recovering larger deformations, although their performance highly depends on having overlapping features in neighboring patches, or require large number of correspondences to enforce local rigidity constraints (Chhatkuli et al. 2014; Taylor et al. 2010; Varol et al. 2009), which can be hard to obtain in practice. In any event, these previous approaches batch process all frames of the sequence at once, after video capture, preventing them from being used on-line and in real-time applications, where NRS*f*M may have an enormous potential. This has been recently addressed in Agudo et al. (2014a, 2012), Paladini et al. (2010) and Tao et al. (2013), which, however, still focus on global models only valid for relatively small deformations (Paladini et al. 2010; Tao et al. 2013) or continuous warps (Agudo et al. 2014a, 2012).

An alternative to statistical and low-rank approaches is to directly model the physical laws that locally govern object kinematics. Drawing inspiration from computer graphics (Popovic and Witkin 1999), there have been several attempts at using these models for tracking non-rigid motion (Metaxas and Terzopoulos 1993) and modeling human activities (Brubaker et al. 2009). Unfortunately, these methods are usually focused to specific types of motion, and their underlying laws rely on non-linear relations complex to optimize. An interesting exception is Salzmann and Urtasun (2011), which directly uses the Newton's second law of motion to build a convex formulation for tracking purposes. This work, though, is not sequential, does not estimate the camera pose, as we do, and depends on priors computed from training data, specially when dealing with complex models such as human motion.

In this paper, we combine the best of global-statistical and local-physical approaches. In particular, we exploit Newton's second law of motion to introduce a force perturbed second-order Markov model that rules the local motion of every particle conforming the shape. The joint dynamics are then optimized using a bundle adjustment (BA) framework, with simple regularization terms that ensure temporal and global spatial consistency of the estimated shape and camera poses. This yields a sequential estimate of the shape and camera poses, while also allowing to progressively learn a low-rank global model of the shape, which is fed back into the optimization scheme. The overall approach is still sequential, fast, can cope with missing data and with different types of deformations such as articulated, isometric and stretchable, without requiring pre-trained data. We demonstrate its effectiveness on a variety of scenarios such as those depicted in Fig. 1, ranging from full body and face human motion capture to 3D reconstruction of organic tissues. In all cases, we show comparable results to competing batch algorithms, but

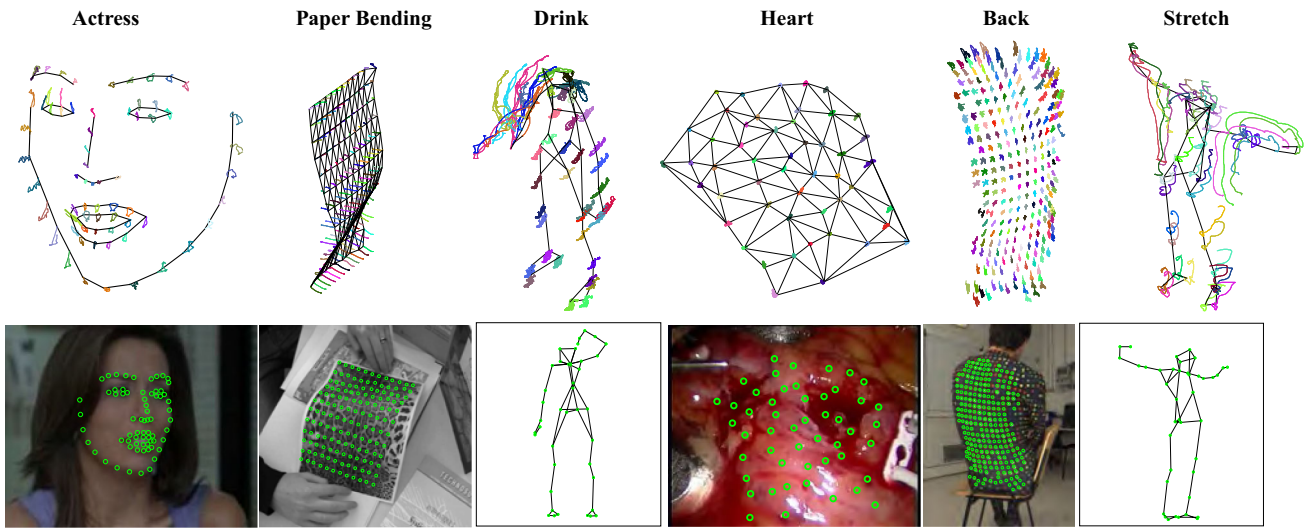


Fig. 1 3D Reconstruction of time-varying shapes using our physically-inspired velocity model for different types of deformations. We show that our approach is applicable in a wide range of domains, from non-rigid motion (for continuous shapes such as faces, back, paper and a beating heart) to articulated motion (drink and stretch). *Top* 3D reconstruction of one specific frame, and particle trajectories (each represented with a different *color*). *Bottom* A specific frame of the input sequence with 2D tracking data. The reader is referred to the experimental section for more details. The figure is best viewed in *color*

Table 1 Qualitative comparison of our approach with other state-of-the-art techniques

Char.	Method								
	EM-PPCA	MP	PTA	CSF2	KSTA	SPM	SBA	BA-FEM	GLSMM
Sequential	✗	✗	✗	✗	✗	✗	✓	✓	✓
Rank	✗	✗	✗	✗	✗	✗	✓	✗	✓
Missing data	✓	✓	✗	✓	✓	✗	✓	✓	✓
Articulated	✓	✓	✓	✓	✓	✓	✗	✗	✓
Learning	✓	✓	✗	✓	✓	✓	✓	✗	✓

We consider the methods: EM-PPCA (Torresani et al. 2008), MP (Paladini et al. 2009), PTA (Akhter et al. 2008), CSF2 (Gotardo and Martínez 2011b), KSTA (Gotardo and Martínez 2011a), SPM (Dai et al. 2012), SBA (Paladini et al. 2010), BA-FEM (Agudo et al. 2014a) and our approach denoted as GLSMM. The comparison is done in terms of whether: the solution is sequential or not (sequential), a specific rank of a deformation model is required to constrain the solution (rank), it can handle missing observations (missing data), it can cope with articulated motion and finally, whether the method can learn a deformation model on the fly (learning)

at a much smaller cost and a potential real-time applicability. Additionally, our approach yields remarkable improvement when compared to other sequential NRS*f*M techniques.

The part of this work regarding the use of local models based on particle dynamics was already presented in Agudo and Moreno-Noguer (2015). Here, we have combined these local constraints with global low-rank shape representations that are progressively and on-line learned. Additional theoretical discussions and mostly, synthetic and real results demonstrating the wide range of scenarios where our approach is applicable, are included in this version.

2 Related Work

NRS*f*M is an inherently ambiguous problem that to be solved requires a priori knowledge of either the nature of the deformations or the camera motion properties. Early NRS*f*M approaches extended the Tomasi and Kanade (1992) factorization algorithm to the non-rigid case by representing deformations as linear combinations of basis shapes under orthographic projection (Bregler et al. 2000; Xiao et al. 2006). On top of this, spatial (Torresani et al. 2008) and temporal (Bartoli et al. 2008; Del Bue et al. 2006; Torresani et al. 2008) smoothness priors have been considered to further limit the solution space. Later, Dai et al. (2012) relaxed the amount of extra prior knowledge by directly imposing a low-rank constraint on the factorization of the measurement matrix. Other approaches have modeled deformation using a low-rank trajectory basis per 3D point (Akhter et al. 2008), including priors on trajectories in terms of 3D point differentials (Valmadre and Lucey 2012) and enforcing smoothness on their paths (Gotardo and Martínez 2011b). One inherent limitation of these methods, is that they are highly sensitive to the number of bases chosen to represent the trajectory, making them very problem specific. Additionally, while being adequate to encode global deformations, low-rank methods' applicability is limited to smoothly deforming objects.

Recently, results from this field have significantly advanced. Stronger deformations have been tackled using piecewise models (Chhatkuli et al. 2014; Fayad et al. 2010; Russell et al. 2011; Taylor et al. 2010), even combining segmentation and reconstruction under local rigidity (Russell et al. 2014), or eliminating the rank dependency by means of Procustean normal distributions (Lee et al. 2013). In Garg et al. (2013), a variational approach integrating a low-rank shape model with spatial smoothness allowed per-pixel dense reconstructions.

In any event, all aforementioned NRS*f*M works are batch and they process all frames of the sequence at once, preventing thus, on-line and real-time computations. While sequential solutions exist for the rigid case (Newcome and Davison 2010; Lim et al. 2011), sequential estimation of deformable objects based only on the measurements up to that moment remains a challenging and unsolved problem. There are just a few attempts along this direction (Agudo et al. 2016, 2014a, 2012; Paladini et al. 2010; Tao et al. 2013). Specifically, Paladini et al. (2010) proposed a 3D-implicit low-rank model to encode the time-varying shape, estimating the remaining model parameters by BA over a temporal sliding window. Based on a similar framework, Tao et al. (2013) proposed an incremental principal component analysis to recursively update the low-rank model. Linear elasticity by means of finite element models was introduced into an extended Kalman filter to encode extensible deformations in real time (Agudo et al. 2012), even computing the full camera trajectory (Agudo et al. 2016). Very recently, Agudo et al. (2014a, b) presented the first approach to reconstruct both sparse and dense 3D shapes in a sequential manner, relying on a linear subspace of basis shapes computed by modal analysis. However, despite being very promising, these meth-

ods are only valid to handle smoothly deforming objects, as is the case of Paladini et al. (2010) and Tao et al. (2013), and cannot be applied to articulated motion (Agudo et al. 2012, 2014b, 2016).

An alternative to these approaches is to consider the object as a system of individual particles and represent global deformation by locally modeling the underlying physical laws that govern each of the particles. This has been typically used in computer graphics for simulation purposes (Baraff 1989; Popovic and Witkin 1999), and further exported to computer vision applications, for non-rigid tracking of surfaces (Metaxas and Terzopoulos 1993) or articulated bodies (Brubaker et al. 2009; Salzmann and Urtasun 2011; Vondrak et al. 2008). Yet, none of these approaches tackles the problem of besides retrieving shape, estimating the camera pose parameters.

Contribution In this paper we propose a way to combine the best of local and global methods, yielding an approach that is able to overcome most of the limitations of previous methods. In particular, our technique: (1) is sequential and efficient, (2) is applicable to articulated bodies and non-rigid surfaces, (3) handles local non-linearities and deformations of different nature including isometric, extensible and breakable surfaces, (4) does not require training data, and (5) can cope discontinuous tracks and missing data. We are not aware of any previous NRS*f*M approach able to simultaneously tackle all these challenges. Table 1 provides a qualitative comparison of our approach with respect to the most relevant state-of-the-art approaches.

3 Overview of the Approach

In this paper, we present an approach that combines the strengths of local-physical models with those of global-statistical shape representations. It has the following main features:

- **Local Physical Model** We model the local behavior of the deformable shape by considering particle dynamics equations. This allows recovering accurate shape representations and dealing with local non-linearities and even discontinuous motions. Also remarkable, is the fact that our model is based on simple classical mechanics equations which do not require using training data.
- **Global Statistical Model** We progressively learn a global low-rank shape model that is used to obtain a coarse but fast approximation of the shape.
- **Integration of Local and Global Models** We integrate both global and local constraints in a coarse-to-fine sequential manner, estimating shape and camera pose upon the arrival of a new image.

