A Shape Completion Component for Monocular Non-Rigid SLAM

Yongzhi Su
DFKI
TU Kaiserslautern

Vladislav Golyanik
MPI for Informatics, Saarland

Nareg Minaskan
DFKI

Sk Aziz Ali
DFKI
TU Kaiserslautern

Didier Stricker
DFKI
TU Kaiserslautern

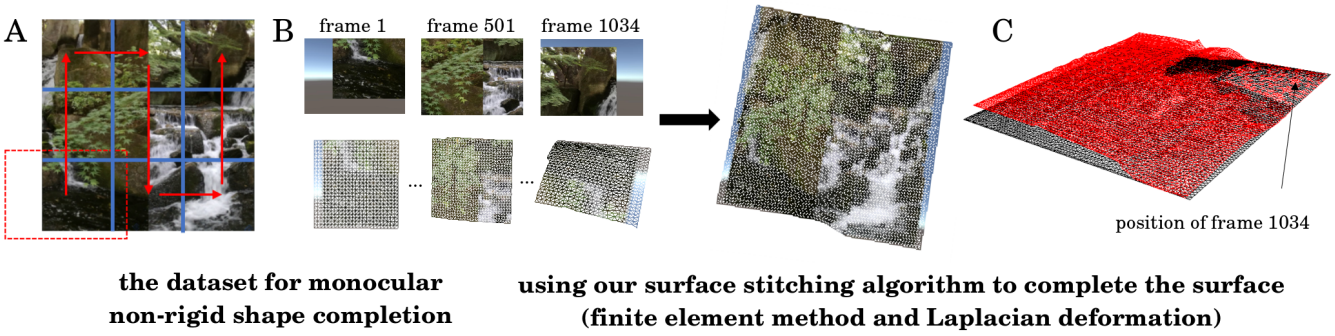

Figure 1. We propose a method for monocular shape completion and a new evaluation dataset. **A**: The trajectory of the moving camera in our dataset for non-rigid surface completion; **B**: Using individual reconstructions for multiple frames and camera poses, our physics-based approach incrementally completes the global surface. For the observed parts, we use finite element method [34], and for the hidden parts, we employ Laplacian deformation modelling [28]. **C**: The complete surface obtained by our algorithm (in red) and the ground truth (in black).

Abstract

We propose a finite element method (FEM) based approach for surface stitching which can be integrated into existing SLAM and NRSfM pipelines for AR applications. Given individual reconstructions and camera poses at different time stamps, our stitching method incrementally completes the surface with a smooth transition between the hidden and the observed parts, so that all the observed parts can be stitched into a single surface. Thanks to the physical modelling, deformations from the observed parts are propagated to the hidden parts enabling an overall high-fidelity and realistic estimate. To keep the computational time in bounds, deformations near the observed parts are computed with FEM, and the remaining region is approximated by Laplacian deformation. We assume that no force is applied to the hidden parts. To evaluate the algorithm, we generate a synthetic dataset with ground truth. In our dataset, the camera observes only a part of the target surface in each frame and moves until the whole target surface is covered. The dataset which will be made publicly available includes the ground truth camera poses and geometries of the whole surface at each time frame. An experimental evaluation of the stitching method with accuracy metrics rounds out the draft.

Index Terms: H.5.1—Information Interfaces and Presentation–Multimedia Information Systems—Artificial, augmented, and virtual realities; I.3.5—Computer Graphics—Computational Geometry and Object Modeling—Physically-based modeling.

1 Introduction and Motivation

Visual simultaneous localisation and mapping (SLAM) techniques are widely used in augmented reality (AR), medical image analysis and robotic navigation. In the last decade, the methods based on bundle adjustment [21] and extended Kalman filter [14] have significantly advanced. Some of them have been extended to produce per-pixel dense reconstructions for hand-held devices [24] or even aerial vehicles [32]. These methods require a single monocular camera and can only handle scenes with rigid objects.

It is challenging to overcome this limitation since the problem of single-body monocular reconstruction under non-rigid deformations is ill-posed and underconstrained [33]. Different shapes in 3D can result in the same observations when projected into 2D. At the same time, non-rigid structure from motion (NRSfM) remains an active research field with remarkable results achieved over the last years. In its core, NRSfM relies on various types of prior knowledge such as assumptions about deformations and camera motion [9, 31] or low-rank shape basis [12, 13, 31]. Recently, NRSfM has been extended to the dense setting [6, 17, 19]. In all of the above methods, the reconstructed surface must always be entirely observed, including possible occasional external and self-occlusions. In many real cases, yet, only parts of the target surface are observed at any given frame. In endoscopic scenarios (*e.g.,* as can be seen in the liver dataset from [3]), only a part of the surface is observed in each frame, and reconstruction of the whole tissue is of high interest for the medical diagnostics or the AR-assisted surgery. Although the surface is observed partially, the surface remains connected in most of the cases (see Fig. 2).

In this work, we take a step towards overcoming the above-mentioned limitations of SLAM and NRSfM systems, see Fig. 1 for an overview. We use global surface connectivity as prior knowledge and demonstrate a solution to the stitching problem with given partial reconstructions and camera poses. The reconstruction of the whole surface does not need to be known in advance. It is worth noting that we exclusively focus on the stitching problem in this paper and assume that partial reconstructions and camera poses are given. In the following, we refer to the *observed part* as a surface in the current frame. We refer to the *hidden part* as a surface which has been already observed and which is currently out of view. Furthermore, we assume that the hidden part is force-balanced and its deformation is caused by the deformation of the observed part (imagine a sheet of paper — if one of its corners is displaced, the whole sheet will be deforming). We model the deformation of the observed part together with hidden part using linear finite element method (FEM). Deformations of the observed part are used as constraints while solving for the deformations of the hidden part.

