Total Estimation from RGB Video: On-line Camera Self-Calibration, Non-Rigid Shape and Motion

Antonio Agudo
Institut de Robòtica i Informàtica Industrial, CSIC-UPC
Barcelona, 08028, Spain
Email: aagudo@iri.upc.edu

***Abstract*—In this paper we present a sequential approach to jointly retrieve camera auto-calibration, camera pose and the 3D reconstruction of a non-rigid object from an uncalibrated RGB image sequence, without assuming any prior information about the shape structure, nor the need for a calibration pattern, nor the use of training data at all. To this end, we propose a Bayesian filtering approach based on a sum-of-Gaussians filter composed of a bank of extended Kalman filters (EKF). For every EKF, we make use of dynamic models to estimate its state vector, which later will be Gaussianly combined to achieve a global solution. To deal with deformable objects, we incorporate a mechanical model solved by using the finite element method. Thanks to these ingredients, the resulting method is both efficient and robust to several artifacts such as missing and noisy observations as well as sudden camera motions, while being available for a wide variety of objects and materials, including isometric and elastic shape deformations. Experimental validation is proposed in real experiments, showing its strengths with respect to competing approaches.**

I. Introduction

The simultaneous 3D reconstruction of a shape structure together with the full 3D trajectory of a RGB camera is a well-studied problem in computer vision and robotics. Early works assumed the observed shape is rigid, achieving robust solutions even in real time. In this context, two type of formulations were proposed: global-optimization approaches based mainly on bundle adjustment [1], [29], [30], and filtering ones such as those based on the Extended Kalman Filter (EKF) [18], [37]. While some degree of success has been obtained in rigid scenarios, retrieving the 3D geometry of the vivid moving real world is still in its infancy. In those cases, the problem is inherently ill-posed since many different 3D representations can have very similar image observations, producing severe ambiguities that can only be avoided by incorporating *the art of the priors* about the camera trajectory and shape deformation. Unfortunately, since including deformation priors is substantially more difficult than using simple rigidity, solving for a deformable shape is very weakly constrained compared with retrieving a rigid structure.

Many efforts have been done in the last decade [6], [9], [17], [38], proposing a wide variety of priors to constrain the solution space. Alternatively, a better representation of the underlying dynamics involved in non-rigid deformations can be obtained through physically-grounded models, such as force-based kinematics [15], inextensibility-based deformations [39], linear [26] or non-linear [20] elastic models, and numerical techniques based on the Finite Element Method (FEM) [3], [8], to name just a few.

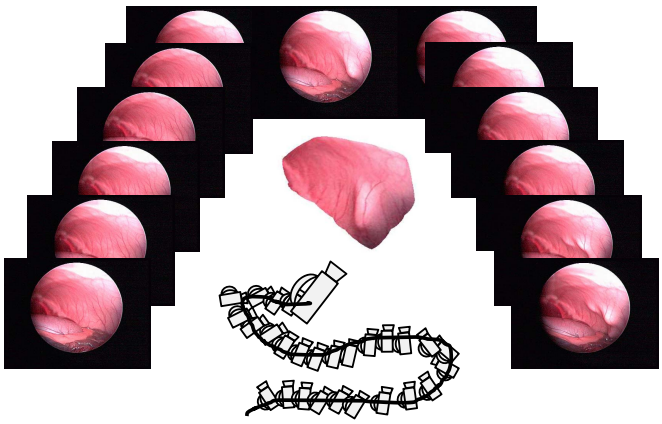

Fig. 1. **Total estimation from RGB video.** In this paper we address the problem of simultaneously and sequentially retrieving camera self-calibration, camera motion, and the 3D reconstruction of a deformable object from a sequence of RGB images. This is a very challenging problem in computer vision and robotics, especially when neither training data nor a calibration pattern are assumed in advance. Medical applications (such as the laparoscopy) are a typical case where the amount of priors to be exploited is reduced, and an on-line estimation is mandatory.

In any event, most of these approaches batch process all images of the video at once, after video capture, preventing them from being used on-line and in real-time applications. In this case, only the observations until the current image can be considered, doing even more complex the problem to be sorted out. To solve this limitation, some works for sequential estimation were proposed [2], [32], sharing a couple of limitations with the previous ones. On the one hand, these formulations relied on an orthographic camera model, which is only a good approximation when the object depth is much smaller than the distance from the camera. On the other hand, the previous approaches used image points that can be observed and tracked over the sequence, being a standard practice to assume these measurements as input. An interesting exception is [8], where a sequential formulation was presented, including the use of a projective-camera model and a strategy in order to solve data association. The only limitation is that the calibration of the camera needs to be known a priori, i.e., to be effectively used, the method requires an off-line step to be calibrated, limiting its applicability in real scenarios. In practice, self-calibration allows the computation from scratch

of some projective parameters, such as the focal length, principal point, and skew; and even including the distortion parameters to improve the metric reconstruction.

In this paper, we propose a solution in order to jointly retrieve camera self-calibration, camera trajectory and the 3D reconstruction of a non-rigid shape (a typical case we handle is displayed in Fig. 1). All of them, it is estimated in a sequential fashion, computing automatically the correspondences between consecutive images, for a full-perspective camera, and without assuming any training data at all nor a calibration pattern. We are not aware of any other approach solving simultaneously the three problems. To this end, we use a Sum-of-Gaussians (SoG) filter in combination with a mechanical model to encode shape deformations, and a motion model to code the camera parameters. We extensively evaluate our approach on real sequences. Moreover, as we will show later, our solution exhibits a good trade-off between global accuracy and generality in comparison with competing techniques.

II. Related work

Estimating the non-rigid 3D shape from a single RGB camera has been an active research area in the past two decades. In the literature, two main classes of techniques have proved most effective so far: template-based formulations and non-rigid structure from motion (NRSfM). On the one side, template-based approaches rely on establishing correspondences with a reference image in which the shape is already known in advance [31], [36]. To avoid ambiguities, additional constraints are included in the optimization, such as the inextensibility, providing very robust solutions but limiting its applicability to inelastic surfaces [31], [39]. More recently, some solutions have been proposed to handle elastic deformations [20], [26] by enforcing physical priors. While these formulations normally use a perspective camera model, the internal parameters have to be known a priori, being a good calibration a key factor to achieve accurate solutions. Enforcing also inextensibility, some works [14], [33] have been also extended to include the focal length in the estimation.

On the other side, NRSfM has been proposed to solve the problem from 2D tracking data in a monocular video (in the literature, feature points are collected in a measurement matrix). Most approaches have incorporated additional priors in different optimization frameworks. The most important prior is to assume the shape to lie in a low-rank subspace [6], [9], [17], [38], incorporating spatial [38] or temporal [13] shape smoothness; by imposing the 3D shapes to be closely aligned [24], or by means of a union of subspaces [40].