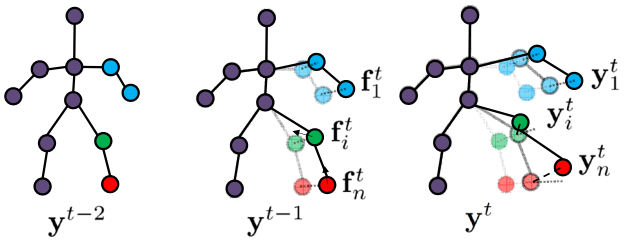


Fig. 2 Force-perturbed motion model for a system of particles. We use a kinematic model derived from Newton's second law of motion. A particle is moving with constant velocity while no forces are acting on it (*blue* particles). External forces $\mathbf{f}^t$ can change the dynamical behavior $\mathbf{d}^t$ of a single particle (*red* and *green* particles), and hence, change the configuration $\mathbf{y}^t$ of the deformable object (Color figure online)

In the next sections we describe each of these ingredients and how they are combined.

4 Local Physical Model

In this section, we first describe some basic concepts from dynamics in classical mechanics, which will then be used to introduce a local motion model for deformable objects approximated as a system of particles.

4.1 Classical Mechanics Motion Model

The local deformation model we propose holds on the Newton's second law of motion, which is satisfied by particles moving in an inertial frame of reference. We next review its general formulation.

We assume our object is represented as an assemblage of n discrete particles (as shown in Fig. 2). Let $\mathbf{y}_i^t \in \mathbb{R}^3$ be the 3D position of the ith particle at a time instant t and m_i its mass, assumed to be constant. When a force $\mathbf{f}_i^t$ is applied to this particle, Newton's second law of motion states that it produces a proportional acceleration $\mathbf{a}_i$:

$$\mathbf{f}_i^t = m_i \mathbf{a}_i^t = m_i \frac{d\mathbf{v}_i^t}{dt} , \tag{1}$$

where $\mathbf{v}_i^t$ is the instantaneous velocity of the particle, and $\mathbf{f}_i^t$ is the sum of all external forces applied to the particle.

In order to derive the formulation of our kinematic model we first approximate the acceleration at time t using backward second-order finite differences:

$$\mathbf{f}_i^t \approx m_i \left[\frac{\mathbf{y}_i^{t-2} - 2\mathbf{y}_i^{t-1} + \mathbf{y}_i^t}{(\Delta t)^2} \right], \tag{2}$$

that relates the current force $\mathbf{f}_i^t$ with the current 3D location $\mathbf{y}_i^t$ and the locations at previous time instances $\mathbf{y}_i^{t-1}$ and $\mathbf{y}_i^{t-2}$. We also considered a wider temporal window by

using a third-order finite difference model.[1] Nevertheless, for the experiments we report in the results section, we did not observe major differences in the reconstruction accuracy, and hence, we kept the second order model, as it is computationally less expensive.

We next extend the model to all the n particles of the deformable object.

Let $\mathbf{y}^t = [(\mathbf{y}_1^t)^\top, \ldots, (\mathbf{y}_n^t)^\top]^\top$ be a $3n$ dimensional vector composed of the 3D locations of all particles at time t; and $\mathbf{f}^t = [(\mathbf{f}_1^t)^\top, \ldots, (\mathbf{f}_n^t)^\top]^\top$ a $3n$ dimensional vector containing all instantaneous forces. We can then re-write Eq. (2) for all the particles using the following linear system:

$$\mathbf{f}^t = \begin{bmatrix}\mathbf{M} & -2\mathbf{M} & \mathbf{M}\end{bmatrix} \begin{bmatrix}\mathbf{y}^{t-2} \\ \mathbf{y}^{t-1} \\ \mathbf{y}^t\end{bmatrix}, \tag{3}$$

where $\mathbf{M}$ is a $3n \times 3n$ diagonal matrix with entries being the masses of each particle. In practice, we omit them and set $\mathbf{M} = \mathbf{I}$, the $3n \times 3n$ identity matrix. We also omit the term Δt in Eq. (2). By doing both these approximations, the forces we estimate will be up to scale, and will be expressed per unit of mass and increment of time, or equivalently, in length units.[2] This lets us to directly relate forces applied to the particles to their displacement. Note that this relation can be obtained without the need to compute any inverse matrix. This is in contrast to other physically-based methods where the inversion of a stiffness matrix can be a computationally expensive step. In our case the 3D position of the particles at time t can be written based on the following dynamical model:

$$\mathbf{y}^t = \mathbf{f}^t + 2\mathbf{y}^{t-1} - \mathbf{y}^{t-2} = \mathbf{f}^t + \mathbf{d}^t, \tag{4}$$

where $\mathbf{d}^t = 2\mathbf{y}^{t-1} - \mathbf{y}^{t-2}$ is a displacement vector. Observe that when $\mathbf{f}^t = \mathbf{0}$ this dynamical model boils down to a second-order Markov model in which each particle will move with a constant velocity $\mathbf{d}^t$ (see the blue particles in Fig. 2). However, when external forces are acting $\mathbf{f}^t \neq \mathbf{0}$, the particles can change their dynamics, accelerating or even reaching the rest. It is worth to point that a similar kinematic model was already used in Agudo et al. (2012), but in contrast to our paper, it was a first order Markov model and used to encode the camera motion, and not to encode the motion of each particle conforming the time-varying shape, as we do here.

[1] A third-order backward model to code the displacement vector can be expressed by considering 4-time instances as $\mathbf{f}_i^t \approx m_i \left[\frac{-\mathbf{y}_i^{t-3} + 4\mathbf{y}_i^{t-2} - 5\mathbf{y}_i^{t-1} + 2\mathbf{y}_i^t}{(\Delta t)^2}\right]$.

[2] $\frac{[\text{force}]}{[\text{mass}][\text{time}]^{-2}} = \frac{[\text{mass}][\text{length}][\text{time}]^{-2}}{[\text{mass}][\text{time}]^{-2}} = [\text{length}]$

4.2 Local Deformation Model for NRS*f*M

We next describe how to employ the proposed dynamic model to simultaneously, and in a sequential manner, estimate deformable shape and camera pose.

Let us consider a deformable object as an ensemble of n particles. At time t we represent the 3D position of all particles with the (previously defined) $3n$ dimensional vector $\mathbf{y}^t$. If we assume an orthographic camera model, the image projection of this object can be written as:

$$\mathbf{P}^t = [\mathbf{p}_1^t, \ldots, \mathbf{p}_n^t] = \mathbf{R}^t\mathbf{Y}^t + \mathbf{T}^t, \tag{5}$$

where $\mathbf{P}^t$ is the $2 \times n$ measurement matrix, $\mathbf{p}_i^t = [u_i^t, v_i^t]^\top$ are the image coordinates of the ith particle, $\mathbf{R}^t$ is a 2×3 truncated version of the rotation matrix, and $\mathbf{T}^t$ is a $2 \times n$ matrix that stacks n copies of the bidimensional translation vector $\mathbf{t}^t$. To represent the 3D shape $\mathbf{Y}^t$, we use a permutation operator $\mathcal{P}(\mathbf{y}^t)$ that rearranges the entries of $\mathbf{y}^t$ into a $3 \times n$ matrix such that the ith column of $\mathbf{Y}^t$ corresponds to the 3D coordinates of the point i.

Problem Formulation Given 2D point tracks up to frame t of a monocular video, our problem consists in sequentially and simultaneously estimating the camera motion $(\mathbf{R}^t, \mathbf{t}^t)$ and the deformable 3D shape $\mathbf{Y}^t$.

To solve this under-constrained problem, we initially represent the deformable object using Eq. (4), which after applying the operator $\mathcal{P}(\cdot)$, can be rewritten as:

$$\mathbf{Y}^t = \mathbf{F}^t + \mathbf{D}^t, \tag{6}$$

where $\mathbf{D}^t = 2\mathbf{Y}^{t-1} - \mathbf{Y}^{t-2}$ is the displacement vector, that at frame t is already known, as it only involves the particles position at previous time instances. Therefore, the current 3D shape estimation is reduced to estimating the forces $\mathbf{F}^t$.

In order to estimate $\mathbf{F}^t$ and the pose parameters $\mathbf{R}^t$ and $\mathbf{t}^t$, we perform a BA over a temporal sliding window on the last frames. This is indeed similar to what was done in other sequential NRS*f*M approaches (Agudo et al. 2014a; Paladini et al. 2010), with the key difference that at this point we do not rely on a low-rank model to parameterize the object deformation. The use of the Newton's second law of motion yields to our method higher generalization properties and major resilience to large non-linear deformations. Indeed, as we will discuss in the following section, we will eventually use a low-rank model to initialize the optimization, but after this initialization is done, the low-rank model does no longer constrain the shape.

Our BA optimization is performed over a temporal window on the last three frames, in which we jointly represent the projection equations as:

$$\begin{bmatrix} \mathbf{P}^{t-2} \\ \mathbf{P}^{t-1} \\ \mathbf{P}^{t} \end{bmatrix} = \begin{bmatrix} \mathbf{R}^{t-2} & & \\ & \mathbf{R}^{t-1} & \\ & & \mathbf{R}^{t} \end{bmatrix} \begin{bmatrix} \mathbf{Y}^{t-2} \\ \mathbf{Y}^{t-1} \\ \mathbf{F}^{t}+\mathbf{D}^{t} \end{bmatrix} + \begin{bmatrix} \mathbf{T}^{t-2} \\ \mathbf{T}^{t-1} \\ \mathbf{T}^{t} \end{bmatrix}. \quad (7)$$

Since the measurement matrix $\mathbf{P}^t$ may contain lost tracks due to occlusions or outliers, we define $\mathcal{V}^t$ as the set of visible points at time t. We then estimate the model parameters by minimizing the following energy function in terms of $\{\mathbf{R}^j, \mathbf{t}^j, \mathbf{F}^t\}$, with $j = \{t-2, t-1, t\}$[3]:

$$\mathcal{E} = \mathcal{E}_{img} + \alpha_p \mathcal{E}_{pose} + \alpha_s \mathcal{E}_{shape} + \alpha_e \mathcal{E}_{ext} \quad (8)$$

where:

$$\mathcal{E}_{img} = \sum_{j=t-2}^{t} \sum_{v \in \mathcal{V}^j} \| \mathbf{p}_v^j - \mathbf{R}^j(\mathbf{q}^j)\mathbf{y}_v^j - \mathbf{t}^j \|_{\mathcal{F}}^2 \quad (9)$$

minimizes the reprojection error of all observed points in $\mathcal{V}^j$. $\| \cdot \|_{\mathcal{F}}$ represents the Frobenius norm and $\mathbf{R}^j$ are the rotation matrices, which are parameterized using quaternions, $\mathbf{R}^j(\mathbf{q}^j)$, to guarantee orthonormality $\mathbf{R}^j {\mathbf{R}^j}^\top - \mathbf{I}_2 = \mathbf{0}$. A second energy term, $\mathcal{E}_{pose}$, serves for regularizing the estimated pose enforcing the rotation matrices and translation vectors of consecutive frames to be similar:

$$\mathcal{E}_{pose} = \sum_{j=t-1}^{t} \| \mathbf{q}^j - \mathbf{q}^{j-1} \|_{\mathcal{F}}^2 + \alpha_t \sum_{j=t-1}^{t} \| \mathbf{t}^j - \mathbf{t}^{j-1} \|_{\mathcal{F}}^2, \quad (10)$$

where α_t is the specific weight for the translation energy term. Similarly, we have introduced a regularization for the shape, to penalize strong variations in consecutive frames:

$$\mathcal{E}_{shape} = \| \mathbf{Y}^t(\mathbf{F}^t) - \mathbf{Y}^{t-1} \|_{\mathcal{F}}^2, \quad (11)$$

where the current shape $\mathbf{Y}^t$ is only function of the estimated force (see Eq. (6)). Finally, we have also considered spatial priors to control the extensibility of the surface. To this end, we regularize the change in the euclidean distance over n_e edges of the object using a Gaussian kernel, where d_e^r represents the initially estimated reference length for edge e, and d_e^t is the length at the current frame:

$$\mathcal{E}_{ext} = \sum_{e=1}^{n_e} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{{d_e^r}^2}{2\sigma^2} \right) |d_e^r - d_e^t(\mathbf{F}^t)|. \quad (12)$$

Note that this prior is not a hard constraint, and hence it still permits non-isometric deformations.