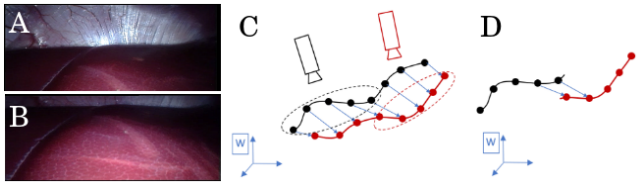

Figure 2. A, B: Sample frames from the liver dataset [3]. **C**: The black-dotted frame is the reconstruction at time t_1, the red-dotted frame is the reconstruction at time t_2. The object deforms between time t_1 and t_2. **D**: With given partial reconstructions, stitching is required to obtain a complete shape. Best viewed in colour.

Since no suitable dataset for evaluating non-rigid shape completion exists, we create a new comprehensive synthetic dataset with global ground truth geometry, camera poses and partially observed geometries generated by physical simulations. Thus, it reflects realistic deformations of a thin surface. The new dataset (Sec. 3) and the proposed shape completion technique (Sec. 4) contribute to the field with a solution and an evaluation methodology for the scarcely addressed problem of non-rigid 3D reconstruction and shape completion from a single moving camera. The main contributions of this work can be summarised as the follows: We present the first technique for monocular surface stitching which can be integrated into SLAM and NRSfM pipelines, in order to make them able to reconstruct the whole non-rigidly deforming surface and incrementally stitch newly incoming parts to already available reconstruction. Our algorithm can be used in augmented reality applications for non-rigid surface recovery, non-rigid SLAM, or predicting the shape of the non-rigid surface during partial occlusion (Sec. 5).

2 Related Work

Different kinds of SLAM algorithms have been developed in the last decade. [14] represent the kind of works, which use extended Kalman filter as the back-end to track the sparse key points. In contrast to [14], Klein and Murray [21] uses non-linear optimisation. They extract as much as possible of the information from multiple keyframes. In contrast to both these methods, Engel *et al.* [15] minimise the photometric disparity error of the pixels in the entire image. Still, all existing SLAM algorithms can only reconstruct scenes with rigid objects. Dense NRSfM aims to reconstruct non-rigid surfaces [6, 8, 17–19, 25], though, in the existing NRSfM algorithms, the surface must always be entirely observed and tracked from a reference frame, including possible external and self-occlusions.

The most closely related methods to our proposed monocular surface stitching are of the class of NRSfM. Our SLAM component can be integrated into the final stage of the NRSfM pipeline to stitch individual non-rigid surfaces under the global connectivity assumption. Following the work by Bregler *et al.* [12], most NRSfM methods represent non-rigid shapes as a linear combination in a low-rank shape basis. Although this approach can capture global deformation effectively, it fails to approximate the surface with multiple stronger local deformations. To solve this problem, the solution based on piece-wise modelling was proposed [16, 26, 30], *i.e.,* the surface is split into multiple overlapping regions, and each region is treated as a local model optimised separately. However, the same point in the overlapping regions can have different positions due to the independent optimisation. Solely relying on geometric fitting cost in the overlapping regions might lead to a physically implausible global reconstruction. As applied to the case of shape completion, deformations of the hidden parts should also follow the physical laws. Thus, we propose to model surfaces using a physical model and estimate deformations of the hidden parts conditioned upon the observed deformations.

Physics-based models have been used for animation and simulation purposes in computer graphics [10, 34] and in computer vision for deformation modelling [20, 22]. These approaches capture small relative deformations well. More accurate simulation can be achieved by using non-linear FEM for large deformations. Along with that, the material properties need to be known, which leads to an additional parameter. Recently, a linear FEM-based approach has been proposed to recover non-linear deformations [5]. Compared to the non-linear FEM, the linear FEM does not require much knowledge of the material properties, because most of the material properties can be factorised out and do not have to be known in advance [7]. In contrast to [7], we are solving the problem of shape completion with FEM, while Adugo *et al.* [7] reconstruct single objects which are uninterruptedly observed by a camera.

In our method, the computational cost grows with the increasing area of the hidden surface part. To keep the computational cost feasible, only deformations of the region that is near to the observed part are calculated with FEM. The remaining region is approximated using Laplacian deformation [28], which can deform a part of a surface without losing the geometric details of the remaining part.

3 Dataset for monocular non-rigid SLAM

For the new dataset, we first generate a 75×75 grid mesh, see Fig. 3. Then, we apply a particle solver and simulate cloth-like deformations [23]. The position of each particle (vertices with mass) is updated after the solver satisfies the equations according to a set of constraints (*e.g.,* distance, bending and collision). The edges of the cloth are fixed and serve as boundary conditions in the simulation.

