A Template-based 3D Reconstruction of Colon Structures and Textures from Stereo Colonoscopic Images

Shuai Zhang, Liang Zhao, *Member, IEEE,* Shoudong Huang, *Senior Member, IEEE,* Qi Hao, and Menglong Ye

***Abstract*—This paper presents a framework for 3D reconstruction of colonic surface using stereo colonoscopic images. Due to the limited overlaps between consecutive frames and the nonexistence of large loop closures during a normal screening colonoscopy, the state-of-art simultaneous localization and mapping (SLAM) algorithms cannot be directly applied to this scenario, thus a colon model segmented from CT scans is used together with the colonosocopic images to achieve the colon 3D reconstruction with high accuracy. The proposed framework includes 3D scan (point cloud with RGB information) reconstruction from stereo images, a visual odometry (VO) based camera pose initialization module, a 3D registration scheme for matching texture scans to the segmented colon model, and a barycentric-based texture rendering module for mapping textures from colonoscopic images to the reconstructed colonic surface. A realistic simulator is developed using Unity to simulate the procedures of colonoscopy and used to provide experimental datasets in different scenarios. Experimental results demonstrate the good performance of the proposed 3D colonic surface reconstruction method in terms of accuracy and robustness. Currently, the framework requires a pre-operative colon model as the template for colon reconstruction and can reconstruct 3D colon maps when the colon has no large deformation and the colon structure is clearly visible. The datasets used in this paper and the developed simulator are made publicly available for other researchers to use[1].**

***Index Terms*—colonoscopy, 3D reconstruction, SLAM, texture, simulator.**

I. Introduction

COLORECTAL cancer is the second most commonly occurring cancer in women and the third most commonly occurring cancer in men all over the world. Colonoscopy is considered as the gold-standard method to detect changes and remove precancerous polyps in the large intestine. However, recent studies report that around 20% of the abnormalities (polyps, abnormal lesions and cancer) are missed [1] and approximately 60% of colorectal cancer cases detected after optical colonoscopy are closely associated with missed polyps and lesions [2].

There are two main reasons for missed abnormalities: i) the areas where abnormalities reside are never visualized by the colonoscopy, often called non-visualization; ii) these areas are inspected but the abnormalities are not recognized. Non-visualization mainly results from the lack of orientation changes of the endoscope to the full circumference of the colon [3] and the occlusion from the structural complexity of colon [4]. Non-recognition is due to the difficulty to detect abnormalities from video alone. Although virtual colonoscopy is a non-invasive, radiographic method of visualizing the colon by flying through the segmented colon model, it has difficulty in detecting $5mm$ or less size lesions and flat lesions and meanwhile the patient will be exposed to a certain dose of radiation [5]. Therefore, the traditional optical colonoscopy will be ultimately needed to detect very small and flat colon lesions and remove polyps or any abnormalities identified from virtual colonoscopy.

If a 3D map of the entire colon internal surface with detailed textures can be reconstructed during the colonoscopy procedures, the following two main potential benefits can be achieved: i) uninspected areas can be shown on this map and gastroenterologists can navigate the endoscope to these missing areas to ensure more colon surfaces are inspected; ii) the detailed textures on the reconstructed map can help gastroenterologists to inspect abnormalities offline.

Reconstructing a complete 3D virtual colon from a sequences of colonic images has to deal with the following technical challenges:

1) Special geometric structure. The human colon has long and narrow tubular structure with many colon folds and a lot of turns, which make it impossible to have large loop closures and difficult to observe the back side of colon folds during a colonoscopy screening. This is the main reason for deficient coverage in a normal colonoscopy and large drift in the colon reconstruction if directly using visual SLAM algorithms;
2) Reconstruction with detailed textures. Detailed texture information is necessary for the gastroenterologists to identify abnormalities. Accurate texture matching requires highly accurate camera pose estimation and highly accurate scan registration which is very difficult to achieve using information from images only;
3) Colonoscopy datasets with ground truth. Complete datasets of colonoscopic images with ground truth of

This work is supported by Australian Research Council Discovery Project (No. DP200100982), the Science and Technology Innovation Committee of Shenzhen City (No. GJHZ20170314114424152) and the Nanshan District Science and Technology Innovation Bureau (No. LHTD20170007).

S. Zhang, L. Zhao and S. Huang are with Centre for Autonomous Systems, University of Technology Sydney, Australia (e-mail: shuai.zhang@student.uts.edu.au).

S. Zhang is also with Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, China.

Q. Hao is with Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, China.

M. Ye is with Hamlyn Centre for Robotic Surgery, Imperial College London, United Kingdom.

[1]https://github.com/zsustc/colon_reconstruction_dataset

camera poses and depths are critical to develop and validate colon reconstruction algorithms. These datasets are challenging to generate since depth sensors and absolute 3D positioning sensors are impractical to be coupled to a colonoscope;

4) Camera motion estimation. In a normal colonoscopy, the tiny camera attached to the end of a colonsocope moves fast with significant view changes and the deformation of colon itself creates a further challenge to recover the camera motion; how to improve the accuracy and robustness of camera motion estimation becomes critical.

In this work, we develop a template-based framework for 3D reconstruction of colon structures and textures from stereo colonoscopic RGB images, which mainly addresses the first three challenges listed above. The template model segmented from preoperative colon CT scans is used as a global reference to increase the stability and accuracy of the camera pose estimation. The joint photometric and geometric optimization pipeline optimizes the camera poses to address the inconsistency of texture matching problem. Based on the estimated camera poses, the point correspondences between the reconstructed scan and the colon model are extracted and used to map the detailed textures from the corresponding image to the registered areas on the colon model. A realistic colonoscopy simulator which can generate binocular colonoscopic RGB images with the ground truth of camera poses is developed to evaluate the proposed framework. The fourth challenge listed above, regarding the deformation of colon structure, has not been well addressed in the current work yet. We will deal with that challenge in our future work.

II. Related Works

Some works have been developed to reconstruct colonic surface based on the advances in computer vision and image processing techniques. Some methods try to generate a 2D visibility map of internal colonic surface, and some other studies have been focused on generating a small portion of the 3D colon surface. Karargyris et al. [6] used shape from shading (SfS) algorithm [7] to compute colon structures from the brightness of colon surface. Koppel [8] et al. and Chen et al. [9] used shape from motion (SfM) algorithm [10] to reconstruct a small portion of 3D colonic surface with textures from multiple sequential colonoscopic images. Kaufman et al. [11] used the SfS algorithm [7] to reconstruct partial colonic surfaces from individual frames, then utilized the camera poses estimated by the SfM [10] to integrate the partial flattened surfaces obtained from several consecutive frames to form a relatively large surface. However, the SfS algorithm would incorrectly express the colon lumen as relatively far surface, not a tubular structure, and the SfM algorithm requires very slow camera motion to estimate the camera poses.

There are other advanced approaches with restrictive assumptions. Zhou et al. [12] adopted an optical flow based method to reconstruct small colon segments with the assumptions that the neighboring folds in an image are not occluded and that the colon fold contours are circular in nature. However, partial occlusion of folds is very common and the transverse, ascending and descending segments of colon have no well circular characteristic. Hong et al. [13] took the advantage of the tubular nature of the colon to estimate colon folds and only reconstructed a colon segment from a single colonoscopic image. Armin et al. [14] fitted a cylinder model to the colon structure generated by 3D pseudo stereo vision and unrolled the fitted model to a 2D band image. Then the estimated camera poses were used as initial values to register these 2D band images together to build a large 2D visibility map, but the generated 2D map was less intuitive than a 3D dense reconstruction. Despite the fact that remarkable progress has been made in this field, all of the research has focused on 3D or 2D surface reconstruction of very small parts of colon.

Currently, the popularization of visual SLAM algorithms which can be classified into sparse [15], [16], semi-dense [17], [18], [19], [20] and dense reconstruction [21], [22], [23] have inspired researchers to apply SLAM technology to recover the 3D structures of the human colon. Chen et al. [24] trained an adversarial depth estimation neural network in a supervised approach from synthetic dataset of a phantom, then input monocular images paired with depth estimation from the neural network to the ElasticFusion [21] to stitch depth images and reconstruct a dense surfel point cloud. However, the metric accuracy of estimated camera poses and reconstruction is not given. Also, it is not suitable to directly apply ElasticFusion on a colonoscopy since it requires slow camera motion. Ma et al. [25] used sparse depth estimated from SfM as a ground truth proxy to train a recurrent neural network for depth and the camera pose estimation. Then the output of the network is passed into the direct sparse odometry SLAM system [17] for refinement.

