1-1-2015

Visual SLAM and Surface Reconstruction for Abdominal Minimally Invasive Surgery

Bingxiong Lin
University of South Florida, bingxiong@mail.usf.edu

Follow this and additional works at: http://scholarcommons.usf.edu/etd

Part of the Robotics Commons

Scholar Commons Citation
Lin, Bingxiong, "Visual SLAM and Surface Reconstruction for Abdominal Minimally Invasive Surgery" (2015). Graduate Theses and Dissertations.
http://scholarcommons.usf.edu/etd/5849

This Dissertation is brought to you for free and open access by the Graduate School at Scholar Commons. It has been accepted for inclusion in Graduate Theses and Dissertations by an authorized administrator of Scholar Commons. For more information, please contact scholarcommons@usf.edu.
Visual SLAM and Surface Reconstruction for Abdominal Minimally Invasive Surgery

by

Bingxiong Lin

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Department of Computer Science and Engineering
College of Engineering
University of South Florida

Major Professor: Yu Sun, Ph.D.
Xiaoning Qian, Ph.D.
Dmitry Goldgof, Ph.D.
Richard Gitlin, Sc.D.
Yuncheng You, Ph.D.

Date of Approval:
March 20, 2015

Keywords: 3D reconstruction, Laparoscope localization, Feature detection, Vessel feature, Tissue deformation

Copyright © 2015, Bingxiong Lin
DEDICATION

To my parents.
ACKNOWLEDGMENTS

First and foremost, I would like to thank my advisor, Yu Sun. Over the past five years, he has taught me how to choose research topics, how to measure the research progress, how to address research problems. He led me to the fascinating 3D Computer Vision when I started my study here, and has helped me grow and explore in this area since then. His trust in my ability has built up my confidence in my research. He has shown a great support on my job hunting and his suggestions are invaluable to me. He has also been a good friend in life and the “coffee time” on Friday afternoon has brought me a lot of fun.

I would also like to thank Xiaoning Qian, who has always been a great teacher and friend to me. His patience and kindness sometimes make me feel that he is my peer friend. His door is always open whenever I have a question. He has given me massive suggestions and comments on how to approach a research problem and how to write a technical paper, and has always been patient with my stupid grammar errors. It is my great honor that he took a flight to Tampa and attended my dissertation defense in person.

Of course, I am also very grateful to other committee members and professors who have helped me in my research. Dr. Dmitry Goldgof has provided many insightful suggestions on my research, especially during the Computer Vision seminars. Dr. Richard Gitlin and Dr. Yuncheng You have also provided many useful comments. Dr. Jaime Sanchez has brought me to the surgery room, explained to me the surgical procedures, and helped me collect data that is critical in this dissertation.

I own my thanks to my labmates and friends, who have helped me during my research and made my life here more colorful. They are Adrian Johnson, Yun Lin, Fillipe Souza, Rajmadhan Ekambaram, Yongqiang Huang, Jie Zhang, Junyi Tu, Haifang Wu, Shaogang Ren, Meng Lv, Yingying Zhang, Mu Zhou, Xiang Zuo, Huixing Zhai, David Paulius, Roger Milton, Baishali Chaudhury, Hao Li, Ran Rui, Tao Wang, Wen Zhao, Lukai Guo, Rui Guo, Wei Yuan, Gang Liu, Dongping Du, and Zunjun Chen.

I would not have pursued a PhD without the supports from my family. My mom, Daibi Yang, has taught me determination and hardwork, which led me through my PhD. My dad, Minlian Lin, has encouraged me to follow my heart. My brothers, Bingqiang Lin and Bingcai Lin, have supported my ideas and decisions.
TABLE OF CONTENTS

LIST OF TABLES iv

LIST OF FIGURES v

ABSTRACT viii

CHAPTER 1 MOTIVATION AND CHALLENGES 1
  1.1 Background and Motivation 1
  1.2 Challenges 2
  1.3 Summary 3

CHAPTER 2 LITERATURE REVIEW 4
  2.1 Note to Reader 4
  2.2 Motivation and MIS Datasets 4
  2.3 Feature Detection and Feature Tracking 6
    2.3.1 Feature Detection 7
    2.3.2 Feature Tracking 8
    2.3.3 Discussion 10
  2.4 Reconstruction without Camera Motion 11
    2.4.1 Stereo Cue 11
    2.4.2 Active Methods 12
    2.4.3 Shading and Shadow Cue 13
    2.4.4 Discussion 14
  2.5 Rigid MIS-VSLAM 14
    2.5.1 Monocular Camera 16
    2.5.2 Stereo Cameras 19
    2.5.3 Discussion 20
  2.6 Dynamic MIS-VSLAM 20
    2.6.1 State-of-the-art DSSFM 21
      2.6.1.1 DSSFM with Monocular Cameras 21
      2.6.1.2 DSSFM with Stereo Cameras 23
    2.6.2 DSSFM in MIS Environment 24
    2.6.3 Moving Instrument Tracking 26
    2.6.4 Discussion 27
CHAPTER 3   VESSEL-BASED IMAGE FEATURE DETECTION  
3.1 Note to Reader   
3.2 Vessel Feature Introduction   
3.3 Vessel Feature Detection   
3.3.1 Method Overview   
3.3.2 Detecting Candidate Branching Points   
3.3.3 Blood Vessel Enhancement: Vesselnness   
3.3.4 Blood Vessel Enhancement: Ridgeness   
3.3.5 Branching Point Detection (RBCT)   
3.3.6 Connected Component Labeling and Non-maximal Suppression   
3.3.7 Branching Segment Detection (RBSD)   
3.3.8 Computational Analysis and Run Time Results   
3.4 Experiments and Results   
3.4.1 In Vivo Datasets and Ground Truths   
3.4.2 Repeatability and Number of Points   
3.4.3 Patch Matching Correctness   

CHAPTER 4   VESSEL-FEATURE-BASED 3D RECONSTRUCTION  
4.1 Traditional Stereo Matching   
4.2 Vessel Feature Stereo Matching   
4.2.1 Branching Point Matching   
4.2.2 Branching Segment Matching   
4.3 Large-area Dense 3D Reconstruction   
4.4 Experiments and Results   
4.4.1 In Vivo Datasets   
4.4.2 Vessel Feature Stereo Matching Accuracy   
4.4.3 Large-area 3D Reconstruction Results   

CHAPTER 5   SHADOW-CASTING-BASED 3D RECONSTRUCTION  
5.1 Note to Reader   
5.2 Introduction   
5.3 Shadow-Scanning-Based 3D Reconstruction   
5.3.1 System Overview   
5.3.2 Extracting Shadow Curves   
5.3.3 Intersecting Shadow Curves with Epipolar Lines   
5.3.4 Field Surface Interpolation   
5.3.5 3D Reconstruction   
5.4 Experiments and Results   
5.4.1 Phantom and ex vivo Images   
5.4.2 Disparity Maps   
5.4.3 3D Reconstruction Results   
5.4.4 Numerical Comparison   
5.4.5 Robustness Analysis
CHAPTER 6 VISUAL SLAM IN A DEFORMING ENVIRONMENT 83
  6.1 Note to Reader 83
  6.2 Introduction 83
    6.2.1 PTAM 84
  6.3 Stereoscope PTAM 84
    6.3.1 Deforming Point Detection 85
    6.3.2 Non-linear Stereo Pose Refinement 87
    6.3.3 Map Initialization 88
    6.3.4 Stereo Bundle Adjustment 88
  6.4 Experimental and Results 88
    6.4.1 Tracking Accuracy 88
    6.4.2 Evaluation with In Vivo Data 91

CHAPTER 7 PERIPHERY AUGMENTATION SYSTEM AND EVALUATION 94
  7.1 Periphery Augmentation System Design 94
  7.2 In Vivo Evaluation 95
    7.2.1 Interface and Point Estimation Procedure 96
    7.2.2 Environment Awareness Measures 98
    7.2.3 Results and Analysis 98

CHAPTER 8 CONCLUSION AND FUTURE DIRECTIONS 101
  8.1 Summary of Contributions 101
  8.2 Future Research Directions 102

