IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2023, London, UK. Cite as RA-L paper.

Direct and Sparse Deformable Tracking

José Lamarca[1], Juan J. Gómez Rodríguez[1], Juan D. Tardós[1], *Fellow, IEEE,* and J.M.M. Montiel[1], *Member, IEEE*

***Abstract*—Deformable Monocular SLAM algorithms recover the localization of a camera in an unknown deformable environment. Current approaches use a template-based deformable tracking to recover the camera pose and the deformation of the map. These template-based methods use an underlying global deformation model. In this paper, we introduce a novel deformable camera tracking method with a local deformation model for each point. Each map point is defined as a single textured surfel that moves independently of the other map points. Thanks to a direct photometric error cost function, we can track the position and orientation of the surfel without an explicit global deformation model. In our experiments, we validate the proposed system and observe that our local deformation model estimates more accurately the targeted deformations of the map in both laboratory-controlled experiments and in-body scenarios undergoing quasi-isometric deformations, with changing topology or discontinuities.**

***Index Terms*—SLAM; Localization; Computer Vision for Medical Robotics**

I. Introduction

VSLAM (Simultanenous Localization And Mapping from Visual sensors) is becoming a mature technology to navigate in human-made environments, being crucial for technologies like augmented reality and autonomous robot operation. Current state-of-the-art VSLAM algorithms [1], [2], [3] strongly rely on scene rigidity. As a consequence, they perform poorly in deforming scenes, e.g. in medical environments.

Since PTAM [4], VSLAM algorithms divide the computation in a tracking and a mapping concurrent threads. The tracking thread computes the camera position *wrt.* the map at frame-rate. In parallel, the mapping thread recovers the structure of the scene with a higher computational cost from some selected frames, so-called keyframes. In the deformable case, both DefSLAM [5] and SD-DefSLAM [6] use a deformable mapping based on a Non-Rigid Structure-from-Motion (NRSfM) [7] to recover the structure of the scene at keyframe rate, and a deformable tracking [8] that estimates simultaneously the camera pose and the deformation of the map for every frame.

The deformable tracking of these previous methods relies on the usage of a mesh that embeds the map points, and it recovers the most likely shape of the mesh according to a deformation model. This deformation model is global i.e. each map point is connected with their neighbours. This shows excellent performance in scenes with a single surface where all the points are indeed connected.

Manuscript received: May, 21, 2022; Revised: July, 20, 2022; Accepted: August, 3, 2022.

This paper was recommended for publication by Editor Sven Behnke upon evaluation of the Associate Editor and Reviewers' comments.

This work was supported by the EU-H2020 grant 863146: ENDOMAPPER, the Spanish government grants PGC2018-096367-B-I00, DPI2017-91104-EXP and the MINECO scholarship BES-2016-078678, and by Aragón government grant DGA_T45-17R.

[1]The authors are with the Instituto de Investigación en Ingeniería de Aragón (I3A), Universidad de Zaragoza, María de Luna 1, 50018 Zaragoza, Spain. E-mail: {jlamarca, jjgomez, tardos, josemari}@unizar.es.

Digital Object Identifier (DOI): see top of this page.

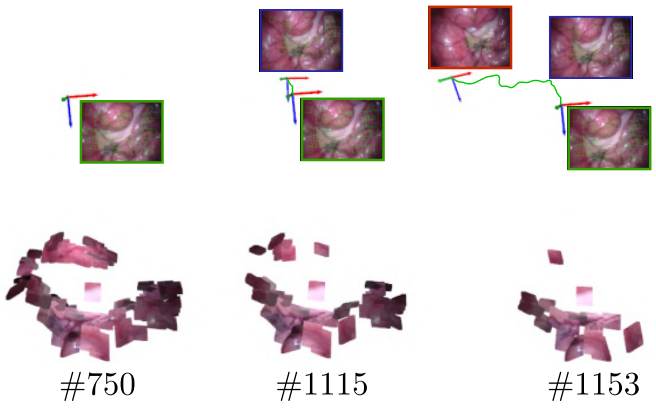


Fig. 1. Direct and Sparse Deformable Tracking processing Hamlyn Dataset sequence 20, results after frames #750, #1115 and #1153. Top: Camera Trajectory in green. Bottom: The map composed of sparse surfels.

However, when points are not connected, like in scenes with several surfaces, non-isometric surfaces, or with topological changes, the global model does not represent properly the deformation of the map, yielding low performance.

In this paper, we propose a novel deformable tracking method that uses local deformation models to treat the map points as independent bodies. Our first contribution is to model the map as a sparse set of 3D moving textured surfels observed by a moving perspective camera. Each surfel is assumed to have independent rigid displacements from the other surfels around its position at rest. The formulation of the surfel is a first-order Taylor approximation of the map point. The main advantage of this approach is that any smooth surface, e.g. cylinders, planes, spheres or discontinuous surfaces, can be represented locally by a plane, independently of its topology.

Our second contribution is to use a direct photometric error resulting from back-projecting the surfel texture. We jointly optimize the 3D position and orientation of the surface to minimize the direct photometric error. In contrast to previous approaches, in our proposed direct deformable tracking there is no hard data association, instead, the final matching is a byproduct of the photometric alignment.

In our experimental section, we prove that our method can deal with discontinuous surfaces and topological changes, and achieves better performance than the tracking methods used in current Deformable Monocular SLAM systems [5], [6], obtaining longer tracks with better geometrical accuracy in medical sequences.

Next, in Sec.II, we discuss in detail the related works in non-rigid reconstruction and VSLAM. In Sec.III, we present our formulation for the surfel. In Sec.IV, we develop our deformable tracking with fixed camera to prove the potential of surfel tracking adapting to different surfaces. In contrast to the previous methods, we propose a fully direct and sparse approach able to recompute the matches during the optimization. In Sec.V, we formulate a world-centric direct deformable tracking to estimate the pose of the camera based on a equilibrium regularizer. Finally, in the last Sec.VI, the results

obtained show a considerable improvement *wrt.* the previous deformable tracking methods both in terms of robustness and accuracy.

II. Related work

Deformable SLAM problem consists in reconstructing a map whose shape is constantly deforming and recovering the camera trajectory *wrt.* the reconstructed map.

The first deformable SLAM method proposed was DynamicFusion [9]. This method proposes a pipeline where the entire shape of map was reconstructed from partial RGB-D observations from different positions. MISSLAM [10] transferred this technique to medical scenarios by using stereo pairs. Concerning monocular SLAM, the lack of depth information significantly entangles the reconstruction problem. The first work to solve Deformable SLAM with monocular cameras was DefSLAM [5]. Like other monocular SLAM systems [1], [2], [3], DefSLAM is composed of two main threads: deformable tracking and mapping. These two components are based on the two main families of non-rigid monocular methods: Non-Rigid Structure-from-Motion for mapping, and template-based techniques for tracking.