In general, these SLAM systems do not need a template, but require slow camera motion and large loop closure to reduce the drift in the camera pose estimation. Although promising results can be achieved for other scenarios, these algorithms are seldom directly applied in colon reconstruction scenarios mainly due to the following reasons. The human colon has a tubular shape and the colon inner space is limited, so the camera attached to the colonoscope is tiny and its field of view is limited by the tubular shape. During a normal colonoscopy screening, the camera moves fast relative to the colonic surface and causes large inter frame motions and less frame overlaps, and this will cause inaccurate or even failed camera pose estimations. Meanwhile, there is no large loop closures in a normal colonoscopy procedure since the colonoscope is withdrawn from the cecum (the distal end of colon) to the rectum (the proximal start of colon) and this will cause a large drift for the camera pose estimation and scene reconstruction. Furthermore, the poor texture of colonoscopic images is a challenge for feature-point-based SLAM methods, e.g., ORB-SLAM2 [15]. All these will lead to misalignment in textures on the reconstructed colon map.

In this work, we aim to develop a framework for reconstructing a 3D map of the internal surface of the colon using stereo colonoscopy. The input of our framework is a sequence of stereo colonoscopic images and a corresponding colon mesh model segmented from pre-operative CT scans, and the final output of the framework is the reconstructed and texturized 3D

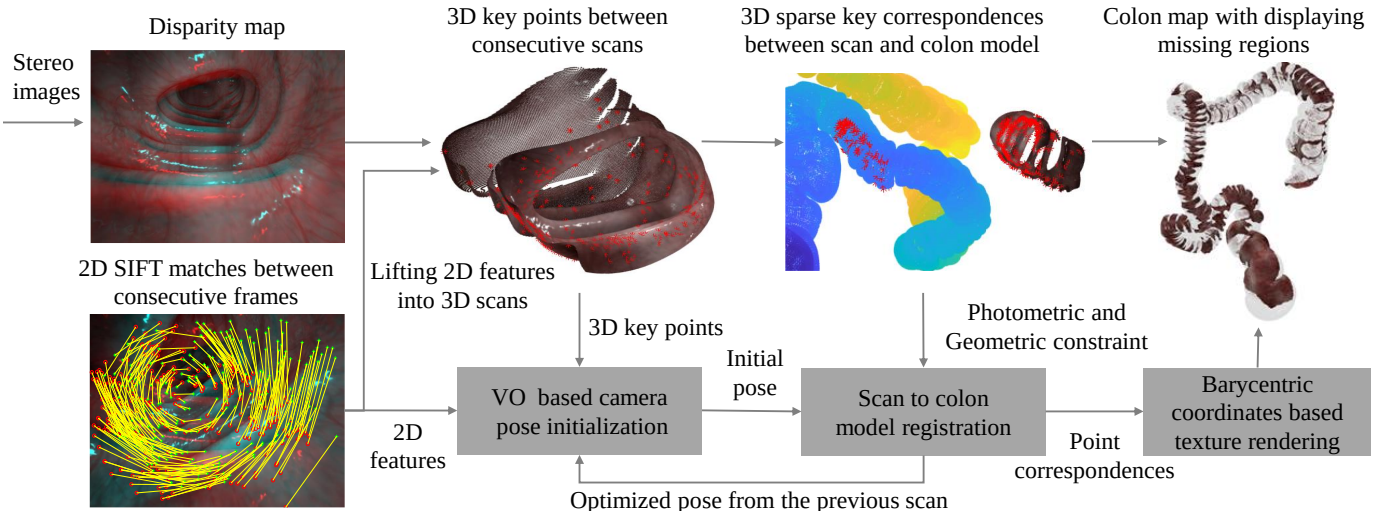


Fig. 1: The framework of reconstructing and texturing 3D colon structures.

colon maps. Specifically, this work will focus on resolving the following problems assuming no much deformation happens:

1) How to robustly estimate the motion of the camera inside a human colon during colonoscopy;
2) How to precisely reconstruct a complete 3D virtual colon map from stereo colonoscopic images;
3) How to map the texture from colonoscopic images to the reconstructed map.

The proposed framework is validated on datasets of different scenarios from the developed colonoscopy simulator and the accuracy of the reconstruction and texture rendering is within $[-0.04, 0.04]$ rad for Euler angles, and $[-0.5, 0.5]$ mm for translation.

The rest of the paper is organized as follows: Section III describes the proposed framework and the development of simulation platform. Section IV presents the technical details of the proposed method. Section V provides validation and experimental results. Section VI concludes the paper and outlines our future work.

III. Framework Overview and Simulation Platform

A. Framework overview

Fig. 1 illustrates the proposed framework for reconstructing and texturing a 3D colon map from stereo colonoscopic images, which includes 3D scan reconstruction from stereo images, VO based camera initialization, geometric and photometric scan to colon model registration and barycentric-based texture rendering.

The developed colonoscopy simulator works in a way similar to a real colonoscopy, it starts to take images during the withdraw processing of the colonoscope, which means the reconstruction processing starts from the distal end of the human colon. Therefore, the 3D colon map is initially reconstructed by the geometric-only ICP registration between the first estimated scan and the colon model. Then, each time when a new frame is incorporated, the relative pose between the current scan and the previous scan is estimated by the VO module. As a result, this relative pose combined with the optimized pose between the previous scan and the colon model estimated in the last step is used as the initial guess of the relative pose between the current scan and the colon model.

This initial guess sets the current scan to a good initial position for registration between itself and the colon model. After that, the developed geometric and photometric based scan registration is applied between the current scan and the colon model. Hence, the pose of current scan is optimized and dense point correspondences between the scan and the vertices of the colon model are established from the proposed registration processing. Based on the established point correspondences, texture coordinates between 2D color images and the colon model are extracted using the barycentric-based mapping algorithm. Section IV will explain all the modules in details.

B. Simulation Platform

To develop algorithms for recovering the 3D structures of the human colon in colonoscopy procedures or to train a depth prediction network for depth estimation of colonoscopic images, both synthetic and real clinical data are crucial. However, due to reasons of patient privacy, human and animal rights, guarantee of operation safety and conflict of interest, it is hard to obtain any dataset with complete or segmental colonoscopic images with corresponding ground truth depth and camera poses. Therefore, we developed a realistic simulator to simulate colonoscopy procedures. To encourage research in the field, we have made the simulator and datasets of synthetic colonoscopy images with corresponding depth and camera poses publicly available [1].

Fig. 2 shows the schematic diagram of the developed colonoscopy simulator framework. The framework mainly consists of 3D colon mesh model segmentation and optimization, creation of a 2D image texture that wraps around the segmented colon model and implementation of the virtual visualization and interaction system.

The triangular 3D colon surface mesh is segmented from a set of 2D colon CT scans using 3D Slicer and then be sculpted

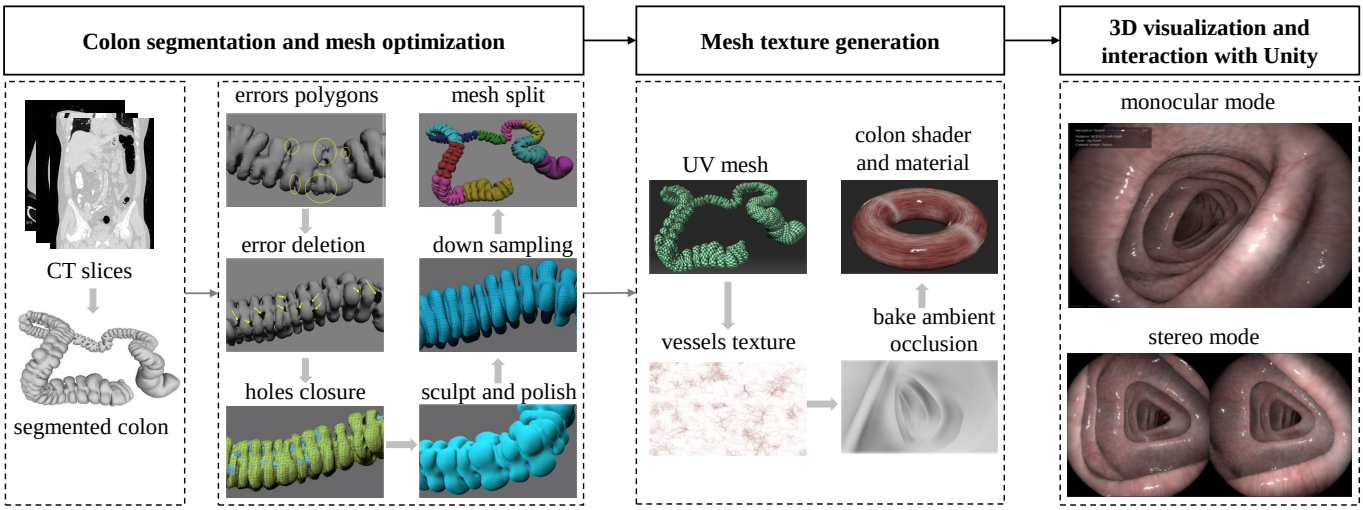


Fig. 2: Schematic diagram of the developed colonoscopy simulator framework. The length of CT segmented colon is about 1.5 meters and its bounding box size is $36cm \times 26cm \times 14.9cm$