However, in contrast to their rigid counterparts [1], [29], previous approaches to NRSfM process all the images at once, remaining the sequential estimation as a challenging problem. Some sequential formulations have emerged in the non-rigid domain [2], [32], providing striking results. Unfortunately, these approaches rely on orthographic camera models, or they are not capable of solving feature tracking and outliers detection in a single process. More recently, in [8] was proposed a sequential solution to recover both inelastic and elastic materials while tracks the feature points. In addition, this approach includes a full projective camera model where the calibration is pre-computed after video capture. Again, while self-calibration has been addressed for decades in rigid shapes [12], [16], [22], [34], its incorporation in non-rigid scenarios is very limited. Table I summarizes a qualitative comparison in terms of available characteristics of our approach and the most relevant competing approaches. It is worth noting that our approach is the only one that has all characteristics.

Meth. \ Feat.	Tracking	Self-Calibration		Process		Shape		
		Focal	Full	Batch	Sequential	Rigid	Non-Rigid	
							Isometric	Elastic
[12], [34]	✓	✓			✓	✓		
[16]	✓		✓		✓	✓		
[19], [25], [38]				✓		✓	✓	✓
[2], [32]					✓	✓	✓	✓
[8]	✓				✓	✓	✓	✓
[14], [33]	✓	✓		✓		✓	✓	
Ours	✓		✓		✓	✓	✓	✓

TABLE I

Qualitative comparison of our approach with respect to competing methods. Our approach is the only one that jointly retrieves 3D reconstruction of both rigid and non-rigid objects (from isometric to elastic deformations), estimates the full self-calibration of the camera in a sequential fashion, and automatically solves the tracking (the measurement matrix in a NRSfM context) in the same loop. Note that [14] is a shape-from-template method, i.e., a 3D template is known a priori to establish feature correspondences.

In this paper we depart from previous work in that our solution can, in a sequential manner, jointly estimate the 3D reconstruction of a non-rigid object (both inelastic and elastic deformations are considered), camera pose, and camera self-calibration. To the best of our knowledge, no previous approach has jointly addressed all these problems in a unified framework, and directly from a monocular video.

III. Revisiting SoG Filter

We next revisit the basics on SOG filtering [11] which will be employed later to propose our filter-based approach for estimating the state vector.

Let us denote a probability density function of $\mathbf{x}$ by $p(\mathbf{x})$. In a general case, we could approximate this function as a combination of G weighted functions, where every function in the subspace is represented by a multivariate Gaussian, such that:

$$p(\mathbf{x}) = \sum_{g=1}^{G} \gamma^g \, \mathcal{N}\left(\mathbf{x}^g; \mathbf{P}^g\right), \tag{1}$$

where $\mathbf{x}^g$ and $\mathbf{P}^g$ represent the mean and covariance matrix, respectively, for the g-th Gaussian. γ^g is a weight coefficient, subject to the conditions $\sum_{g=1}^{G} \gamma^g = 1$ and $\gamma^g \geq 0$. It is worth mentioning that $p(\mathbf{x})$ could represent any probability density function, being theoretically well-approximated for high values of G.

As it can be seen in Eq. (1), $p(\mathbf{x})$ can be updated by modifying everyone of the Gaussian probability density functions (pdfs) in the combination. For instance, every Gaussian distribution can come from an EKF filter [23], where both mean and covariance are estimated by means of the use of

new observations in a prediction-update strategy. Then, the SOG algorithm combines several EKF solutions (a filter bank) running in parallel as it is displayed in Fig. 2.

IV. Self-Calibration Non-Rigid SOG

Our key contribution is to present a novel technique for simultaneously estimating the shape of a non-rigid object, the full trajectory of a moving camera, and its auto-calibration. To do this, we embed a FEM formulation that encodes shape deformations within the Bayesian framework of a SOG. This combination will provide a mechanism to sort out the problem in a unified manner. As it was commented before, our SOG filter exploits a bank of EKF filters for Bayesian estimation.

A. Bank of EKF Filters

To improve understanding, in this subsection IV-A we will drop super-indexes ^{g} to denote the g-th EKF component, since all definitions that we consider are the same for every filter.

1) Problem Formulation: The state of the camera is represented by a 18-dimensional vector, considering both intrinsic and extrinsic parameters such as:

$$\mathbf{m} = [\mathbf{c}^\top, \mathbf{r}^\top, \mathbf{q}^\top, \mathbf{v}^\top, \boldsymbol{\omega}^{\mathcal{C}\top}]^\top \ , \tag{2}$$

where $\mathbf{c}$ includes the internal calibration parameters, $\mathbf{r}$ and $\mathbf{q}$ denote the position and orientation quaternion, respectively, in order to express the pose of the camera relative to the world coordinate system $\mathcal{W}$. Eventually, $\mathbf{v}$ and $\boldsymbol{\omega}$ are the linear and angular velocities relative to $\mathcal{W}$ and to a frame $\mathcal{C}$ fixed to the camera, respectively.

In addition to the state of the camera, we also consider the surface of a non-rigid object that it is represented by means of a triangulated mesh with N vertices $\mathbf{g}_n = [x_i, y_i, z_i]^\top$ concatenated in a $3N$ vector $\mathbf{y} = [\mathbf{g}_1, \dots, \mathbf{g}_N]^\top$. As we propose to solve for the full 3D camera trajectory, without loss of generality, we assume that $R << N$ of these points are rigid, i.e., they always remain steady. It is worth pointing out that without this assumption, we could not disambiguate between camera motions and rigid displacements of the object [4], [7].

To jointly estimate both camera and shape, we define by $\mathbf{m}_k$, $\mathbf{y}_k$, and $\mathcal{I}_k$ the camera state vector, the 3D mesh state configuration, and the input image at frame k. Our problem consists in using the current state and the input image $\mathcal{I}_{k+1}$ at frame $k+1$, to retrieve both $\mathbf{m}_{k+1}$ and $\mathbf{y}_{k+1}$. To this end, the state vector for every EKF filter is represented by $\mathbf{x} = [\mathbf{m}^\top, \mathbf{y}^\top]^\top$, i.e., the full state vector includes both camera (calibration and location) and the 3D point locations. Upon the arrival of a new input frame, the corresponding estimation is iteratively updated and can be written as $\hat{\mathbf{x}} = [\hat{\mathbf{m}}^\top, \hat{\mathbf{y}}^\top]^\top$, denoting by $\mathbf{P}$ its covariance matrix. We next describe the main ingredients of this process, introducing the dynamic models we use to predict the state vector.