REFERENCES 104

APPENDICES 118
  Appendix A Permission to Reproduce Material 119
LIST OF TABLES

Table 2.1 Summary of publicly available MIS datasets. 6
Table 2.2 Summary of different approaches in DSSFM. 22
Table 2.3 Dynamic visual SLAM in MIS. 24
Table 2.4 Summary of the state-of-the-art methods in MIS-VSLAM. 25
Table 3.1 Eigenvalues analysis towards vessel and branching points($0 < \lambda_1 < \lambda_2$). 31
Table 3.2 Run time of different steps in RBSD. 41
Table 3.3 Parameters of state-of-the-art feature detectors used in this chapter. 42
Table 3.4 Summary of the adopted datasets. 43
Table 3.5 The total number of points in all branching segments detected by RBSD. 46
Table 4.1 Summary of datasets used here. 59
Table 4.2 The disparity errors and 3D position errors of the propose vessel feature matching method across the six scenes. 60
Table 5.1 Disparity error of the four methods over the five experiments. 80
Table 5.2 3D position error of the four methods over the five experiments. 80
Table 5.3 The percentage of pixels in image with valid disparity value. 81
Table 6.1 Mean error and variance of the tracking results. 89
Table 7.1 The overall mean and standard deviation of the angle errors and its Student’s t-test results. 99
Table 7.2 The overall mean and standard deviation of the pixel errors and its Student’s t-test results. 99
**LIST OF FIGURES**

| Figure 1.1  | The diagram of a typical abdominal MIS setup. | 2 |
| Figure 2.1  | The dichotomy of MIS-VSLAM methods based on camera motions and scene types. | 5 |
| Figure 2.2  | Illustration of MIS-VSLAM methods in rigid and static scenes. | 15 |
| Figure 2.3  | Illustration of MIS-VSLAM methods in dynamic or deforming scenes. | 21 |
| Figure 3.1  | Illustration of vessel features: branching point (detected by RBCT) and branching segment (detected by RBSD). | 30 |
| Figure 3.2  | Overview of the branching point and branching segment detection. | 30 |
| Figure 3.3  | Illustration of eigenvalues of Hessian matrix \((0 < \lambda_1 < \lambda_2)\). | 32 |
| Figure 3.4  | Example of vessel enhancement using Frangi vesselness. | 33 |
| Figure 3.5  | Comparison of a) binary ridge image and b) our ridgeness image. | 35 |
| Figure 3.6  | Typical example of a circle test at a branching point on a) raw image, b) vesselness image, c) binary ridge image, and d) ridgeness image. | 36 |
| Figure 3.7  | Ridgeness value of pixels along the circle around a branching point. | 37 |
| Figure 3.8  | Candidate branching points a) before and b) after RBCT. | 37 |
| Figure 3.9  | Illustration of the two-pass test vessel tracing process. | 39 |
| Figure 3.10 | Illustration of branching segment detection. | 40 |
| Figure 3.11 | Sample images of the seven *in vivo* video clips. | 43 |
| Figure 3.12 | Illustration of selected ground truth point correspondences (green) across different views in scene3. | 44 |
| Figure 3.13 | Repeatability scores of different methods in the seven scenes. | 46 |
| Figure 3.14 | The number of feature points detected by different feature point detectors in the seven scenes. | 47 |
Figure 3.15  Patch matching correctness of different methods in the seven scenes.

Figure 3.16  One example of correlation-based patch matching using “fixed range image search” with feature detector RBCT.

Figure 4.1  One example of stereo images before (top) and after rectification (down).

Figure 4.2  The detected branching points are shown as cyan dots.

Figure 4.3  Diagram of vessel vectors

Figure 4.4  Illustration of local adjusting for candidate branching points.

Figure 4.5  Illustration of branching point matching results.

Figure 4.6  Illustration of vessel pixel search range (epipolar band) in the right image.

Figure 4.7  One example of the vessel-pixel matching results.

Figure 4.8  Illustration of the vessel pixel matching with one mis-matched point (red solid rectangle).

Figure 4.9  Illustration of vessel pixel matching results on the ridge mask.

Figure 4.10  One example of recovered 3D vessels from a pair of stereo images.

Figure 4.11  Exemplar images (left channel) of the five datasets.

Figure 4.12  The recovered 3D vessel network by integrating 3D vessels from different views.

Figure 4.13  The untextured and textured 3D models that are recovered based on the obtained 3D vessel network.

Figure 5.1  Outline of the proposed method with four major steps.

Figure 5.2  One example of a) the difference image and b) the shadow-mask image with shadow area shown as white.

Figure 5.3  Illustration of accumulated-shadow-mask image.

Figure 5.4  Shadow curves before and after LWR.

Figure 5.5  Illustration of intersection of shadow curve and epipolar lines.

Figure 5.6  Mapping defined by a pair of curves and epipolar lines.

Figure 5.7  Mapping defined by two pairs of curves and epipolar lines.

Figure 5.8  Images of the expriment setup.
Figure 5.9   Diagram of the experiment setup. 73
Figure 5.10   Example images of different targets. 74
Figure 5.11   Comparison of disparity maps of different methods. 76
Figure 5.12   3D reconstruction results of flat plane, intestine phantom, lung phantom, heart phantom and porcine liver. 78
Figure 5.13   3D reconstruction results of different methods over five experiments. 79
Figure 5.14   Illustration of predicted points in right image with ground truth. 80
Figure 5.15   Robustness analysis. 82
Figure 6.1    Outline of our stereoscope PTAM. 85
Figure 6.2    Phantom experiment setup. 89
Figure 6.3    Comparison of stereo tracking accuracy with ground truth. 90
Figure 6.4    3D trajectories of the stereoscope tracked by our method over three videos. 92
Figure 6.5    Typical 3D feature points detected in one frame (zoom in for detail). 93
Figure 6.6    A bladder model reconstructed from Computerized Tomography (CT) was augmented in the colon surgery video. 93
Figure 7.1    The outline of the process of building the periphery augmentation system. 95
Figure 7.2    Illustration of the point estimation for the periphery augmentation system (top) and the original MIS video (bottom). 97
Figure 7.3    One example of ground-truth points (green) and user-estimated points (blue) on the periphery augmentation image. 97
Figure 7.4    The obtained errors per subject per round: a) angle errors, b) pixel errors. 100
Figure 7.5    Histograms of a) angle errors and b) pixel errors for each subject in each round of point tests. 100
ABSTRACT

Depth information of tissue surfaces and laparoscope poses are crucial for accurate surgical guidance and navigation in Computer Assisted Surgeries (CAS). Intra-operative Three Dimensional (3D) reconstruction and laparoscope localization are therefore two fundamental tasks in CAS. This dissertation focuses on the abdominal Minimally Invasive Surgeries (MIS) and presents laparoscopic-video-based methods for these two tasks.

Different kinds of methods have been presented to recover 3D surface structures of surgical scenes in MIS. Those methods are mainly based on laser, structured light, time-of-flight cameras, and video cameras. Among them, laparoscopic-video-based surface reconstruction techniques have many significant advantages. Specifically, they are non-invasive, provide intra-operative information, and do not introduce extra-hardware to the current surgical platform. On the other side, laparoscopic-video-based 3D reconstruction and laparoscope localization are challenging tasks due to the specialties of the abdominal imaging environment. The well-known difficulties include: low texture, homogeneous areas, tissue deformations, and so on. The goal of this dissertation is to design novel 3D reconstruction and laparoscope localization methods and overcome those challenges from the abdominal imaging environment.

Two novel methods are proposed to achieve accurate 3D reconstruction for MIS. The first method is based on the detection of distinctive image features, which is difficult in MIS images due to the low-texture and homogeneous tissue surfaces. To overcome this problem, this dissertation first introduces new types of image features for MIS images based on blood vessels on tissue surfaces and designs novel methods to efficiently detect them. After vessel features have been detected, novel methods are presented to match them in stereo images and 3D vessels can be recovered for each frame. Those 3D vessels from different views are integrated together to obtain a global 3D vessel network and Poisson reconstruction is applied to achieve large-area dense surface reconstruction.
The second method is texture-independent and does not rely on the detection of image features. Instead, it proposes to mount a single-point light source on the abdominal wall. Shadows are cast on tissue surfaces when surgical instruments are waving in front of the light. Shadow boundaries are detected and matched in stereo images to recover the depth information. The recovered 3D shadow curves are interpolated to achieve dense reconstruction of tissue surfaces.

One novel stereoscope localization method is designed specifically for the abdominal environment. The method relies on RANdom SAmple Consensus (RANSAC) to differentiate rigid points and deforming points. Since no assumption is made on the tissue deformations, the proposed methods is able to handle general tissue deformations and achieve accurate laparoscope localization results in the abdominal MIS environment.

With the stereoscope localization results and the large-area dense surface reconstruction, a new scene visualization system, periphery augmented system, is designed to augment the peripheral areas of the original video so that surgeons can have a larger field of view. A user-evaluation system is designed to compare the periphery augmented system with the original MIS video. 30 subjects including 4 surgeons specialized in abdominal MIS participate the evaluation and a numerical measure is defined to represent their understanding of surgical scenes. T-test is performed on the numerical errors and the null hypothesis that the periphery augmented system and the original video have the same mean of errors is rejected. In other words, the results validate that the periphery augmented system improves users’ understanding and awareness of surgical scenes.
CHAPTER 1
MOTIVATION AND CHALLENGES

1.1 Background and Motivation

Different from traditional open-cavity surgeries, Minimally Invasive Surgeries (MIS) do not introduce large incisions and therefore benefit patients with smaller trauma, shorter hospitalizations, less pain, and lower infection risks. In abdominal MIS, the abdomen is insufflated with carbon dioxide to allow enough operating room for surgical tasks. A typical MIS setup with one laparoscope and one surgical instrument is shown in Fig. 1.1. Monocular cameras and stereo cameras are normally attached at the tip of the laparoscopes to provide real-time videos of surgical scenes.