The first approaches of NRSfM were formulated using statistical models [11], [12], [13]. A low dimensional basis model is used to obtain the configuration of the 3D points for several images. The problem has been formulated with different regularizers, e.g. spatial [12], [14], temporal [13], or spatio-temporal [15]. The main weakness of these methods is the assumption of orthographic camera model, not suitable for VSLAM due to the noticeable perspective effects in many targeted scenes where close-ups are dominant. Recent geometric methods have been proved to work with perspective cameras under the assumption of local isometry in the surface [16], [17], [7], [18], [19]. The method proposed in [7] was the base of the deformable mapping in [5] due to its ability to naturally handle occlusions and missing data.

Template-based techniques recover the deformation of the scene from a single-image relying in a known textured surface and a deformation model. The 3D shape at rest of the textured surface is the so-called template. In the deformable SLAM approaches, the template is used to estimate the deformation of the map during tracking. The main difference between these methods is the representation of the surface and its deformation model. Among the analytic solutions, one of the most extended assumptions is that the surface is isometric. In other words, the geodesic distance between points in the surface is preserved during the entire sequence. Isometry for shape-from-template –SfT– has been proven to be well-posed and to quickly evolve to stable and real-time analytical solutions [20], [21], [22]. On the other hand, energy-based methods [23] are numerical approaches that jointly minimize the shape deformation energy *wrt.* the shape-at-rest and the reprojection error for the current image correspondences. These optimization methods are well suited to implement sequential data association with robust kernels to deal with outliers.

The mentioned methods consider the camera static and usually reconstruct small objects that move in the camera field of view. The deformable tracking methods estimate the camera pose in addition to the deformation of the map. Usually, this is done by constraining the problem with boundary conditions [24], [8]. The deformable tracking for deformable monocular SLAM [5], [6] was built on top of [8], in which the tracking represents the map as a continuous triangular mesh with a global deformation model that penalizes stretching and bending. This global model does not allow topological changes or discontinuous surfaces. In contrast, we propose a local model for the deformable tracking.

In this paper, we formulate a deformable tracking that represents the map points of the surface separately as unconnected surfels -surface element- and jointly estimate the surfel's pose and deformation for each frame and the pose of the camera. Each surfel has a local deformation model being able to represent more general disconnected shapes of the scene and movements, and avoid the usage of a global deformation model. One of the closest approach was the scene flow technique proposed in [25], that uses surfels to track some points of the scene, however they rely in a multi-camera setup, while we use a monocular camera.

Piecewise methods are local techniques where the non-rigid object is a collection of pre-defined patches that move independently as rigid objects. The first work in using this strategy was [26], imposing a 3D global consistency in overlapping points. A relaxation to the piecewise rigid constraint was given by [27], assuming each patch deforms with a quadratic physical model accounting for linear and bending deformations. All these methods required an initial patch segmentation and the number of overlapping points, to this end [28] optimize the number of patches and overlapping through an energy-based optimization. In contrast, [18] constructs a triangular mesh, connecting all the points, and considering each triangle as being locally rigid, being able to deal with topological changes. Our method belongs to this family of methods, but in contrast, we do not assume that the points are overlapping.

Our previous work [6] is a semi-direct method that replaces the feature-based tracking of [5] with a multiscale Lucas-Kanade tracker, resulting in an improvement of the track lengths and reconstruction accuracy. In this work, we take advantage of direct photometric error to recover the 3D relative position of the surface points. Direct methods use the photometric error and have been proven extremely accurate in the rigid SLAM case [2], [3] and other NRSfM works like [29].

III. Formulation

This section is devoted to formalize the parametrization of a surfel and the photometric equations describing its observation by a projective camera.

A. Notation

Bold letters represent vectors or matrices ($\mathbf{J}$, $\mathbf{X}$). Scalars will be represented by light lowercase letters (u), image brightness functions by light uppercase letters (I). Superindex t denotes the frame in which the estimation is done. Subindex i identifies the surfel. Subindex p refers to pixel coordinates in reference local to the surfel. To simplify the index notation all the scene points coordinates are in the world reference. Camera poses are represented as transformation matrices $\mathbf{T}_{cw} \in \mathbf{SE}(3)$, transforming the coordinates of point from the world frame into the camera frame.

B. Surfel parametrization

Assuming a continue and derivable C^1 surface, a point $\mathbf{X}_i^t$ is represented by a surfel $\mathbf{S}_i^t$ contained in the tangent space of the surface

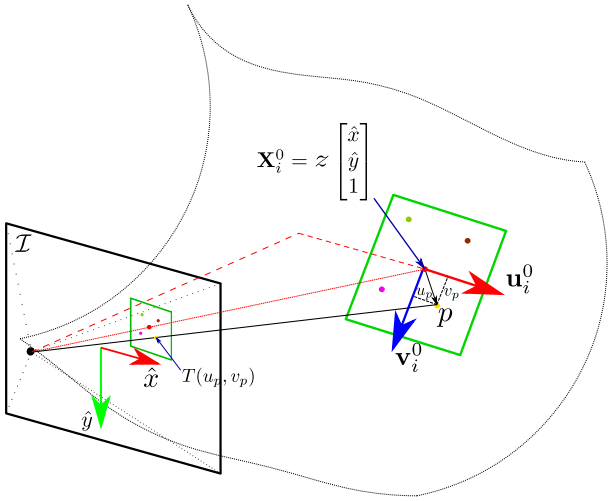


Fig. 2. Parametrization of a surfel in the initial image. Coordinates of the surface u and v correspond to the normalized coordinates in the image $\hat{x}$ and $\hat{y}$. We obtain z from the depth image, and we estimate the tangent space vectors $\mathbf{u}_i^t$ and $\mathbf{v}_i^t$ as the directional derivatives in the image coordinates $\mathcal{I}$.

at the point. Thus, a generic 3D point p belonging to the surfel can be parametrized using two local coordinates u_p and v_p around $\mathbf{X}_i^t$:

$$\mathbf{S}_i^t(u_p,v_p) = \mathbf{X}_i^t + \mathbf{J}_i^t \begin{bmatrix} u_p \\ v_p \end{bmatrix} \tag{1}$$

$$\mathbf{J}_i^t \quad = \quad \begin{bmatrix} \mathbf{u}_i^t & \mathbf{v}_i^t \end{bmatrix} \in \mathbb{R}^{3\times 2} \tag{2}$$

where $\mathbf{J}_i^t$ is the so-called Jacobian matrix whose columns are a pair of vectors forming a base of the tangent space. As described in Eq. (6), $\mathbf{X}_i^t$ and $\mathbf{J}_i^t$ are defined for each frame in terms of the corresponding values at $t=0$, $\mathbf{X}_i^0$ and $\mathbf{J}_i^0$, whose initialization from the first image is described next.