2) Camera and Surface Motion Models: As it was introduced before, the camera state vector can be split in two terms: the vector $\mathbf{c}$ to represent the intrinsic parameters and the rest of the elements to encode the extrinsic ones and the dynamic. On the one hand, $\mathbf{c}$ is a 5-dimensional vector and includes the focal length α, the principal point coordinates (β_x, β_y), and two parameters to model radial distortion (k_1, k_2), as it was done in the literature [16]. On the other hand, to model the extrinsic parameters, we use a constant velocity model [8], [18], introducing an impulse of both linear and angular velocities at every frame step Δt as $\Delta\mathbf{v} = \dot{\mathbf{v}}\Delta t$ and $\Delta\boldsymbol{\omega}^{\mathcal{C}} = \dot{\boldsymbol{\omega}}^{\mathcal{C}}\Delta t$, respectively. $\dot{\mathbf{v}}$ and $\dot{\boldsymbol{\omega}}^{\mathcal{C}}$ denote unknown linear and angular acceleration variables, and are modeled by a zero-mean Gaussian distribution with covariance matrix $\mathbf{Q}_\mathbf{m}$. Finally, the camera state function $\mathbf{m}_{k+1} \equiv \mathbf{m}_{k+1}(\mathbf{m}_k, \mathbf{0}, \Delta\mathbf{v}, \Delta\boldsymbol{\omega}^{\mathcal{C}})$ is represented by:

$$\mathbf{m}_{k+1} = \begin{bmatrix} \alpha_{k+1} \\ \beta_{x_{k+1}} \\ \beta_{y_{k+1}} \\ k_{1_{k+1}} \\ k_{2_{k+1}} \\ \mathbf{r}_{k+1} \\ \mathbf{q}_{k+1} \\ \mathbf{v}_{k+1} \\ \boldsymbol{\omega}^{\mathcal{C}}_{k+1} \end{bmatrix} = \begin{bmatrix} \alpha_k \\ \beta_{x_k} \\ \beta_{y_k} \\ k_{1_k} \\ k_{2_k} \\ \mathbf{r}_k + (\mathbf{v}_k + \Delta\mathbf{v})\Delta t \\ \mathbf{q}_k \times \mathbf{q}((\boldsymbol{\omega}^{\mathcal{C}}_k + \Delta\boldsymbol{\omega}^{\mathcal{C}})\Delta t) \\ \mathbf{v}_k + \Delta\mathbf{v} \\ \boldsymbol{\omega}^{\mathcal{C}}_k + \Delta\boldsymbol{\omega}^{\mathcal{C}} \end{bmatrix} , \tag{3}$$

where $\mathbf{q}((\boldsymbol{\omega}^{\mathcal{C}}_k + \Delta\boldsymbol{\omega}^{\mathcal{C}})\Delta t)$ indicates the quaternion defined by the rotation vector $(\boldsymbol{\omega}^{\mathcal{C}}_k + \Delta\boldsymbol{\omega}^{\mathcal{C}})\Delta t$.

To model the surface localization, the object state is encoded using a FEM model with unknown Gaussian forces. Following [8], we can introduce a compliance matrix $\mathbf{C}_k$ to correlate all the points in the map. Let us consider that an unknown force vector $\Delta\mathbf{f}$ is applied on the shape, producing a displacement field coded by the vector $\Delta\mathbf{d}$. Both terms can be related by using the compliance matrix as $\Delta\mathbf{d} = \mathbf{C}_k\Delta\mathbf{f}$

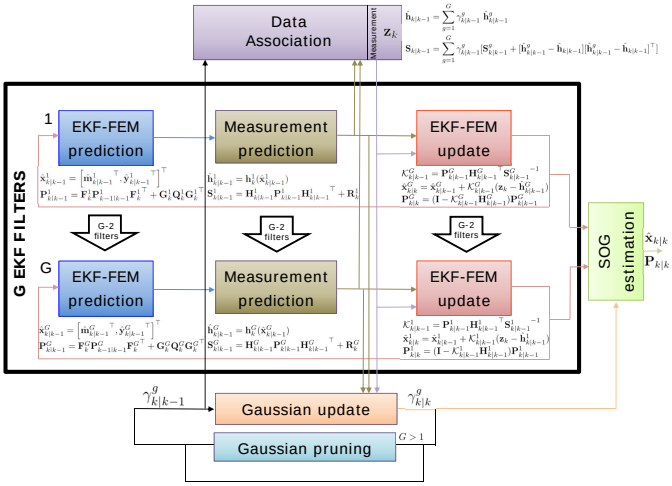

Fig. 2. **Non-Rigid SOG filter.** Our non-rigid SOG filter can be applied directly to a monocular video to jointly estimate camera trajectory, camera self-calibration and the 3D reconstruction of a deformable object. To this end, the SOG filter is composed of a bank of EKF filters (state prediction, measurement prediction and updating are performed for every of them), that Gaussianly are combined to produce an overall solution. Data association is also solved automatically in the same loop by combining the contribution of every filter in the bank. The most important equations in our model are displayed for the k-th frame. To improve efficiency, bad filters in term of weight contribution are pruned. Best viewed in digital version.

(recall that its inverse, the stiffness matrix $\mathbf{K}_k$, may map displacements into forces).

With these ingredients, the surface configuration $\mathbf{y}_k$ at a time step k, and its associated compliance matrix $\mathbf{C}_k$, can be used to obtain the new state estimation via the state function:

$$\mathbf{y}_{k+1} \equiv \mathbf{y}_{k+1}\left(\mathbf{y}_k, \Delta\mathbf{f}\right) = \mathbf{y}_k + \mathbf{C}_k\Delta\mathbf{f}\ , \tag{4}$$

where $\Delta\mathbf{f}$ is assumed to be a random variable with zero mean and Gaussian distribution. As it can be seen, we recompute the compliance matrix $\mathbf{C}_k$ at each iteration, thus being adapted to the deforming geometry of the structure. In order to compute $\mathbf{C}_k$, we follow the FEM model proposed by [8] that is available for both inelastic and elastic materials by means of the use of normalized Gaussian forces. Thanks to this model, we can correct accumulative errors produced by the inherent linearization of every EKF in our filter bank, that might cause drifting problems. Finally, it is necessary to associate a covariance matrix $\mathbf{Q}_\mathbf{y}$ to this non-rigid model, whose elements encode deformation variances except for the rigid points where the entries will be zero.

3) Measurement Model: We next describe how the process of observing the deformable points is modeled in a generic image frame. Considering the 3D coordinates of a point expressed in the world coordinate system $\mathcal{W}$, $\mathbf{g}_i = [x_i, y_i, z_i]^\top$, we initially use the extrinsic components ($\mathbf{q}$ and $\mathbf{r}$) of the camera state vector to compute $\mathbf{g}_i^\mathcal{C}$, the expected position of the feature in the local coordinate system of the camera $\mathcal{C}$ is:

$$\mathbf{g}_i^\mathcal{C} = [x_i^\mathcal{C}, y_i^\mathcal{C}, z_i^\mathcal{C}]^\top = \mathbf{O}^\top(\mathbf{g}_i - \mathbf{r}), \tag{5}$$

where $\mathbf{O}$ denotes the rotation matrix corresponding to the quaternion $\mathbf{q}$. The measurement function $\mathbf{b}_i\left(\mathbf{m}, \mathbf{g}_i\right)$ computes the 2D projection of $\mathbf{g}_i^\mathcal{C}$ onto the image, knowing the camera and shape estimations in the current state vector:

$$\mathbf{b}_i \equiv \mathbf{b}_i\left(\mathbf{m}, \mathbf{g}_i\right) = \left[\begin{array}{c} \beta_x - \alpha\frac{x_i^\mathcal{C}}{z_i^\mathcal{C}} \\ \beta_y - \alpha\frac{y_i^\mathcal{C}}{z_i^\mathcal{C}} \end{array}\right]. \tag{6}$$