Laparoscopic videos provide surgeons with live feedback and allow surgeons to precisely manipulate surgical instruments. However laparoscopic videos are two-dimensional (2D) in nature and have several undesirable restrictions. First, 2D images do not have explicit depth information and therefore surgeons have to estimate the depth based on their experience. Additionally, a laparoscope has a narrow field of view, which sometimes makes it difficult for surgeons to understand the position and orientation of the laparoscope and the surgical instruments [1]. More importantly, the most significant limitation of an endoscopic video is that it is unable to provide information about critical tissue structures underneath organ surfaces. For example, to identify the hidden ureters in colon surgeries, surgeons have to spend a large amount of extra time dissecting tissues.

To overcome the above limitations of laparoscopes, this dissertation focuses on the study of two fundamental technologies: video-based intra-operative 3D reconstruction of surgical scenes and laparoscope localization in abdominal MIS. 3D reconstruction of surgical scene provide surgeons with explicit depth information and allow more accurate instrument control. Laparoscope localization can help surgeons determine where the instruments are with respect to the human anatomy. After proper registration with pre-
operative data, such as Computed Tomography (CT), the laparoscope localization can potentially enable the augmentation of hidden structures on the video.

1.2 Challenges

Multiple methods of video-based 3D reconstruction and laparoscope localization have been proposed in the literature. However, the large-area 3D reconstruction and the laparoscope localization in the abdominal environment in real time remain open challenges. The difficulties are mainly from the special environment of the abdominal MIS. First, compared with general images taken in a man-made environment, MIS images are low-texture and usually contain homogeneous areas and specular reflections due to the smooth and wet tissue surface [2,3]. These properties significantly affect the performance of the state-of-the-art feature-point detection methods. Without reliable feature-point correspondences, many feature-based 3D reconstruction and visual Simultaneous Localization and Mapping (SLAM) [4–6] methods developed in Computer Vision do not perform well. Second, surgical scenes are non-rigid in nature and may have deformations due to respiration or interaction with surgical instruments. The moving surgical instruments may also cause occlusion problems. The real-time laparoscope localization in a deforming environment with surgical-instrument occlusions is challenging.
1.3 Summary

Chapter 2 first introduces basic image feature detection and tracking methods. It then discusses those 3D reconstruction methods for MIS without considering camera motions. Later, state-of-the-art visual SLAM methods for the abdominal MIS are summarized.

In Chapter 3, a new type of image feature that is based on blood vessels is proposed specifically for the MIS imaging environment. Novel methods are presented to detect those vessel-based image features.

Chapter 4 focuses on introducing novel matching methods of vessel-based image features for stereo images. Additionally, a method is explained to integrate 3D vessels recovered at different views together to obtain a large 3D vessel network. Based on the 3D vessel network, Poisson reconstruction [7] is applied to obtain large-area dense 3D reconstruction results.

In Chapter 5, a new texture-independent 3D reconstruction method is introduced for low-texture tissue surfaces. The method proposes to mount a single-point light source on the abdominal wall and cast shadows on tissue surfaces by waving surgical instruments in front of the light. The cast shadows are used to recover the 3D structure of the tissue surfaces.

In Chapter 6, a visual SLAM system is designed specially for stereo cameras in the abdominal MIS environment. The system relies on RANdom SAmple Consensus (RANSAC) to overcome general tissue deformations and obtain accurate endoscope localization results.

In Chapter 7, based on the large-area dense 3D reconstruction results from Chapter 4 and laparoscope localization results in Chapter 6, a periphery augmentation system is designed to provide surgeons a larger field of view that helps surgeons better understand surgical scenes.
CHAPTER 2

LITERATURE REVIEW

2.1 Note to Reader

Portions of this chapter were accepted for publication in the International Journal of Medical Robotics and Computer Assisted Surgery (IJMRCAS) by Wiley on March 23, 2015.

2.2 Motivation and MIS Datasets

Intra-operative 3D reconstruction and Visual SLAM in the low-texture and deforming abdominal environment is a difficult task. Many related research efforts have been made in the literature. In this chapter, those related research efforts are discussed and the state-of-the-art methods are summarized. First of all, the existing research work in image feature detection and image feature tracking are discussed. The discussion is focused on how these detection and tracking methods are designed to overcome the special difficulties from MIS images, such as low contrast, specular reflections, and smokes. Additionally, laparoscopic-video-based 3D surgical scene reconstruction methods without the estimation of camera motion are introduced and are summarized based on the vision cues, such as stereo, structured light, and shadow. Note that, in addition to the challenges from feature detection, 3D reconstruction methods in MIS must overcome extra difficulties from surgical-instrument occlusion, a small baseline of stereo cameras, and the constrained environment. Moreover, the camera motion is estimated during the 3D reconstruction, and the scene is assumed to be rigid or static.

With the rigid scene assumption, Visual SLAM becomes tractable, and many methods have been presented in Computer Vision and Robotics. These methods and how they are adopted in MIS to overcome the special difficulties are discussed. Finally, the most difficult problem is considered – visual SLAM in dynamic and deforming MIS scenes, which is referred to as MIS-VSLAM here. This research problem is
similar to the Non-Rigid Structure from Motion (NRSFM) in the Computer Vision. Different approaches have been presented to tackle the problem from different perspectives. Those methods are summarized, and their key ideas are explained. The dichotomy of MIS-VSLAM methods based on camera motions and scene types is shown in Fig. 2.1.

Public MIS datasets are valuable to the community, and multiple MIS datasets have been collected and made available. Hamlyn Centre Laparoscopic/Endoscopic Video Datasets [8] contained a large collection of MIS videos for different organs, including lung, heart, colon, liver, spleen, and bowel. The videos in [8] included a variety of endoscope motions and tissue motions. Bartoli provided a uterus dataset in [9], which contains tissue deformation caused by instrument interactions. The corresponding ground-truth homography mappings are publicly available. Puerto-Souza and Mariottini [10] provided the Hierarchical Multi-Affine (HMA) feature matching toolbox for MIS images, which contained 100 image pairs representing various imaging conditions, such as instrument occlusion, fast cameras, and organ motion.

Stereo videos with surgical instruments moving in front of the liver were made publicly available with ground truth information of the poses and positions of those instruments in [11]. The Johns Hopkins Uni-
Table 2.1: Summary of publicly available MIS datasets.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Sensor</th>
<th>Video/ image</th>
<th>Scene</th>
<th>Scene motion</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hamlyn [8]</td>
<td>Mono.&amp; stereo</td>
<td>Video</td>
<td>Abdomen</td>
<td>Rigid &amp; deforming</td>
<td>Varied</td>
</tr>
<tr>
<td>JIGSAWS [12]</td>
<td>Stereo</td>
<td>Video</td>
<td>Lab</td>
<td>Deforming</td>
<td>640*480</td>
</tr>
<tr>
<td>Open-CAS [13, 14]</td>
<td>Stereo</td>
<td>Image</td>
<td>Liver</td>
<td>Rigid</td>
<td>720*576</td>
</tr>
<tr>
<td>Sznitman [16]</td>
<td>Mono.</td>
<td>Image</td>
<td>Pelvic</td>
<td>Deforming</td>
<td>640*290</td>
</tr>
<tr>
<td>DRIVE [18]</td>
<td>Mono.</td>
<td>Image</td>
<td>Retina</td>
<td>Rigid</td>
<td>565*584</td>
</tr>
</tbody>
</table>

University Intuitive Surgical, Inc., Gesture and Skill Assessment Working Set (JIGSAWS) [12] contained stereo videos of three elementary surgical tasks on a bench-top model: suturing, knot-tying, and needle-passing. The goal of the JIGSAWS dataset was to study and analyze surgical gestures. The Open-CAS [13, 14] collected multiple datasets for validating and benchmarking CAS, including liver simulation, liver registration, and liver 3D reconstruction. There are also multiple retinal datasets that are publicly available, including STructured Analysis of the Retina (STARE), Digital Retinal Images for Vessel Extraction (DRIVE), and Retinal Vessel Image set for Estimation of Widths (REVIEW). A summary of these datasets is shown in Table 2.1.

2.3 Feature Detection and Feature Tracking

Image feature detection and feature tracking are fundamental steps in many applications, such as structure and pose estimation, deformation recovery, and augmented reality. Many well-known feature detectors and feature descriptors have been presented. In this section, different feature detection and feature tracking methods are introduced, and how they are adapted for MIS images is discussed.
2.3.1 Feature Detection

Depending on what information is used, feature detection methods can be broadly classified into three categories: intensity-based detectors, first-derivative-based detectors, and second-derivative-based detectors. In the first category, feature detectors are mostly based on pixel intensity comparisons. In Features from Accelerated Segment Test (FAST) [20], Rosten *et al.* replaced the disk with a circle and detected corner points by identifying the pattern of a continuously bright or dark segment along the circle. Different from FAST, Mair *et al.* introduced a new circle pattern and used a binary decision tree for the corner classification [21].