[3] Note that although $\mathbf{R}^j$ and $\mathbf{t}^j$ for $j = \{t-2, t-1\}$ are allowed to change while optimizing the pose and shape at frame t, their value is not propagated back in time. That is, our approach remains purely sequential.

The proposed approach relies on a few hyper-parameters: the regularization weights α_p, α_t, α_s and α_e in Eq. (8) and the standard deviation σ in Eq. (12). All these parameters are determined empirically using a validation sequence, and kept constant for the rest of all experiments. Specifically, we will set $\sigma = 0.1$, and the regularization weights will be adjusted to bring the error of each energy term in Eq. (8) to a similar order of magnitude.

4.3 Non-linear Optimization

We optimize the energy $\mathcal{E}(\mathbf{R}^j, \mathbf{t}^j, \mathbf{F}^t)$ in Eq. (8) using sparse Levenberg–Marquardt. Note, again, that in contrast to competing approaches (Bregler et al. 2000; Dai et al. 2012), we can deal with missing data and do not require all points to be tracked throughout the whole sequence.

Since we estimate a perturbation force per point, the complexity of our BA algorithm is dominated by the solution of the linearized system. This system is governed by a Jacobian matrix $\mathbf{J}_{\mathcal{E}}$ of size $N_c \times (3n + 6w)$, where w is the size of the temporal window, $w = 3$ in our case. N_c are the number of constraints introduced by the four terms of the energy function $\mathcal{E}$, including the total number of visible observations over the temporal window, and the number of constraints to enforce pose and shape smoothness. For instance, in Fig. 3(left) we depict the structure of the Jacobian corresponding to the *stretch* sequence, in which $n = 41$ and $N_c = 302$, yielding a matrix of size 302×141. Note that this matrix is very sparse, with only 4.53% of non-null elements.

Solving the BA problem requires computing the Hessian matrix, approximated by $\mathbf{H}_{\mathcal{E}} = \mathbf{J}_{\mathcal{E}}^\top \mathbf{J}_{\mathcal{E}}$, of size $(3n + 6w) \times (3n + 6w)$. This matrix multiplication can be done very efficiently,[4] by exploiting the high degree of sparsity of $\mathbf{J}_{\mathcal{E}}$. Indeed, the most computationally demanding step of the BA is in inverting $\mathbf{H}_{\mathcal{E}}$, which is an almost fully dense matrix, as seen in Fig. 3(right). Computing this inverse can be done in $\mathcal{O}((n + w)^3)$ time, which considering that $n >> w$ boils down to a $\mathcal{O}(n^3)$ cost. With these values, we may achieve real-time performance for models with less than $n = 100$ points. For instance, in the experiments we report in the next section, we achieve a frame rate of about 5 fps when dealing with a model of approximately 40 points. Since these results are obtained with non-optimized Matlab code, they can still be significantly speeded up.

[4] The computational complexity of the product $\mathbf{A}^\top \mathbf{A}$, where $\mathbf{A}$ is a sparse $m \times n$ matrix with n_{nz} non-zero elements is $\mathcal{O}(n_{nz} + m + n)$, that is, it depends linearly on n_{nz}, the row size m and column size n of the matrix, but is independent of the product mn. See: http://es.mathworks.com/help/matlab/math/sparse-matrix-operations.html#f6-13058.

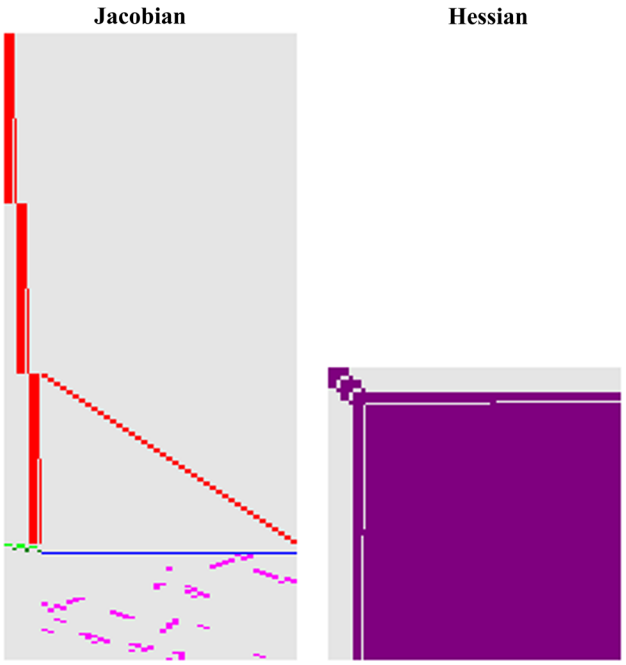


Fig. 3 Structure of Jacobian and Hessian matrices. We sketch the patterns of the Jacobian $\mathbf{J}_{\mathcal{E}}$ and Hessian $\mathbf{H}_{\mathcal{E}}$ matrices for the *stretch* sequence. In this case, the number of 3D points of the shape is $n = 41$ and no missing tracks are assumed. The number of links between particles is 51. This yields a total of $N_c = 302$ constraints for the Jacobian matrix, corresponding to the number of rows. They are split into the reprojection error term (shown in *red*), the pose (*green*) and shape (*blue*) smoothness priors and the extensibility (*magenta*) term. The number of columns of $\mathbf{J}_{\mathcal{E}}$ corresponds number of unknown shape and pose parameters, which is $3n + 18 = 141$ in this case. Note, that the Jacobian is very sparse, and only 4.53% of its elements are non-null. In contrast, the approximation $\mathbf{J}_{\mathcal{E}}^{\top}\mathbf{J}_{\mathcal{E}}$ to the Hessian matrix is almost fully dense (Color figure online)

4.4 Shape at Rest and Per Frame Initialization

Since the minimization of our energy function is highly non-convex, a very important element will refer to the initialization required at each frame. In Sect. 6 we will discuss how this initialization is performed using a coarse approximation of the shape provided by a global model that we iteratively learn and refine.

Additionally, we need to estimate the shape at rest and the initial pose values of the first frames. For this purpose, we follow Agudo et al. (2014a, b) and Paladini et al. (2010), and assume that the sequence contains a few initial frames where the object does not undergo large deformations. We use a standard practice done in NRS*f*M, that is running a rigid factorization algorithm (Marques and Costeira 2008) on these first frames to obtain a shape and pose estimate. Let us denote by $\mathbf{s}_0$ the shape at rest. Once this initialization is done, we then run our approach, which just for the first incoming image uses the assumption that $\mathbf{y}^{t-2} = \mathbf{y}^{t-1}$, i.e., it assumes each particle has null velocity.

5 Global Statistical Model

The main contribution of our work is that the optimization we just described is performed at a local level, for each particle, and we just consider constraints induced by their close neighborhood. This is in contrast to most existing NRS*f*M approaches, that typically apply constraints in a global manner, usually in the form of low-rank models. For batch methods (Dai et al. 2012; Del Bue et al. 2006; Garg et al. 2013; Torresani et al. 2008), these low-rank models are learned after processing all frames of the sequence at once. For sequential ones (Paladini et al. 2010; Tao et al. 2013), the low-rank model is incrementally learned.

In this paper we show that we can also use a low-rank model as a soft-constraint, but in contrast to other sequential approaches (Agudo et al. 2014a), we do not assume any initial generic mode, and learn them from scratch, and progressively learn them upon the arrival of new data.

The scheme for building and growing this low-rank model is very simple. Let us assume that at frame t, a shape basis $\mathbf{S} = [\mathbf{s}_1, \ldots, \mathbf{s}_r]$ of r modes is available. The modes $\mathbf{s}_i$ are $3n$-dimensional shape vectors. We then estimate the input shape $\mathbf{y}^t$ using the procedure described in the previous section. This shape can be approximated in terms of a low-rank model:

$$\hat{\mathbf{y}}^t = \mathbf{s}_0 + \mathbf{S}\boldsymbol{\psi}^t \ , \tag{13}$$

where $\mathbf{s}_0$ is the shape at rest and $\boldsymbol{\psi}^t$ is an r-dimensional vector with the weights of the linear combination. Denoting by $(\cdot)^{\dagger}$ the pseudoinverse operation, these weights are computed as:

$$\boldsymbol{\psi}^t = (\mathbf{S}^{\top}\mathbf{S})^{-1}\mathbf{S}(\mathbf{y}^t - \mathbf{s}_0) = \mathbf{S}^{\dagger}(\mathbf{y}^t - \mathbf{s}_0). \tag{14}$$

Given the current estimated shape $\mathbf{y}^t$, we then define its geometric error as the vector difference between $\mathbf{y}^t$ and $\hat{\mathbf{y}}^t$, its best approximation using the low-rank model:

$$\mathbf{g}^t = \mathbf{y}^t - \hat{\mathbf{y}}^t \ . \tag{15}$$

If the magnitude of this error is above a certain threshold ϵ, we then incorporate the geometric error into the basis. That is:

$$\textbf{if } \|\mathbf{g}^t\| > \epsilon \textbf{ then}$$
$$\mathbf{S} \leftarrow [\mathbf{S}, \mathbf{g}^t/\|\mathbf{g}^t\|]. \tag{16}$$

Note that we are just incorporating into the low-rank model the part of $\mathbf{y}^t$ (i.e., a 3D displacement) which cannot be modeled with the current basis. By doing this, we avoid augmenting the basis with redundant information.

Additionally, it is worth to point that since the estimation of $\mathbf{y}^t$ using local constraints is robust to missing observations, the estimation of the global basis, which takes as inputs the estimations of the local model, will also be robust to missing data.

6 Initializing Local Optimization with Global Constraints

The energy function we have defined in Eq. (8) involves seven different parameters within a temporal window of three frames: $\mathbf{R}^{t-2}, \mathbf{R}^{t-1}, \mathbf{R}^{t}, \mathbf{t}^{t-2}, \mathbf{t}^{t-1}, \mathbf{t}^{t}$ and $\mathbf{F}^{t}$. Upon the arrival of a new image, and its associated measurement matrix $\mathbf{P}^{t}$, these parameters need to be given an initial value. In particular $\mathbf{R}^{t-2}$, $\mathbf{R}^{t-1}$, $\mathbf{t}^{t-2}$ and $\mathbf{t}^{t-1}$ are initialized to the values we have estimated when evaluating frames $t-2$ and $t-1$. The translation vector $\mathbf{t}^{t}$ is simply initialized to the mean of the measurement matrix $\mathbf{P}^{t}$.