In addition, in order to compensate for radial distortion, we introduce a first order radial distortion model [27]. The undistorted projective coordinates $\mathbf{b}_i$ can be computed from the distorted ones $\mathbf{b}_i^d \equiv (u_d, v_d)^\top$ acquired by the camera as:

$$\mathbf{b}_i(\mathbf{b}_i^d) = \left[\begin{array}{c} \beta_x + (u_d - \beta_x)(1 + k_1 r_d^2 + k_2 r_d^4) \\ \beta_y + (v_d - \beta_y)(1 + k_1 r_d^2 + k_2 r_d^4) \end{array}\right], \tag{7}$$

where $r_d = \sqrt{(d_x(u_d - \beta_x))^2 + (d_y(v_d - \beta_y))^2}$ with (d_x, d_y) the pixel size. Without loss of generality, we assume square pixels, i.e., $d_x \equiv d_y$, tuning this value from the camera specifications.

The measurement equations for the visible q mesh vertexes are stacked together into a unique non-linear measurement function of the state vector as:

$$\hat{\mathbf{h}}_{k|k-1}(\hat{\mathbf{x}}_{k|k-1}) = \left[\mathbf{b}_1 \quad \dots \quad \mathbf{b}_i \quad \dots \quad \mathbf{b}_q\right]^\top. \tag{8}$$

A zero-mean Gaussian error with diagonal 2×2 covariance matrix $\Sigma_{\mathbf{b}_i}$ is assigned to every measurement. The overall measurement noise covariance $\mathbf{R}_k$ is built by assembling the previous covariances into a block diagonal matrix.

4) Jacobian Computation: The proposed sequential monocular non-rigid SOG-FEM algorithm needs information about how the Jacobian matrices for every EKF are assembled. Considering our motion model described in section IV-A2, the Jacobian matrices of the dynamic model can be defined as:

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{I}_5 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_3 & \mathbf{0} & \Delta t\,\mathbf{I}_3 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \frac{\partial \mathbf{q}_{k+1}}{\partial \mathbf{q}_k} & \mathbf{0} & \frac{\partial \mathbf{q}_{k+1}}{\partial \boldsymbol{\omega}_k^\mathcal{C}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_3 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_3 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{3n} \end{bmatrix}, \tag{9}$$

$$\mathbf{G}_k = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \Delta t\,\mathbf{I}_3 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{\partial \mathbf{q}_{k+1}}{\partial \Delta\boldsymbol{\omega}^\mathcal{C}} & \mathbf{0} \\ \mathbf{I}_3 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_3 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{C}_k \end{bmatrix}, \tag{10}$$

where $\mathbf{I}_n$ denotes a $n \times n$ identity matrix. Let $\mathbf{n} = [\Delta\mathbf{v}^\top, \Delta{\boldsymbol{\omega}^\mathcal{C}}^\top, \Delta\mathbf{f}^\top]^\top$ be the state vector noise whose covariance matrix $\mathbf{Q}$ is block diagonal, and it can be composed of $\mathbf{Q}_\mathbf{m}$ and $\mathbf{Q}_\mathbf{y}$, respectively. The full prediction stage is summarized in Fig. 2-left.

Considering the set of q measurements of Eq. (8), the Jacobian matrix $\mathbf{H}_k$ can be expressed as:

$$\mathbf{H}_k = \left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right] = \begin{bmatrix} \frac{\partial \mathbf{b}_1}{\partial \mathbf{m}} & \frac{\partial \mathbf{b}_1}{\partial \mathbf{y}_1} & \mathbf{0} & \cdots & \mathbf{0} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{\partial \mathbf{b}_i}{\partial \mathbf{m}} & \mathbf{0} & \mathbf{0} & \frac{\partial \mathbf{b}_i}{\partial \mathbf{y}_i} & \mathbf{0} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{\partial \mathbf{b}_q}{\partial \mathbf{m}} & \mathbf{0} & \cdots & \mathbf{0} & \frac{\partial \mathbf{b}_q}{\partial \mathbf{y}_n} \end{bmatrix}, \tag{11}$$

that it will be used in the measurement prediction stage and in the EKF-FEM update (see Fig. 2-middle/right).

B. Data Association

In order to perform data association we proceed as follows. We first predict the image coordinates $\mathbf{b}_i^g$ of every keypoint feeding the current prediction estimate $\hat{\mathbf{x}}_{k|k-1}^g$ into the measurement model in Eqs. (6)-(7), to obtain the predicted measurements in Eq. (8). In addition, the Jacobian of this function, $\mathbf{H}_k^g$ in Eq. (11), is used to compute the uncertainty of the prediction, represented by the innovation covariance $\mathbf{S}_{k|k-1}^g = \mathbf{H}_{k|k-1}^g \mathbf{P}_{k|k-1}^g {\mathbf{H}_{k|k-1}^g}^\top + \mathbf{R}_k^g$. Finally, the predicted measurements for every EKF are Gaussianly combined to obtain $\hat{\mathbf{h}}_{k|k-1}$ and $\mathbf{S}_{k|k-1}$ as:

$$\hat{\mathbf{h}} \equiv \hat{\mathbf{h}}_{k|k-1} = \sum_{g=1}^{G} \gamma_{k|k-1}^g\ \hat{\mathbf{h}}_{k|k-1}^g\,,$$

$$\mathbf{S}_{k|k-1} = \sum_{g=1}^{G} \gamma_{k|k-1}^g [\mathbf{S}_{k|k-1}^g + [\hat{\mathbf{h}}_{k|k-1}^g - \hat{\mathbf{h}}][\hat{\mathbf{h}}_{k|k-1}^g - \hat{\mathbf{h}}]^\top],$$

that we use to define the ellipses in the image plane where a guided search of matches is performed.

C. Gaussian Update

The contribution of every EKF filter is Gaussianly combined by means of the weight coefficient $\gamma^g_{k|k-1}$, that are updated for every frame. To this end, we obtain an innovation mean for every EKF defined by $\mathbf{i}^g_{k|k-1} = \mathbf{z}_k - \mathbf{h}^g_k(\hat{\mathbf{x}}^g_{k|k-1})$, where $\mathbf{z}_k$ denotes the current observations and $\mathbf{h}^g_k(\hat{\mathbf{x}}^g_{k|k-1})$ indicates the predicted ones as a function of the predicted state $\hat{\mathbf{x}}^g_{k|k-1}$. Finally, every weight coefficient at frame k can be updated as:

$$\gamma^g_{k|k} = \frac{\gamma^g_{k|k-1}\ \mathcal{N}(\mathbf{i}^g_{k|k-1}; \mathbf{S}^g_{k|k-1})}{\sum_{g=1}^{G} \gamma^g_{k|k-1}\ \mathcal{N}(\mathbf{i}^g_{k|k-1}; \mathbf{S}^g_{k|k-1})}. \tag{12}$$

Once the weight coefficients are known (see Gaussian update stage in Fig. 2), an overall mean and covariance for the SOG filter can be represented as:

$$\hat{\mathbf{x}}_{k|k} = \sum_{g=1}^{G} \gamma^g_{k|k}\ \hat{\mathbf{x}}^g_{k|k}\,, \tag{13}$$

$$\mathbf{P}_{k|k} = \sum_{g=1}^{G} \gamma^g_{k|k}\,[\mathbf{P}^g_{k|k} + [\hat{\mathbf{x}}^g_{k|k} - \hat{\mathbf{x}}_{k|k}][\hat{\mathbf{x}}^g_{k|k} - \hat{\mathbf{x}}_{k|k}]^\top], \tag{14}$$

which it can be seen as the global estimation of our algorithm. Note that, though, we will never use this estimation in the bank of EKF filters.