In the second category, the first derivatives along the x- and y- coordinates in the raw image, namely $I_x, I_y$, reflect the intensity change and can be used to detect object structures, such as edges and boundaries. Most methods in this category are based on the eigenvalues of the auto-correlation matrix [22]. Harris and Stephens [22] proposed a measure based on those eigenvalues to detect image patches that are likely to be corners. Shi and Tomasi [23] argued that $\lambda_1$ itself was a good indicator for corners. Mikolajczyk and Schmid [24] extended the Harris corner detector in scale space and proposed the Harris-Affine detector, which had better invariance property under affine transformation. The Anisotropic Feature Detector (AFD) exploited the anisotropism and gradient information to detect interest points [25, 26].

In the third category, feature detectors exploit the second derivatives of raw images to detect interest points defined by blobs and ridges. Most methods in this category are based on the analysis of the Hessian matrix. In the Hessian-affine detector [27], the determinants of the Hessian matrices were calculated for all pixels, and the local maxima were selected as feature points. Lowe approximated the Laplacian of Gaussian with the Difference of Gaussian (DoG) [28] and built a pyramid image space to detect interest points. The Speeded Up Robust Features (SURF) feature detector replaced the Gaussian filters with box filters to obtain a faster speed [29]. It has been reported that general feature point detectors do not perform well in MIS images [25].

It is well known that the performance of feature detectors is determined by multiple parameters, such as the standard deviation of Gaussian smoothing, the discrete quantization of orientation, and the number of bins in the histogram of orientation. Most of the above-mentioned feature detection methods require manual parameter tuning based on personal experience. Stavens *et al.* [30] proposed an unsupervised method,
which learned those parameters from video sequences. In [30], Harris corners [22, 23] were detected and tracked by Lucas-Kanade (LK) optical flow [31], and the patches were stored as training data. The idea of treating feature matching as a classification problem was first introduced by Lepetit et al. [32]. The synthesized images were used to generate local feature point patches as training data for classification. In early work [33, 34], randomized trees were used as the classifier. Later, it was shown that the good performance of feature matching was mainly from the randomized binary tests rather than the randomized tree classifier and, hence, simple Semi-Naive Bayesian classifier was adopted [35, 36].

2.3.2 Feature Tracking

To track feature points, the target feature points are usually represented by their local patches. Based on local patch representations, tracking methods can be broadly classified into two categories: intensity-based tracking and descriptor-based tracking. In the first category, each feature point is directly represented by the intensity values of the pixels in its local square patch. By assuming that each pixel has a constant intensity, the well-known LK tracking method [31] compares and matches image patches in successive frames using the Sum of Squared Difference (SSD). To incorporate temporal information, many methods exploit the motion constraints and estimate the probabilities of matches, such as in Extended Kalman Filter (EKF) [6]. In the MIS environment, the tissues might have deformations and the surgical instruments might cause occlusion problems. Mountney et al. proposed an on-line learning mechanism and treated the tracking as a classification problem [37]. The Thin Plate Spline (TPS) model was successfully applied in [38] to track a region of a deforming surface. Richa et al. [39] extended [38] to track the heart surface with stereo cameras. Many other tracking methods were introduced and compared in [26].

The recovery of the heart motion is a fundamental task in cardiac surgeries, and feature tracking using stereo images from stereoscopic laparoscopes has shown promising results. Note that there are two kinds of feature matching with stereoscopic laparoscopes: temporal matching for successive frames and spatial matching between left and right images. Typically, feature points are detected in both left and right images, and feature points in the first frame are matched temporally with successive frames to enable tracking. In [40], Stoyanov et al. used a Shi-Tomasi detector [23] and an MSER descriptor [41] to perform spatial matching. The LK tracking [31] framework was used to track the initial features, and the intensity infor-
formation of both stereo images was used during the estimation of the warp [40]. It was reported in [42] that the use of the LK tracking framework was not very stable due to the large tissue motion and the fact that some feature points were not well tracked. In [43], feature-based methods [40] and Scale Invariant Feature Transform (SIFT) [28] are combined with intensity-based methods [39] to generate a hybrid tracker for the robustness purpose.

Since pixel intensities used in the first category are sensitive to lighting conditions, most of these methods make it difficult to track features across large viewpoint changes. On the other hand, in the second category, feature tracking methods are reliant on feature descriptors to represent feature points. Many feature descriptors have been presented, such as SURF [29], SIFT [28] and Binary Robust Independent Elementary Features (BRIEF) [44]. Feature descriptors are usually normalized and processed to overcome problems such as illumination and appearance changes; therefore, they are usually more robust than the intensity comparison in LK-based tracking. Due to the special imaging conditions in the MIS environment, descriptor-based feature matching is not robust with large viewpoint changes. To overcome this problem, different methods have been introduced to exploit the geometry property of the tissue surface. Puerto-Souza et al. clustered feature points into different groups, and the local area of each cluster was assumed to be planar [10, 45, 46]. A comprehensive study on the evaluation of different feature descriptors on MIS images was reported in [47].

One major challenge of descriptor-based tracking is the time-consuming calculation and matching of descriptors. Currently, without special hardware such as Graphics Processor Units (GPU), the SIFT feature extraction is still difficult for achieving real-time speed. Recently, from the speed point of view, Yip et al. proposed a significant tracking-by-detection method that achieved a speed of 15-20 Hz on a MIS scene with tissue deformation and instrument interaction [48, 49]. The major novelty of the presented method in [48, 49] is that a feature list is dynamically maintained and updated, which makes it robust to the large deformation and occlusion. In [48], for speed consideration, the Star detector [50] implementation of the Center surround Extremas (CensurE) [51] feature detector and BRIEF descriptor were used. To further speed up the tracking process, the prior information of the surgical scene, such as the small camera motion and small scale change, was exploited to reduce the unnecessary feature comparisons [48]. An extensive comparison of tracking accuracy and speed among Star + BRIEF, SIFT, and SURF was provided in [49].

9
To evaluate feature detection and feature tracking, one key task is to generate the ground truth point correspondences across multiple views. To obtain the ground truth point correspondences, typically, experienced human subjects are trained to select the same scene point in multiple images. However, the ground truth information sometimes might not be accurate enough due to the manual selection process. To minimize the ground truth error, Maier-Hein et al. [52] extended a crowd-sourcing-based method to generate reference correspondences for endoscopic images. The correspondence error was reduced from 2 pixels to 1 pixel after applying crowd sourcing [53].

After the ground truth point correspondences are generated for each frame, the evaluation of feature point tracking can be successfully carried out [26]. To evaluate the feature point detection, traditional methods such as [27] usually rely on planar scenes so that global homography mappings are available. Klippenstein and Zhang [54] estimated the fundamental matrices between the first frame and other frames and defined the distances of feature points to the epipolar lines as the error for feature tracking. Different feature detectors and feature matching methods have been compared in [54]. However, the mappings used in [54] are not bijective and, therefore, the definition of error is not accurate. Selka et al. [55] reported a forward-backward tracking method for evaluation of both feature detectors and feature tracking. In [55], the MIS video sequence was re-organized into the order \((I_0, I_2, \ldots, I_{n-2}, I_n, I_{n-1}, \ldots, I_3, I_1, I_0)\). Those points that were detected in both the first and last frames were called robust points, and the percentage of robust points was used to represent the performance of feature detector and feature tracking.

2.3.3 Discussion

Feature detection and feature tracking are well-studied topics in Computer Vision. However, distinctive feature detection, matching, and tracking for endoscopic images are still challenging due to the specialty of the endoscopic imaging environment, such as poor texture, bleeding, smoke, and moving light sources. One future research direction is to exploit the special structures shown in laparoscopic images, such as blood vessels and blood dots caused by surgical instruments. Since light sources are mounted at the tip of a laparoscope, the light illumination is non-uniform and increases the difficulty in finding the image-point correspondences. It is interesting to look into how to remove or reduce the influence from this inhomogeneous
illumination from the endoscopic lighting. Another promising research direction is to integrate supervised learning techniques into feature detection and tracking, such as the work in [37].

2.4 Reconstruction without Camera Motion

In this section, 3D reconstruction methods without the consideration of camera motion are introduced. These methods are separated into different categories based on the vision cues applied.