To generate deformation on the surface of the cloth, we use random force vectors. The magnitude of the force was tuned such that it a) does not destroy the mesh topology and b) causes moderate deformations of the surface. The camera moves along a pre-defined path and records images of the patches. At each stationary position, the camera stays for 50 frames, because NRSfM usually requires accumulated motion and deformation cues of at least 30-50 frames. For each camera position, we extract the whole mesh, the partial mesh of every patch (which the camera currently observes) as well as the patch images. We use Unity [1] for the simulation. All measurements obtained after the simulation comprise the new dataset.

The new dataset suits well for the evaluation of existing and emerging monocular surface reconstruction methods. It can be used for evaluation of the shape completion methods (the main scope of this paper), template-based surface recovery and to train a neural network for non-rigid reconstruction, in the spirit of the recently proposed IsMO-GAN [27].

4 Our Surface Completion Approach

Our target is to incrementally complete the surface while some regions leave and enter the field of view. We assume that the hidden (force-balanced) parts are conditioned upon the deformation propagated from the observed regions. Under this assumption, we model surface deformations with FEM. First, we introduce a deformation model with continuum mechanics in Sec. 4.1 and then describe our FEM solution Sec. 4.2. Finally, we present our approach to non-rigid surface stitching in Sec. 4.3 including details on the handling of hidden parts with Laplacian deformation modelling in Sec. 4.4.

4.1 Deformation Model with Continuum Mechanics

Similar to [7], we model surface deformations with continuum mechanics, *i.e.,* relate the force applied to the surface with the caused deformations. The FEM model considers the linearly elastic object in Fig. 4.A referred to a 3D rectangular Cartesian coordinate system $\boldsymbol{C} = \{\mathrm{x}, \mathrm{y}, \mathrm{z}\}$. A volumetric force $\mathbf{f_c}$ acts on the surface, and it gets internally stressed due to the prescribed loading conditions. The surface states can be expressed in vector notation in terms of the

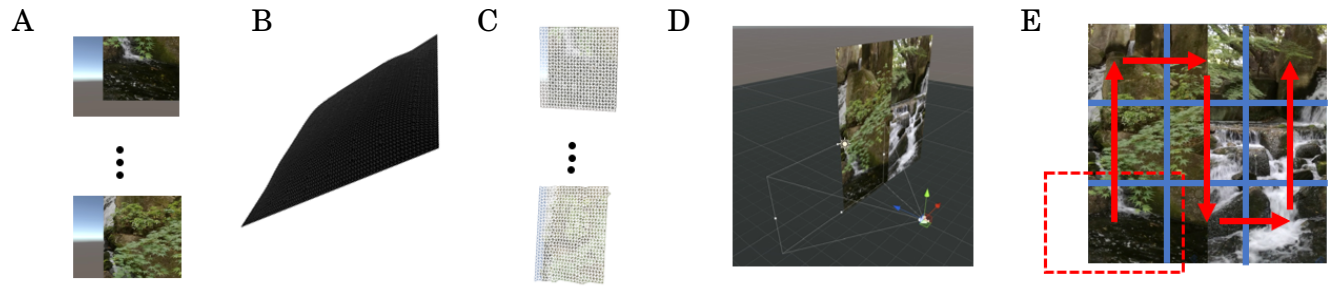

Figure 3. **A**: Sample images from the dataset. **B**: Whole groud truth surface. **C**: Sample groud truth geometries from the dataset (represented as mesh). **D**: Third-person views of the simulated surface and camera positions in Unity [1]. **E**: The global surface is divided into nine patches. Every patch corresponds to stationary locations of the camera which are substituted one by the other. The red arrows show the camera trajectory during the image capture.

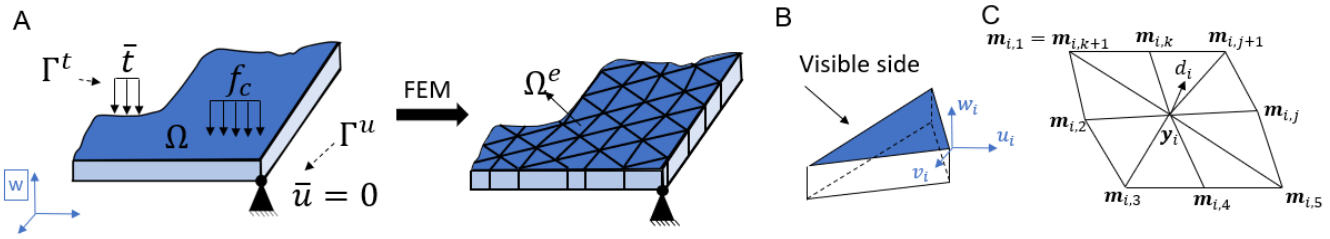

Figure 4. **A**: A volumetric force $\mathbf{f_c}$ acts on the surface Ω, and the Neumann boundary conditions are expressed as $\bar{\boldsymbol{t}}$ and $\bar{\boldsymbol{u}}$. The problem can be solved with FEM by dividing the surface into small patches Ω^e. **B**: The 3D wedge elements are used to model the patch Ω^e, which has a visible side and a invisible back side. **C**: The k triangles to which the node $\boldsymbol{y_i}$ belongs. This figure is inspired by [7].