and polished using the software Softimage and ZBrush. To texture the colon mesh and make the simulator as realistic as possible, the UV-mapping tool Unfold3D is used to create the UV ("U" and "V" denote the axes of the 2D texture image) map for the colon mesh and this UV map will be used in the Unity platform for applying the vessels texture over the mesh. Then, seamless and tillable textures of the blood vessels, perlin noise and mucous are created in the Photoshop and added into the 2D texture image. The blood vessels textures are extracted from the real colon images which are downloaded from Google Images directly. After that, the created UV map is imported into Softimage to bake ambient occlusion into the mesh vertices colors and this can help to mix some deep shadow and realistic look to the shader and material of the colon. Last, the visualization and interaction interface is mainly built by the Playmaker plugin of Unity3D. The Playmaker uses Finite State Machines (FSMs) to add functions to a game object. In our simulator, the game object Action Manager with many FSMs is used to trigger important events on the colon mesh model. The events mainly include user interface load, virtual camera creation, camera path creation, start and stop camera movement, manually control of camera, data capture and saving, etc. There are also other plugins of Unity that have been used in the simulator development, "Easy Save 3" is used to provide save and load functions to the simulator, the "StandaloneFileBrowser" is used for creating file open/save browser window, "Post Processing Stack" is used to add some visual effects such as Motion Blur, Vignette, Bloom and Grain and "AmplifyShaderEditor" is used to create the shader of surface of the colon. Therefore, the visualization module can create 3D virtual visualization environment of the colon model and provide volume-based rendering of endoscopic views during the virtual camera's flight through the colon model. For the interaction part, it allows the user to manually control cameras using buttons on a keyboard and output simulated colonoscopic images together with ground truth of camera poses and image depths. To prevent the camera from moving through the colonic surface, a mesh collider which roughly defines the shape of the colon mesh is built for the purposes of physical collisions.

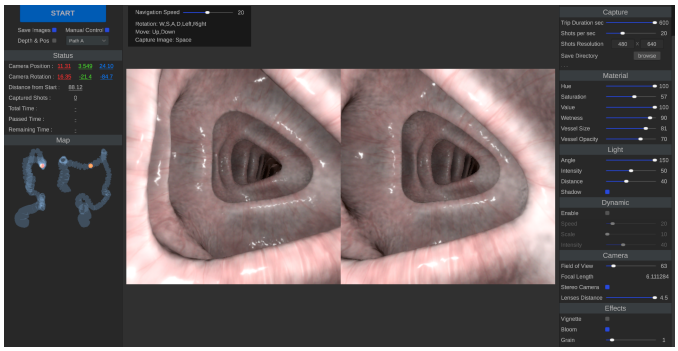


Fig. 3: The interface of the developed colonoscopy simulator.

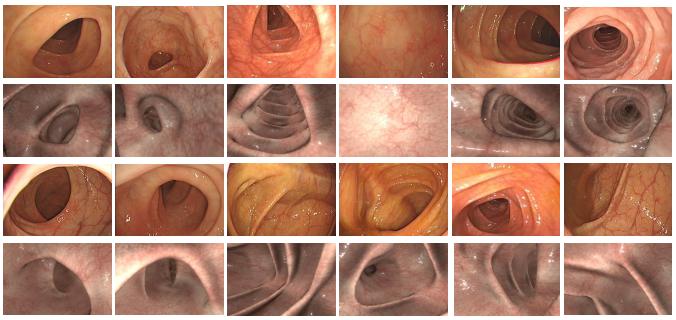


Fig. 4: Simulated and real colonoscopic images. The first and third rows show real images generated from a colonoscopy, the second and the last row show simulated images generated from our simulator.

Fig. 3 shows the interface of the developed colonoscopy simulator. In the user interface of the simulator, some specific parameters of the monocular or stereo camera or the visualization environment can be adjusted, such as the hue, saturation

and wetness of colon inside, and the light source configuration for specular reflection. The field view of camera can be set within $[50°, 150°]$, and corresponding range of focal length is $[1, 8]$ mm, the baseline of stereo camera can be set within $[0.5, 4.5]$ mm.

Fig. 4 shows visual comparison between real colonoscopic images with clearly structure and the simulated images generated by the developed simulator. The motion blur and image distortion effects are currently not taken into consideration in the developed simulator.

IV. Methodology

A. 3D scan reconstruction from stereo images

The semi-global matching algorithm (SGM) [26] is used as the scan reconstruction method. First, a disparity map is computed from a pair of rectified stereo images using SGM algorithm. Then, the 3D coordinates of the pixel points in the camera coordinate frame are computed to reconstruct a 3D scan and each 3D scan has one to one correspondence to a corresponding 2D image. Fig. 5 shows an example of ground truth scans and corresponding reconstructed 3D scans, respectively.

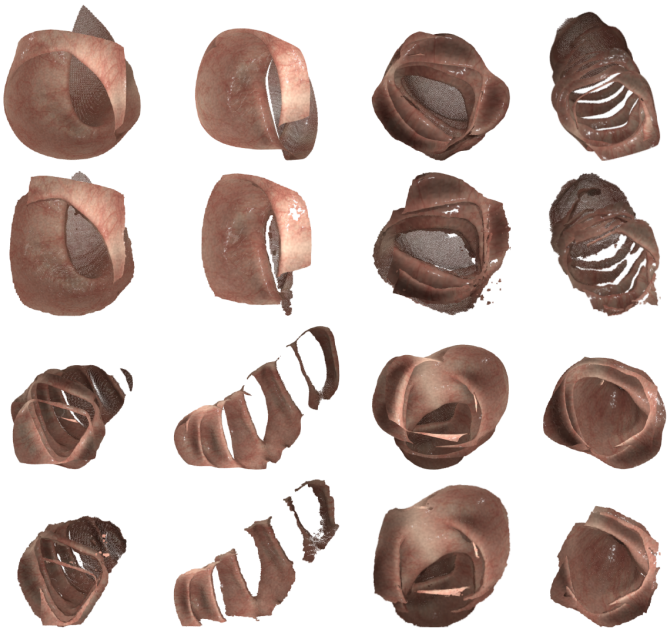


Fig. 5: Examples of reconstructed scan and ground truth. The first and third rows show ground truth scans, the second and last rows show corresponding scans from stereo.

B. Sparse key correspondences and camera pose initialization

In the VO based camera motion initialization module, first, two disparity maps are computed from the current and the previous pairs of stereo images and the corresponding two 3D scans can be computed from the disparity maps. Then, 2D Scale-Invariant Feature Transform (SIFT) features are extracted and matched between the consecutive left images. For an accurate motion estimation, Random Sample Consensus (RANSAC) is used to remove outliers from the set of 2D SIFT feature correspondences. After that, these 2D SIFT features are migrated into 3D scans by tracing the pixel indices of these 2D SIFT points in their corresponding 3D scans, and a set of 3D key point correspondences (anchor points) between the two scans are acquired. In our experiments, around 100-200 successfully matched SIFT features between two consecutive images could be obtained and the mean rate of outliers is around 8-9%.

Since the 3D-to-2D method is more accurate than 3D-to-3D methods [27] and the RANSAC algorithm can help to remove outliers. After acquiring 2D SIFT feature point correspondences and corresponding 3D anchor point correspondences from the SIFT approach. The perspective-three-point (P3P) algorithm in conjunction with the RANSAC algorithm are applied [28] on 3D-to-2D point correspondences to estimate the camera motion robustly. This relative pose between the current scan and the previous scan is then combined with the optimized pose of the previous scan and used as the initial pose of the current scan in the scan-to-model registration processing described in Section IV-C.

C. Scan to colon model registration

One can build the 3D colon map by incrementally registering all the scans together, but the errors of poses estimation accumulate during scan to scan registration. Also, only the geometric constraint applied on the registration causes inconsistency of texture matching in the overlapping region of two scans. To address these problems, we formulate an objective function by combining the geometric constraint and the photometric feature constraint:

$$E(T) = (1 - \sigma)E_G(T) + \sigma E_F(T), \tag{1}$$

where $E_G(T)$ is the geometric term of the objective function and the $E_F(T)$ is the photometric feature term provided by the pair-wise 3D sparse anchor points generated from 2D SIFT features described in Section IV-B, $\sigma \in [0, 1]$ is the weight that balances the two terms. Here "$photometric$" is used to express that these constraints are from the texture information instead of the geometric structure. Our goal is to find the optimal transformation T that best aligns the reconstructed scan to the colon model.