2.4.1 Stereo Cue

Stereo laparoscopes have become widely used in robotic surgery platforms, such as the da Vinci system, to provide 3D views for surgeons. Since no extra hardware is required, reconstruction using stereo laparoscopes has been considered as one of the most practical approaches for MIS [56]. Lau et al. used the Zero-mean Sum of Squared Difference (ZSSD) for stereo matching and, later, the heart surface was estimated using the B-spline-based method [57]. Kowalczuk et al. [58] evaluated the stereo reconstruction results of the operating field with porcine experiments. Recently, Stoyanov et al. proposed a novel stereo matching algorithm for MIS images, which was robust to specular reflections and surgical instrument occlusion. They proposed to first establish a sparse set of correspondences of salient features and then propagate the disparity information of those salient features to nearby pixels [56]. The propagation in [56] was based on the assumption that the disparity values of the nearby pixels in MIS images were usually very similar since many tissue organ surfaces are locally smooth. Stereo reconstruction of the liver surface is known to be difficult because of the homogeneous texture. Totz et al. [59] proposed a semi-dense stereo reconstruction method for liver surface reconstruction, which adopted a coarse-to-fine pyramidal approach and relied on GPU to exploit the parallelism. In [60], semi-dense stereo reconstruction results [56] from different viewpoints were merged to obtain large-area 3D reconstruction results of the surgical scene based on camera localization results from [61]. In [62], the local surface orientation was estimated based on the constraints from the endoscope camera and light sources and then fused with the semi-dense reconstruction from [56] to generate a gaze-contingent dense reconstruction. Stoyanov [63] reported a 3D scene flow method to estimate the structure and deformation of the surgical scene by imposing spatial and temporal constraints.
Distinctive feature points can be matched in stereo images to obtain a set of sparse 3D points. To achieve dense reconstruction results of tissue surfaces, different methods have been proposed to incorporate geometrical constraints of tissue surfaces. Richa et al. tracked feature points over stereo images and obtained the 3D positions of those feature points based on triangulation [43]. Later, the sparse 3D points were chosen as the control points in a TPS model, and a dense 3D shape was estimated [43]. Bernhardt et al. [64] analyzed the surgical scenes and presented three criteria for the stereo matching to remove outliers. After the outliers were discarded, the holes were filled with the median of their neighboring pixels’ values [64]. Chang et al. first obtained a coarse reconstruction using the Zero-mean Normalized Cross-Correlation (ZNCC) and then refined the disparity function using a Huber – $L^1$ variational functional [65].

2.4.2 Active Methods

Most of the above methods are dependent on the texture of tissue surfaces to establish feature point correspondences for reconstruction. These methods become unstable if tissue surfaces are poorly textured. To overcome this problem, many methods aim to actively project special patterns, using laser stripes or structured light, onto tissue surfaces and build correspondences based on those patterns. To obtain 3D position through triangulation, the Euclidean transformation between the monocular camera and the light source needs to be accurately calibrated, and the system after calibration has to be fixed during the whole reconstruction procedure.

Different methods have been proposed to project laser stripes on organ surfaces for reconstruction. In [66], a laser stripe was projected in the laparoscopic environment to measure intra-corporeal targets. To measure the 3D shape of the surgical site in real time, a laser-scan endoscope system with two ports was designed in [67]. For the calibration of this system, infrared markers were placed at the ends of both the camera device and the laser device and tracked by the OPTOTRAK system [67]. The root mean square error of measurements among those markers was reported to be 0.1 mm [67].

Instead of using laser stripes, other methods project an encoded light pattern on tissue surfaces. Different light patterns have been designed to establish the correspondences between the camera and the projector [68,69]. To recover the dynamic internal structure of the abdomen in real time, Albitar et al. [70] developed a new monochromatic pattern composed of three primitives: disc, circle, and strip. The images were processed
to detect and discriminate the primitives, whose spatial neighborhood information was used to establish correspondences between the captured image and the known pattern [70]. The developed system was able to project 29*27 primitives on an area of size 10*10 cm [70]. Later, Maurice et al. designed a new spatial-neighborhood-based framework to generate coded patterns with 200*200 features using the mean Hamming distance [71].

One major challenge of using either laser or structured light is that the whole 3D scanning system is usually too large to fit into the current MIS setup [72]. To overcome this size problem, Schnalz et al. designed a very tiny endoscopic 3D scanning system composed of a catadioptric camera and a sliding projector [72]. The sensor head in the scanning system had a diameter of 3.6 mm and a length of 14 mm [72]. The system was specifically designed for a tubular environment and was able to obtain the 3D depth at 30 fps with a working cylindrical volume of about 30 mm in length by 30 mm in diameter [72]. Clancy et al. [73] designed another tiny structured lighting probe with a 1.7 mm diameter. In their system, a set of points are projected, and each point is assigned a unique wavelength.

Recently, the Time-of-Flight (TOF) camera sensor has become popular for 3D reconstruction. Penne et al. [74] designed an endoscope system with a TOF camera sensor. Haase et al. [75] proposed a method to fuse structures recovered from different frames of a TOF sensor to obtain large-area reconstruction results. More details regarding the TOF-camera-based reconstruction methods can be found in [76].

### 2.4.3 Shading and Shadow Cue

As one of the well-studied 3D reconstruction methods in Computer Vision, Shape-From-Shading (SFS) is very appealing to researchers because it does not require extra hardware in MIS. Many researchers have attempted to apply SFS to recover the shape from a monocular camera [77]. Wu et al. first extended the SFS problem to a perspective camera and near-point-light sources and then applied it to reconstruct the shape of bones from near-lighting endoscopic video [78]. The application of SFS in MIS is difficult and has multiple restrictions. To begin with, endoscopic images do not satisfy the common assumptions required by SFS: Lambertian reflectance and uniform albedo [63]. Additionally, SFS generally is not possible to recover a complete 3D surface with one lighting condition because each pixel has only one intensity measurement, which is not enough to recover the surface orientation that has two degrees of freedom [79]. Therefore, mul-
tiple lighting conditions with a constant viewing direction are required to theoretically achieve a complete surface recovery, which is commonly known as Photometric Stereo (PS) [79]. Please refer to [79, 80] for more details about PS.

During the MIS procedure, shadows cast by the surgical instruments or the tissue itself are good sources of visual cue for reconstruction. Researchers also are interested in generating optimal shadows for MIS surgeries in terms of contrast and location of shadow-casting illumination [81]. In [82], an “invisible shadow” was generated by a secondary light source and was detected and enhanced to provide a depth cue.

2.4.4 Discussion

In stereo reconstruction, because of the similarity of left and right images from stereo cameras, feature point matching between the two channels is relatively easy, and a sufficient number of feature point correspondences can be established if rich texture is available. Currently, one of the main challenges in stereo reconstruction for MIS is how to obtain dense reconstruction results. Interesting future research directions include building suitable models for tissue surfaces and integrating laparoscope motions, such as the work in [83]. Active methods are able to obtain accurate 3D information without depending on tissue texture and, therefore, are attractive to researchers. The main drawback of the active methods is the requirement of extra hardware in current surgical platforms. In the future, it will be necessary to design very small-scale hardware that is compatible with the MIS surgical platform. Meanwhile, how to generate optimal structure patterns for MIS is also an important research topic [71]. Methods based on defocus have also shown the ability to recover the 3D structure of tissue surfaces [84] and need further investigation. To better apply SFS in MIS, a more advanced reflectance model for the laparoscopic environment is needed [76].

2.5 Rigid MIS-VSLAM

In the previous section, no camera motion is considered during the 3D reconstruction process and, hence, the motion cannot be recovered or used. In practice, the endoscopic cameras are usually moving during the MIS procedure, and the motion can be used to recover the 3D structure. Additionally, knowledge of the camera pose is crucial to help surgeons better understand the surgical environment. For example, accurate camera tracking is necessary for safe navigation and instrument control during the endoscopic Endonasal...
Skull Base Surgery (ESBS) [85]. Many external endoscope tracking methods that rely on passive optical markers have been presented and have been used to track the location of an endoscope relative to CT. Shahidi et al. [86] reported millimeter tracking accuracies of a marker-based external tracking system. Lapeer et al. [87] showed that sub-millimeter accuracy was still difficult to achieve. Mirota et al. [85] presented an endoscope-tracking method that relies on the video content only and achieved accuracy at one millimeter. Compared with external-marker-based tracking systems, the video-based endoscope localization has the advantage that no marker or external system is needed. In addition, the passive markers might be blocked from the tracking system during surgery and cause tracking failures. Therefore, the combination of external tracking and video-based tracking potentially can offer more robust tracking results.