The initialization of $\mathbf{R}^{t}$ and $\mathbf{F}^{t}$ is a bit trickier, and is precisely at this point where we integrate global and local constraints (see Fig. 4). The idea is to first initialize the camera rotation assuming the deformation is spanned by the estimated linear subspace. This is done by iterating between the rotation matrix $\mathbf{R}^{t}$ and the weights of the subspace $\boldsymbol{\psi}^{t}$. Once these parameters are fixed, estimating $\mathbf{F}^{t}$ is straightforward. We next detail these steps.

The initialization of $\mathbf{R}^{t}$ is by itself an iterative process. We first start by estimating the rotation matrix that yields the best fit of $\mathbf{Y}^{t}$ onto the current observations $\mathbf{P}^{t}$, assuming just a rigid motion. That is, we initially seek to retrieve the value of $\mathbf{R}^{t}$ such that:

$$\arg\min_{\mathbf{R}^{t}} \sum_{\nu \in \mathcal{V}^{t}} \|\mathbf{p}_{\nu}^{t} - \mathbf{R}^{t}\mathbf{y}_{\nu}^{t} - \mathbf{t}^{t}\|_{\mathcal{F}}^{2} \tag{17}$$

where all parameters but $\mathbf{R}^{t}$ are known. Recall that $\mathbf{R}^{t}$ is a 2×3 truncated matrix, which can be computed from a full rotation matrix $\mathbf{Q}^{t} \in \mathrm{SO}(3)$ using $\mathbf{R}^{t} = \boldsymbol{\Pi}\mathbf{Q}^{t}$, and where $\boldsymbol{\Pi}$ is the orthographic camera matrix. In order to solve Eq. (17), while ensuring the resulting $\mathbf{Q}^{t}$ to lie on SO(3) group, we have followed a standard Newton algorithm for optimizing on manifolds (Ma et al. 1999; Shaji and Chandran 2008), which usually converges in one single iteration. We refer the reader to these for further details.

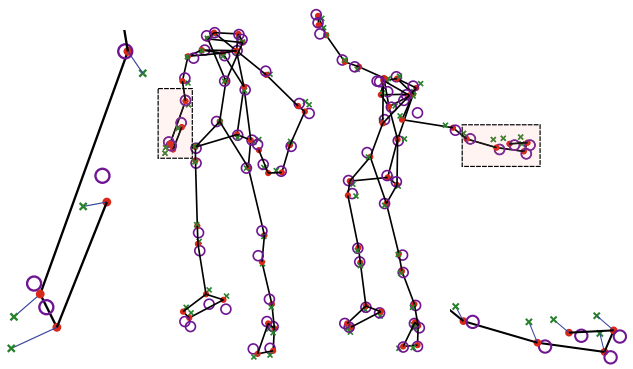


Fig. 4 Global-to-local optimization. 3D reconstruction of two frames for the *stretch* sequence using our global-to-local approach. The learned low-rank model is used to obtain a coarse solution of the deformation (*green* crosses), and then the local-physical model is applied to refine the solution (*red dots*). The zoomed views of specific joints are shown on the side of each figure. The ground truth is represented with *purple circles* (Color figure online)

Once we have an initial estimate for $\mathbf{R}^{t}$, we compute an initial solution for the shape, and constrain it to lie on the linear low-rank model we have learned. To do this, let $\mathbf{S}_{\nu}$ and $\mathbf{s}_{0,\nu}$ be the νth 3D point on all vectors of the subspace and on the shape at rest, respectively. We estimate the modal weights $\boldsymbol{\psi}^{t}$ using the following minimization over all set of visible point $\mathcal{V}^{t}$:

$$\arg\min_{\boldsymbol{\psi}^{t}} \sum_{\nu \in \mathcal{V}^{t}} \|\mathbf{p}_{\nu}^{t} - \mathbf{R}^{t}\left(\mathbf{s}_{0,\nu} + \mathbf{S}_{\nu}\boldsymbol{\psi}^{t}\right) - \mathbf{t}^{t}\|_{\mathcal{F}}^{2} . \tag{18}$$

This can be solved in closed-form using a Cholesky factorization:

$$\boldsymbol{\psi}^{t} = \left(\sum_{\nu \in \mathcal{V}^{t}} \mathbf{R}^{t}\mathbf{S}_{\nu}\right)^{-1} \left(\sum_{\nu \in \mathcal{V}^{t}} \left(\mathbf{p}_{\nu}^{t} - \mathbf{R}^{t}\mathbf{s}_{0,\nu} - \mathbf{t}^{t}\right)\right) \tag{19}$$

Equations (17) and (18) are alternated in order to compute an initial value for $\mathbf{R}^{t}$ consistent with the low-rank model. Finally, after convergence, we can initialize $\mathbf{f}^{t}$ (or equivalently the matrix $\mathbf{F}^{t}$) by applying the proposed physics-based model $\mathbf{f}^{t} = \mathbf{y}^{t} - \mathbf{d}^{t}$, where the shape is obtained from $\mathbf{y}^{t} = \mathbf{s}_{0} + \mathbf{S}\boldsymbol{\psi}^{t}$.

The outcome of this iterative procedure is an initialization of the pose and shape parameters, assuming a smooth camera motion and a global deformation model for the shape. This is then further refined based on the local deformation model defined in Eq. (8). Figure 5 shows the progressive reduction of the reprojection error after each of these stages.

7 Experimental Evaluation

In this section we present experimental results for different types of deformations, including articulated and non-rigid motion (some examples are shown in Fig. 1). A video summarizing all results can be found in.[5] We provide both qualitative results and quantitative evaluation, where we compare our method to several state-of-the-art approaches. In particular, we report the standard 3D reconstruction error, which is computed as:

$$\epsilon_{3\mathcal{D}} = \frac{1}{n_f}\sum_{t=1}^{n_f} \frac{\|\tilde{\mathbf{Y}}^{t} - \tilde{\mathbf{Y}}_{GT}^{t}\|_{\mathcal{F}}}{\|\tilde{\mathbf{Y}}_{GT}^{t}\|_{\mathcal{F}}}, \tag{20}$$

where $\tilde{\mathbf{Y}}^{t}$ is the estimated 3D reconstruction, $\tilde{\mathbf{Y}}_{GT}^{t}$ is the corresponding ground truth, and n_f is the total number of

[5] http://www.iri.upc.edu/people/aagudo.

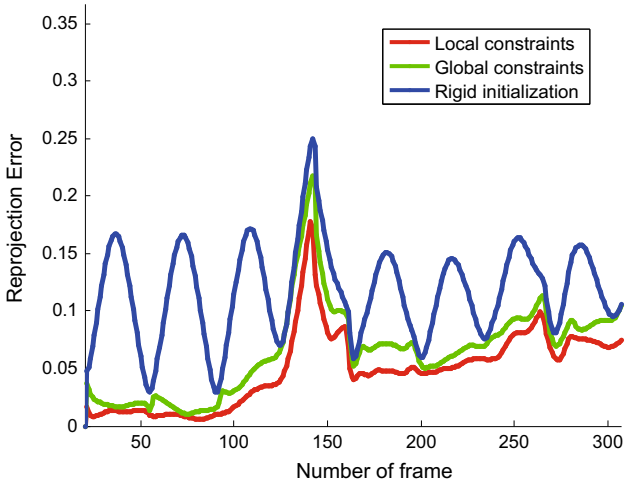


Fig. 5 Optimization steps. Reprojection error per frame of a mocap sequence. *Blue line* initial error after initializing the pose and shape at frame t with the results of frame $t-1$. *Green line* error obtained after optimizing shape and pose considering just global shape constraints. *Red line* error obtained after optimizing shape and pose considering local shape constraints. Each step is initialized with the output of the previous one (Color figure online)

non-rigid frames in the sequence (i.e., we do not consider the initial rigid frames used to estimate the shape at rest). The $\epsilon_{3\mathcal{D}}$ is computed after aligning the estimated 3D shape with the 3D ground truth using Procrustes analysis over all frames.

7.1 Motion Capture Data

We first evaluate our method on several existing datasets with 3D ground truth. We use the following motion capture sequences: *Drink*, *Stretch* and *Yoga* from Akhter et al. (2008), for evaluating articulated motion; the face deformation sequences *Jacky* and *Face*, from Torresani et al. (2008) and Paladini et al. (2009), respectively; and finally the synthetic bending *Shark* sequence from Torresani et al. (2008).

We compare our GLSMM approach (from Global-to-Local Sequential Motion Model) against GSMM and LSMM which correspond to our approaches but just considering either global or local constraints, respectively), and against eight state-of-the-art methods, both batch and sequential approaches. Among the batch algorithms we consider: EM-PPCA (Torresani et al. 2008), the Metric Projections (MP) (Paladini et al. 2009), the DCT-based 3D point trajectory (PTA) (Akhter et al. 2008), the Kernel Shape Trajectory Approach (KSTA) (Gotardo and Martínez 2011a), the Column Space Fitting (CSF2) (Gotardo and Martínez 2011b) and the block matrix method for SPM (Dai et al. 2012). We also consider the following sequential methods: Sequential BA (SBA) (Paladini et al. 2010), and the BA with Finite Elements formulation (BA-FEM) of Agudo et al. (2014a). The parameters of these methods were set in accordance with their original papers. We exactly use the same initialization for our proposed method, SBA (Paladini et al. 2010) and BA-FEM (Agudo et al. 2014a).