C. Surfel initialization

We assume the scene surface is defined by means of the depth function: $z(\hat{x},\hat{y}) : \mathbb{R}^2 \to \mathbb{R}$ in terms of the normalized retina coordinates $\hat{x},\hat{y}$. This depth function can be provided by a depth sensor (RGB-D camera or stereo rig). Then, $\mathbf{X}_i^0$ and $\mathbf{J}_i^0$ are estimated as:

$$\mathbf{X}_i^0 = z(\hat{x},\hat{y}) \begin{bmatrix} \hat{x} \\ \hat{y} \\ 1 \end{bmatrix} \tag{3}$$

and

$$\mathbf{J}_i^0 = \begin{bmatrix} z + \hat{x}\frac{\partial z}{\partial \hat{x}} & \hat{x}\frac{\partial z}{\partial \hat{y}} \\ \hat{y}\frac{\partial z}{\partial \hat{x}} & z + \hat{y}\frac{\partial z}{\partial \hat{y}} \\ \frac{\partial z}{\partial \hat{x}} & \frac{\partial z}{\partial \hat{y}} \end{bmatrix} \tag{4}$$

For the experiments, we initialize surfels in the interest points extracted with Shi-Tomasi [30].

D. Photometric error

We denote the projection function as $\pi(\cdot) : \mathbb{R}^3 \to \mathbb{R}^2$. For our experiments, we use the pinhole camera model. Note that this can be easily substituted by any other camera model.

We optimize the difference between the intensities of points in the surfel and the intensities in their reprojections in the current image:

TABLE I
DEFORMATION TENSOR F_i^t FOR DIFFERENT LOCAL DEFORMATION MODELS.

	Isometry	Conformal	Equiareal	General
F_i^t	$\mathbb{I}_2$	$s\mathbb{I}_2$	$\begin{bmatrix} \alpha & \beta \\ \beta & \frac{1+\beta}{\alpha} \end{bmatrix}$	$\begin{bmatrix} \alpha & \beta \\ \beta & \gamma \end{bmatrix}$
Variables	-	s	α,β	α,β,γ

$$\mathcal{P}_i^t = \sum_p \left(\alpha_i^t I^t\left(\pi\left(\mathbf{T}_{cw}\mathbf{S}_i^t(u_p,v_p)\right)\right) + \beta_i^t - T(u_p,v_p)\right)^2 \tag{5}$$

where $\mathbf{T}_{cw}$ is the pose of the camera with respect to the world. $\mathbf{S}_i^t(u_p,v_p)$ is in the world reference. We compensate the illumination changes by means of a gain (α_i^t) and a bias (β_i^t) per surfel and per image. That allows us to synthesize the deformed surfel into the image, thus our error function takes into account the local deformation.

We define a symmetric uniform grid in the surfel local coordinates that is reprojected into the inital image to extract the surfel texture $T(u_p,v_p)$ parameterized by u_p and v_p.

IV. DIRECT AND SPARSE CAMERA TRACKING WITH STATIC CAMERA

Let's assume in this section that the camera is fixed and the initial values of the textured surfels are given in advance, and we want to estimate the deformation for each incoming image. With our formulation, the initial surfel is defined by its initial position $\mathbf{X}_i^0$, its Jacobian $\mathbf{J}_i^0$ and its texture $T(u_p,v_p)$.

The geometrical transformation of the surfel is expressed as:

$$\mathbf{S}_i^t(u_p,v_p) = (\mathbf{X}_i^0 + \mathbf{t}_i^t) + \mathbf{R}_i^t \mathbf{J}_i^0 \mathbf{F}_i^t \begin{bmatrix} u_p \\ v_p \end{bmatrix} \tag{6}$$

where $\mathbf{t}_i^t \in \mathbb{R}^3$ is the translation of the surfel, $\mathbf{R}_i^t \in \mathbf{SO}(3)$ is the rotation of the surfel modeled by the 3 parameters of its Lie algebra, and $\mathbf{F}_i^t \in \mathbb{R}^{2\times 2}$ is a symmetric matrix that represent the deformation tensor. Its diagonal components represent the stretching of the tangent vectors, and its off-diagonal element models the angle change between these vectors, i.e. the shearing.

As seen in Table I, the most restrictive local deformation is isometric. This constraint is equivalent to a rigid movement of the surfel. When the surfel deformation is not bounded the first ambiguity that we focus on arises:

Growing map ambiguity. The depth component of the translation of the surfel and the surfel size can be coupled in such a way that changing its depth and size produces the same image.

Proof. We define an μ factor that transforms the position and the deformation of the surfel as:

$$(\mathbf{X}_i^0 + \mathbf{t}_i^t) \quad = \quad \mu \mathbf{X}_i^0 \tag{7}$$

$$\mathbf{F}_i^t \quad = \quad \begin{bmatrix} \mu & 0 \\ 0 & \mu \end{bmatrix} \tag{8}$$

$$\hat{\mathbf{S}}_i^t(u_p,v_p) \quad = \quad \mu \mathbf{X}_i^0 + \mu \mathbf{R}_i^t \mathbf{J}_i^0 \begin{bmatrix} u_p \\ v_p \end{bmatrix} = \mu \mathbf{S}_i^t(u_p,v_p) \tag{9}$$

Under perspective projection any surfel $\hat{\mathbf{S}}_i^t(u_p,v_p)$ multiplied by μ produces the same image $\pi(\mathbf{S}_i^t(u_p,v_p)) = \pi(\hat{\mathbf{S}}_i^t(u_p,v_p))$. □

To solve this ambiguity, we impose local isometry within the surfel. Isometry is a distance preserving transformation. We propose two alternatives to code the isometry, as a hard constraint or as a soft constraint.