An extensive discussion on external tracking is beyond the scope of this dissertation. More detail regarding external tracking is available in [87]. From here forward, we focus on video-based camera tracking. In this section, the surgical scene is assumed to be rigid (static), and methods that simultaneously estimate the 3D structure and the camera motion are introduced. The illustration of MIS-VSLAM methods in rigid scenes is shown in Fig. 2.2.
In Computer Vision, many methods in Structure from Motion (SFM) have been proposed to estimate the sparse 3D structure of a rigid scene from a set of images taken at different locations [88, 89]. The technique has been scaled up successfully to a large dataset with millions of images taken from Internet [89]. Also, much research has applied SFM in MIS to expand the field-of-view for surgeons and recover a wide area of 3D structures [90, 91]. The results of SFM are greatly based on the ability of establishing correspondences between widely-separated cameras, which requires robust wide-baseline feature matching, such as SIFT on images of man-made buildings. However, the wide-baseline feature matching is difficult for low-contrast MIS images. Hu et al. presented a method to alleviate this problem and apply it in Totally Endoscopic Coronary Artery Bypass (TECAB) surgery [42]. In [42, 92], a genetic/evolutionary algorithm was proposed to overcome the missing data problem during the LK tracking. Another drawback of SFM is that it processes all images together to optimize the 3D structures and the cameras’ poses. One benefit of this global batch optimization is that the recovered structure and camera poses can achieve high accuracy. However, the number of parameters is large and the optimization requires expensive computation, which makes the system impractical for the real-time purpose. To reduce the difficulty, in [93], the laparoscope was externally tracked to provide camera poses in the optimization of SFM.

Different from SFM, in Robotics, one main task is to achieve real-time camera localization. Robotics researchers treat the camera as a sensor observing and moving in an explored or unexplored environment, and the problem is normally termed as “visual SLAM.” SLAM is a well-studied topic in Robotics and has been applied to the automatic navigation of mobile robots in an unexplored environment. A comprehensive survey paper about SLAM can be found in [4, 5]. Originally, SLAM was designed for range sensors, such as laser range finder and sonar systems, which obtain 3D information with uncertainty directly from the sensor reading. Different from that, a monocular camera is a bearing-only sensor that needs at least two measurements from different locations to calculate the 3D information. However, the availability of camera and rich information in each image has made the camera a popular sensor for SLAM.

2.5.1 Monocular Camera

Burschka et al. [94, 95] proposed an early framework to simultaneously estimate 3D structure and camera pose based on endoscopic video. However, the estimation of camera poses in [94, 95] is performed
frame by frame using the correspondences detected in successive frames, which might lead to a significant accumulated error. To overcome the aforementioned difficulty of feature matching in MIS images, Wang et al. first applied Singular Value Decomposition (SVD) matching [96] on SIFT points to obtain more but less accurate correspondences, which were further refined by a novel method called Adaptive Scale Kernel Consensus (ASKC) [97]. With the feature correspondences from successive frames, the method in [97] maintained a 3D feature point list and tracked the camera at each frame. Mori et al. designed a visual SLAM system specifically for a bronchoscope [98], in which the motion of the bronchoscope was initially estimated based on optical flow and was later refined by intensity-based image registration.

The seminal work of Davison [6] is the first significant real-time system that successfully applied the Extended Kalman Filter-SLAM (EKF-SLAM) framework for a hand-held monocular camera. In [6], feature points were detected by the Shi-Tomasi operator [23] and represented as 2D square patches. The measurement model for the monocular camera first initialized a 3D line when a new feature point was observed and then calculated the 3D position of the feature point when it was observed the next time [6]. Since Davison’s system updates the pose and the map at each frame, it can maintain only a small number (typically fewer than 100) of landmarks.

Multiple methods have been introduced to improve Davison’s monocular camera EKF-SLAM framework. To overcome the delayed initialization problem of a feature point in [6, 99], Civera et al. [100, 101] presented an inverse depth parameterization method to unify the initialization and tracking of both close and distant points. Civera et al. [102, 103] further integrated the RANdom SAmples Consensus (RANSAC) method into the EKF-SLAM framework [6, 99] to estimate inliers of feature point matches and presented the 1-point RANSAC method. With the prior information of camera poses, only one sample was needed to initialize the model estimation in the RANSAC process and, therefore, the RANSAC computation can be greatly reduced [102, 103]. Based on the inverse depth parameterization [100, 101], Grasa et al. [104, 105] successfully combined the 1-Point RANSAC method [102] and randomized list re-localization [106] together so that the system was robust to the challenges from the MIS environment, such as sudden camera motion and surgical instrument occlusion. In a more extensive evaluation of the system from [105], more than 15 human ventral hernia repair surgeries were reported in [107], in which the scale information was
obtained from the clinch of the surgical instrument. The measurements of the main hernia axes were chosen to represent the accuracy of the reconstruction and the ground truth was measured by tape [107].

In SFM, the time-consuming bundle adjustment has been shown to be very effective in simultaneously optimizing 3D structure and camera poses. To apply the bundle adjustment in a real-time system, different methods have been reported and discussed to reduce the computational burden of the bundle adjustment in Robotics. The local bundle adjustment was used in [108] to achieve accurate reconstruction results and simultaneously reduce the computation. Later, Klein and Murray introduced the breakthrough work, Parallel Tracking and Mapping (PTAM), which was able to robustly localize the camera in real time and recover the 3D positions of thousands of points in a desktop-like environment [109]. Due to the fact that the camera-pose update with a fixed map is much more efficient than the map update with known camera poses, Klein and Murray proposed to separate the tracking and mapping into two parallel threads. To achieve a real-time speed, the tracking thread was given a higher priority than the mapping thread [109]. In the mapping thread, the time-consuming bundle adjustment optimization [110, 111] was run to refine the stored 3D points and camera poses [109]. The benefits of separating tracking and mapping include more robust camera tracking and more accurate 3D point positions.

Based on the results of camera tracking from PTAM, many research efforts [112–114] have been proposed to generate a consistent dense 3D model in real time. In [112], the 3D points from PTAM was triangulated to build a base mesh using MSCRBF [115]. This base mesh was then used to generate a synthesized image, which was compared with the real images captured by the camera at the same position to iteratively polish the dense model [112]. In [112], TV-L1 optical flow [116] was applied to establish the correspondences between synthesized images and real images. The dense model from [112] was later used to improve the camera tracking in [117]. Instead of using variational optical flow as in [112], Graber et al. adopted the multi-view plane-sweep to perform 3D reconstruction with the high-quality depth map fusion [113]. Based on the PTAM and the work of Graber et al. [113], Wendel et al. [114] developed a live dense volumetric reconstruction for micro aerial vehicles.

After the recovery of the 3D structure from monocular endoscopic videos, researchers have attempted to register the recovered 3D structures with the pre-operative data. Burschka et al. [94,95] proposed to register the recovered 3D points with a pre-operative CT model to achieve accurate navigation in sinus surgeries.
To obtain accurate navigation for ESBS surgeries, Mirota et al. [118] introduced a new registration method, which was later applied in [85, 119], to register the 3D point cloud from [97] with the CT data.

2.5.2 Stereo Cameras

Stereo cameras have gained popularity recently in robot-assisted surgeries, such as the da Vinci system [61, 63]. Compared with the bearing-only sensor of the monocular camera, stereo cameras can get the 3D locations of landmarks directly from a single measurement. Early work of SLAM with stereo cameras on a mobile robot can be found in [120–122]. Currently, researchers in this area are interested in applying stereo SLAM in a large environment and overcoming the corresponding problems, such as time consumption and loop closure [123–127]. Mountney et al. [128] applied the stereo SLAM technique in the MIS environment to track the stereoscope and reconstruct a sparse set of 3D points. A Shi-Tomasi feature point detector was used to find interest points, which were represented by 25*25 patches and tracked using ZSSD correlation [128]. A “constant velocity and constant angular velocity” model was adopted to describe the endoscope motion [128].

A stereo EKF-SLAM framework developed by Mountney [128] has been widely used since it was introduced. Noonan et al. presented a newly-designed stereoscopic fiberscope imaging system and applied the stereo EKF-SLAM from [128] in the limited-resolution videos generated by the imaging system [129]. To get a larger field-of-view for surgeons, Mountney and Yang integrated the output of the sparse 3D points and camera tracking from [128] into the dynamic view expansion system [130] and textured the 3D mesh with past and current images [131]. Warren et al. [132] pointed out that disorientation was a major challenge in the Natural Orice Transluminal Surgery (NOTES). They used an Inertial Measurement Unit (IMU) attached at the tip of the endoscope to stabilize the image horizontally [132]. The stabilized images were further integrated into the dynamic view expansion system [131] to provide a more realistic navigation [132]. Totz et al. reported that the sparse 3D mesh generated from the Mountney et al. stereo SLAM was not rich enough to represent the real 3D shape of the scene, which caused visual artifacts in the final textured-mapped 3D model [60]. To overcome this problem, Totz et al. used the sparse 3D points to register a couple of semi-dense 3D surfaces from stereo reconstruction [56] together to generate a larger and more accurate 3D model, which resulted in more consistent rendering results with dynamic view expansion [60].
2.5.3 Discussion

In a rigid MIS environment with rich texture on tissue surfaces, visual SLAM has shown to work well in estimating camera poses and recovering a sparse set of 3D points [107]. However, the visual SLAM greatly relies on the successful extraction of distinctive image features. Therefore, further studies are needed to extract a distinctive image feature for MIS images. On the other hand, recently, a new visual SLAM framework [133, 134] was presented that does not require the detection of image feature points. The system exploits and reconstructs each pixel with valid image gradients. This framework does not rely on image feature points and can be useful for the MIS environment.