displacement and volumetric force with electrostatic Navier-Cauchy equations [34] as

$$\frac{E}{2(1+\nu)(1-2\nu)}\nabla(\nabla\cdot\boldsymbol{u})+\frac{E}{2(1+\nu)}\nabla^2\boldsymbol{u}+\boldsymbol{f}_c=0 \text{ in } \Omega, \quad (1)$$

where $\boldsymbol{u}=[u_x,u_y,u_z]^T$ is the unknown 3D displacement field. This expression includes the gradient operator $\nabla=[\delta/\delta\mathrm{x},\,\delta/\delta\mathrm{y},\,\delta/\delta\mathrm{z}]^T$, the divergence operator $\nabla\cdot\boldsymbol{u}=\delta u_x/\delta\mathrm{x}+\delta u_y/\delta\mathrm{y}+\delta u_z/\delta\mathrm{z}$, and the Laplacian operator $\nabla^2(\cdot)$ (the divergence of the gradient). Material properties for the modelled isotropic elastic solid are described using Young's modulus E and the Poisson's ratio ν. A displacement vector (Dirichlet conditions) $\boldsymbol{u}=\bar{\boldsymbol{u}}$ on Γ^u , or a stress $\boldsymbol{t}$ vector (Neumann conditions) $\boldsymbol{t}=\bar{\boldsymbol{t}}$ on Γ^t expresses the required boundary condition of this equation. The boundary is defined as $\Gamma=\Gamma^u+\Gamma^t$, with $\bar{\boldsymbol{u}}$ and $\bar{\boldsymbol{t}}$ being a prescribed displacement and traction field, respectively.

4.2 FEM-Based Solution

While the partial Eq. (1) does not have an analytical solution in most cases, numerical methods such as FEM can be applied for an approximate solution. The idea of FEM is to divide a surface Ω (see Fig. 4) into a finite set of small patches, whose deformations are easier solvable. The patches are denoted as Ω^e and defined by the nodal points, which has the form of $\boldsymbol{y_i}=[x_i,y_i,z_i]^T$. The nodal displacement vector and nodal force vector of the $\boldsymbol{i}$-th nodal point can be expressed as $\boldsymbol{a_i}=[u_i,v_i,w_i]^T$ and $\boldsymbol{f_i}=[f_{xi},f_{yi},f_{zi}]^T$. Based on the nodal displacements of every nodal point of a patch, the displacement vector $\boldsymbol{u}$ of any point in the patch can be approximated as a weighted sum of piecewise shape basis functions N_i:

$$\boldsymbol{u}(x,y,z)=\sum_i N_i\boldsymbol{a}_i. \quad (2)$$

Eq. (1) can be formulated as a classic linear global FEM system with the linear FEM approximation:

$$\mathbf{K}\boldsymbol{a}=\boldsymbol{f}, \quad (3)$$

where $\boldsymbol{a}=[\boldsymbol{a_0},\ldots,\boldsymbol{a_n}]^T$ and $\boldsymbol{f}=[\boldsymbol{f_0},\ldots,\boldsymbol{f_n}]^T$ are the 3D global displacement vector and the force vector of n nodal points. $\mathbf{K}$ is the global stiffness matrix, which can be obtained by assembling from the associate element stiffness matrix $\mathbf{K^e}$. The element of the stiffness matrix $\mathbf{K^e}$ can be calculated as

$$\mathbf{K^e}=\int_{\Omega^e}\mathbf{B}^T\mathbf{D}\mathbf{B}\,d\Omega^e, \quad (4)$$

where $\mathbf{D}(E,\nu)$ is the behaviour matrix for isotropic linear materials and $\mathbf{B}$ is the strain-displacement matrix that depends on the type of discretisation [10, 34]. Note that the behaviour matrix $\mathbf{D}$ is proportional to E, so the element stiffness matrix $\mathbf{K^e}$ is also proportional to E. See App. A for more details. In this paper, the 3D wedge elements defined using six nodal points are used to model the patches of non-rigid shape (see Fig. 4). The surface is modelled as a single layer of these elements. The elements are opaque, which means the camera cannot see one side of the element. We consider the 3D surface reconstruction in the point cloud representation as the nodal points in the visible side and generate a triangulated mesh. To ensure that the equations have a solution, the normals of the triangles should be oriented consistently, and every single point should be connected in the mesh. The invisible nodal point $\boldsymbol{y_{hi}}$ at the back side should share the same normal with a correspondent visible nodal $\boldsymbol{y_{vi}}$. The position of an invisible nodal point can be expressed as

$$\boldsymbol{y_{hi}}=h\boldsymbol{d_i}+\boldsymbol{y_{vi}}, \quad (5)$$

where h is a fixed value that corresponds to the surface thickness. The normal unit vector $\boldsymbol{d_i}$ of each visible nodal point is weighted

according to the corresponding triangle area computed by a cross product (see Fig. 4):

$$\boldsymbol{d_i} = \frac{\sum_{j=1}^{k}(\boldsymbol{m_{ij}} - \boldsymbol{y_i} \times (\boldsymbol{m_{ij+1}} - \boldsymbol{y_i})}{\|\sum_{j=1}^{k}(\boldsymbol{m_{ij}} - \boldsymbol{y_i} \times (\boldsymbol{m_{ij+1}} - \boldsymbol{y_i})\|}, \tag{6}$$

where $\boldsymbol{m_{ij}} \in \{\boldsymbol{m_{i1}}, \boldsymbol{m_{i2}}, \cdots, \boldsymbol{m_{ik}}\}$ are the neighbouring nodes defining k triangles to which the node $\boldsymbol{y_i}$ belongs (see Fig. 4).