Isometry as hard constraint implies the transformation of the surfel only as a rigid body motion, in other words, the deformation matrix, $\mathbf{F}_i^t = \mathbb{I}_2$. The motion is defined by 6 parameters (3 for translation + 3 for rotation).

$$\mathbf{J}_i^t = \mathbf{R}_i^t \mathbf{J}_i^0 \ ; \ \mathbf{R}_i^t \in \mathbf{SO}(3) \tag{10}$$

Thus, our cost function is defined only by the photometric error (Eq. (5)):

$$\mathbf{t}_i^t, \mathbf{R}_i^t = \underset{\mathbf{t}_i^t, \mathbf{R}_i^t, \alpha_i, \beta_i}{\operatorname{argmin}} \ \mathcal{P}_i^t \tag{11}$$

In the case of a soft constrain we penalize the stretching and shearing of the surfel. It is formulated through the tangent plane $\mathbf{J}_i^t$. We define a deformation energy quadratic error as:

$$\mathcal{I}_i^t = \left\| \mathbf{F}_i^t - \mathbb{I}_2 \right\|_2^2 \tag{12}$$

The soft constraint is modeled by means of the deformation energy coded by 3 additional parameters defining the symmetric matrix $\mathbf{F}_i^t$. In other words, the surfel can stretch and shear if it explains better the image, but it tends to stay as close as possible to its original shape. $\mathcal{I}_i^t$ is a scalar that penalises the shearing and stretching.

Finally, the optimization is a combination of the forward-compositional photometric error and the deformation energy. The deformation energy regularization is weighted by a constant $\omega_{\mathcal{I}}$,

$$\mathbf{t}_i^t, \mathbf{R}_i^t, \mathbf{F}_i^t = \underset{\mathbf{t}_i^t, \mathbf{R}_i^t, \mathbf{F}_i^t, \alpha_i, \beta_i}{\operatorname{argmin}} \ \mathcal{P}_i^t + \omega_{\mathcal{I}} \mathcal{I}_i^t \tag{13}$$

All the errors considered in (11, 13) are quadratic, so it can be solved as a non-linear least-squares problem. We propose Levenberg–Marquardt (LM) optimization [31]. The LM algorithm is a trust-region method that combines a Gauss-Newton and steepest descend. The step control is defined through the damping factor λ that weights both methods, λ also allows to control the step size. The Hessian is approximated as $H \approx J^\top J$.

During the optimization, the data association between the images is changed boosting the accuracy, however the convergence basin of the photometric optimization is small. We propose an strict step size control to avoid leaving the convergence basin. We confine the step to an ellipsoidal trust region defined by the diagonal matrix $D_w = \operatorname{diag}(H)$. We apply a step policy where λ is limited to be ≥ 1 during the first steps to avoid long steps when far from the minimum. In a subsequent stage λ is allowed to be reduced in order to benefit from the Gauss-Newton quadratic convergence.

Due to the different units in translation, rotation and deformation components of the optimized vector $\delta\mathbf{x}$, we would have a poor conditioning of the H matrix. Thus, we propose to use a diagonal scaling preconditioner matrix $D_s(i,i) = \frac{1}{\sqrt{s_i}}$ to avoid numerical issues, being s_i the diagonal values of $(H + \lambda D_w)$. At each iteration the $\delta\mathbf{x}$ is then estimated as:

$$D_s(H + \lambda D_w) D_s \delta\mathbf{x}^* = -D_s J^\top \mathbf{r} \tag{14}$$

$$\delta\mathbf{x} = D_s \delta\mathbf{x}^* \tag{15}$$

Our experiments validate that this preconditioning is a key factor to achieve performance.

In addition, to avoid mismatches due to discontinuities and light reflections, we saturate the photometric error. We also carry out a multi-scale optimization to increase the convergence basin observing that in case of temporal discontinuities or fast movements the algorithm becomes much more robust.

We detect the outlier surfels using a threshold in the Zero Normalized Cross correlation (ZNCC) between the texture of the surfel and the texture of its reprojection because it is illumination invariant. If the ZNCC drops under a threshold the surfel can be assumed as badly tracked and the corresponding observation is marked as an outlier.

The algorithm complexity is linear in the number of points since each new point would suppose a new optimization and cubic in the number of pixels per surfel since the Jacobian block of the surfel is dense and it increases one row per new pixel included.

V. Direct and Sparse deformable tracking

Deformable tracking algorithm takes as input the textured surfels and the initial camera pose. Then, it estimates the deformation of the map and the camera pose *wrt.* the map.

Floating map ambiguity. Surfel position and camera pose are coupled and can be varied producing the same projection of the pixel in the image.

Proof. Eq. (6) corresponding to a static camera can be rewritten as:

$$\begin{bmatrix} \mathbf{S}_i^t(u_p, v_p) \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_i^t & \mathbf{X}_i^0 + \mathbf{t}_i^t \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{J}_i^0 \mathbf{F}_i^t \begin{bmatrix} u_p \\ v_p \end{bmatrix} \\ 1 \end{bmatrix} \tag{16}$$

$$= \mathbf{T}_{wi}^t \begin{bmatrix} \mathbf{J}_i^0 \mathbf{F}_i^t \begin{bmatrix} u_p \\ v_p \end{bmatrix} \\ 1 \end{bmatrix} \tag{17}$$

where $\mathbf{T}_{wi}^t \in \mathbf{SE}(3)$ represents the frame of the surfel at instant t with respect to world reference (the static camera pose). In case of a moving camera, the coordinates of the patch's pixels with respect to the camera will be:

$$\begin{bmatrix} {}^c\mathbf{S}_i^t(u_p, v_p) \\ 1 \end{bmatrix} = \mathbf{T}_{cw}^t \mathbf{T}_{wi}^t \begin{bmatrix} \mathbf{J}_i^0 \mathbf{F}_i^t \begin{bmatrix} u_p \\ v_p \end{bmatrix} \\ 1 \end{bmatrix} \tag{18}$$

Thus, a small perturbation of the camera pose $\mathbf{d}_c \in \mathfrak{se}(3)$ is indistinguishable from a small perturbation of the surfel pose $\mathbf{d}_i \in \mathfrak{se}(3)$, provided that:

$$\mathbf{d}_c \oplus (\mathbf{T}_{cw}^t \mathbf{T}_{wi}^t) = (\mathbf{T}_{cw}^t \mathbf{T}_{wi}^t) \oplus \mathbf{d}_i \tag{19}$$

where $\oplus$ represents the composition of a member of $\mathbf{SE}(3)$ and a member of $\mathfrak{se}(3)$ [32]. Both perturbations are related through the adjoint matrix:

$$\mathbf{d}_c = \mathrm{Ad}_{(\mathbf{T}_{cw}^t \mathbf{T}_{wi}^t)} \mathbf{d}_i \tag{20}$$

□

Therefore, the absolute movement of a surfel observed from the camera is coupled with the movement of the camera and it is not observable without further constrains.