Table 2 summarizes the results. As expected, the version of our approach that considers both local and global constraints, outperforms by a large margin the version that only considers global constraints, and by a smaller margin the version with local constraints. Additionally, our GLSMM method sequentially learns an incremental low-rank model, which is not possible by the other two modalities. GLSMM also consistently outperforms the other sequential methods, specially SBA (Paladini et al. 2010) while being more generally applicable than BA-FEM (Agudo et al. 2014a), that cannot model articulated motion. Our results are indeed comparable to batch methods, where all frames need to be available in advance. Note that trajectory-based methods were proposed to exploit the time-varying evolution of a single point,

Table 2 Quantitative comparison on motion capture sequences

Seq.	Process Met.										
	Batch						Sequential				
	EM-PPCA	MP	PTA	CSF2	KSTA	SPM	SBA	BA-FEM	GSMM	LSMM	GLSMM
Drink	5.56 (5)	4.14 (6)	1.38 (13)	1.14 (6)	**0.94 (12)**	1.60 (12)	11.25 (12)	–	4.48	1.93	**1.92**
Jacky	**1.80 (5)**	2.74 (5)	2.69 (3)	1.93 (5)	2.12 (4)	1.82 (7)	2.90 (16)	3.43 (15)	2.84	2.80	**2.79**
Face	7.30 (9)	3.77 (7)	5.79 (2)	6.34 (5)	6.14 (8)	**2.67 (9)**	6.92 (27)	6.89 (2)	4.82	4.49	**4.33**
Stretch	13.72 (15)	8.13 (5)	3.85 (8)	2.46 (8)	2.00 (7)	**1.86 (11)**	17.61 (20)	–	6.52	5.76	**5.65**
Yoga	11.89 (14)	12.98 (8)	2.42 (8)	1.84 (7)	2.12 (7)	**1.65 (10)**	15.84 (20)	–	7.89	6.65	**6.65**
Shark	1.82 (2)	9.34 (23)	5.91 (6)	1.09 (5)	**1.03 (3)**	6.29 (2)	8.81 (5)	–	6.89	6.99	**6.73**

Bold values indicate the best solutions in every case

Reconstruction error $\epsilon_{3\mathcal{D}}$[%] for batch methods EM-PPCA (Torresani et al. 2008), MP (Paladini et al. 2009), PTA (Akhter et al. 2008), CSF2 (Gotardo and Martínez 2011b), KSTA (Gotardo and Martínez 2011a) and SPM (Dai et al. 2012); and for sequential methods SBA (Paladini et al. 2010), BA-FEM (Agudo et al. 2014a) and our approach denoted as GLSMM. We also report the results of GSMM and LSMM, our implementations when just considering global or local constraints, respectively. For low-rank based methods, we chose the basis rank (in brackets) that yielded the lowest $\epsilon_{3\mathcal{D}}$ error

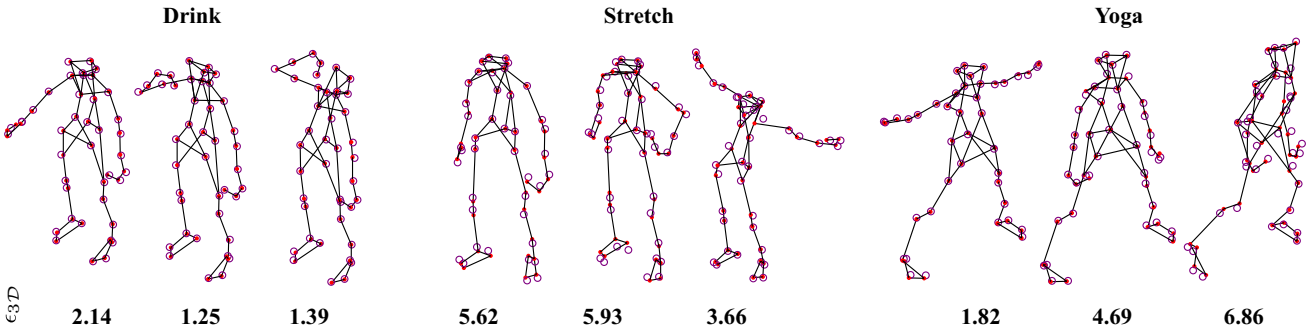


Fig. 6 Motion capture sequences. 3D reconstruction results for three sample frames in each of the human mocap sequences (*drink*, *stretch* and *yoga*). We also show the corresponding 3D reconstruction error. *Red dots* correspond to the shape estimated with our approach, and *purple circles* are the ground truth. Below each result we display the corresponding reconstruction error (Color figure online)

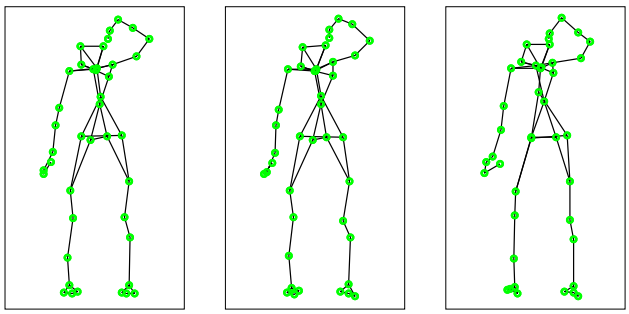
so a batch estimation with all frames available is required. Additionally, most of these methods are very sensitive to the choice of the specific rank of the deformation model. We do not require any of this fine tuning. Figure 6 shows the 3D reconstruction results on several frames of some of the mocap sequences.

7.1.1 Robustness to Noisy Observations

We also use the mocap sequences to evaluate the robustness of our approach against noise in the observations. For this purpose, we corrupt the observations using Gaussian noise with standard deviation $\sigma = \frac{\rho}{100}\gamma$, where ρ controls the amount of noise. The parameter γ represents the maximum distance of an image point to the centroid of all the points. The results are summarized in Table 3. Note that the noise also corrupts the initialization, thus changing the reference lengths d_e^r used in the extensibility energy term of Eq. (12). Nevertheless, the solution we estimate is very stable even for large levels of noise.

Table 3 Quantitative results against noisy observations

Seq. \ ρ	0	0.5	1	1.5	2
Drink	1.92	1.98	2.19	2.44	2.71
Jacky	2.79	2.85	3.14	3.58	4.08
Face	4.33	4.35	4.41	4.60	4.62
Stretch	5.65	5.68	5.72	5.82	5.85
Yoga	6.65	6.80	7.14	7.04	7.40
Shark	6.73	6.66	6.94	7.98	8.90



Top Reconstruction error $\epsilon_{3D}[\%]$ for mocap sequences under noisy measurements using the proposed approach. The level of noise is parameterized by ρ. *Bottom* To give meaning to the noise values, we represent the same input frame under different amounts of noise. Observe that for $\rho = 2$ there are remarkable differences w.r.t. the ground truth ($\rho = 0$), especially on the configuration of the legs, hips and right hand

7.1.2 Robustness to Initialization

Regarding the initialization, we have carried out two types of studies. On one side, we have evaluated the robustness of the approach to the initial value of the force parameter $\mathbf{F}^t$. By setting it to zero in all mocap sequences, we have observed that the reconstruction results remain virtually the same, compared to the initialization strategy describe in Sect. 6. There is, however, an increase in the number of iterations required to converge. In particular, the convergence time has grown according to the following values: *Drink* (25.73%); *Jacky* (6.36%); *Face* (2.45%); *Stretch* (36.45%); *Yoga* (58.79%); *Shark* (33.47%).

Additionally, we have also evaluated the stability of the approach to inaccuracies of the shape at rest computed using a rigid factorization (Sect. 4.4). In particular we have considered the mocap sequences with larger non-rigid components in the initial frames (i.e., stretch, yoga and drink) and have produced different initializations with increasing amount of frames. The more frames used for initialization, the more non-rigid component was included in these frames, and hence, the rigid factorization provided worse initial reconstructions. The amount of non-rigid component is quantified by $\|\mathbf{U}_{\text{non-rigid}}\|_{\mathcal{F}}/\|\mathbf{Y}_{\text{GT}}\|_{\mathcal{F}}$, where $\mathbf{U}_{\text{non-rigid}}$ represents the 3D non-rigid deviation of the shape fed to the non-rigid algorithm with respect to the initial shape $\mathbf{Y}_{\text{GT}}$. The results are reported in Table 4. As expected, the amount of non-rigid deformation grows with the number of frames used for the initialization. Nevertheless, this has almost no effect on the

Table 4 Quantitative evaluation of our approach with respect to the number of frames used to initialize, and the corresponding non-rigid motion in [%]

Stretch sequence								
Number of frames	5	10	15	20	25	30	35	40
Non-rigid motion[%]	2.29	4.15	6.75	9.22	9.95	9.98	9.99	9.99
$\epsilon_{3\mathcal{D}}$[%]	5.94	6.00	5.80	5.65	5.52	5.48	5.44	5.40
Yoga sequence								
Number of frames	5	10	15	20	25	30	35	40
Non-rigid motion[%]	0.43	1.05	1.73	2.46	3.22	4.00	4.77	5.51
$\epsilon_{3\mathcal{D}}$[%]	6.01	6.46	6.57	6.65	6.81	6.96	7.09	7.32
Drink sequence								
Number of frames	5	10	15	20	25	30	35	40
Non-rigid motion[%]	0.15	0.28	0.46	0.94	1.65	2.48	3.38	4.31
$\epsilon_{3\mathcal{D}}$[%]	1.90	1.91	1.92	1.92	1.94	1.94	1.93	1.93

Observe that our solution remains stable even when the initial frames contain non-rigid motion

Table 5 3D Reconstruction error $\epsilon_{3\mathcal{D}}$[%] for motion capture sequences when adding ±25% noise in optimization weights

Seq.	−25%	Original error	+25%
Drink	1.87	1.92	2.25
Jacky	2.79	2.79	2.82
Face	4.19	4.33	4.82
Stretch	6.24	5.65	6.46
Yoga	6.85	6.65	6.59
Shark	6.26	6.73	7.29

accuracy of our algorithm for the rest of the sequence, which remains very stable.

7.1.3 Parameter Tuning

The contribution of each energy term in the cost function of Eq. (8) can be controlled by means of the weights α_p, α_t, α_s. We tuned these parameters on the *stretch* sequence, and used the same values for the rest of sequences. Yet, these parameters do not need to be carefully tuned. In Table 5 we report the 3D reconstruction error over all mocap sequences after changing these weights by a ±25% their original value. Observe that the reconstruction results barely change.

7.2 Real Videos

We next present qualitative evaluation on seven different real sequences that demonstrate the appropriateness of our approach for shape recovery in varying situations, going from surfaces undergoing smooth continuous warps to abrupt deformations produced by a newspaper being torn apart. We also use these videos to provide a qualitative evaluation against missing observations (structured and random) and a quantitative comparison with respect to state-of-the-art techniques when 3D ground truth is available.

7.2.1 Actress Sequence

The *Actress* sequence is made of 102 frames showing a woman simultaneously talking and moving her head. We rely on the sequence tracks from Bartoli et al. (2008), and as is also done in sequential methods (Agudo et al. 2014a; Paladini et al. 2010), we use the first 30 frames to compute the initial model. Figure 7, shows the 3D reconstruction we obtain,

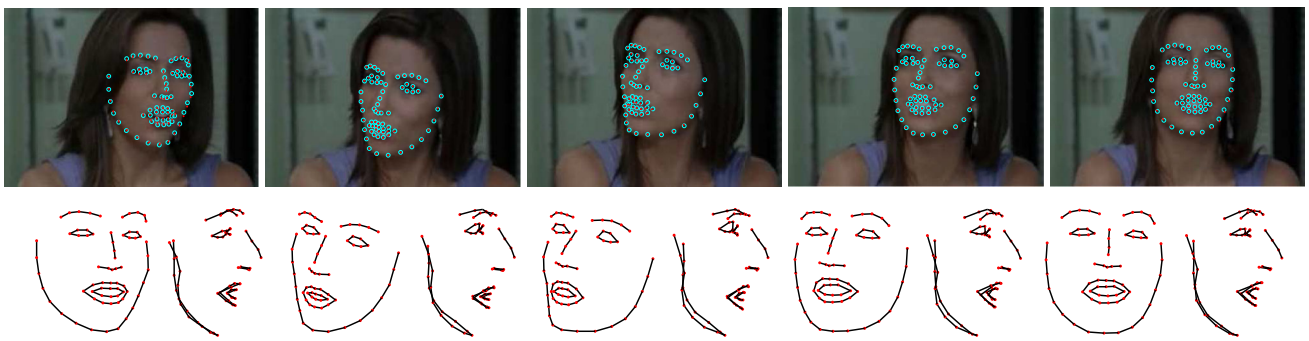


Fig. 7 Actress sequence. *Top* Frames #31, #48, #66, #84 and #91 with 2D tracking data and reprojected 3D shape represented by *cyan circles* and *red dots*, respectively. *Bottom* Original viewpoint and side views of our 3D reconstruction (Color figure online)