The geometric term $E_G(T)$ sums all the squared distances between each source point $\mathbf{s}_i = [s_{ix}, s_{iy}, s_{iz}, 1]^T$ in a scan and the tangent plane at its closest point $\mathbf{d}_i = [d_{ix}, d_{iy}, d_{iz}, 1]^T$ in the colon model:

$$E_G(T) = \sum_i ((T \cdot \mathbf{s}_i - \mathbf{d}_i) \bullet \mathbf{n}_i)^2 \tag{2}$$

where $\mathbf{n}_i = [n_{ix}, n_{iy}, n_{iz}, 0]^T$ is the unit normal vector at $\mathbf{d}_i$, and "$\bullet$" denotes the dot product.

Similarly, the photometric term $E_F(T)$ sums all the point-to-point distances between the 3D anchor point $\mathbf{s}_j^f = [s_{jx}^f, s_{jy}^f, s_{jz}^f, 1]^T$ in a scan and its corresponding 3D anchor point $\mathbf{d}_j^f = [d_{jx}^f, d_{jy}^f, d_{jz}^f, 1]^T$ in the colon mesh, provided in Section IV-B:

$$E_F(T) = \sum_j (T \cdot \mathbf{s}_j^f - \mathbf{d}_j^f) \bullet (T \cdot \mathbf{s}_j^f - \mathbf{d}_j^f). \tag{3}$$

D. Optimization

A 3D rigid transformation ΔT can be expressed as:

$$\Delta T = \mathbf{t}(t_x, t_y, t_z) \cdot R_z(\gamma) \cdot R_y(\beta) \cdot R_x(\alpha) \tag{4}$$

where $\mathbf{t}(t_x, t_y, t_z) = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$, t_x, t_y, t_z are the corresponding translation component along x-axis, y-axis and z-axis, respectively. $R_x(\alpha)$, $R_y(\beta)$ and $R_z(\gamma)$ are rotations of α, β and γ radians around the x-axis, y-axis and z-axis, respectively. When $\alpha, \beta, \gamma \approx 0$, then ΔT is approximated by:

$$\Delta T \approx \begin{bmatrix} 1 & -\gamma & \beta & t_x \\ \gamma & 1 & -\alpha & t_y \\ -\beta & \alpha & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5}$$

Therefore, we minimize the objective function $E(T)$ in (1) by iteratively linearizing the rigid transformation matrix T [29]. Thus, at the k^{th} iteration, T is approximated by a linear function of $\mathbf{x}$:

$$T \approx \begin{bmatrix} 1 & -\gamma & \beta & t_x \\ \gamma & 1 & -\alpha & t_y \\ -\beta & \alpha & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot T^k \tag{6}$$

where $\mathbf{x} = [\alpha, \beta, \gamma, t_x, t_y, t_z]^T$, T^k is the transformation estimated in the last iteration and is used to update all the source points described in (3). Then, we can rewrite (1) and compute $\mathbf{x}$ by solving the following least squares problem:

$$\min_{\mathbf{x}} (1-\sigma)|A_1\mathbf{x} - \mathbf{b}_1|^2 + \sigma|A_2\mathbf{x} - \mathbf{b}_2|^2 \tag{7}$$

where $(A_1|\mathbf{b}_1)$ and $(A_2|\mathbf{b}_2)$ are the augmented matrices of the liner expression of $E_G(T)$ and $E_F(T)$ respectively, both evaluated at T^k. Given N_1 pairs of point correspondences in $E_G(T)$, A_1 is an N_1 by 6 matrix and $\mathbf{b}_1$ is an N_1 by 1 vector:

$$A_1 = \begin{bmatrix} [\hat{\bar{\mathbf{s}}}_1 \times \hat{\mathbf{n}}_1]^T, \hat{\mathbf{n}}_1^T \\ \cdots \\ [\hat{\bar{\mathbf{s}}}_i \times \hat{\mathbf{n}}_i]^T, \hat{\mathbf{n}}_i^T \\ \cdots \\ [\hat{\bar{\mathbf{s}}}_{N_1} \times \hat{\mathbf{n}}_{N_1}]^T, \hat{\mathbf{n}}_{N_1}^T \end{bmatrix}, \mathbf{b}_1 = \begin{bmatrix} [\mathbf{d}_1 - \bar{\mathbf{s}}_1] \bullet \mathbf{n}_1 \\ \cdots \\ [\mathbf{d}_i - \bar{\mathbf{s}}_i] \bullet \mathbf{n}_i \\ \cdots \\ [\mathbf{d}_{N_1} - \bar{\mathbf{s}}_{N_1}] \bullet \mathbf{n}_{N_1} \end{bmatrix} \tag{8}$$

where $\mathbf{n}_i = [\hat{n}_i^T, 1]^T$, $\bar{\mathbf{s}}_i = T^k \cdot \mathbf{s}_i$, $\bar{\mathbf{s}}_i = [\hat{\bar{\mathbf{s}}}_i^T, 1]^T$ and "$\times$" denotes the cross product.

Similarly, given N_2 pairs of point correspondences in $E_F(T)$, A_2 is a $3N_2$ by 6 matrix and $\mathbf{b}_2$ is a $3N_2$ by 1 vector:

$$A_2 = \begin{bmatrix} A_{21}^T \\ \cdots \\ A_{2j}^T \\ \cdots \\ A_{2N_2}^T \end{bmatrix}, \mathbf{b}_2 = \begin{bmatrix} \mathbf{b}_{21}^T \\ \cdots \\ \mathbf{b}_{2j}^T \\ \cdots \\ \mathbf{b}_{2N_2}^T \end{bmatrix}, \tag{9}$$

where

$$A_{2j} = \begin{bmatrix} 0 & \bar{s}_{jz}^f & -\bar{s}_{jy}^f & 1 & 0 & 0 \\ -\bar{s}_{jz}^f & 0 & \bar{s}_{jx}^f & 0 & 1 & 0 \\ \bar{s}_{jy}^f & -\bar{s}_{jx}^f & 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{b}_{2j} = \begin{bmatrix} d_{jx}^f - \bar{s}_{jx}^f \\ d_{jy}^f - \bar{s}_{jy}^f \\ d_{jz}^f - \bar{s}_{jz}^f \end{bmatrix}, \bar{\mathbf{s}}_j^f = T^k \cdot \mathbf{s}_j^f. \tag{10}$$

In each iteration, we calculate the augmented matrices, solve the least squares problem in (7), and update T by applying the incremental transformation $\mathbf{x}$ to T^k using (4). In the next iteration, we re-linearize T at T^{k+1} and repeat the process. Once the optimization processing is finished, the optimal pose is estimated and point correspondences between the scan and vertices of the colon model are established for texture rendering described in Section IV-E.

E. Texture mapping using barycentric coordinates

One can assign RGB color data from points in each scan to the corresponding vertices in the colon mesh, then color each pixel of a triangle face by interpolating between the colors of the three vertices in the colon mesh model. However, the texture in triangle faces will be blurry since the vertices in the colon mesh are much sparser than the point cloud in the scans and one vertex in the colon mesh may correspond to multiple points in a scan. Thus in this paper, as each 3D point in a scan

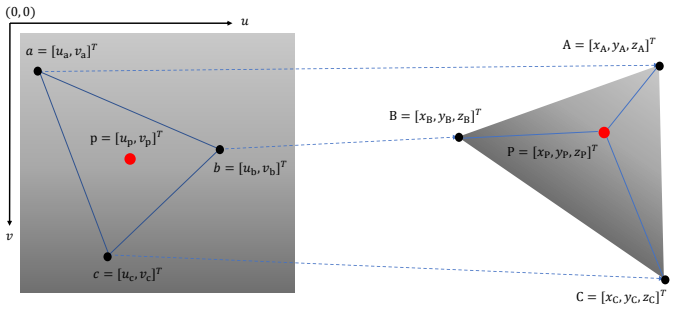


Fig. 6: Barycentric coordinates based texture mapping

corresponds to a 2D pixel in a 2D image when reconstructing the scan, we can extract a triangular texture region $\triangle\mathbf{abc}$ (where $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$ are the 2D locations of the three vertices of the triangle) in 2D images for each triangle $\triangle\mathbf{ABC}$ (where $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ are the corresponding 3D vertices of the triangle face) in the colon mesh, which is shown in Fig. 6. After that, we use the barycentric-based mapping technique [30] to map pixel color from the 2D texture region $\triangle\mathbf{abc}$ to the 3D triangle $\triangle\mathbf{ABC}$ face as

$$\mathbf{P} = \begin{bmatrix} \mathbf{A} & \mathbf{B} & \mathbf{C} \end{bmatrix} \cdot \lambda, \ \mathbf{p} = \begin{bmatrix} \mathbf{a} & \mathbf{b} & \mathbf{c} \end{bmatrix} \cdot \lambda \tag{11}$$

where $\lambda = [\lambda_1, \lambda_2, \lambda_3]^T$ with $\lambda_1 + \lambda_2 + \lambda_3 = 1$ indicates the barycentric coordinates of an arbitrary point $\mathbf{P}$ inside the triangle $\triangle\mathbf{ABC}$. Then, $\mathbf{P}$'s corresponding texture coordinates $\mathbf{p}$ can be determined by (11).