D. Gaussian Pruning

Assuming that the final pdf should follow a unimodal Gaussian distribution with small covariance, we gradually reduce the number of Gaussian pdfs in our model that obtained a low weight factor $\gamma^g_{k|k}$ [23]. That can be done via a sequential probability ratio test, improving so the efficiency of our algorithm. Considering every Gaussian in the SOG filter, we can define a null H_0 and an alternative H_1 hypothesis, in order to denote when a Gaussian filter represents the true state or not, respectively. To this end, we define Wald's boundaries [10] in terms of decision errors of false alarm ψ_a and missed detection ψ_b, accepting the null hypothesis if:

$$\prod_{k=1}^{K} \frac{\mathcal{L}^g_k(H_0)}{\mathcal{L}^g_k(H_1)} > \frac{1-\psi_b}{\psi_a}\,, \tag{15}$$

or the alternative one if:

$$\prod_{k=1}^{K} \frac{\mathcal{L}^g_k(H_0)}{\mathcal{L}^g_k(H_1)} < \frac{\psi_b}{1-\psi_a}\,, \tag{16}$$

where $\mathcal{L}^g_k(H_0)$ and $\mathcal{L}^g_k(H_1)$ represent the likelihoods of the data under hypothesis H_0 and H_1, respectively, and they can be computed as:

$$\mathcal{L}^g_k(H_0) = \mathcal{N}(\mathbf{i}^g_{k|k-1}; \mathbf{S}^g_{k|k-1}),$$

$$\mathcal{L}^g_k(H_1) = \sum_{a=1;a\neq g}^{G} \frac{\gamma^a_{k|k}}{\sum_{b=1;b\neq g}^{G} \gamma^b_{k|k}}\,\mathcal{N}(\mathbf{i}^a_{k|k-1}; \mathbf{S}^a_{k|k-1}).$$

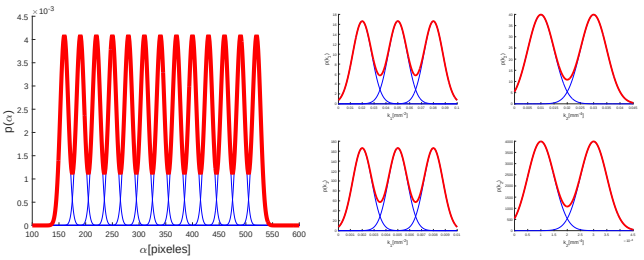

Fig. 3. **Probability density function for calibration parameters.** In all cases, it is represented the bank of filters that is used in our experiments. **Left:** $p(\alpha)$ for the focal length. **Middle-Right:** $p(k_1)$ and $p(k_2)$ for the first and second distortion parameters, respectively: sensors in general scenarios (top), the rest of cameras (bottom). See numerical axis to observe the scale deviation. For $\{\beta_x, \beta_y\}$, we assume a unimodal Gaussian distribution.

E. Initialization

Following [8], we initialize with a null force model in the first frames of the monocular video, i.e., the initial surface is computed by assuming a rigid map $\mathbf{y}^g_{k+1} = \mathbf{y}^g_k$ setting the covariance matrix as $\mathbf{Q}^g_{\mathbf{y}} = \mathbf{0}$ for the g-th EKF filter. In this part, we use inverse depth parametrization [28] from the detected FAST interest points [35] to encode the map locations (see estimations on the left in Figs. 4-8 to visualize initializations), transforming them to euclidean coordinates when the points are observed with enough parallax. Once the initialization is done, a Delaunay's tessellation of 3D points is performed to obtain a soup of irregular triangles. Camera covariance $\mathbf{Q}^g_{\mathbf{m}}$ is set to a constant diagonal matrix, a standard practice in rigid SLAM formulations [18].

V. Experimental Evaluation

We now present experimental results on real RGB videos, providing both quantitative or qualitative evaluation. Unfortunately, as we saw in the qualitative evaluation in table I, we cannot provide a quantitative comparison with respect to other approaches since none of them can solve the full problem directly from video. First of all, we define the SOG filter that we use in our experiments, in spite of using different cameras for acquisition. Particularly, we use a dataset that were acquired with four different cameras, where two of them are used in general scenarios and the rest to visualize closer ones.

For the focal length α, we consider an interval from 160 to 520 pixels, dividing it into 14 Gaussians with standard deviation of 7.5 pixels and a separation between means of 30 pixels. In a similar manner, we do to model the distortion parameters k_1 and k_2, considering for the first one an interval from 0.002 to 0.008 mm^{-2} into 3 Gaussians with standard deviation of 0.0008 mm^{-2} and a separation between means of 0.003 mm^{-2}, and an interval from 0.0001 to 0.0003 mm^{-4} into 2 Gaussians with standard deviation of 0.00005 mm^{-4} and a separation between peaks of 0.0002 mm^{-4} for the second one. For the sensors that observed general scenarios, the distortion parameters are considerably bigger and we scaled them maintaining the same number of filters. The corresponding probability density function for everyone of these parameters is displayed in Fig. 3, showing in the top

Param. \ Data	Non-Critical Motion Sequences								Critical Motion Sequences					
	Indoor		Loop Closing		Silicone Cloth		Laparoscopy		Pure Rotation		Pure Translation		Parallel Optical Axis	
	Off-line	On-line (Ours)	Off-line	On-line (Ours)	Off-line	On-line (Ours)	Off-line	On-line (Ours)	Off-line	On-line (Ours)	Off-line	On-line (Ours)	Off-line	On-line (Ours)
α[pixels]	194.10	195.24±1.27	196.90	196.97±0.53	312.89	309.30±0.30	280.91	274.36±0.32	194.10	211.65±12.90	194.10	204.14±4.44	194.10	202.84±8.31
β_x[pixels]	160.20	158.94±0.92	153.50	159.14±1.41	157.66	158.60±0.11	184.48	166.00±0.18	160.20	158.68±6.07	160.20	156.47±3.48	160.20	158.89±7.08
β_y[pixels]	128.90	128.85±0.99	130.80	131.22±1.19	121.32	119.21±0.11	133.48	136.06±0.17	128.90	121.48±6.92	128.90	129.14±3.13	128.90	116.51±5.98
k_1[mm^{-2}]	.0623	.0661±.0023	.0693	.0721±.0028	.0094	.0056±.0002	.0054	.0078±.0004	.0623	.0626±.0073	.0623	.0676±.0048	.0623	.0679±.0109
k_2[mm^{-4}]	.0139	.0122±.0008	.0109	.0107±.0007	.00011	.00036±.00003	.00026	.0004±.00004	.0139	.0098±.0024	.0139	.0088±.0015	.0139	.0121±.0032