To avoid the floating map ambiguity, additionally to the isometry constrain, we propose to soft-constrain each surfel position around an equilibrium position $\mathbf{X}_{e_i}$ with the regularizer $\mathcal{E}_i^t$:

$$\mathcal{E}_i^t = \left(\mathbf{X}_i^t - \mathbf{X}_{e_i}\right)^\top \Sigma_i^{-1} \left(\mathbf{X}_i^t - \mathbf{X}_{e_i}\right) \tag{21}$$

Constraining the pose of 3 or more non-aligned surfels, we constraint the 6 dof of the map rigid motion, providing a reference for the camera trajectory estimation. With this approach, the camera codes in its trajectory most of the relative rigid motion between camera and environment. We can understand the camera movement in our approach as the global rigid movement, and the deformation of the surfels as movements around that equilibrium. Σ_i is the covariance that the surfels can reach in their movement.

If the trajectory of the points along the sequence is known in advance, the equilibrium point can be estimated as their average position and its covariance. In the case that the position and covariance are unknown, we approximate $\mathbf{X}_{e_i}$ as the original position $\mathbf{X}_0^t$ and select a heuristic covariance with the expected movement. Lower covariances lead to more rigid interpretation.

Similarly to Sec. IV, it is possible code the isometry as a hard or as a soft constraint. The optimization for the hard constraint case is:

$$\mathbf{X}_i^t, \mathbf{J}_i^t, \mathbf{T}_{cw} = \underset{\mathbf{X}_i^t, \mathbf{J}_i^t, \alpha, \beta, \mathbf{T}_{cw}}{\operatorname{argmin}} \sum_{i \in \mathcal{X}} \mathcal{P}_i^t + \omega_\mathcal{E} \mathcal{E}_i^t \tag{22}$$

The movement of the camera is defined by using Lie algebra of $\mathbf{SE}(3)$. We linearize in the solution for each step and update the pose each step as:

$$\hat{\mathbf{T}}_{cw} = \exp(\zeta)\mathbf{T}_{cw} \tag{23}$$

The optimization is done by using Levenger-Marquard. We again need to scale the parameters through D_s and control the step with λD_w.

VI. Experiments

We evaluate the performance of the two proposed methods: Sparse Deformable Tracking with and without static camera, in quasi-rigid, quasi-isometric and changing-topology deformation. We use sequences of laboratory-controlled scenes from CVLab [33] and sequences of intracorporeal scenes selected from the Hamlyn Dataset [34]. We use the first depth image (or stereo pair) to initialise the surfels, i.e. the position, Jacobian and texture of each surfel. Notice that our system is monocular, hence our algorithm only processes the gray images obtained with the RGB-D camera or with the left camera of the stereo rig.

We report the 3D RMS error with respect to the Euclidean distance between the estimated point position and the ground truth position for each frame:

$$e_{\text{rms}} = \sqrt{\frac{\sum_i \|\hat{X}_i^t - X_i^t\|^2}{n}} \tag{24}$$

A video with the results is provided as supplementary material.

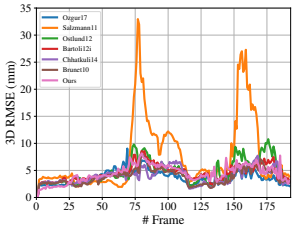
A. Tuning

1) ***Surfel size****:* Our primary assumption is that any surface can be locally approximated by the tangent plane. The accuracy of the approximation deceases with the distance to the centre of the surfel, hence it decreases with the surfel size. In contrast, bigger surfels allow more accurate estimates of the surfel geometry. Thanks to the saturation policy that we apply, we have noticed that even big surfels can accurately estimate the surfel geometry. We chose a surfel size of $\approx$23 pixels experimentally. In experiments in the Kinect paper dataset, we have observed that the error is reduced for bigger surfels, even if they do not fully accomplish the planarity assumption. Too big surfels lead to problems with spatial discontinuities in the scene.

2) ***Multi-scale****:* The convergence basin of the photometric methods is around one pixel, using multi-scale increases it to more that one pixel in the finest scale. We use the solution of a coarse scale as the initial guess of the next finer scale. In the kinect paper and T-shirt datasets, we found several missing frames. That precludes the convergence for many surfels if only the finest scale is used. Using 3 scales, the algorithm converges despite the missing frames.

3) ***Outlier rejection****:* We evaluate the ZNCC method to classify inliers as points that have converged correctly in the optimization. Positives are inliers and negatives are outliers. The ground truth of the correct tracks are classified through a threshold in the RMSE *wrt.* the ground truth surfel trajectory. We show the ROC curve in Fig. 3 right *wrt.* varying the ZNCC threshold. Ideally, the more up to the left the curve is, the better the classifier is. We finally select a value for the ZNCC of 0.95 for the experiments in the Kinect dataset, and 0.85 for the Hamlyn dataset.

4) ***Soft vs. Hard isometric constraint****:* In Sec.IV we have discussed two ways of constraining the deformation of the surfel. We have validated them in different sequences. We have observed that a soft constrained deformation model does not improve the accuracy of the system. Isometry seems to be a good local approach for the local deformation of the surfels, and thanks to treating the points individually we can recover very different global deformations. We use the isometry as a hard constrain (Eq. 11 and Eq. 22) in all the rest of experiments.



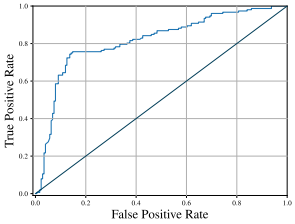


Fig. 3. Left: Comparison in kinect Paper dataset from CVLab. Our method tracks individual surfels with similar accuracy than template-based methods that assume surface continuity. Right: Outlier detection, ROC curve *wrt.* the ZNCC threshold

TABLE II
METHODS COMPARISON IN KINECT PAPER DATASET

	[35]	[36]	[23]	[37]	[38]	[39]	DSDT-SC
Automatic Data Association	-	-	✓	✓	-	✓	✓
Discontinuous	-	-	-	-	-	-	✓
Direct	-	-	-	-	-	-	✓
e_{rms} (mm)	4.56	3.86	7.47	3.78	3.97	4.82	4.02

B. Kinect Paper dataset

In Fig. 3 and Table II, we benchmark our method Direct and Sparse Deformable Tracking with Static Camera (DSDT-SC) against other state-of-the-art template-based methods: **Ozgur17** [37],