TABLE I: A brief summary of experimental data for evaluating the proposed framework

Case	Frames	Path	Case	Frames	Path
0	6000	manual	8	362	manual
1	259	auto	9	279	manual
2	260	auto	10	192	manual
3	260	auto	11	618	fully
4	260	auto	12	859	fully
5	260	auto	13	679	fully
6	260	auto	14	339	fully
7	845	manual	15	150	fully

The resolution of all collected colonoscopic images is 640×480, the baseline of stereo camera is set to $4.5mm$ and the camera field of view is set to 74° with corresponding focal length $4.969mm$. "auto" represents that datasets are automatically captured on the simulator-planned camera flight paths; "manual" represents that datasets are manually captured by three people with different clinical skills; "fully" represents that datasets are captured on the designed camera flight paths that aim to fully recover the internal surface of the colon.

V. Results

In the experiments, we begin with showing the limitations of the state-of-art SLAM algorithms Kintinuous [22], ElasticFusion [21], KinectFusion [23], ORB-SLAM2 [15] and StereoDSO [20] to colonoscopic datasets captured in scenarios simulating the real normal colonoscopy screening as well as in scenarios where the camera is operated with very slow camera motion. Then, we validate the robustness and accuracy of the proposed framework using 15 different datasets collected in different scenarios using the developed colonoscopy simulator. Finally, an in-vivo video sequence is used to demonstrate the practicality of the proposed framework. Note that the experiments with state-of-the-art RGB-D SLAM algorithms are not trying to make comparisons, but to show the limitations of these methods when applied to colonoscopic images. The summary of experimental datasets is shown in Table I.

A. Evaluation of RGB-D and stereo SLAM systems on colonoscopic images

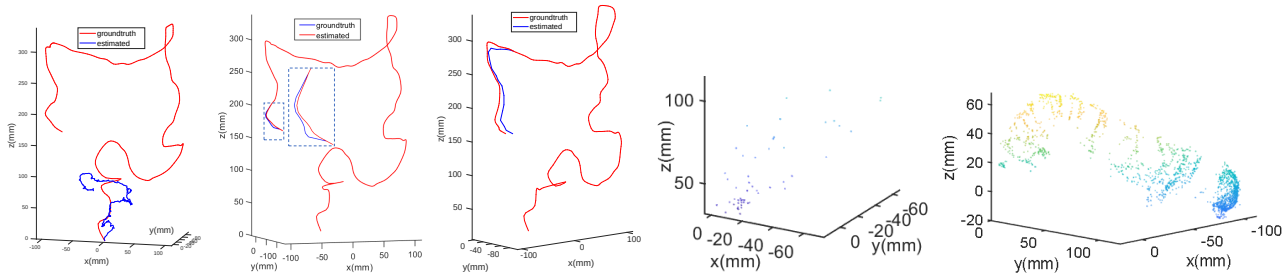
We run all the SLAM algorithms in offline mode. For RGB-D SLAM algorithms Kintinuous, ElasticFusion and KinectFusion, the images from the left camera together with the corresponding ground truth depth are used. The paired stereo color image sequences are input into ORB-SLAM2 and StereoDSO to reconstruct maps. The datasets captured from a normal colonoscopy scenario (Cases 7 to 9) are first used and all the SLAM algorithms fail. Fig. 7 illustrates the failure using Case 8. Since the major working principle of KinectFusion relies heavily on feature matching step using ICP, it fails when the camera moves fast during the normal colonoscopy procedures. For the voxel-based Kintinuous and surfel-based ElasticFusion, fast camera motion violates the assumption behind projective data association and hinders tracking performance, so their estimated trajectories suffer from very large errors which create many outlier points and no map is generated. StereoDSO extracts candidate points from the first frame in initialization and fails to track them in the following key frames. It keeps resetting until the last segment of the colon, and thus only generates a very short trajectory with sparse point clouds. This also happens to ORB-SLAM2, it only obtains a small segment of sparse map. Therefore, the experiment results show that these SLAM systems are not suitable for map reconstruction using images from normal colonoscopy procedures.

Then, we collect a large complete set of colonoscopic image sequences with very unrealistically slow camera motions (Case 0). It contains 6k pairs of stereo color and depth images. Fig. 8 shows comparison of the ground truth trajectory and estimated trajectories from the different SLAM algorithms, and the reconstructed maps are shown in Fig. 9. It shows that Kintinuous performs poorly because the trajectory is long and has a lot of turns as well as the camera is forward facing. ElasticFusion recovers the main topological structures but the estimated trajectory is very wrong. KinectFusion is very easy to lose tracking and only able to reconstruct a small segment of the colon map. The initialization of StereDSO is slow and unstable if there are only little rotations without relatively large translations. The estimated trajectory of StereoDSO has large drift and the obtained map is unacceptable. ORB-SLAM2 can obtain a reasonable trajectory with drift but it only built a sparse map. The evaluation results show that these stereo or RGB-D SLAM algorithms are not directly suitable for 3D reconstruction in colonoscopy even with the unrealistic very slow camera motion.

B. Colon 3D reconstruction on simulator-planned camera flight paths

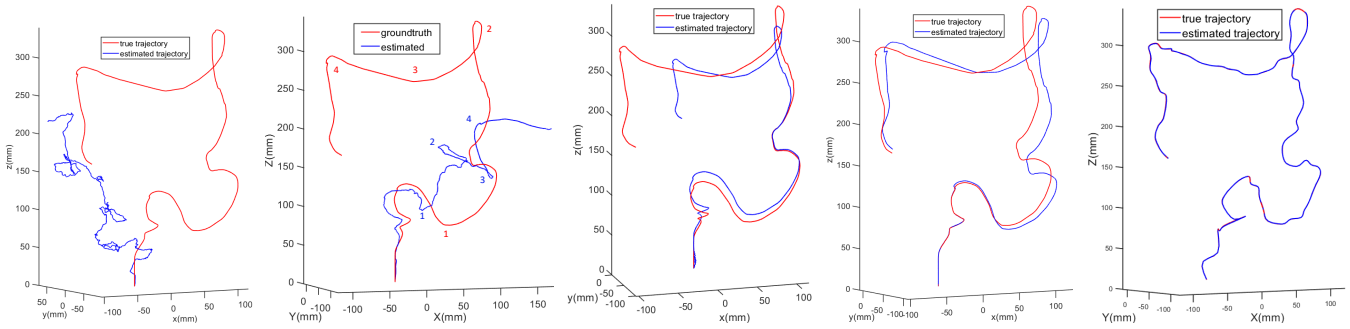
In this and the following two subsections, we evaluate our algorithm using datasets collected in different scenarios. Six planned camera flight paths are generated by the simulator to automatically guide the camera through the entire colon lumen and we record Case 1 to 6 of the experimental datasets. Fig. 11 (a) shows the trajectory of camera flight path in Case 1 and Fig. 11 (b) shows the reconstructed complete colon map with detailed textures. Fig. 11 (c) and Fig. 11 (d) illustrate the registration errors between scans from stereo images and colon model using the proposed joint optimization algorithm. The Euler angle errors along X, Y and Z axis are within $[-0.04, 0.04]$ rad and the translation errors along X, Y, Z are within $[-0.5, 0.5]$ mm, respectively. Fig. 11 (e) and Fig. 11 (f) show the Euler and translation error distributions on datasets Case 1 to Case 6, respectively, which validates the robustness and accuracy of the proposed method. For each scan to colon model registration, the algorithm takes 50 iterations on average to converge. Fig. 10 shows the comparison between several textured regions which are reconstructed by the proposed method and the actually seen regions, their textures are slightly different as the field of view of a scan is smaller than the corresponding pair of stereo images.

It is noted that at least 40% of colon internal surface are missed in the colonoscopy procedures, especially the opposite sides of the colon wall, since the camera always keep forward moving during the simulator-planned flight.