TABLE II

CAMERA SELF-CALIBRATION QUANTITATIVE EVALUATION. THE TABLE REPORTS THE CALIBRATION RESULTS BY USING AN OFF-LINE APPROACH (A CALIBRATION PATTERN IS NEEDED) BASED ON NON-LINEAR OPTIMIZATION; AS WELL AS OUR ON-LINE ESTIMATION FOR BOTH NON-CRITICAL AND CRITICAL MOTION SEQUENCES. IN ADDITION, OUR SOLUTION INCLUDES AN 95% UNCERTAINTY ESTIMATION.

the filter for sensors in general scenarios and in the bottom the rest. As in real cases k_1 has a bigger influence in the solution, we use more filters than to model k_2. Finally, as the measurement equation is linear with respect to the optical center $\{\beta_x, \beta_y\}$, both parameters are modeled by one single Gaussian, whose mean is located in the middle of the image with standard deviations of 3.3 pixels in every coordinate. Considering altogether, the final SOG filter is composed of 84 filters. We also fix $\psi_a = 0.01$ and $\psi_b = 0.05$ for all experiments. We next evaluate our approach on real videos where rigid and both local and global deformations appear, as well as including some critical motion sequences where the auto-calibration is ambiguous.

A. Non-Critical Motion Sequences

First, we use four non-critical motion sequences denoted as: *Indoor*, *Loop Closing*, *Silicone Cloth*, and *Laparoscopy*. The first two were taken with a hand-held 320 $\times$ 240 IEEE1394 camera in general indoor conditions, representing one of them a challenging indoor loop closing scenario [16]. The third were provided by [5], where a hand-held camera with resolution 320 $\times$ 240 is observing an elastic silicone cloth fixed to a circular stretcher. The last one is a *laparoscopy* sequence provided by [8], where a rabbit abdominal cavity is observed by a hand-held endoscope. The sequence consists in 400 frames of resolution 288 $\times$ 384, and contains a combination of sudden camera motions and strong deformations that often washes out the observations. Before acquiring the video sequences, another video was recorded with a calibration pattern that was used to compute an off-line calibration by non-linear optimization where the matches were provided by hand, and we will use for validation. In contrast, data association was automatically solved in our on-line approach as was discussed in Sec. IV-B, being the 2D ellipses we obtain represented in Figs. 4-5-8-top along with some input images.

Although our algorithm provides one estimation per frame, the calibration becomes to converge as the frames arrive, especially when the number of filters is reduced to one (this usually happens after processing between 50 and 100 images). Table II reports a quantitative comparison between our on-line estimation and the off-line baseline. As it can be seen, our results for the four cameras are very accurate even computing automatically the correspondences. In the *Indoor* sequence we can observe a correlation between the estimations of k_1 and k_2, showing the difficulty to recover these values. This can be also seen in the *Silicone Cloth* and *Laparoscopy* sequences, where the estimation of these values is very challenging due to the strong deformations. It is worth noting that the estimation of β_x is worse in the *Loop Closing* sequence, since a *cycle-torsion* motion is marginal in this case. In any case, despite being some complex scenarios to be measured, our solution is competitive with respect to off-line approaches while it estimates the rest of parameters. Additionally, our approach can also estimate both camera trajectory and shape structure. A general estimation of 3D shape and motion is displayed in Fig. 4 for general indoor scenarios. As it can be seen, our algorithm recovers properly the loop trajectory in the *Loop Closing* sequence.

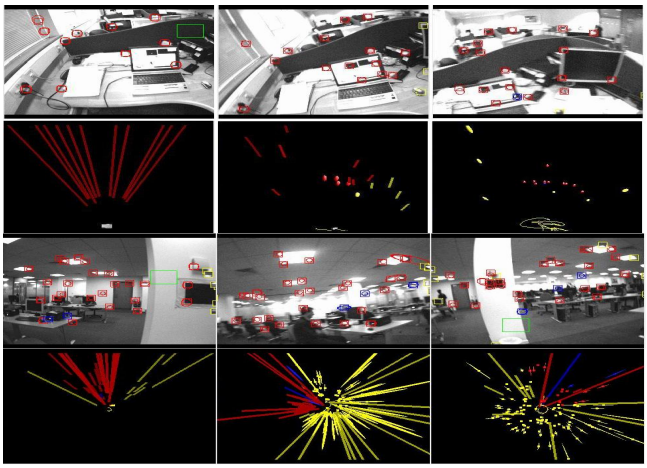

Fig. 4. **On-line camera self-calibration, shape and motion for *Indoor* and *Loop Closing* sequences.** In both cases, the same information is displayed. **Top:** Some input images and estimated location of the points of interest, with their associated uncertainty (red, blue and yellow ellipses mean matched, non-matched and non-visible points, respectively). **Bottom:** Global representation of 3D shape and camera trajectory (in yellow). As it can be seen, when a point is observed in just one frame or with low parallax, its uncertainty is big (see left part in both cases).

Figure 5-middle/bottom displays the corresponding 3D reconstruction results for three images in the *Silicone Cloth* sequence (a ground truth from stereo vision is available for these frames). To improve visualization, we also provide two cross sectional views where we represent our estimation with uncertainty, and the corresponding ground truth. If we compare the mean of our uncertainty distributions with the ground truth, we obtain a mean 3D reconstruction error of 3.96 mm, which is very accurate considering the diameter of the silicone cloth, around 200 mm, and the distance to the camera, around 700 mm. In any case, this error is slightly greater than the 2.5 mm reconstruction error reported by EKF-FEM [8], being, though, compatible with the removal of priors since EKF-FEM needs a