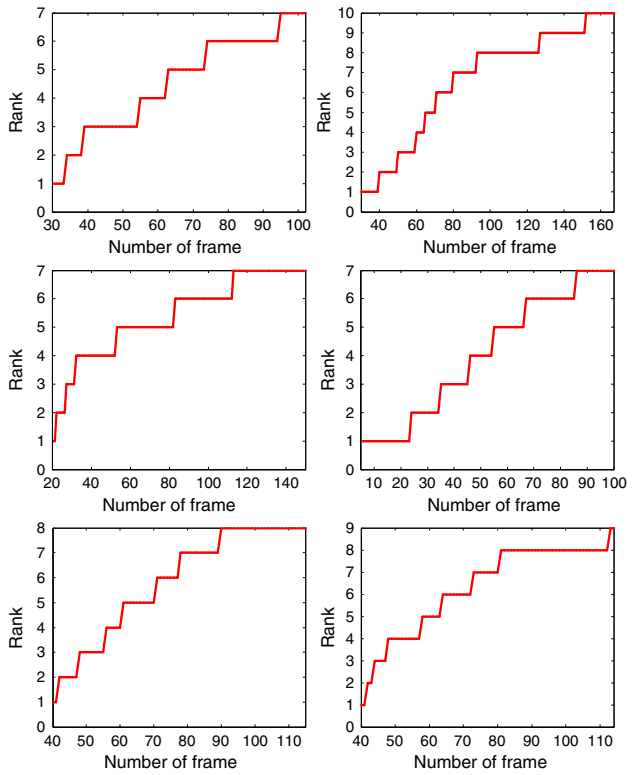


Fig. 8 Learning an incremental low-rank shape model. We represent the rank global model for each frame of the sequences. *Top* Actress and Tear sequence. *Middle* Back and Bending sequence. *Bottom* ASL1 and ASL2 sequence. It is worth point out that for the Tear experiment, the value the of rank strongly increases when the paper is split in two parts (Color figure online)

rotated according to the estimated rotation matrices, that is comparatively very similar to those obtained by Agudo et al. (2014a) and Paladini et al. (2010). However, we model the deformation at a local level, and this will allow us to code more non-linear deformations as we will show in subsequent examples. In Fig. 8 we plot the number of bases needed to represent the global model, reaching a maximum number of 7.

7.2.2 Tear Sequence

The *Tear* sequence (Taylor et al. 2010) contains 167 frames of a paper being split in two parts. We use the point tracks provided by (Russell et al. 2011). Again, the first 30 frames of the sequence are used to initialize the model. For this specific experiment we set the weight α_e of the extensibility term in Eq. (8) to zero, to allow the model to be split in two, without the need of exactly knowing the edges that suffer the cut.

Since the deformation on this video is very local, it was originally tackled using piecewise techniques (Taylor et al. 2010; Russell et al. 2011). Our particle-based approach also handles this type of deformation without much difficulty. Figure 9 shows a few 3D reconstructions obtained with our approach and with CSF2 (Gotardo and Martínez 2011b) using a low-rank basis of dimension 5. Although both solutions are similar, CSF2 renders the cut before the actual separation in two parts is produced. This is because this method processes all frames at once, which can produce certain de-synchronization between the actual 2D observations and the retrieved shape. Interestingly, note it Fig. 8 how the rank of the global model rapidly increases when the paper is split in two, between frames #40 and #90.

Table 6 Quantitative comparison on back sequence

Seq.	Process Met.					
	Batch					Sequential
	PQ	CSF2	KSTA	NOM	EM-PND	GLSMM
Back	15.20[a]	8.80 (2)	9.33 (2)	9.17[a]	8.10	**7.63**

Bold value indicates the best solutions in every case
Reconstruction error $\epsilon_{3\mathcal{D}}[\%]$ for batch methods PQ (Fayad et al. 2010), CSF2 (Gotardo and Martínez 2011b), KSTA (Gotardo and Martínez 2011a), NOM (Russell et al. 2011) and EM-PND (Lee et al. 2013)
[a] Numbers for PQ and NOM baselines are from (Russell et al. 2011). We denote our approach as GLSMM. For low-rank based methods, we chose the basis rank (in brackets) that yielded the lowest $\epsilon_{3\mathcal{D}}$ error

7.2.3 Back Sequence

The *Back* sequence consists of 150 frames showing the back of a person deforming sideways and flexing. We use the sparse point tracks of Russell et al. (2011) and the first 20 frames to compute the initial model. For this experiment, we also have the 3D ground truth obtained from stereo, and which we use for comparison.

In this case, we compare against the following batch methods: piecewise quadratic model (PQ) of Fayad et al. (2010), the network of overlapping models using also quadratic models (NOM) (Russell et al. 2011), the Procustean normal distribution model (EM-PND) (Lee et al. 2013), and the trajectory-shape-based methods CSF2 (Gotardo and Martínez 2011b) and KSTA (Gotardo and Martínez 2011a) which obtained very good performance in the mocap experiments of the previous section. A summary of the results is reported in Table 6. For this real experiment, we obtain a mean reconstruction error $\epsilon_{3\mathcal{D}}[\%]$ of 7.63, outperforming even batch state-of-the-art methods. In addition, recall that our solution is sequential, while all other approaches are batch. Figure 10 shows a few 3D reconstructed frames obtained with our approach and with CSF2 (Gotardo and Martínez 2011b). This is one of the batch methods with better performance in the mocap experiments of the previous section, specially under significant changes of the camera rotation, like those produced in this particular experiment. We observe, however, that this approach suffers from certain reconstruction errors, especially in regions reconstructed as convex while they should be concave (the natural shape of a back is dominated by a global concave configuration). In

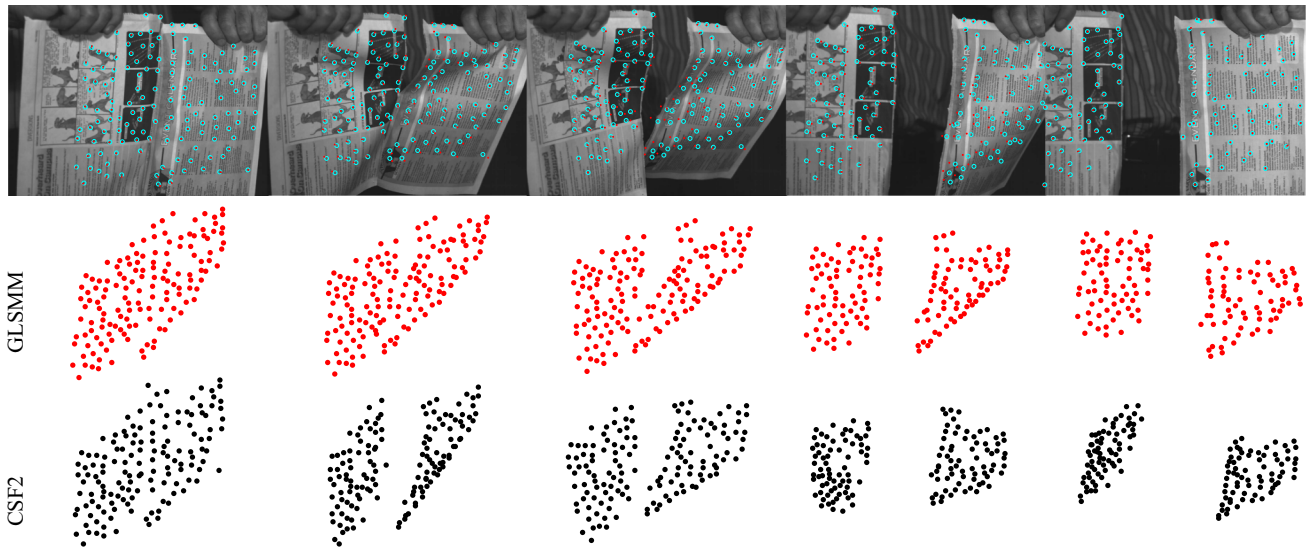


Fig. 9 Tear sequence. *Top* Frames #31, #52, #64, #82 and #123 with 2D tracking data and reprojected 3D shape represented by *cyan circles* and *red dots*, respectively. *Bottom* 3D views of the reconstructed shape using our approach and CSF2 (Gotardo and Martínez 2011b). Note that the batch method CSF2 assumes the paper starts splitting before it actually happens (Color figure online)

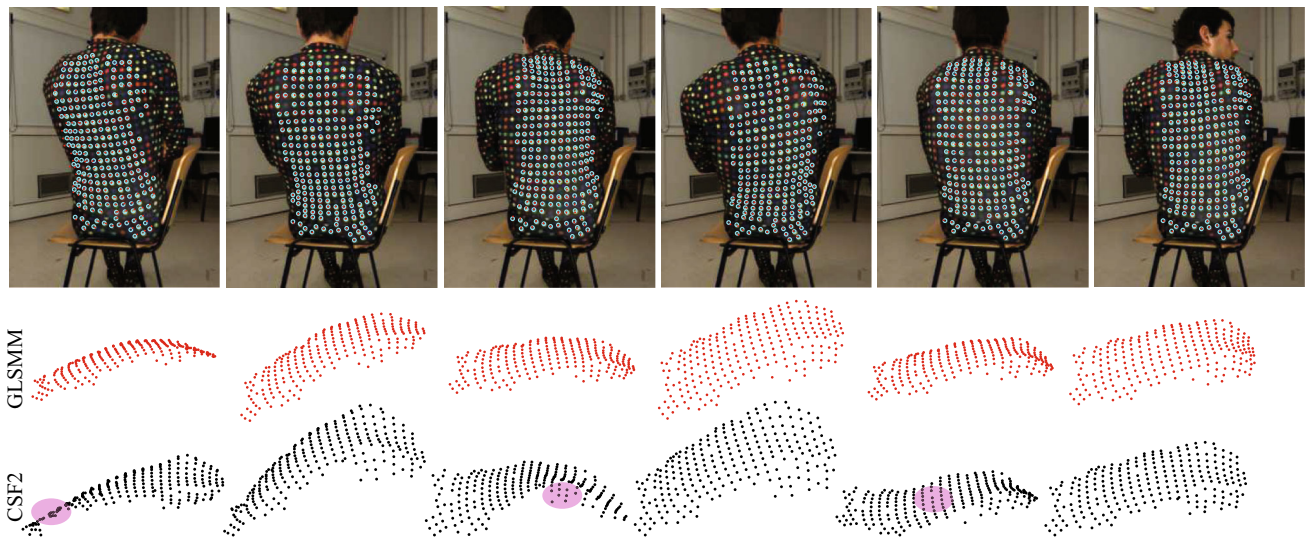


Fig. 10 Back sequence. *Top* Frames #30, #53, #82, #113, #137 and #148 with 2D tracking data and reprojected 3D shape with *cyan circles* and *red crosses*, respectively. *Bottom* 3D views of the reconstructed shape using our sequential method, CSF2 (Gotardo and Martínez 2011b), that batch processes all frames simultaneously. In *magenta* we highlight small artifacts of the reconstruction (Color figure online)

Fig. 10 we highlight in magenta these regions which do not seem very realistically plausible.