The nodal point $\boldsymbol{y_i}$ in Cartesian coordinates can be expressed in natural coordinates $\boldsymbol{\xi_i} = [\eta_i, \xi_i, \zeta_i]^T$ to simplify the integration of Eq. (4), with $\xi, \eta \in [0,1]$ and $\zeta \in [-1,1]$. The Jacobian matrix of the transformation (see App. A for more details) can be calculated as

$$\mathbf{J} = \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{\xi}}. \tag{7}$$

Eq. (4) can be expressed as

$$\mathbf{K^e} = \int_{-1}^{1}\int_{0}^{1}\int_{0}^{1-\xi} \mathbf{B}^T\mathbf{D}\mathbf{B}\|\mathbf{J}\|\, d\eta\, d\xi\, d\zeta, \tag{8}$$

which can be approximated using integration points [29]. In this paper, we use three-point hammer-integration to approximate the integral over the triangulation with $(\zeta, \xi) \in \left\{\left(\frac{1}{6}, \frac{2}{3}\right), \left(\frac{2}{3}, \frac{1}{6}\right), \left(\frac{1}{6}, \frac{1}{6}\right)\right\}$, and three points Gaussian integration to approximate the integral over the η direction with $\eta \in \{-0.7746, 0, 0.7746\}$, which means nine points in total are used to approximate the integral over 3D wedge elements Ω^e.

Next, thanks to the symmetry characteristics of the global stiffness matrix $\mathbf{K}$, Eq. (1) can be rewritten as

$$\begin{bmatrix} \mathbf{K_{vv}} & \mathbf{K_{vh}} \\ \mathbf{K_{vh}^T} & \mathbf{K_{hh}} \end{bmatrix} \begin{bmatrix} \boldsymbol{a^v} \\ \boldsymbol{a^h} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f^v} \\ \boldsymbol{f^h} \end{bmatrix}. \tag{9}$$

We assume the force is only applied on the visible side of the surface [7] and Eq. (9) can be simplified with $\boldsymbol{f^h} = [0, \cdots, 0]^T$ as:

$$\mathbf{K^v}\boldsymbol{a^v} = \boldsymbol{f^v}, \tag{10}$$

where

$$\mathbf{K^v} = \left[\mathbf{K_{vv}} - \mathbf{K_{vh}}(\mathbf{K_{hh}})^{-1}(\mathbf{K_{vh}})^T\right] \tag{11}$$

is the implicated global matrix.

4.3 Surface Completion using FEM

In our approach, the surface is completed by stitching all 3D reconstructions $F_0 \cdots F_n$, where F_n is the reconstruction at the n-th frame. Assuming the camera provides sufficient frame rate, there is always an overlap between two consecutive frames F_{i-1} and F_i, with $i \in [1, n]$. The reconstruction of the complete surface G_0 is initialised as F_0. Starting from F_1, each frame F_i is stitched into the completed surface G_{i-1}, and the state after stitching is saved as G_i.

In our FEM model, the surface is considered to be sufficiently elastic, which means, the deformation is always proportional to the applied force. In the real world, objects have an elastic limit (*e.g.,* a sheet can be torn). Our model does not allow to predict either if the sheet is torn, or the state of the sheet after tearing. So, we assume that no force is applied to the hidden part of G_{i-1}, and displacements in the hidden part are caused by the displacements in the overlapping regions. In many practical scenarios, this is a realistic assumption.

To solve Eq. (10), 3D point correspondences in the overlapping region between every pair of consecutive frames are required. We obtain the correspondences by projecting geometries into the image plane and establishing correspondences in the image plane (in the projection, 2D-3D correspondences are known). In this paper, we take advantage of known point indexes in the ground truth which are kept unchanged for the whole surface. This assumption will be relaxed in future work.

In the following, we explain how Eq. (10) can be solved with some known displacements and prove that E can be factorised out. Eq. (10) can be rewritten as

$$\begin{bmatrix} \boldsymbol{K_0^v} & \cdots & \boldsymbol{K_j^v} & \cdots & \boldsymbol{K_m^v} \end{bmatrix} \begin{bmatrix} a_0^v \\ \vdots \\ a_j^v \\ \vdots \\ a_m^v \end{bmatrix} = \begin{bmatrix} f_0^v \\ \vdots \\ f_j^v \\ \vdots \\ f_m^v \end{bmatrix}, \tag{12}$$

where $\boldsymbol{K_j^v} = [K_{0j}^v, \cdots, K_{mj}^v]^T$, with $j \in [0, m]$. The dimension m is equal to three times the number of the nodal points. If the displacement of a_j is known, the j-th row and column of $\mathbf{K^v}$ can be removed without affecting the solution of a_k^v by rewriting $\boldsymbol{f^v}$ as $\boldsymbol{f^v}$ - $a_j^v\boldsymbol{K_j^v}$ with $k \in \{k = 0, \cdots, m;\, k \neq j\}$. As the corresponding force of the unknown displacement was initialized to 0, after removing rows of all known displacements, the elements in the resultant force (every row on the right side of the Eq. 12) can be seen as the sum of a constant times an element of the global $\mathbf{K^v}$ matrix. As we have mentioned before, $\mathbf{K^e}$ is proportional to E and, therefore, the elements of global $\mathbf{K^v}$ matrix are proportional to E. Since the elements on both sides of the remaining equation are proportional to E, E can be factorised out. In this paper, we set E to 1.