2.6 Dynamic MIS-VSLAM

The assumption of rigid scenes in the previous section might not be valid in a general MIS environment. This section focuses on the general problem of MIS-VSLAM, which is termed as dynamic MIS-VSLAM, as illustrated in Fig. 2.3. In a typical MIS environment, the tissue surfaces might undergo non-rigid deformation caused by heartbeats, breathing, and surgical-instrument interaction. Meanwhile, multiple surgical instruments might be moving dynamically in the scene and cause occlusion problems. Correspondingly, there are two fundamental tasks for dynamic MIS-VSLAM: the theoretical treatment of tissue deformation [61, 63] and the moving-instrument tracking. The first task is similar to the recovery of surface deformation with a moving camera, which has been an active research topic in Computer Vision and belongs to the broader Non-Rigid Structure from Motion (NRSFM) [135]. NRSFM has been proposed to analyze non-rigid scenes, such as smooth surfaces, articulated bodies, and piecewise rigid surfaces [136]. The general problem of NRSFM is considered to be ill-posed if arbitrary deformations are allowed [136]. In an MIS environment, the smooth tissue surfaces cast additional constraints to the general NRSFM and, therefore, the problem becomes less difficult. This section focuses on the introduction of the two essential tasks in dynamic MIS-VSLAM: NRSFM with deforming tissue surfaces, termed Deforming Surface SFM (DSSFM), and dynamic surgical instrument tracking.
2.6.1 State-of-the-art DSSFM

Many approaches have been presented to tackle the problem of DSSFM and can be broadly classified into two categories based on whether a monocular camera or stereo cameras are used. Those approaches are summarized in Table 2.2.

2.6.1.1 DSSFM with Monocular Cameras

Different from rigid SFM, each point in DSSFM can deform due to both global rigid motion and local deformation, which are difficult to differentiate. Therefore, different constraints of the deformation from the inherent geometry of the shape have been introduced [135, 137, 138, 142, 146, 147]. It is normally considered that the work of Bregler et al. [137] is the first approach that successfully extended Tomasi and Kanades’ factorization method [148] to non-rigid scenes. In [137], the idea of representing a 3D shape as a linear combination of a set of basis shapes was introduced, which greatly reduced the number of unknown parameters. This idea of linear combination of basis shapes has been widely adopted since it was introduced. Most following research has focused on the convergence of the optimization by adding spatial and temporal smoothness constraints [142]. One impractical assumption of the Bregler et al. method is the scaled ortho-
Table 2.2: Summary of different approaches in DSSFM. Each column from the second to the fifth represents: sensor types, optimization types, assumption of rigid points or not, and the camera model. “orth” is short for orthographic.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Sensor</th>
<th>Batch/ Sequential</th>
<th>Rigid</th>
<th>Camera model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xiao [138]</td>
<td>Mono.</td>
<td>Batch</td>
<td>No</td>
<td>Perspective</td>
</tr>
<tr>
<td>DelBue [139]</td>
<td>Mono.</td>
<td>Batch</td>
<td>Yes</td>
<td>Perspective</td>
</tr>
<tr>
<td>Wang [140]</td>
<td>Mono.</td>
<td>Batch</td>
<td>Yes</td>
<td>Affine</td>
</tr>
<tr>
<td>Hartley [141]</td>
<td>Mono.</td>
<td>Batch</td>
<td>No</td>
<td>Perspective</td>
</tr>
<tr>
<td>Paladini [142]</td>
<td>Mono.</td>
<td>Sequential</td>
<td>No</td>
<td>Orthographic</td>
</tr>
<tr>
<td>DelBue [143]</td>
<td>Stereo</td>
<td>Batch</td>
<td>No</td>
<td>Affine</td>
</tr>
<tr>
<td>Bartoli [144]</td>
<td>3D sensor</td>
<td>Batch</td>
<td>No</td>
<td>–</td>
</tr>
<tr>
<td>Llado [145]</td>
<td>Stereo</td>
<td>Batch</td>
<td>Yes</td>
<td>Perspective</td>
</tr>
</tbody>
</table>

Orthographic camera model. This camera model assumes that images are taken at a long distance from the objects. This restriction was later removed to allow the usage of more general perspective cameras and obtain the closed-form solution for linear basis shape models [138, 141].

Many DSSFM methods assume that all points are under non-rigid deformation, such as a piece of cloth under perturbation. In practice, a scene generally contains both rigid and non-rigid points – a common scenario in an MIS environment as well. In the publicly-available laparoscopic MIS image datasets [8], only those tissue organs that were interacted by surgical instruments display large deformation; other tissue surfaces mostly have small deformations that can sometimes be treated as rigid objects. Del Bue et al. introduced the idea of assuming the existence of both rigid and non-rigid points [139]. For a monocular camera, Del Bue et al. [139] used the RANSAC algorithm to segment rigid and non-rigid points based on the criterion that only rigid points could satisfy epipolar geometry. The purpose of the Del Bue et al. method is to estimate the 3D shape of human faces, where there are many fewer rigid points than non-rigid ones. The small percentage of rigid points requires a large number of sampling in the RANSAC process, which greatly slows the segmentation. In [139], to speed up the RANSAC process, Degree of Non-rigidity (DoN) was calculated for each point as the prior information and was used to guide the sampling of RANSAC. DoN was defined based on the observation that 3D positions of non-rigid points change from time to time and, hence, have larger variances than the rigid ones [139].
One major challenge of many existing monocular DSSFM methods is the expensive time consumption of the final non-linear optimization. Motivated by the significant real-time performance of PTAM [109], Paladini et al. proposed the first work to separate model-based camera tracking and model updating [142]. To enable sequential model updating, a sequential framework was presented that increased the degrees of freedom of basis shape whenever the current shape model was not able to represent a new shape [142]. Based on the dense 2D correspondences from [149], Garg et al. [136] formulated DSSFM as a variational energy minimization problem to estimate the 3D structure of deformable surface from a monocular video sequence.

2.6.1.2 DSSFM with Stereo Cameras

Since the relative pose between the stereo cameras is fixed, the factorization method [137] was extended to stereo cameras by stacking the constraints from each camera together [143, 150]. A novel method of decomposing the measurement matrix to get stereo camera pose and 3D shape was presented in [143, 150]. Assuming that correspondences are available between stereo cameras, the 3D positions of those points for each frame can be obtained through triangulation. Therefore, the input to the DSSFM becomes 3D point tracks rather than 2D point tracks as in the monocular case. With 3D point tracks as input, Llado et al. [145] extended the rigid and non-rigid point segmentations from a monocular camera to stereo cameras based on the fact that only rigid points satisfied a global Euclidean transformation. After RANSAC estimation, the classification of rigid points and non-rigid points was further refined based on the accumulated 3D registration errors, which were large for non-rigid points and small for rigid ones [145].

Another significant stereo DSSFM method was presented by Bartoli [144]. In [144], Bartoli first learned the basis shapes by maximum likelihood, and then the learned basis shapes were used to estimate the stereo rig’s poses as well as the configuration weights. There is a major difference between the Llado et al. method [145] and Bartoli methods [144]. In [145], Llado et al. estimated the poses, basis shapes, and configuration weights altogether by non-linear optimization, which minimized the re-projection error. Bartoli proposed to learn the basis shapes first from a sequence of 3D shapes and then minimized the 3D registration error to estimate basis shapes and configuration weights [144]. Besides stereo cameras, a multi-camera setup [151]
Table 2.3: Dynamic visual SLAM in MIS.

<table>
<thead>
<tr>
<th>Scene</th>
<th>Monocular camera</th>
<th>Stereo cameras</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rigid</td>
<td>[98], [94], [95], [104], [97], [85], [119]</td>
<td>[128], [129], [60], [131], [132], [152]</td>
</tr>
<tr>
<td>Deforming</td>
<td>[153], [105], [154]</td>
<td>[155], [156]</td>
</tr>
</tbody>
</table>

also has been considered to solve the DSSFM problem. Even though many methods have been presented, DSSFM is still considered a very difficult problem and remains an open challenge to researchers.

2.6.2 DSSFM in MIS Environment

Currently, most DSSFM methods assume that all 3D points are correctly detected and tracked in each frame. This assumption is impractical in reality, because the feature matching might contain mismatches due to noise. In an MIS environment, the low-contrast images, non-rigid deformation of organs, and dynamic moving of surgical instruments further complicate this issue. Despite these difficulties, different methods have been proposed to simplify the problem by adding practical constraints from the MIS environment, as summarized in Table 2.3. Some significant methods were chosen as representative; their properties are displayed in Table 2.4. This section first introduced the methods proposed to overcome tissue deformations and then discussed methods designed to track moving objects.