7.2.4 Paper Bending Sequence

We next present the results on a *Paper Bending* sequence of 100 frames already used in Bartoli et al. (2008). In this experiment we show a qualitative evaluation under the presence of randomly distributed missing data, which our BA-based approach can naturally handle. In particular, we add a pattern of 20% of missing data in the measurement matrix. In Fig. 11, we show our 3D reconstruction results with and without missing observations, being in both cases very similar and visually correct. For this experiment, we include the reconstruction result obtained with KSTA, the batch approach proposed by Gotardo and Martínez (2011a), using a basis of rank 2. Note however that the performance of this algorithm drops significantly, even without the presence of outliers. This is due, as pointed in Garg et al. (2013), to the fact that trajectory-based algorithms become unstable when dealing with small camera

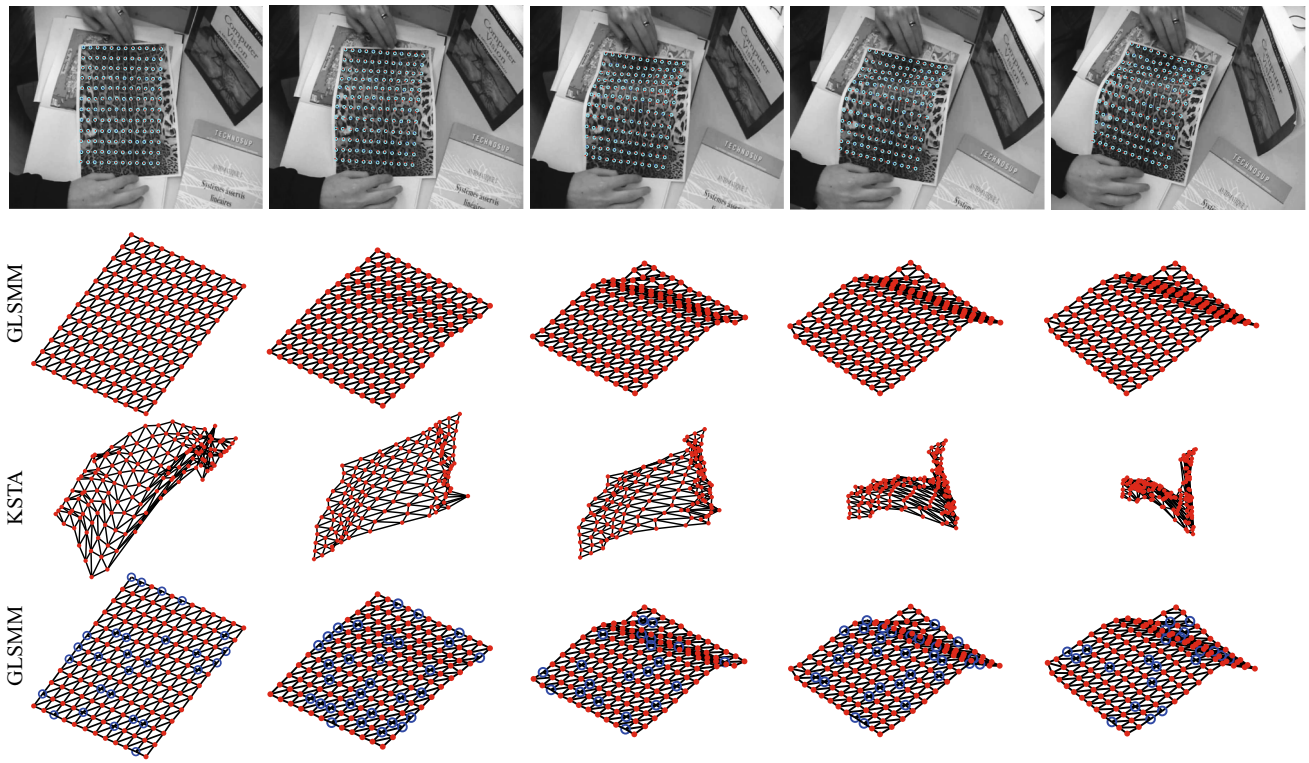


Fig. 11 Paper Bending sequence. *Top* Input frames #20, #40, #60, #80 and #100. The 2D input tracks are displayed as *cyan circles*, and the reprojected 3D points (after estimating the shape with our approach) are shown as *red* and *blue dots*. *Blue dots* correspond to missing data. *Bottom* The next three rows show the 3D view of the reconstructed shape obtained with our sequential GLSMM method without missing data, using the KSTA (Gotardo and Martínez 2011a) algorithm that batch processes all frames simultaneously, and again using our approach but considering a random pattern of 20% of missing measurements. Since this sequence only shows small changes in the rotation, KSTA (Gotardo and Martínez 2011a) becomes highly non-stable, even without the presence of missing data (Color figure online)

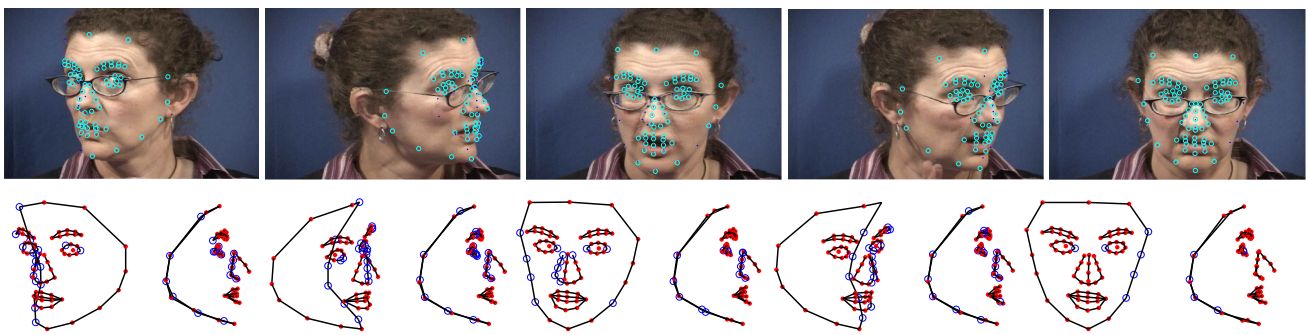


Fig. 12 ASL1 sequence. *Top* Frames #45, #67, #77, #92 and #115 with 2D tracking data and reprojected 3D shape represented as *cyan circles* and *red dots*, respectively. Missing points are shown in *blue*. *Bottom* Original viewpoint and side views of the 3D reconstruction obtained with our approach (Color figure online)

rotations, as is the case of this experiment. Similar results are obtained using CSF2 (Gotardo and Martínez 2011b).

7.2.5 ASL Sequences

In this subsection we test two sequences taken from the American Sign Language (ASL) dataset, which show a person moving the head while talking and hand gesturing. We use these sequences to provide a qualitative evaluation of our approach under the presence of structured missing data, due to self-occlusions produced by the interference of the hands or lack of visibility of certain regions due to the motion of the head. The *ASL1* sequence consists of 115 frames and 77 feature points, with a 17.4% of missing data. The *ASL2* sequence consists of 114 frames and also 77 feature points, with a 11.5% of missing data. In both cases, we use the sparse point tracks of Gotardo and Martínez (2011a) and the first 40 frames to compute the initial model. In Figs. 12 and 13,

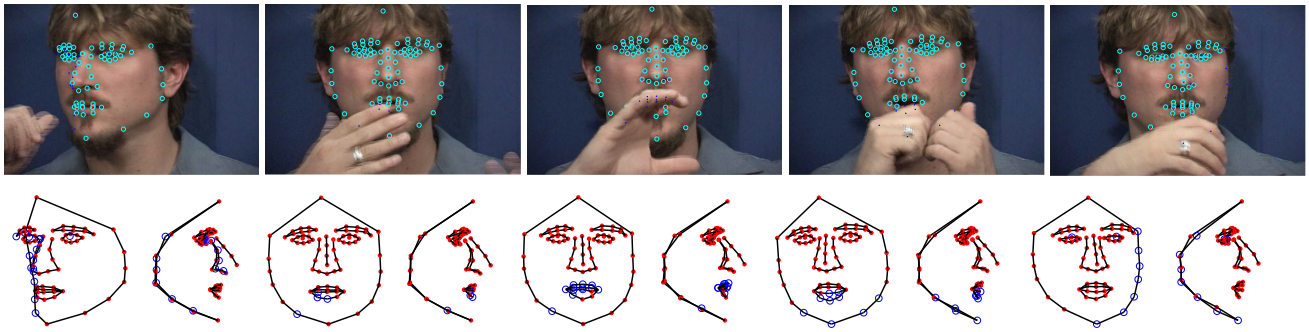


Fig. 13 ASL2 sequence. *Top* Frames #41, #58, #70, #92 and #114 with 2D tracking data and reprojected 3D shape represented as *cyan circles* and *red dots*, respectively. Missing points are shown in blue. *Bottom* Original viewpoint and side views of the 3D reconstruction obtained with our approach (Color figure online)

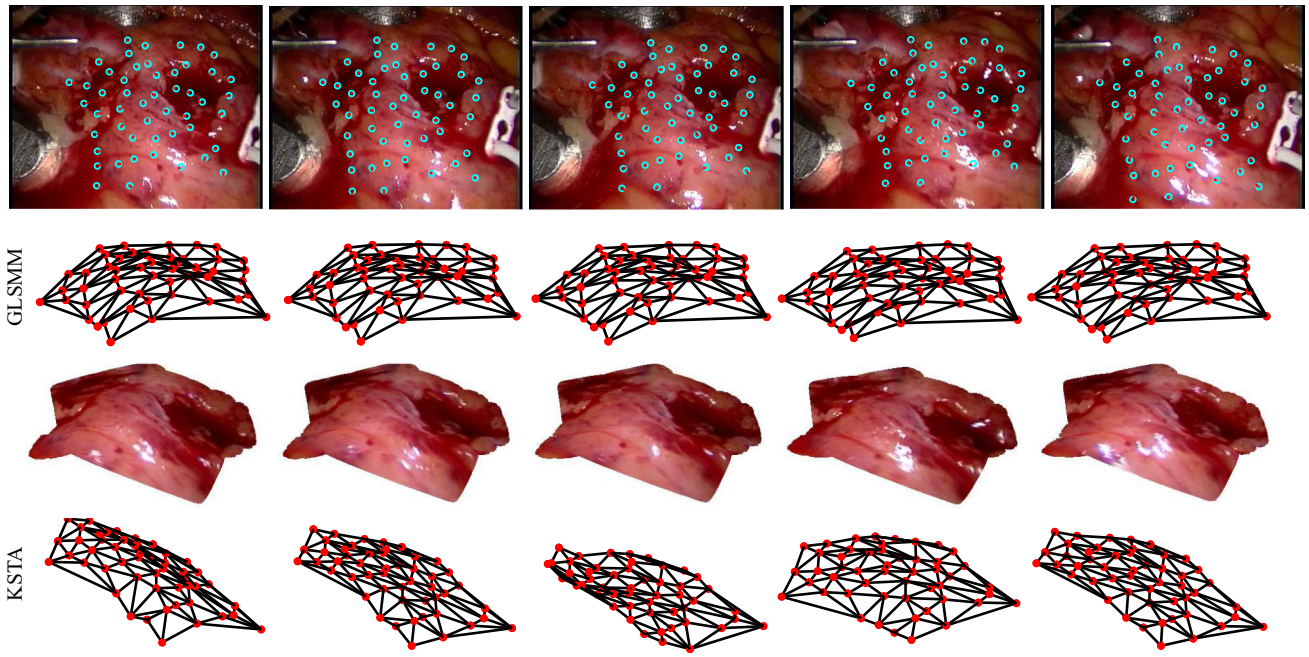


Fig. 14 Beating heart sequence. *Top* Frames #34, #49, #55, #66 and #79 with 2D tracking data and reprojected 3D shape with *cyan circles* and *red dots*, respectively. *Middle* 3D reconstruction of the shape by using a wire model, and other views with the original texture. *Bottom* Same views using KSTA (Gotardo and Martínez 2011a). Again, the 3D reconstruction of this approach seems to be highly non-stable, because the sequence is acquired with small camera rotation. Best viewed in *color*

we show that our approach can handle this type of structured occlusions. This is in contrast to other state-of-the-art approaches, such as SPM (Dai et al. 2012), which cannot handle these artifacts.