Thus, the deformations of the observed part can be used as constraints while solving for the deformations of the hidden part.

4.4 Handling of Hidden Parts

To speed up the computations, only the deformation of the region that is near to the observed part is calculated with FEM. We use orthographic projection to define the region in the completed surface G_{i-1} whose deformation should be calculated using FEM. The 3D reconstruction of frame F_i and completed surface G_{i-1} is projected into the i-th image plane. A polygon referring to a 2D convex hull of F_i is estimated and extended to ≈1.3 times of its original area. Point displacements in G_{i-1}, whose projection fall within this polygon, is evaluated with FEM.

FEM allows to calculate the deformations according to the laws of physics but with a high computational cost, and the Laplacian deformation [28] can calculate the deformation fast and without losing the geometric details of the surface. The principle of Laplacian deformation is to minimise the energy function expressed as:

$$E(\boldsymbol{y'}) = \underbrace{\sum_{p} ||\mathscr{L}(\boldsymbol{y}_p) - \mathscr{L}(\boldsymbol{y'}_p)||^2}_{\text{vertices belong to hidden part}} + \underbrace{\sum_{q} ||\boldsymbol{y'}_q - (\boldsymbol{y}_q + \boldsymbol{a_q^v})||^2}_{\text{vertices inside projected polygon}}, \tag{13}$$

where $\mathscr{L}$ denotes Laplacian vertex coordinates of G_{i-1}, and $\boldsymbol{y'}$ denotes the vertex coordinates after Laplacian deformation. We expect to realistically simulate the observed areas while the shape of the invisible part is less certain. So, it is a reasonable combination, to use FEM for the approximation of the deformation near the observed areas and the Laplacian deformation for the remaining part.

Besides, prior knowledge about the surface could be added to our stitching algorithm. Since the boundary of our dataset is fixed, we could set their displacement as 0 both in FEM calculation and Laplacian deformation. To prove that our algorithm can complete a surface without any prior, we do not use this prior in this paper.

5 Experimental Evaluation

We implement FEM with the third-party library *eigen* [2]. We also use the third-party library *itk* [4] for the Laplacian deformation,

Figure 5. A, B: The completed global surface at frames 350 and 1034. **C**, **top**: The completed surface of frame 81 with $\nu = 0.3$ and $h = 0.1$; **middle**: The global surface of frame 81 with $\nu = 0.2$ and $h = 0.1$; **bottom**: The ground truth.

and this operation can be accomplished within 0.5 seconds. In our experiment, it takes in total ∼40 seconds to stitch a newly incoming frame to the already available reconstruction with the Intel Core i7 Processor (4x 2.67 GHz). The runtime can be further improved by using a dedicated FEM library.

We evaluate the proposed algorithm on our new dataset (Sec. 3). Since we focus only on the stitching problem in this paper, we believe that a successful experiment with our synthetic dataset with simulated surface deformations is convincing to prove the effectiveness of our algorithm. Thus, given individual reconstructions and camera poses at different moments, a surface is completed frame by frame. To compare the completed surface reconstructed by our algorithm against the ground truth, in the first step, we translate both point clouds so that their centroids coincide with the origin of the coordinate system. Next, we report the 3D error between the completed surface and the ground truth, which is defined as

$$e_{3D} = \frac{\left\|\mathbf{S_{GT}}^{\mathbf{i}} - \mathbf{S}^{\mathbf{i}}\right\|_{\mathscr{F}}}{\left\|\mathbf{S_{GT}}^{\mathbf{i}}\right\|_{\mathscr{F}}}, \tag{14}$$

where $\mathbf{S_{GT}}^{\mathbf{i}}$ and $\mathbf{S}^{\mathbf{i}}$ are the ground truth and the completed surface of the i-th frame, respectively, and $\|\cdot\|_{\mathscr{F}}$ stands for Frobenius norm.

Note that the thickness and ν of the surface are not available as a prior. We choose $h = 0.1$ and $\nu = 0.2$ as the default setting. The bigger the ν, the more incompressible is the material [11]. This means that the volume of the 3D wedge elements is harder to change with increasing ν. The boundary of the surface in our dataset is fixed. Thus, the deformation of the surface causes the change of its volume. Recall that in our FEM solution, we assume that no external force is applied to the hidden part. Therefore, its volume is harder to be changed than the volume of the observed part. We choose a smaller ν to make it more compressible. The surface with smaller ν is significantly smoother and fits the ground truth better (see Fig. 5).

The recovered surface can be divided into three types of regions: 1) ground truth reconstruction from the observed part, 2) the nearby hidden part, which is predicted with FEM, and 3) the remainder of the hidden part, which is approximated with Laplacian deformation. According to e_{3D} summarised in Fig. 6, the error in the region which includes the first and second region types (the grey line), fluctuates due to the uncertainty in the hidden part. However, it is independent of the area of the hidden part and is accurate. While the error of the whole recovered surface (the blue line), which includes all types of regions, is proportional to the area of the hidden part.

6 Conclusion

We introduce a new difficult problem, *i.e.,* monocular non-rigid surface completion, and propose a first physically-inspired approach to address it. Experiments show that our method obtains a smooth, complete, physically plausible and accurate global surface, given camera poses and locally observed surface parts from our synthetic non-rigid surface stitching dataset. Even if the forces applied to the hidden parts are unknown, the parts of the surface calculated with FEM are accurately stitched, and the observed geometry is preserved. To keep the runtime in the feasible bounds, we apply Laplacian deformation modelling for the hidden parts. In future work, we plan to integrate our algorithm into a framework with automatic camera pose estimation and monocular surface regression and test it for real endoscopic data.