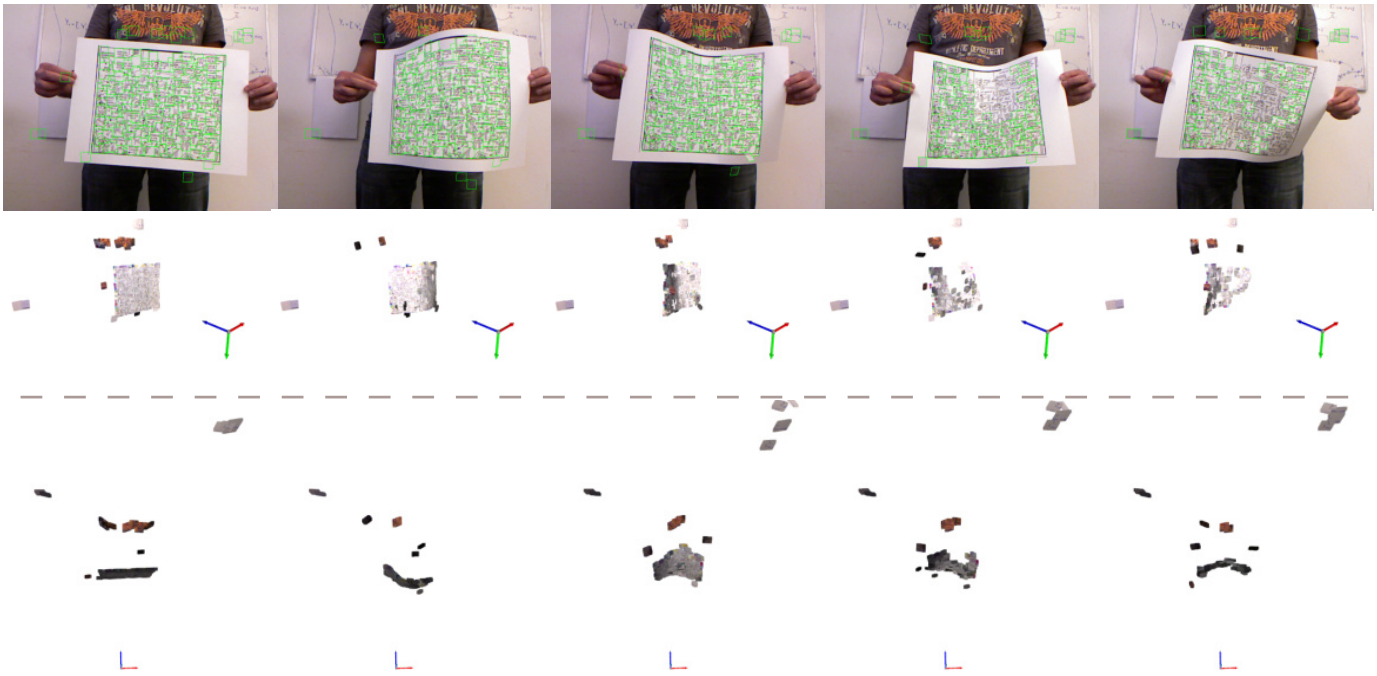


Fig. 4. Results of Direct Sparse Deformable tracking with Static Camera (DSDT-SC) in Kinect Paper Dataset. Top to bottom rows: RGB image with reprojected surfels, perspective view and top view. Even if the surfels are estimated independently the entire reconstruction displays an homogeneous consistency. Kinect Paper dataset, frames #0, #60, #110, #160 and #180. (see the entire sequences in supplementary material).

Salzmann11 [23], **Ostlund12** [39], **Bartoli12** [35], **Chhatkuli14** [38], **Brunet10** [36]. The characteristics of each method can be found in Table II. Although [35], [36] and [38] work with a local parametrization they assume continuity when they estimate the warp between images and when they integrate the final 3D surface. None of the other methods we compare with uses direct photometric error to recover the deformation or deal with discontinuities. We use the kinect paper dataset that shows a continuous paper under isometric deformations and we evaluate in the paper area. In contrast to the rest of the algorithms, our method does not assume a continuous deformation or a globally isometric deformation. Despite this, the quality of the surface deformation recovered with our algorithm is similar to the one obtained by the rest of methods, validating that our method can achieve state-of-the-art performance in continuous isometric scenes without assuming an explicit global model.

In contrast to the rest of the methods, our proposal can cope with discontinuous surfaces. As shown in Fig. 4a, our map includes surfels not only on the paper area, but also on the t-shirt, jeans and whiteboard in the background. As we have not assumed any regularizer between the individual surfels, we are able to track all surfels in the different surfaces. Discontinuous surfaces lead to other challenges like occlusions of the patches. Our optimization is also able to manage this kind of situations through the saturation of the photometric error and the outlier detection based on ZNCC.

C. Medical scenes

Now, we analyse the two proposed algorithms in our targeted medical sequences. We have tested our system in the laparoscopic sequences 6, 20 and 21 from Hamlyn dataset [34]. We evaluate our world-centric (DSDT) approach (Sec. V) where the camera can move and our camera-centric approach with static camera (DSDT-SC) (Sec. IV). We compare both methods against our tracking with a static (i.e. rigid) map (DSDT-SM), the deformable tracking from SD-DefSLAM [6] and the rigid tracking of ORBSLAM [40]. All the methods are initialized with a stereo pair in the same frame and no mapping is performed, tracking the initial map without refining or extending it. Table III shows the final results, that are discussed next.

Quasi-rigid scene. Sequence 6 (from frame #50) is an abdominal exploration where the scene remains almost rigid. It has a planar topology in the area where the camera closes up, and a small discontinuity due to a nerve. The texture is minimal except for the veins. This scene has a wide change of point of view what makes it challenging, allowing us to evaluate the matching stage.

Comparing ORBSLAM with our rigid approach (DSDT-SM), we are able to process more frames and with a lower error. This is because we are able to synthesise the view of the surfels for bigger view-point changes. Both rigid methods failed earlier than the deformable methods (SD, DSDT-SC and DSDT) because the scene is not completely rigid.

Our full deformable tracking DSDT can process 300 frames before tracking loss with an RMS error close to 3 mm. In contrast, DSDT-SC without moving camera can process a similar number of frames but with a much bigger error. We conclude that the regularizer (Eq. 21) added in the deformation tracking gives hints to the optimizer to move the camera to reduce the error what results in better performance. SD-DefSLAM tracks a similar number of frames with a slightly lower error than our methods since it assumes a global deformation model that is close to the actual scene deformation.

Quasi-isometric scene. Sequence 20 from Hamlyn (from frame #750) is also an abdominal exploration, but in this case the scene is quasi-isometrically deforming while the camera explores three organs with independent non-rigid movements (see Fig. 1).