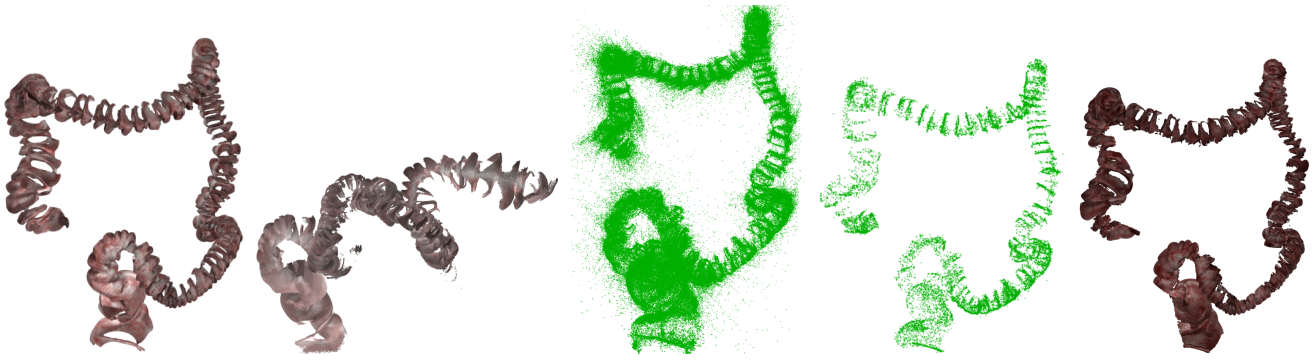
(a) ElasticFusion (b) StereoDSO (c) ORB-SLAM2 (d) StereoDSO (e) ORB-SLAM2

Fig. 7: Trajectories and maps estimated from SLAM systems on Case 8 with normal camera motion: (a) The trajectory estimated from ElasticFusion suffers from large errors; (b) StereoDSO only obtains the trajectory of the last part of the colon; (c) ORB-SLAM2 only obtains the trajectory of the last part of the colon; (d) StereoDSO obtains sparse point clouds; (e) ORB-SLAM2 obtains a small segment map corresponds to its trajectory.



(a) Kintinuous (b) ElasticFusion (c) StereoDSO (d) ORB-SLAM2 (e) Our method.

Fig. 8: Comparison of the ground truth trajectory and estimated trajectories on Case 0 with very slow camera motion: (a) Kintinuous suffers from large errors; (b) ElasticFusion recovers the main topological structures, the turns numbered 1, 2, 3 and 4 in estimated trajectory correspond to the turns numbered 1, 2, 3 and 4 in the ground truth trajectory, respectively; (c) The initialization of StereoDSO is unstable and it recovers a complete trajectory with large drift; (d) The initialization of ORB-SLAM2 is more stable than StereoDSO and it obtains a relatively good trajectory with drift; (e) Our method can achieve very accurate result.



(a) The ground truth (b) ElasticFusion (c) StereoDSO (d) ORB-SLAM2 (e) Ours

Fig. 9: Reconstructed maps on Case 0 with very slow camera motion: (a) The ground truth map; (b) ElasticFusion recovers the main topological structure of the colon; (c) StereoDSO recovers a complete semi-dense map with large drift; (d) ORB-SLAM2 obtains sparse map; (e) Our result is close to the ground truth.

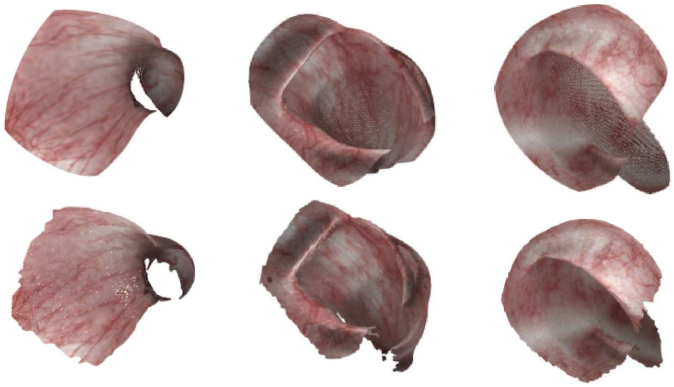


Fig. 10: Examples of texture region comparison. The first row shows the ground truth texture regions and the second row shows the corresponding reconstructed texture regions.

C. Colon 3D reconstruction on manually flown paths

To simulate the real colonoscopy procedures by clinicians with different skills, datasets of Case 7, 8 and 9 are manually collected by three people with different level of clinical skills after training. Fig. 12 (a), (b) and (c) show the estimated camera trajectories of Case 7 to 9, respectively. The camera in Case 7 is flown through the entire colon lumen and the images are taken from the forward, side and opposite view of the colon. For the camera in Case 8, it took images from the forward views and some side views of the colon. Very similar to the real colonoscopy procedures, the camera in Case 7 and Case 8 are operated with sudden changes of rotation and translation. By contrast, the trajectory of the camera in Case 9 is smooth and the least number of images are taken. Fig. 12 (d), (e) and (f) show the reconstructed and texturized colon maps, respectively. We can find that the reconstructed map from Case 7 is more complete than Case 8 and Case 9 because a large amount of colon internal surface is covered. Although the map from Case 8 is slightly more complete than Case 9, there are still many areas that are uninspected, especially the opposite sides of colon folds. The registration error distributions on Case 7, 8 and 9 are shown in Fig. 15. The errors in Case 9 are relative small compared to Case 7 and 8 because its camera motion is smooth and there are certain overlapped areas between each pair of consecutive frames. Overall, all the Euler angle errors and translation errors are relatively small.

D. Colon 3D reconstruction on fully inspected colon

The last evaluation is conducted on datasets of Case 10 to 15 which are manually collected and aimed to validate the ability of the proposed 3D reconstruction framework to fully recover the internal colon surface.

As shown in Fig. 13, six segments of camera flight paths (from Case 10 to Case 15) are designed to fully inspect the internal surface of anatomical segments (Rectum, Sigmoid, Descending, Transverse, Ascending and Cecum) of the human colon, respectively. To inspect as much area as possible of the internal surface of the colon and simulate the real colonoscopy procedures, the camera is manually flown to inspect from

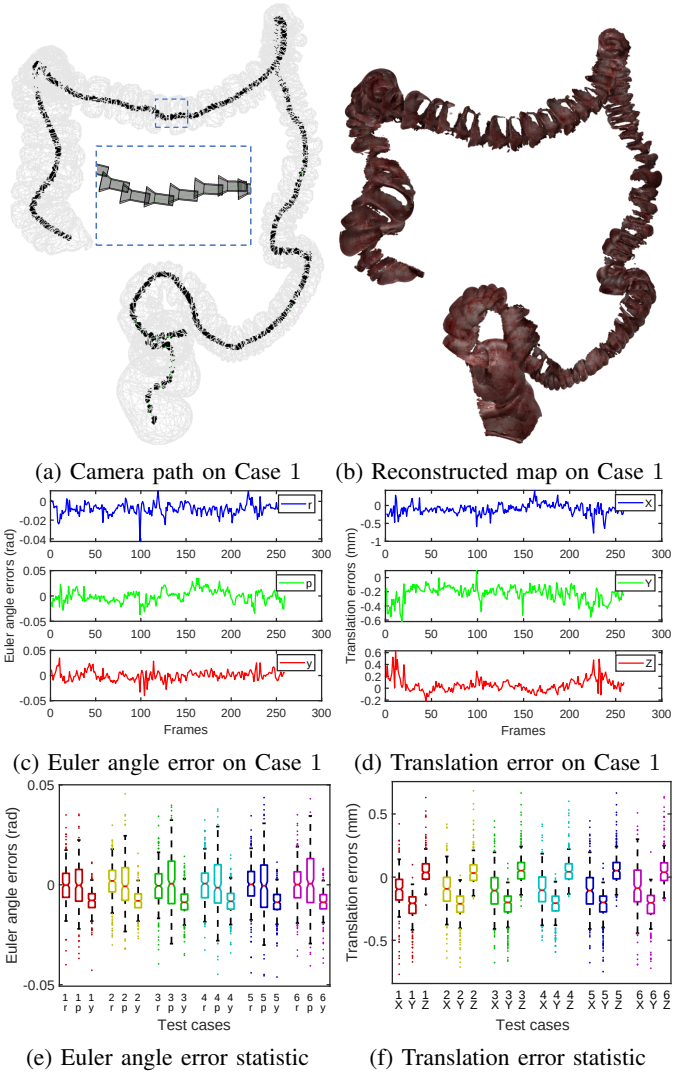
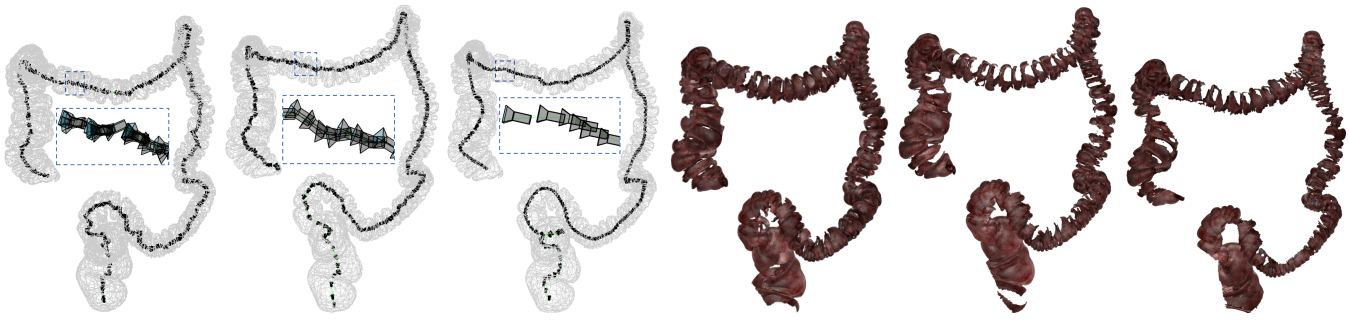


Fig. 11: 3D reconstruction results on the simulator-planned cases, r, p and y represent roll, pitch and yaw angles along axis X, Y and Z axis respectively.

the forward, side and opposite views of the colon segments with challenging conditions including large changes of viewing angles and close distance to the colon surface. Fig. 14 shows very complete colon maps with detailed textures and Fig. 15 shows the mean registration errors of X, Y, Z axis, which demonstrates the capability and high accuracy of 3D reconstruction with fully recovery of internal colon surface.