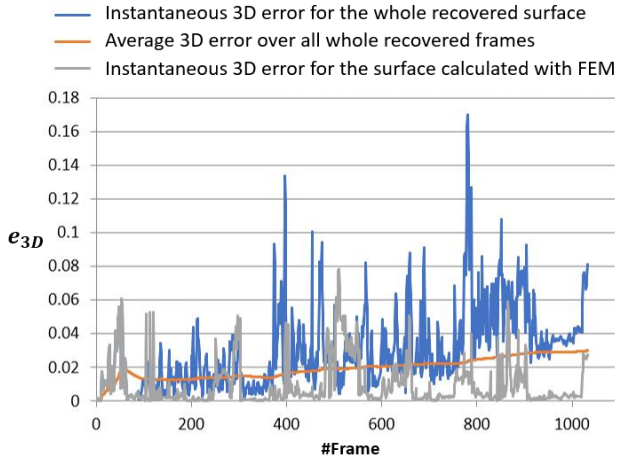

Figure 6. e_{3D} as a function of the number of integrated surfaces corresponding to individual frames. The values at frame 1034 (the last frame) are the final errors for the entire completed global surface.

Our **supplementary material** contains a video demonstrating surface stitching with the proposed algorithm. You can download the dataset for non-rigid surface completion on our web page.

Acknowledgements

This work was supported by the projects DYNAMICS (01IW15003) and VIDETE (01IW18002) of the German Federal Ministry of Education and Research (BMBF). We are thankful to Jiayi Wang and Sikang Yan for the useful feedback and discussions.

A Appendix

Here, we provide more details about the functions and matrices used in FEM. According to the isoparametric concept, any points $\boldsymbol{y}$ in a patch can be mapped from the natural into the Cartesian coordinate system with

$$\boldsymbol{y}(x,y,z) = \sum_i N_i(\eta,\xi,\zeta)\,\boldsymbol{y_i}(x_i,y_i,z_i)\ , \tag{15}$$

where $\boldsymbol{y_i}$ is the nodal point of the patch, N_i denotes the shape basis function and $\mathbf{J} = \partial \boldsymbol{y}/\partial \boldsymbol{\xi}$ is the Jacobian matrix of this transformation. In this paper, we use 3D wedge elements, whose shape basis function N_i can be expressed in the natural coordinate system as

$$\begin{cases} N_1^b = \frac{1}{2}(1-\xi-\eta)(1-\zeta) \\ N_2^b = \frac{1}{2}\xi(1-\zeta) \\ N_3^b = \frac{1}{2}\eta(1-\zeta) \\ N_4^b = \frac{1}{2}(1-\xi-\eta)(1+\zeta) \\ N_5^b = \frac{1}{2}\xi(1+\zeta) \\ N_6^b = \frac{1}{2}\eta(1+\zeta) \end{cases} . \tag{16}$$

Suppose the vector $\boldsymbol{N_e}$ is defined as

$$\boldsymbol{N_e} = [N_1^b, N_2^b, N_3^b, N_4^b, N_5^b, N_6^b]^T. \tag{17}$$

The derivative of N_i can be transformed into the Cartesian coordinate system by

$$\begin{bmatrix} \boldsymbol{N}_{e,x}^T \\ \boldsymbol{N}_{e,y}^T \\ \boldsymbol{N}_{e,z}^T \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \boldsymbol{N}_{e,\xi}^T \\ \boldsymbol{N}_{e,\eta}^T \\ \boldsymbol{N}_{e,\zeta}^T \end{bmatrix}. \tag{18}$$

The strain-displacement matrix $\mathbf{B}$ can be calculated as

$$\mathbf{B} = \begin{bmatrix} \boldsymbol{N}_{e,x}^T \otimes [1,0,0]^T \\ \boldsymbol{N}_{e,y}^T \otimes [0,1,0]^T \\ \boldsymbol{N}_{e,z}^T \otimes [0,0,1]^T \\ \boldsymbol{N}_{e,x}^T \otimes [0,1,0]^T + \boldsymbol{N}_{e,y}^T \otimes [1,0,0]^T \\ \boldsymbol{N}_{e,y}^T \otimes [0,0,1]^T + \boldsymbol{N}_{e,z}^T \otimes [0,1,0]^T \\ \boldsymbol{N}_{e,x}^T \otimes [0,0,1]^T + \boldsymbol{N}_{e,z}^T \otimes [1,0,0]^T \end{bmatrix}. \tag{19}$$

The behaviour matrix $\mathbf{D}$ can be expressed as

$$\mathbf{D} = C \begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix}, \tag{20}$$

with the constant $C = \dfrac{E}{(1+\nu)(1-2\nu)}$, where E denotes the Young's modulus and ν denotes the Poisson's ratio.

REFERENCES

[1] https://unity3d.com/.

[2] eigen. `https://eigen.tuxfamily.org/dox/index.html`.

[3] Endoscopic video datasets of ICL. http://hamlyn.doc.ic.ac.uk/vision/.

[4] itk. `https://itk.org/ITK/project/about.html`.