TABLE III
3D RMS ERROR AND # OF FRAMES PROCESSED IN MEDICAL SEQUENCES

		Rigid map		Deformable map		
		ORB[40]	DSDT-SM	SD[6]	DSDT-SC	DSDT
6	RMSE	4.85	3.26	2.72	9.24	3.17
	# Fr.	128	200	286	334	300
20	RMSE	1.37	1.37	4.68	3.09	2.9
	# Fr.	220	210	252	350	500
21	RMSE	-	-	6.19	1.81	1.30
	# Fr.	-	-	323	321	300

ORB: ORBSLAM; **DSDT-SM**: Direct Sparse with static map; **SD**: Semi-direct DefSLAM; **DSDT-SC**: Direct Sparse Deformable Tracking with static camera; **DSDT**: Direct Sparse Deformable Tracking

Modelling the movement of the camera we are able to recover many points when revisiting known areas, being able to process a higher number of frames with the full DSDT method than with its fixed-camera counterpart DSDT-SC. In contrast, SD-DefSLAM assumes a global model that does not correspond with the observed scene, therefore it misses a big part of points when quasi-isometric deformation happens (it got lost in frame #252 only a little later than the rigid methods). Rigid methods (ORBSLAM and DSDT-SM) only survive in a small area of the scene being badly conditioned and losing camera tracking.

Independent bodies scene. In sequence 21 (from frame #750) the camera images two lobes of a liver moving as independent bodies, one lobe sliding over the other (Fig. 5).

As shown in Table III and Fig. 5, the rigid methods fail to track both lobes. SD-DefSLAM can track some of the points but with a high RMSE because its isometric deformation model cannot code the deformation actually observed. Thanks to our formulation, the proposed deformable tracking can process global non-isometric deformations, being able to cope with deformations from independent bodies.

Also, we can deal with wrong initialization and occlusions thanks to our matching strategy that includes a saturation threshold in the photometric error and a ZNCC outlier detector.

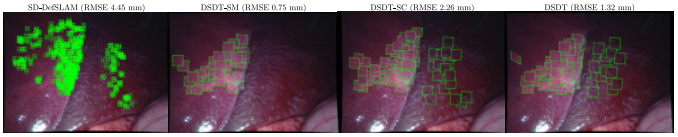


Fig. 5. Results in Hamlyn sequence 21, with independent-body motion. Left to right: DSDT-SM (w. static map), SD-DefSLAM, DSDT-SC (w. static camera) and DSDT.

Changing Topology scene. In addition, we have also evaluated our system in a changing topology sequence. In this sequence, a paper is split while recorded with a static camera.

As shown in Fig. 6, thanks to the local parametrization, we can track the patches even with a global changing typology. Once the two parts of the paper are split, we can still recover the local deformations of the two surfaces. The RMS error in the entire sequence is 11mm.

The sequence shows strong illumination changes due to the fluorescent light flickering. The gain α_i^t and bias β_i^t, included in Eq.5, model properly the illumination changes and cope with the challenging sequence.

VII. CONCLUSION

We have proposed a novel approach for deformable tracking in deformable SLAM. Each map point is modeled as a 3D surfel that is a local approximation of the scene surface. The deformations of the map are modeled through the movement of these surfels. In contrast to the previous deformable tracking methods we have proposed to remove any connection between 3D map points.

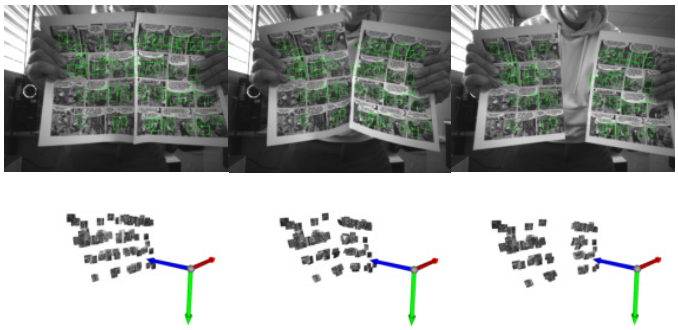


Fig. 6. Changing-topology scene results. Top Row: Image with tracked surfels. Lower Row: 3D position of the surfels and camera frame. Left to right: frames #0, #75 and #110

We have proved experimentally that the local model for the deformable tracking can perform similarly to the state-of-the art methods and can perform more robustly and more accurately than the global methods in scenes composed of discontinuous surfaces, or with global non-isometric deformations. In addition, we reassert the potential of the direct methods over the feature-based equivalents.

Future work could extend this deformable tracking into a deformable mapping able to reconstruct scenes composed of discontinuous surfaces, or with global non-isometric deformations. With this two algorithmic components, it will be possible to create a new generation of monocular Deformable SLAM algorithms able to work in a wider range of scenarios.

REFERENCES

[1] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An accurate open-source library for visual, visual-inertial and multi-map SLAM," *IEEE Transactions on Robotics*, vol. early acces, pp. 1–17, 2021.
[2] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 3, pp. 611–625, 2017.
[3] J. Zubizarreta, I. Aguinaga, and J. M. M. Montiel, "Direct sparse mapping," *IEEE Transactions on Robotics*, vol. 36, no. 4, pp. 1363–1370, August 2020.
[4] G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *ISMAR*, 2007.
[5] J. Lamarca, S. Parashar, A. Bartoli, and J. M. M. Montiel, "DefSLAM: Tracking and mapping of deforming scenes from monocular sequences," *IEEE Transactions on Robotics*, vol. 37, no. 1, pp. 291–303, 2021.
[6] J. J. G. Rodríguez, J. Lamarca, J. Morlana, J. D. Tardós, and J. M. Montiel, "SD-DefSLAM: Semi-direct monocular slam for deformable and intracorporeal scenes," in *ICRA*, 2021.
[7] S. Parashar, D. Pizarro, and A. Bartoli, "Isometric non-rigid shape-from-motion with riemannian geometry solved in linear time," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 10, pp. 2442–2454, 2017.
[8] J. Lamarca and J. M. M. Montiel, "Camera tracking for SLAM in deformable maps," in *4th Inter. Workshop on Recovering 6D Object Pose. In ECCVw*, 2018.
[9] R. A. Newcombe, D. Fox, and S. M. Seitz, "DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time," in *CVPR*, 2015.
[10] J. Song, J. Wang, L. Zhao, S. Huang, and G. Dissanayake, "MIS-SLAM: Real-time large-scale dense deformable SLAM system in minimal invasive surgery based on heterogeneous computing," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 4068–4075, 2018.
[11] C. Bregler, A. Hertzmann, and H. Biermann, "Recovering non-rigid 3D shape from image streams," in *CVPR*, 2000.
[12] Y. Dai, H. Li, and M. He, "A simple prior-free method for non-rigid structure-from-motion factorization," *International Journal of Computer Vision*, vol. 107, no. 2, pp. 101–122, 2014.

[13] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade, "Trajectory space: A dual representation for nonrigid structure from motion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 7, pp. 1442–1456, 2011.

[14] R. Garg, A. Roussos, and L. Agapito, "Dense variational reconstruction of non-rigid surfaces from monocular video," in *CVPR*, 2013.