7.2.6 Beating Heart Sequence

Finally, we also show that our approach can be appropriate to handle medical imaging, where sequential, real time approaches are of paramount importance. For this purpose we present the *Beating Heart* sequence, consisting of 79 frames, acquired during bypass surgery, and which was provided by Garg et al. (2013). We use a sparse version of 50 feature points to show the generality of our approach. In Fig. 14, we represent the 3D reconstruction for this challenging sequence, which could be obtained at about 5 frames per second. We also provide a qualitative comparison with KSTA Gotardo and Martínez (2011a) using a basis of rank 6. Again, the small camera motion of this sequence seems to significantly impact the performance of the approach (Fig. 8).

8 Conclusion

In this paper we have exploited Newton's second law of motion to model the non-rigid deformation of an object represented by a system of particles. We have introduced this

simple physics-based dynamical model into a BA framework, yielding an approach that allows to simultaneously and on-the-fly recover camera motion and time-varying shape. We have also used this approach to progressively learn a low-rank global model of the whole shape, which is fed back to the optimization framework in order to further constrain the local dynamics of the particles. Our system can handle different types of deformations, including articulated, non-rigid, isometric and extensible cases. Additionally, we do not require any training data and the overall solution is remarkably fast. All our claims have been experimentally validated in mocap and real sequences of a large variety of scenarios going from articulated human body, medical images of a beating heart sequence, and even a piece of paper that is split in two. In all cases we have shown similar performance to computationally intensive batch approaches, and being remarkably more accurate than other state-of-the-art sequential approaches. Regarding real-time capability, our approach ensures that the computational cost per frame is bounded and does not grow with the number of frames. We believe our method is a suitable groundwork for later exploitation in real-time applications. Our future work is oriented to generalize our model to full perspective projection cameras and incorporating feature tracking and outlier detection into a single process.

Acknowledgements We would like to thank the anonymous reviewers for their insights and comments that have significantly contributed to improving this manuscript. This work has been partially supported by the Spanish Ministry of Science and Innovation under Project RobInstruct TIN2014-58178-R; by a scholarship FPU12/04886 from the Spanish MECD; and by the ERA-net CHISTERA Projects VISEN PCIN-2013-047 and I-DRESS PCIN-2015-147. The authors also thank Chris Russell, Lourdes Agapito and Paulo Gotardo for making their data available.

References

Agudo, A., & Moreno-Noguer, F. (2015). Simultaneous pose and non-rigid shape with particle dynamics. In *Conference on computer vision and pattern recognition*, pp. 2179–2187

Agudo, A., Calvo, B., Montiel, & J. M. M. (2012). Finite element based sequential Bayesian non-rigid structure from motion. In *Conference on computer vision and pattern recognition*, pp. 1418–1425

Agudo, A., Agapito, L., Calvo, B., & Montiel, J. M. M. (2014a). Good vibrations: A modal analysis approach for sequential non-rigid structure from motion. In *Conference on computer vision and pattern recognition*, pp. 1558–1565

Agudo, A., Montiel, J. M. M., Agapito, L., & Calvo, B. (2014b). Online dense non-rigid 3D shape and camera motion recovery. In *British machine vision conference*

Agudo, A., Moreno-Noguer, F., Calvo, B., & Montiel, J. M. M. (2016). Sequential non-rigid structure from motion using physical priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *38*(5), 979–994.

Akhter, I., Sheikh, Y., Khan, S., & Kanade, T. (2008). Non-rigid structure from motion in trajectory space. In *Neural information processing systems*, pp. 41–48

Baraff, D. (1989). Analytical methods for dynamic simulation of non-penetrating rigid bodies. In *Conference on computer graphics and interactive techniques*, pp. 223–232

Bartoli, A., Gay-Bellile, V., Castellani, U., Peyras, J., Olsen, S., & Sayd, P. (2008). Coarse-to-fine low-rank structure-from-motion. In *Conference on computer vision and pattern recognition*, pp. 1–8

Brand, M. (2001). Morphable 3D models from video. In *Conference on computer vision and pattern recognition*, pp. 456–463

Bregler, C., Hertzmann, A., & Biermann, H. (2000). Recovering non-rigid 3D shape from image streams. In *Conference on computer vision and pattern recognition*, pp. 690–696

Brubaker, M., Sigal, L., & Fleet, D. (2009). Estimating contact dynamics. In *International conference on computer vision*, pp. 2389–2396

Chhatkuli, A., Pizarro, D., & Bartoli, A. (2014). Non-rigid shape-from-motion for isometric surfaces using infinitesimal planarity. In *British machine vision conference*

Dai, Y., Li, H., & He, M. (2012). A simple prior-free method for non-rigid structure from motion factorization. In *Conference on computer vision and pattern recognition*, pp. 2018–2025

Del Bue, A., Llado, X., & Agapito, L. (2006). Non-rigid metric shape and motion recovery from uncalibrated images using priors. In *Conference on computer vision and pattern recognition*, pp. 1191–1198

Fayad, J., Agapito, L., & Del Bue, A. (2010). Piecewise quadratic reconstruction of non-rigid surfaces from monocular sequences. In *European conference on computer vision*, pp. 297–310

Garg, R., Roussos, A., & Agapito, L. (2013). Dense variational reconstruction of non-rigid surfaces from monocular video. In *Conference on computer vision and pattern recognition*, pp. 1272–1279

Gotardo, P. F. U., & Martínez, A. M. (2011a). Kernel non-rigid structure from motion. In *International conference on computer vision*, pp. 802–809

Gotardo, P. F. U., & Martínez, A. M. (2011b). Non-rigid structure from motion with complementary rank-3 spaces. In *Conference on computer vision and pattern recognition*, pp. 3065–3072

Koh, W., Narain, R., & O'Brien, J. F. (2014). View-dependent adaptive cloth simulation. In *ACM SIGGRAPH/Eurographics symposium on computer animation*, pp. 159–166

Lee, M., Cho, J., Choi, C. H., & Oh, S. (2013). Procrustean normal distribution for non-rigid structure from motion. In *Conference on computer vision and pattern recognition*, pp. 1280–1287

Lim, J., Frahm, J., & Pollefeys, M. (2011). Online environment mapping. In *Conference on computer vision and pattern recognition*, pp. 3489–3496

Ma, Y., Kosecka, J., & Sastry, S. (1999). Optimization criteria and geometric algorithms for motion and structure estimation. *International Journal on Computer Vision*, *44*(3), 219–249.

Maier-Hein, L., Groch, A., Bartoli, A., Bodenstedt, S., Boissonnat, G., Chang, P. L., et al. (2014). Comparative validation of single-shot optical techniques for laparoscopic 3D surface reconstruction. *IEEE Transactions on Medical Imaging*, *33*(10), 1913–1930.

Marques, M., & Costeira, J. (2008). Optimal shape from estimation with missing and degenerate data. In *Workshop on motion and video computing*, pp. 1–6

Metaxas, D., & Terzopoulos, D. (1993). Shape and nonrigid motion estimation through physics-based synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *15*(6), 580–591.

Moreno-Noguer, F., & Porta, J. M. (2011). Probabilistic simultaneous pose and non-rigid shape recovery. In *Conference on computer vision and pattern recognition*, pp. 1289–1296

Newcome, R., & Davison, A. J. (2010). Live dense reconstruction with a single moving camera. In *Conference on computer vision and pattern recognition*, pp. 1498–1505

Paladini, M., Del Bue, A., Stosic, M., Dodig, M., Xavier, J., & Agapito, L. (2009). Factorization for non-rigid and articulated structure using metric projections. In *Conference on computer vision and pattern recognition*, pp. 2898–2905

Paladini, M., Bartoli, A., & Agapito, L. (2010). Sequential non rigid structure from motion with the 3D implicit low rank shape model. In *European conference on computer vision*, pp. 15–28

Park, H. S., Shiratori, T., Matthews, I., & Sheikh, Y. (2010). 3D reconstruction of a moving point from a series of 2D projections. In *European conference on computer vision*, pp. 158–171

Popovic, Z., & Witkin, A. (1999). Physically based motion transformation. In *Conference on computer graphics and interactive techniques*, pp. 11–20

Russell, C., Fayad, J., & Agapito, L. (2011). Energy based multiple model fitting for non-rigid structure from motion. In *Conference on computer vision and pattern recognition*, pp. 3009–3016

Russell, C., Yu, R., & Agapito, L. (2014). Video pop-up: Monocular 3D reconstruction of dynamic scenes. In *European conference on computer vision*, pp. 583–598

Salzmann, M., & Urtasun, R. (2011). Physically-based motion models for 3D tracking: A convex formulation. In *International conference on computer vision*, pp. 2064–2071

Shaji, A., & Chandran, S. (2008). Riemannian manifold optimisation for non-rigid structure from motion. In *Workshop on non-rigid shape analysis and deformable image alignment*, pp. 1–6

Tao, L., Mein, S. J., Quan, W., & Matuszewski, B. J. (2013). Recursive non-rigid structure from motion with online learned shape prior. *Computer Vision and Image Understanding*, *117*(10), 1287–1298.

Taylor, J., Jepson, A.D., & Kutulakos, K.N. (2010). Non-rigid structure from locally-rigid motion. In *Conference on computer vision and pattern recognition*, pp. 2761–2768

Tomasi, C., & Kanade, T. (1992). Shape and motion from image streams under orthography: A factorization approach. *International Journal on Computer Vision*, *9*(2), 137–154.

Torresani, L., Hertzmann, A., & Bregler, C. (2008). Nonrigid structure-from-motion: Estimating shape and motion with hierarchical priors. *Transactions on Pattern Analysis and Machine Intelligence*, *30*(5), 878–892.

Valmadre, J., & Lucey, S. (2012). General trajectory prior for non-rigid reconstruction. In *Conference on computer vision and pattern recognition*, pp. 1394–1401

Varol, A., Salzmann, M., Tola, E., & Fua, P. (2009). Template-free monocular reconstruction of deformable surfaces. In *International conference on computer vision*, pp. 1811–1818

Vondrak, M., Sigal, L., & Jenkins, O.C. (2008). Physical simulation for probabilistic motion tracking. In *Conference on computer vision and pattern recognition*, pp. 1–8

Xiao, J., Chai, J., & Kanade, T. (2006). A closed-form solution to non-rigid shape and motion. *International Journal on Computer Vision*, *67*(2), 233–246.