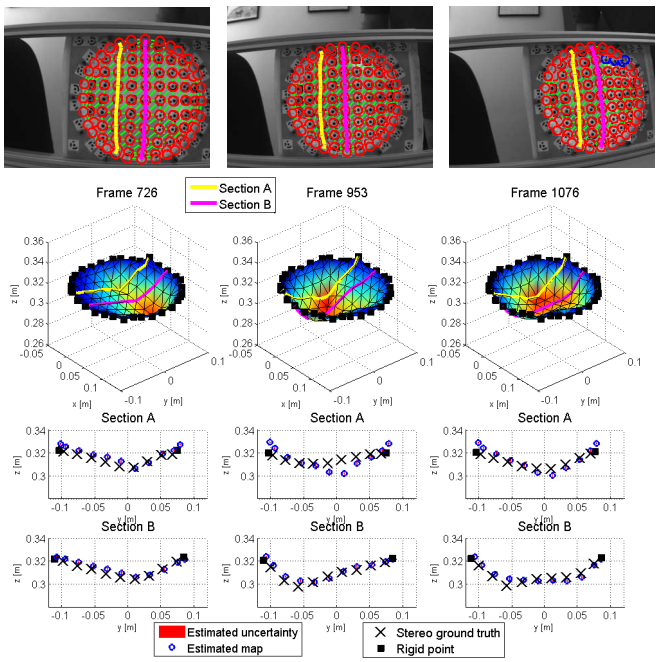

Fig. 5. **3D reconstruction on the *Silicone Cloth* sequence. Top**: Input frames and estimated location of the points of interest, with their associated uncertainty (red and blue ellipses mean matched and non-matched points, respectively). Note that in some frames, a few points could not be matched due to lack of visibility or some deformations. The color lines indicate cross-sectional views that will be represented below. **Middle**: General view of our 3D reconstruction shape. The degree of extensibility of the mesh, compared to the rest shape, is color-coded. Bluish regions are isometrically deformed, while reddish areas have undergone larger elastic deformations. **Bottom**: Two cross sections of the reconstructed shape, in which we show, for every 3D point, both the estimated location (blue circles) with its 95% confidence region (red ellipses), and the corresponding ground truth (black crosses).

calibrated camera instead of estimating it as we do here. Fig. 6 shows the 3D camera trajectory estimation, highlighting the camera location in the three frames considered above. Finally, we consider the *Laparoscopy* sequence. Following [8], we select the points located far apart from the deformed region to be assumed as rigid. This is a very challenging scenario, where a self-calibrated camera is mandatory in practice. Figure 7 shows some input images together with the feature points that we use to solve the problem, as well as the corresponding 3D estimation we obtain.

B. Critical Motion Sequences

We now consider three 600-frame critical motion sequences where it is not possible to completely determine the internal parameters [21], and they were acquired with the IEEE1394 camera. The sequences are: 1) *Pure Rotation* where a fixed camera is rotating over its optical center, i.e., without assuming any translation, 2) *Pure Translation over Optical axis* where the camera moves along the optical axis (the successive optical centers are co-linear), and 3) *Parallel Optical Axis* where the camera follows a linear trajectory while the optical-axis orientation remains, i.e., the successive optical axes are parallel. Our

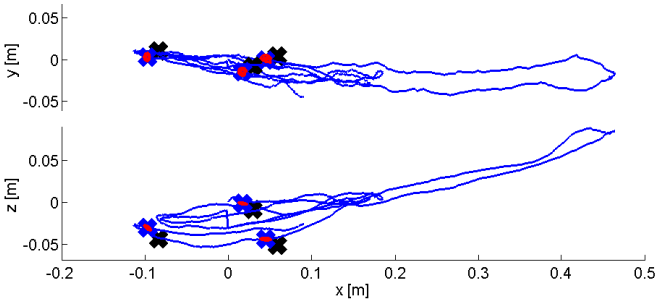

Fig. 6. **Camera trajectory estimation on the *Silicone Cloth* sequence.** Two views, (X-Y) and (X-Z), are displayed. In both cases, it is highlighted the camera location with a 95% confidence for three selected camera locations (see the corresponding shape estimation in Fig. 5) by means of blue crosses, and the ground truth by black ones.

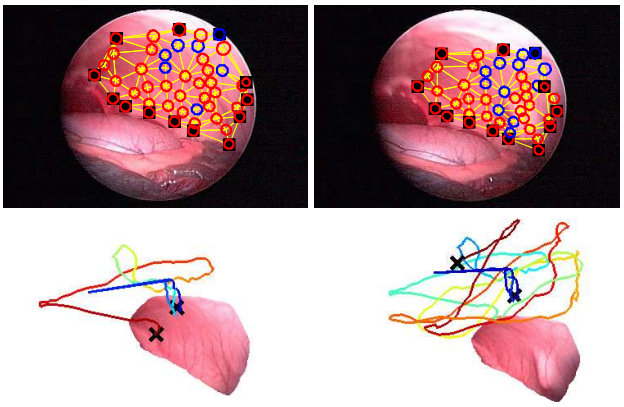

Fig. 7. **Joint 3D reconstruction and trajectory on the *Laparoscopy* sequence. Top:** Two images with the 2D estimated mesh. Matched and non-matched feature points are represented by red and blue ellipses, respectively. **Bottom**: An overall estimation.

calibration results are shown in table II, and the corresponding 3D shape and motion in Fig. 8. For *Pure Rotation*, the 3D reconstruction is not theoretically possible due to lack of parallax, but in practice, due to noisy observations, our algorithm produces a wrong estimation (see the green points in Fig. 8-top, where two arbitrary points with very different depths are estimated quite close). For *Pure Translation over Optical axis* and *Parallel Optical Axis* sequences, as it is confirmed by our results, the principal point $\{\beta_x, \beta_y\}$ cannot be estimated properly, obtaining estimations with bigger uncertainties with respect to non-critical motions. As a consequence, the rest of uncertainty estimations are also bigger, even though the camera trajectories seem to be visually correct.

VI. Conclusion

In this paper we have proposed the first algorithm to jointly retrieve camera self-calibration, camera motion, and the 3D reconstruction of a non-rigid object, all of them, from a monocular video and without assuming any training data at all. To this end, we have embedded a physical model to encode deformations, solved by means of finite elements, along with a dynamic model to code the camera parameters. These ingredients are exploited in a SOG filter that is composed of a bank of EKF filters, being them Gaussianly combined

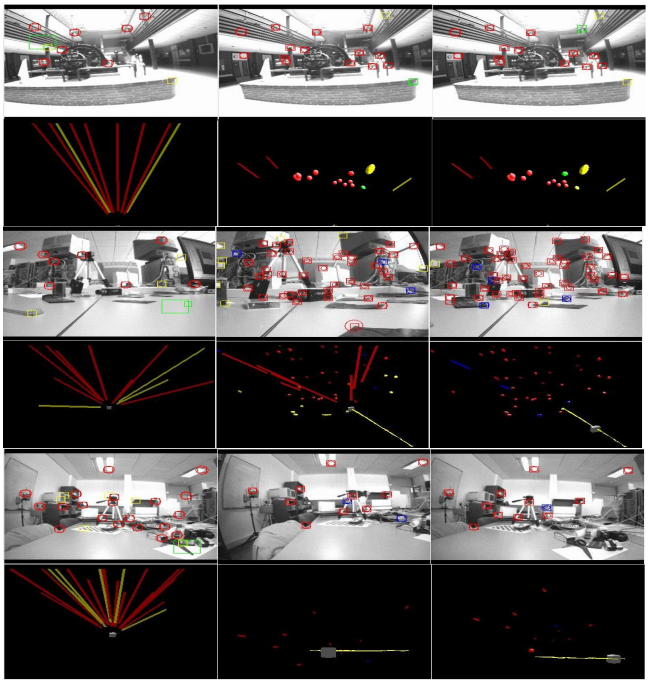

Fig. 8. **Critical motion sequences: *Pure Rotation*, *Pure Translation* and *Parallel Optical axis*, respectively.** See caption in Fig. 4.

to provide a Bayesian estimation. Thanks to our model, we can handle a large variety of deformations without knowing their material properties. We have experimentally evaluated our approach on real sequences, for rigid, isometric and elastic deformations. While our approach provides joint competitive 3D reconstructions and motion estimation in comparison with competing techniques, it can calibrate the camera from scratch without considering any calibration pattern. An interesting avenue for future research is to validate our formulation in real time at frame rate.