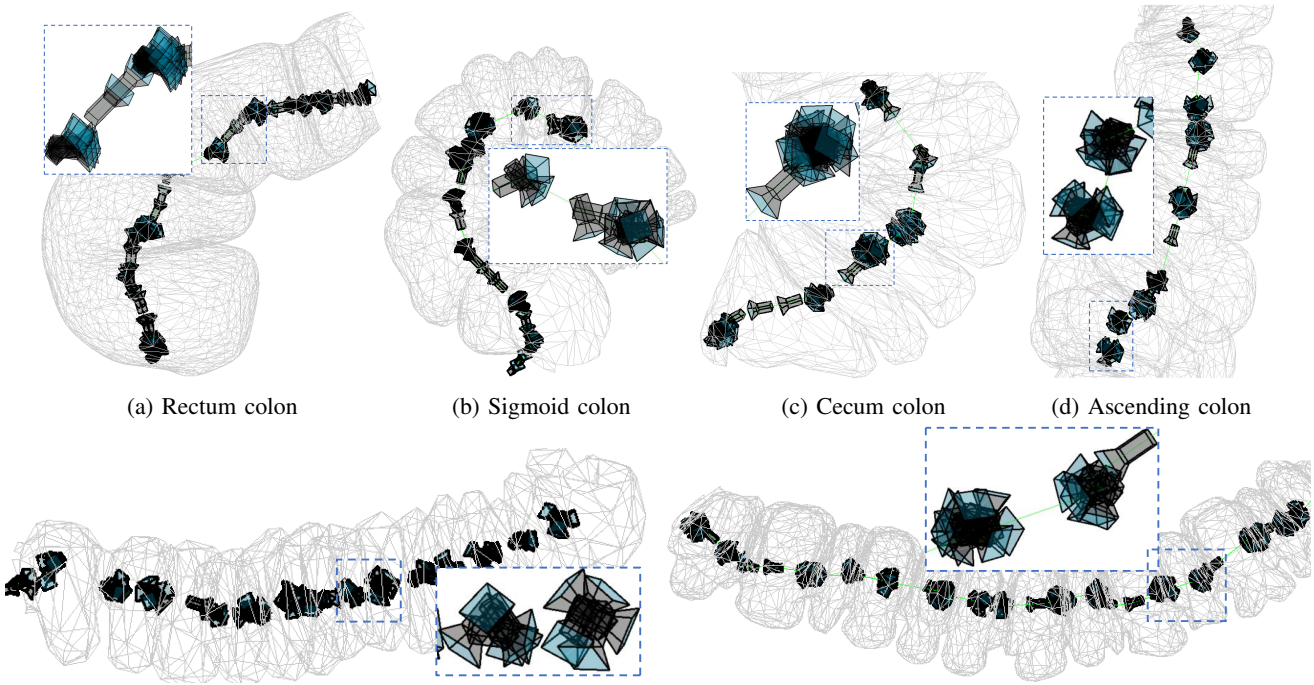
E. In-Vivo Experiments

We also show some preliminary results using in-vivo dataset to demonstrate the practicality of the proposed framework. The synthetic colonoscopy images with ground truth of depths are used to train a supervised convolutional neural network for monocular depth estimation, then the trained network is used to predict depth for the real colonoscopy images [31]. The predicted depth images are dense and we can reconstruct 3D scan for each real monocular colonoscopy image. Fig. 16 shows the reconstructed map of the colon chunk with structures and



(a) Case 7 (b) Case 8 (c) Case 9 (d) Case 7 (e) Case 8 (f) Case 9

Fig. 12: 3D reconstruction results on manually flown cases: (a), (b) and (c) show the camera flight path on Case 7, 8 and 9, respectively; (d), (e), (f) show the corresponding reconstructed colon map on Case 7, 8 and 9, respectively.



(a) Rectum colon (b) Sigmoid colon (c) Cecum colon (d) Ascending colon

(e) Descending colon (f) Transverse colon

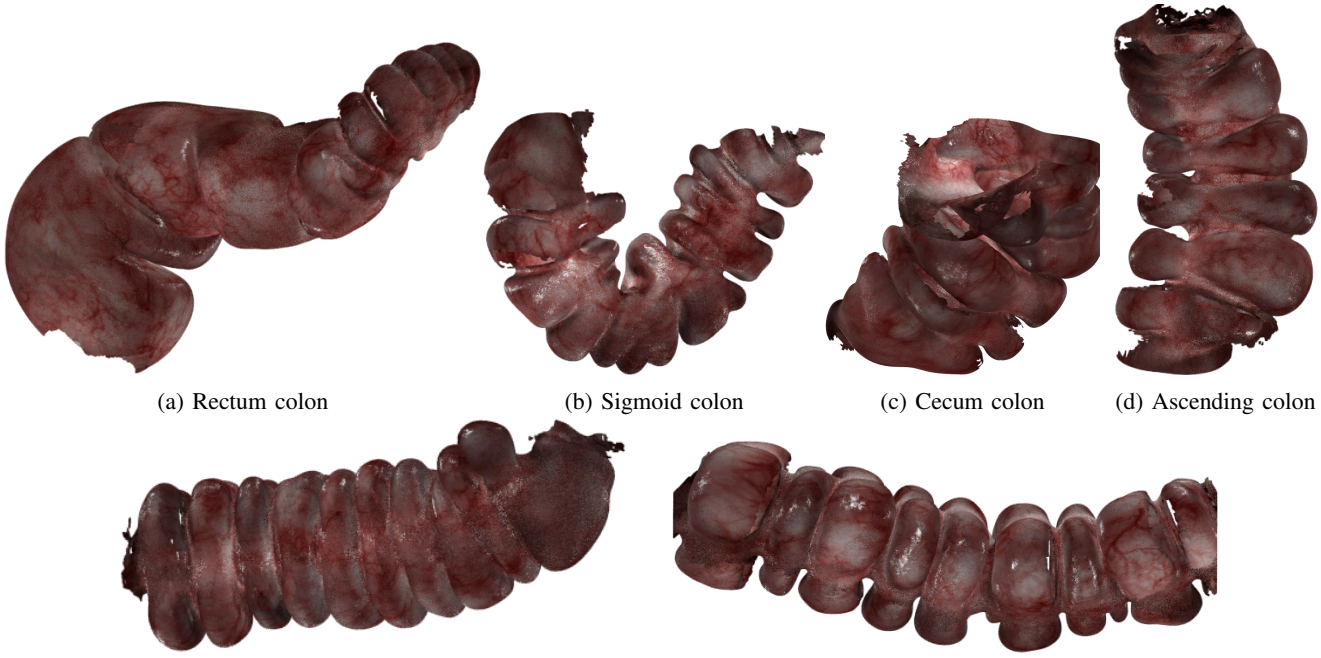
Fig. 13: Designed camera flight trajectories to fully inspect the colon.

textures. However, the quality of the reconstructed map is not as good as that in the simulation experiments. The degradation is mainly caused by errors of predicted depth images and the deformation of the real colon. In our future work, we will further improve our framework to better handle in-vivo data.

VI. Conclusion and Future Work

This paper presents a framework for 3D reconstruction of colon structures and detailed textures from stereo colonoscopic images. A colon model segmented from CT is used together with the colonoscopic images to achieve high quality reconstruction results. A realistic colonoscopy simulator has been developed and the proposed framework is validated using 15 different datasets generated from the simulator. Experimental results have demonstrated the high accuracy and robustness of the proposed framework. Also, an in-vivo dataset is used to show the potential clinical applications in colonoscopy procedures.