[15] P. F. Gotardo and A. M. Martinez, "Kernel non-rigid structure from motion," in *ICCV*, 2011.

[16] A. Chhatkuli, D. Pizarro, and A. Bartoli, "Non-rigid shape-from-motion for isometric surfaces using infinitesimal planarity." in *BMVC*, 2014.

[17] A. Chhatkuli, D. Pizarro, T. Collins, and A. Bartoli, "Inextensible non-rigid shape-from-motion by second-order cone programming," in *CVPR*, 2016.

[18] J. Taylor, A. D. Jepson, and K. N. Kutulakos, "Non-rigid structure from locally-rigid motion," in *CVPR*, 2010.

[19] S. Vicente and L. Agapito, "Soft inextensibility constraints for template-free non-rigid reconstruction," in *ECCV*, 2012.

[20] A. Bartoli, Y. Gérard, F. Chadebecq, T. Collins, and D. Pizarro, "Shape-from-template," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 10, pp. 2099–2118, 2015.

[21] A. Chhatkuli, D. Pizarro, A. Bartoli, and T. Collins, "A stable analytical framework for isometric shape-from-template by surface integration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 5, pp. 833–850, 2017.

[22] T. Collins and A. Bartoli, "Locally affine and planar deformable surface reconstruction from video," in *International Workshop on Vision, Modeling and Visualization*, 2010.

[23] M. Salzmann and P. Fua, "Linear local models for monocular reconstruction of deformable surfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 931–944, 2011.

[24] 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.

[25] F. Devernay, D. Mateus, and M. Guilbert, "Multi-camera scene flow by tracking 3-d points and surfels," in *CVPR*, 2006.

[26] A. Varol, M. Salzmann, E. Tola, and P. Fua, "Template-free monocular reconstruction of deformable surfaces," in *ICCV*, 2009.

[27] J. Fayad, L. Agapito, and A. Del Bue, "Piecewise quadratic reconstruction of non-rigid surfaces from monocular sequences," in *ECCV*, 2010.

[28] C. Russell, J. Fayad, and L. Agapito, "Energy based multiple model fitting for non-rigid structure from motion," in *CVPR*, 2011.

[29] R. Yu, C. Russell, N. D. Campbell, and L. Agapito, "Direct, dense, and deformable: template-based non-rigid 3d reconstruction from rgb video," in *ICCV*, 2015.

[30] Jianbo Shi and Tomasi, "Good features to track," in *In CVPR*, 1994.

[31] J. Nocedal and S. Wright, *Numerical optimization*. Springer Science & Business Media, 2006.

[32] J. Sola, J. Deray, and D. Atchuthan, "A micro lie theory for state estimation in robotics," *arXiv preprint arXiv:1812.01537*, 2018.

[33] A. Varol, M. Salzmann, P. Fua, and R. Urtasun, "A constrained latent variable model," in *CVPR*, 2012.

[34] P. Mountney, D. Stoyanov, and G.-Z. Yang, "Three-dimensional tissue deformation recovery and tracking," *IEEE Signal Processing Magazine*, vol. 27, pp. 14–24, 2010.

[35] A. Bartoli, Y. Gérard, F. Chadebecq, and T. Collins, "On template-based reconstruction from a single view: Analytical solutions and proofs of well-posedness for developable, isometric and conformal surfaces," in *CVPR*, 2012.

[36] F. Brunet, R. Hartley, A. Bartoli, N. Navab, and R. Malgouyres, "Monocular template-based reconstruction of smooth and inextensible surfaces," in *ACCV*, 2010.

[37] E. Özgür and A. Bartoli, "Particle-SfT: A provably-convergent, fast shape-from-template algorithm," *International Journal of Computer Vision*, vol. 123, no. 2, pp. 184–205, 2017.

[38] A. Chhatkuli, D. Pizarro, and A. Bartoli, "Stable template-based isometric 3d reconstruction in all imaging conditions by linear least-squares," in *CVPR*, 2014.

[39] J. Östlund, A. Varol, D. T. Ngo, and P. Fua, "Laplacian meshes for monocular 3D shape recovery," in *ECCV*, 2012.

[40] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

José Lamarca (Zaragoza,Spain,1992) received a Bachelor's and M.S. Degree in Industrial Engineering (mention in Robotics and Computer Vision) from Universidad de Zaragoza, where he recently graduated as PhD. in System Engineering and Computer Science in the I3A Robotics, Perception and Real-Time Group. Currently, he works as researcher in the ARKit team in Apple Inc. His research interests focus in real-time visual SLAM solutions for both rigid and deformable environments.

Juan J. Gómez Rodríguez received a Bachelor's Degree in Informatics Engineering (mention in Computing) and Master's in Biomedical Engineering (mention in Information and Communication Technologies in Biomedical Engineering) from Universidad de Zaragoza, where he is currently working towards the PhD. degree with the I3A Robotics, Perception and Real-Time Group. His research interests are real-time visual SLAM for both rigid and deformable environments. He received an honorable mention to the King-Sun Fu Memorial IEEE Transactions on Robotics Best Paper Award in 2021, for the paper describing ORB-SLAM3.

Juan D. Tardós (Huesca, Spain, 1961) received the M.S. and Ph.D. degrees in electrical engineering from the University of Zaragoza, Spain, in 1985 and 1991, respectively. He is Full Professor with the Departamento de Informática e Ingeniería de Sistemas, University of Zaragoza, where he is in charge of courses in machine learning and SLAM. His research interests include SLAM, perception and mobile robotics. He received the King-Sun Fu Memorial IEEE Transactions on Robotics Best Paper Award in 2015 for the paper describing the monocular SLAM system ORB-SLAM and an honorable mention to the same award in 2021, for the paper describing ORB-SLAM3.

J.M.M. Montiel (Arnedo, Spain, 1967) received the M.S. and PhD degrees in electrical engineering from Universidad de Zaragoza, Spain, in 1992 and 1996, respectively. He has been awarded several Spanish MEC grants to fund research with the University of Oxford, U.K., and Imperial College London, U.K.

He is currently a full professor with the Departamento de Informática e Ingeniería de Sistemas, Universidad de Zaragoza, where he is in charge of perception and computer vision research grants and courses. His interests include real-time visual SLAM for rigid and non-rigid environments, and the transference of this technology to robotic and non- robotic application domains. He has received several awards, including the 2015 King-Sun Fu Memorial IEEE Transactions on Robotics Best Paper Award and an honorable mention to the same award in 2021. Since 2020 he coordinates the EU FET EndoMapper grant to bring visual SLAM to intracorporeal medical scenes.