Acknowledgment: This work has been partially supported by the UPC project MESSI MdM-IP-2019-01, the Spanish Ministry of Science and Innovation under project HuMoUR TIN2017-90086-R, and by the Spanish State Research Agency through the MDM Seal of Excellence to IRI MDM-2016-0656.

REFERENCES

[1] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski, "Building Rome in a day," in *ICCV*, 2009.
[2] A. Agudo, L. Agapito, B. Calvo, and J. M. M. Montiel, "Good vibrations: A modal analysis approach for sequential non-rigid structure from motion," in *CVPR*, 2014.
[3] A. Agudo, B. Calvo, and J. M. M. Montiel, "FEM models to code non-rigid EKF monocular SLAM," in *ICCVW*, 2011.
[4] ——, "3D reconstruction of non-rigid surfaces in real-time using wedge elements," in *ECCVW*, 2012.
[5] ——, "Finite element based sequential bayesian non-rigid structure from motion," in *CVPR*, 2012.
[6] A. Agudo and F. Moreno-Noguer, "Learning shape, motion and elastic models in force space," in *ICCV*, 2015.
[7] A. Agudo, F. Moreno-Noguer, B. Calvo, and J. M. M. Montiel, "Real-time 3D reconstruction of non-rigid shapes from single moving camera," *CVIU*, vol. 153, no. 12, pp. 37–54, 2016.
[8] ——, "Sequential non-rigid structure from motion using physical priors," *TPAMI*, vol. 38, no. 5, pp. 979–994, 2016.
[9] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade, "Trajectory space: A dual representation for nonrigid structure from motion," *TPAMI*, vol. 33, no. 7, pp. 1442–1456, 2011.
[10] D. Alspach and H. Sorenson, "Sequential tests of statistical hypothesis," *The Annals of Mathematical Statistics*, vol. 16, no. 2, pp. 117–186, 1945.
[11] ——, "Nonlinear bayesian estimation using gaussian sum approximation," *TAC*, vol. 17, no. 4, pp. 439–448, 1972.
[12] A. Azarbayejani and A. P. Pentland, "Recursive estimation of motion, structure, and focal length," *TPAMI*, vol. 17, no. 6, pp. 562–575, 1995.
[13] A. Bartoli, V. Gay-Bellile, U. Castellani, J. Peyras, S. Olsen, and P. Sayd, "Coarse-to-fine low-rank structure-from-motion," in *CVPR*, 2008.
[14] A. Bartoli, D. Pizarro, and T. Collins, "A robust analytical solution to isometric shape-from-template with focal length calibration," in *ICCV*, 2013.
[15] M. Brubaker, L. Sigal, and D. Fleet, "Estimating contact dynamics," in *ICCV*, 2009.
[16] J. Civera, D. R. Bueno, A. J. Davison, and J. Montiel, "Camera self-calibration for sequential bayesian structure from motion," in *ICRA*, 2009.
[17] Y. Dai, H. Li, and M. He, "A simple prior-free method for non-rigid structure from motion factorization," in *CVPR*, 2012.
[18] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," *TPAMI*, vol. 29, no. 6, pp. 1052–1067, 2007.
[19] P. F. U. Gotardo and A. M. Martinez, "Computing smooth time-trajectories for camera and deformable shape in structure from motion with occlusion," *TPAMI*, vol. 33, no. 10, pp. 2051–2065, 2011.
[20] N. Haouchine and S. Cotin, "Template-based monocular 3D recovery of elastic shapes using lagrangian multipliers," in *CVPR*, 2017.
[21] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.
[22] N. Keivan and G. Sibley, "Online SLAM with any-time self-calibration and automatic change detection," in *ICRA*, 2017.
[23] N. Kwok, G. Dissanayake, and Q. Ha, "Bearing-only SLAM using a SPRT based gaussian sum filter," in *ICRA*, 2005.
[24] M. Lee, J. Cho, C. H. Choi, and S. Oh, "Procrustean normal distribution for non-rigid structure from motion," in *CVPR*, 2013.
[25] M. Lee, J. Cho, and S. Oh, "Consensus of non-rigid reconstructions," in *CVPR*, 2016.
[26] A. Malti, A. Bartoli, and R. Hartley, "A linear least-squares solution to elastic shape-from-template," in *CVPR*, 2015.
[27] E. M. Mikhail, J. S. Bethel, and J. C. McGlone, *Introduction to Modern Photogrammetry*. John Wiley & Sons, 2001.
[28] J. M. M. Montiel, J. Civera, and A. J. Davison, "Unified inverse depth parametrization for monocular SLAM," in *RSS*, 2006.
[29] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd, "Generic and real-time structure from motion using local bundle adjustment," *IMAVIS*, vol. 27, no. 8, pp. 1178–1193, 2009.
[30] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: A versatile and accurate monocular SLAM system," *TRO*, vol. 31, no. 5, pp. 1147–1163, 2015.
[31] J. Ostlund, A. Varol, and P. Fua, "Laplacian meshes for monocular 3D shape recovery," in *ECCV*, 2012.
[32] M. Paladini, A. Bartoli, and L. Agapito, "Sequential non rigid structure from motion with the 3D implicit low rank shape model," in *ECCV*, 2010.
[33] S. Parashar, A. Bartoli, and D. Pizarro, "Self-calibrating isometric non-rigid structure-from-motion," in *ECCV*, 2018.
[34] G. Qian and R. Chellappa, "Bayesian self-calibration of a moving camera," *CVIU*, vol. 95, no. 3, pp. 287–316, 2004.
[35] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *ECCV*, 2006.
[36] M. Salzmann and P. Fua, "Reconstructing sharply folding surfaces: A convex formulation," in *CVPR*, 2009.
[37] P. Tanskanen, T. Naegeli, M. Pollefeys, and O. Hilliges, "Semi-direct EKF-based monocular visual-inertial odometry," in *IROS*, 2015.
[38] L. Torresani, A. Hertzmann, and C. Bregler, "Nonrigid structure-from-motion: estimating shape and motion with hierarchical priors," *TPAMI*, vol. 30, no. 5, pp. 878–892, 2008.
[39] S. Vicente and L. Agapito, "Soft inextensibility constraints for template-free non-rigid reconstruction," in *ECCV*, 2012.
[40] Y. Zhu, D. Huang, F. de la Torre, and S. Lucey, "Complex non-rigid motion 3D reconstruction by union of subspaces," in *CVPR*, 2014.