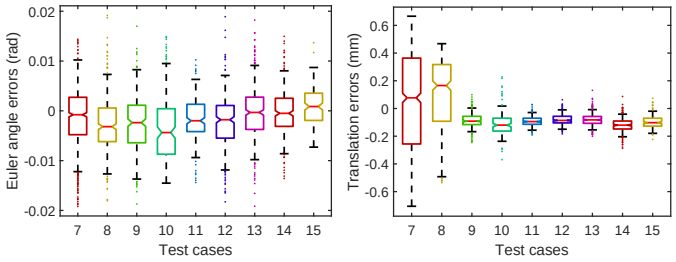
Although very promising results have been achieved, there are a few limitations in the current work. One is that the proposed framework uses stereo images since the depth information computed from stereo matching method is needed. To apply our framework to 3D reconstruction using monocular colonoscopic images, one way is to predict the depth in monocular images using deep learning based method. However, the achievable reconstruction accuracy is expected to be reduced. Another limitation is that the non-rigid characteristic of the real colon will cause some degradations such as inaccuracy in estimating image depth and recovering camera motion. In the future, we will also improve the proposed framework with the capability of overcoming colon deformation using a general template and non-rigid structure-from-motion based approaches. We are aiming to develop robust reconstruction



(a) Rectum colon (b) Sigmoid colon (c) Cecum colon (d) Ascending colon

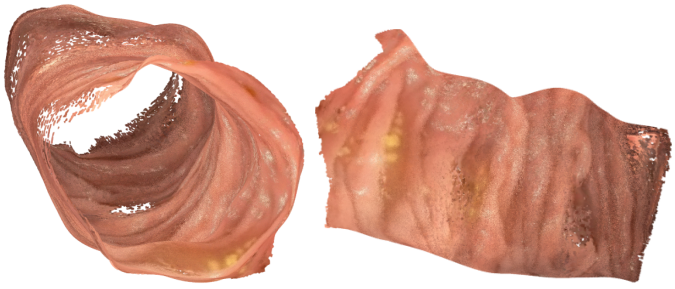
(e) Descending colon (f) Transverse colon

Fig. 14: 3D reconstruction results on the fully inspected colon.



(a) Euler angle error statistic (b) Translation error statistic

Fig. 15: Mean reconstruction errors of Case 7 to Case 15.



(a) (b)

Fig. 16: 3D reconstruction of a real colon chunk: (a) and (b) show the reconstructed colon chunk from the front view and the side view, respectively.

algorithms using clinical colonoscopic images once the colon deformation can be effectively dealt with.

REFERENCES

[1] A. Leufkens, M. Van Oijen, F. Vleggaar, and P. Siersema, "Factors influencing the miss rate of polyps in a back-to-back colonoscopy study," *Endoscopy*, vol. 44, no. 05, pp. 470–475, 2012.

[2] C. M. le Clercq, M. W. Bouwens, E. J. Rondagh, C. M. Bakker, E. T. Keulen, R. J. de Ridder, B. Winkens, A. A. Masclee, and S. Sanduleanu, "Postcolonoscopy colorectal cancers are preventable: a population-based study," *Gut*, vol. 63, no. 6, pp. 957–963, 2014.

[3] J. D. Waye, "Difficult colonoscopy," *Gastroenterology & Hepatology*, vol. 9, no. 10, p. 676, 2013.

[4] H. Zhu, M. Barish, P. Pickhardt, and Z. Liang, "Haustral fold segmentation with curvature-guided level set evolution," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 2, pp. 321–331, 2012.

[5] A. Guinigundo, "Is the virtual colonoscopy a replacement for optical colonoscopy?" in *Seminars in oncology nursing*, vol. 34, no. 2. Elsevier, 2018, pp. 132–136.

[6] A. Karargyris and N. Bourbakis, "Three-dimensional reconstruction of the digestive wall in capsule endoscopy videos using elastic video interpolation," *IEEE transactions on Medical Imaging*, vol. 30, no. 4, pp. 957–971, 2010.

[7] B. K. Horn, "A method for obtaining the shape of a smooth opaque object from one view," *PhD thesis, Massachusetts Institute of Technology, Cambridge, 1970*, 1970.

[8] D. Koppel, C.-I. Chen, Y.-F. Wang, H. Lee, J. Gu, A. Poirson, and R. Wolters, "Toward automated model building from video in computer-assisted diagnoses in colonoscopy," in *Medical Imaging 2007: Visualization and Image-Guided Procedures*, vol. 6509. International Society for Optics and Photonics, 2007, p. 65091L.

[9] C.-I. Chen, D. Sargent, and Y.-F. Wang, "Modeling tumor/polyp/lesion structure in 3d for computer-aided diagnosis in colonoscopy," in *Medical Imaging 2010: Visualization, Image-Guided Procedures, and Modeling*, vol. 7625. International Society for Optics and Photonics, 2010, p. 76252F.

[10] H. C. Longuet-Higgins, "A computer algorithm for reconstructing a scene from two projections," *Nature*, vol. 293, no. 5828, pp. 133–135, 1981.

[11] A. Kaufman and J. Wang, "3d surface reconstruction from endoscopic videos," in *Visualization in Medicine and Life Sciences*. Springer, 2008, pp. 61–74.

[12] J. Zhou, A. Das, F. Li, and B. Li, "Circular generalized cylinder fitting for 3d reconstruction in endoscopic imaging based on mrf," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, 2008, pp. 1–8.

[13] D. Hong, W. Tavanapong, J. Wong, J. Oh, and P. C. De Groen, "3d reconstruction of virtual colon structures from colonoscopy images,"

Computerized Medical Imaging and Graphics, vol. 38, no. 1, pp. 22–33, 2014.

[14] M. A. Armin, G. Chetty, H. De Visser, C. Dumas, F. Grimpen, and O. Salvado, "Automated visibility map of the internal colon surface from colonoscopy video," *International Journal of Computer Assisted Radiology and Surgery*, vol. 11, no. 9, pp. 1599–1610, 2016.

[15] 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.

[16] G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *2007 6th IEEE and ACM international symposium on mixed and augmented reality*. IEEE, 2007, pp. 225–234.

[17] 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.

[18] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *European conference on computer vision*. Springer, 2014, pp. 834–849.

[19] C. Forster, Z. Zhang, M. Gassner, M. Werlberger, and D. Scaramuzza, "Svo: Semidirect visual odometry for monocular and multicamera systems," *IEEE Transactions on Robotics*, vol. 33, no. 2, pp. 249–265, 2016.

[20] R. Wang, M. Schworer, and D. Cremers, "Stereo dso: Large-scale direct sparse visual odometry with stereo cameras," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3903–3911.

[21] T. Whelan, R. F. Salas-Moreno, B. Glocker, A. J. Davison, and S. Leutenegger, "Elasticfusion: Real-time dense slam and light source estimation," *The International Journal of Robotics Research*, vol. 35, no. 14, pp. 1697–1716, 2016.

[22] T. Whelan, M. Kaess, M. Fallon, H. Johannsson, and J. Leonard, "Kintinuous: Spatially extended kinectfusion," *In Workshop on RGB-D: Advanced Reasoning with Depth Cameras, in conjunction with Robotics: Science and Systems*, 2012.

[23] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *2011 10th IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 2011, pp. 127–136.

[24] R. J. Chen, T. L. Bobrow, T. Athey, F. Mahmood, and N. J. Durr, "Slam endoscopy enhanced by adversarial depth prediction," *arXiv preprint arXiv:1907.00283*, 2019.

[25] R. Ma, R. Wang, S. Pizer, J. Rosenman, S. K. McGill, and J.-M. Frahm, "Real-time 3d reconstruction of colonoscopic surfaces for determining missing regions," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2019, pp. 573–582.

[26] H. Hirschmuller, "Accurate and efficient stereo processing by semi-global matching and mutual information," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2. IEEE, 2005, pp. 807–814.

[27] F. Fraundorfer and D. Scaramuzza, "Visual odometry: Part i: The first 30 years and fundamentals," *IEEE Robotics and Automation Magazine*, vol. 18, no. 4, pp. 80–92, 2011.

[28] X.-S. Gao, X.-R. Hou, J. Tang, and H.-F. Cheng, "Complete solution classification for the perspective-three-point problem," *IEEE transactions on pattern analysis and machine intelligence*, vol. 25, no. 8, pp. 930–943, 2003.

[29] S. Rusinkiewicz and M. Levoy, "Efficient variants of the icp algorithm," in *Proceedings Third International Conference on 3D Digital Imaging and Modeling*. IEEE, 2001, pp. 145–152.

[30] O. Weber, M. Ben-Chen, C. Gotsman, and K. Hormann, "A complex view of barycentric mappings," in *Computer Graphics Forum*, vol. 30, no. 5. Wiley Online Library, 2011, pp. 1533–1542.

[31] I. Alhashim and P. Wonka, "High quality monocular depth estimation via transfer learning," *arXiv preprint arXiv:1812.11941*, 2018.