[5] A. Agudo, J. Montiel, B. Calvo, and F. Moreno-Noguer. Mode-shape interpretation: Re-thinking modal space for recovering deformable shapes. In *Winter Conference on Applications of Computer Vision (WACV)*, 2016.

[6] A. Agudo, J. M. M. Montiel, L. Agapito, and B. Calvo. Online dense non-rigid 3d shape and camera motion recovery. In *British Machine Vision Conference (BMVC)*, 2014.

[7] A. Agudo, F. Moreno-Noguer, B. Calvo, and J. Montiel. Real-time 3d reconstruction of non-rigid shapes with a single moving camera. *Computer Vision and Image Understanding (CVIU)*, 153:37–54, 2016.

[8] M. D. Ansari, V. Golyanik, and D. Stricker. Scalable dense monocular surface reconstruction. In *International Conference on 3D Vision (3DV)*, pp. 78–87, 2017.

[9] A. Bartoli, V. Gay-Bellile, U. Castellani, J. Peyras, S. Olsen, and P. Sayd. Coarse-to-fine low-rank structure-from-motion. In *Computer Vision and Pattern Recognition (CVPR)*, 2008.

[10] K. Bathe and H. Saunders. *Finite element procedures in engineering analysis*. American Society of Mechanical Engineers, 1984.

[11] W. Becker and D. Gross. *Mechanik elastischer Körper und Strukturen*. Springer-Verlag, 2013.

[12] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3d shape from image streams. In *Computer Vision and Pattern Recognition (CVPR)*, vol. 2, pp. 690–696, 2000.

[13] Y. Dai, H. Li, and M. He. A simple prior-free method for non-rigid structure-from-motion factorization. *International Journal of Computer Vision (IJCV)*, 107(2):101–122, 2014.

[14] A. J. Davison. Real-time simultaneous localisation and mapping with a single camera. In *International Conference on Computer Vision (ICCV)*, 2003.

[15] J. Engel, T. Schöps, and D. Cremers. Lsd-slam: Large-scale direct monocular slam. In *European Conference on Computer Vision (ECCV)*, pp. 834–849. Springer, 2014.

[16] J. Fayad, L. Agapito, and A. Del Bue. Piecewise quadratic reconstruction of non-rigid surfaces from monocular sequences. In *European Conference on Computer Vision (ECCV)*, pp. 297–310, 2010.

[17] R. Garg, A. Roussos, and L. Agapito. Dense variational reconstruction of non-rigid surfaces from monocular video. In *Computer Vision and Pattern Recognition (CVPR)*, pp. 1272–1279, 2013.

[18] V. Golyanik, A. Jonas, and D. Stricker. Consolidating segmentwise non-rigid structure from motion. In *Machine Vision Applications (MVA)*, 2019.

[19] V. Golyanik and D. Stricker. Dense batch non-rigid structure from motion in a second. In *Winter Conference on Applications of Computer Vision (WACV)*, pp. 254–263, 2017.

[20] Y. Kita. Elastic-model driven analysis of several views of a deformable cylindrical object. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 18(12):1150–1162, 1996.

[21] G. Klein and D. Murray. Parallel tracking and mapping for small ar workspaces. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 225–234, 2007.

[22] T. McInerney and D. Terzopoulos. A dynamic finite element surface model for segmentation and tracking in multidimensional medical images with application to cardiac 4d image analysis. *Computerized Medical Imaging and Graphics*, 19(1):69–83, 1995.

[23] M. Müller, B. Heidelberger, M. Hennix, and J. Ratcliff. Position based dynamics. *Journal of Visual Communication and Image Representation*, 18(2):109–118, 2007.

[24] R. A. Newcombe and A. J. Davison. Live dense reconstruction with a single moving camera. In *Computer Vision and Pattern Recognition (CVPR)*, pp. 1498–1505, 2010.

[25] C. Russell and L. Agapito. Dense non-rigid structure from motion. In *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT)*, 2012.

[26] C. Russell, J. Fayad, and L. Agapito. Energy based multiple model fitting for non-rigid structure from motion. In *Computer Vision and Pattern Recognition (CVPR)*, pp. 3009–3016, 2011.

[27] S. Shimada, V. Golyanik, C. Theobalt, and D. Stricker. IsMo-GAN: Adversarial learning for monocular non-rigid 3d reconstruction. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2019.

[28] O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C. Rössl, and H.-P. Seidel. Laplacian surface editing. In *Eurographics Symposium on Geometry Processing*, pp. 175–184, 2004.

[29] A. H. Stroud. *Approximate calculation of multiple integrals*. Prentice-Hall, 1971.

[30] J. Taylor, A. D. Jepson, and K. N. Kutulakos. Non-rigid structure from locally-rigid motion. In *Computer Vision and Pattern Recognition (CVPR)*, pp. 2761–2768, 2010.

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

[32] A. Wendel, M. Maurer, G. Graber, T. Pock, and H. Bischof. Dense reconstruction on-the-fly. In *Computer Vision and Pattern Recognition (CVPR)*, pp. 1450–1457, 2012.

[33] J. Xiao, J.-x. Chai, and T. Kanade. A closed-form solution to non-rigid shape and motion recovery. In *European Conference on Computer Vision (ECCV)*, pp. 573–587, 2004.

[34] O. C. Zienkiewicz and R. L. Taylor. *The finite element method*, vol. 3. McGraw-hill London, 1977.