Many researchers have attempted to reduce tissue deformations by re-arranging or segmenting the videos. Hu et al. [154] applied the Probabilistic Principal Component Analysis (PPCA)-based NRSFM [135] to reconstruct a beating heart surface and estimate the camera poses. To reduce the complexity from deformation, the video sequence was re-arranged, and the images of the same heart cycles were chosen to reduce tissue deformation [154]. In the method of [154], some feature points may be lost during the tracking, and it is unclear how this problem is compensated. Collins et al. [153] argued that tissue motion was small within a couple of frames and, hence, could be treated as rigid. With this assumption, Collins et al. [153] presented a method to divide the video sequence into small segments, and the motion within each segment was approximated as rigid.

Researchers also observed that some tissue deformations might follow certain periodic patterns, such as respiration and heartbeats. These periodic patterns can be learned and used as constraints to overcome
Table 2.4: Summary of the state-of-the-art methods in MIS-VSLAM. The third column represents optimization frameworks. “ASKC” refers to [97]. “Factor.” means matrix factorization method of rigid SFM. “NSSD” refers to normalized SSD. “Segment.” is short for segmentation. “Regis.” represents pre-operative registration.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Burschka [95]</td>
<td>Rigid</td>
<td>Sequ.</td>
<td>ASKC</td>
<td>Mono.</td>
<td>Segment.</td>
<td>SSD</td>
<td>Sinus</td>
<td>Yes</td>
</tr>
<tr>
<td>Wang [97]</td>
<td>Rigid</td>
<td>Sequ.</td>
<td>ASKC</td>
<td>Mono.</td>
<td>SIFT+SVD</td>
<td>SIFT</td>
<td>Sinus</td>
<td>No</td>
</tr>
<tr>
<td>Mirota [119]</td>
<td>Rigid</td>
<td>Sequ.</td>
<td>ASKC</td>
<td>Mono.</td>
<td>SIFT+SVD</td>
<td>SIFT</td>
<td>Sinus</td>
<td>Yes</td>
</tr>
<tr>
<td>Hu [92]</td>
<td>Rigid</td>
<td>Batch</td>
<td>Factor.</td>
<td>Both</td>
<td>–</td>
<td>LK</td>
<td>Heart</td>
<td>Yes</td>
</tr>
<tr>
<td>Mountney [128]</td>
<td>Rigid</td>
<td>Sequ.</td>
<td>EKF-SLAM</td>
<td>Stereo</td>
<td>Shi-Tomasi</td>
<td>NSSD</td>
<td>Abdomen</td>
<td>No</td>
</tr>
<tr>
<td>Totz [60]</td>
<td>Rigid</td>
<td>Sequ.</td>
<td>EKF-SLAM</td>
<td>Stereo</td>
<td>Shi-Tomasi</td>
<td>NSSD</td>
<td>Abdomen</td>
<td>No</td>
</tr>
<tr>
<td>Hu [154]</td>
<td>Deform.</td>
<td>Batch</td>
<td>NRSFM</td>
<td>Mono.</td>
<td>–</td>
<td>LK</td>
<td>Heart</td>
<td>No</td>
</tr>
<tr>
<td>Grasa [105]</td>
<td>Deform.</td>
<td>Sequ.</td>
<td>EKF-SLAM</td>
<td>Mono.</td>
<td>FAST</td>
<td>NSSD</td>
<td>Abdomen</td>
<td>No</td>
</tr>
<tr>
<td>Mountney [155]</td>
<td>Deform.</td>
<td>Sequ.</td>
<td>EKF-SLAM</td>
<td>Stereo</td>
<td>Shi-Tomasi</td>
<td>[37]</td>
<td>Abdomen</td>
<td>No</td>
</tr>
</tbody>
</table>

the challenges from tissue deformations. Mountney et al. [155] presented a SLAM framework for the MIS environment with periodic tissue deformations. In [155], the liver motion was described by a periodic respiration model and learned by temporally tracking the 3D points on the liver surface using stereo cameras. The learned respiration model was later integrated into the EKF framework for more accurate prediction of camera poses. However, the assumption of periodic motion is not valid for all tissues; for example, the tissue motion caused by the interaction of surgical instruments is mostly not periodic.

With the development and availability of miniaturized microelectromechanical systems, researchers have been trying to use inertial sensors to further improve visual SLAM performance. Giannarou et al. presented a novel method, adaptive Unscented Kalman Filter (UKF), to exploit the data from an IMU [158]. The IMU data were combined with visual information to achieve better camera pose estimation for deformable scenes in MIS [158].
2.6.3 Moving Instrument Tracking

In visual SLAM, the dynamic moving objects cause failure to the camera localization and, therefore, should be properly tracked. In robotics, this problem is generally referred to as SLAM and Moving Object Tracking (SLAMMOT), which deals with dynamic environments containing moving objects, such as human and cars. Wang et al. reported the first work that successfully detected and tracked moving objects within a visual SLAM system [159]. In [160], a mathematical framework was introduced and a general solution was provided to the problem of SLAMMOT. Recently, Lin and Wang presented a stereo-camera-based approach for SLAMMOT, which overcame the observability issue that was common in monocular approaches [161]. Zou et al. [162] presented the first work that applied visual SLAM with multiple independent cameras in a dynamic environment, in which it was shown that with multiple cameras, the rigid and moving points could be classified based on the re-projection distance. Also, each camera’s pose and the 3D locations of moving points can be successfully recovered by considering nearby cameras’ observations of the landmarks [162].

In MIS, different techniques have been introduced to track the dynamically-moving surgical instruments. In a typical MIS setup, the instruments are inserted through small incisions and their motions are, therefore, greatly restricted. Voros et al. measured the 3D position of the insertion point of an instrument and exploited the 3D instrument model to constrain the search space and achieve accurate instrument detection [163, 164]. Allan et al. [11] argued that the estimated trocar positions might be inaccurate due to trocar and patient movement. They proposed a probabilistic supervised classification method, which did not require the estimation of the trocar positions [11]. Their method first detected pixels belonging to surgical instruments and then estimated the pose of those instruments [11].

Endoscope-video-based object tracking has many applications. In [165], a suturing needle was tracked, and 3D cue information was augmented in the video to help surgeons better understand the poses of the needle. Jayaratne et al. introduced a method to track the ultrasound probe using the standard monocular endoscopic camera so that the magnetic tracking could be obviated [166]. They presented an EKF framework to establish the correspondences and estimated the pose of the ultrasound probe [166].
2.6.4 Discussion

To overcome the difficulties in dynamic MIS-VSLAM, it is essential to exploit the prior information of surgical scenes and use them as constraints. Since tissue surfaces are smooth and have special deforming properties, one important research topic is to learn bio-mechanical models of tissue deformations. Organs and tissues have specific shapes and biological properties, which greatly restrict how they would deform. Those bio-mechanical models are usually similar among different people and can be learned before the surgery.

In the abdominal area specifically, a large portion of the surgical scene remains relatively still during the whole surgery procedure, such as the abdominal walls. These rigid areas can be pre-identified and used to separate camera-pose estimation and deformation recovery. Regarding instrument tracking, the 3D models of surgical instruments can be obtained and used as the prior for instrument tracking.

Video-based camera tracking and 3D reconstruction rely on robust image feature detection and matching. However, some tissue surfaces do not have a distinctive texture. For example, the texture of the liver surface is mostly repetitive, which makes image feature detection difficult. In this case, the extra information from tissue organs is needed. For instance, contours of the liver can be accurately detected and matched to its 3D model from preoperative data to estimate its pose and deformations. On the other hand, it might be necessary to actively project patterns on tissue surfaces to build correspondences for 3D reconstruction. Therefore, structured-lighting-based methods, such as depth sensors, are very promising directions to solve the low-texture problem.
CHAPTER 3
VESSEL-BASED IMAGE FEATURE DETECTION

3.1 Note to Reader

This chapter was published in the IEEE Transactions on Biomedical Engineering (TBME) [167]. Permission to reproduce the work in this dissertation is included in Appendix A.

3.2 Vessel Feature Introduction

Image feature detection methods introduced in the previous chapter are usually designed for man-made environments, such as tables, chairs, and buildings. Those image features are referred to as “general image features” in this chapter. Abdominal MIS images are taken inside the human abdomen and therefore have very special properties. For example, MIS images are generally low-texture and contain abundant specular reflections, homogeneous areas, smokes, and so on.

To overcome the difficulties mentioned above, many research results have been presented. Feature detectors and descriptors designed for MIS images to overcome tissue deformation were presented in [25, 26, 47]. Puerto-Souza and Mariottini proposed the novel hierarchical multi-affine (HMA) and adaptive multi-affine (AMA) algorithms [10, 46] to improve the feature matching performance for endoscopic images. They also developed a dense feature matching method to recover the locations of image features on tissue surfaces [10]. Tissue surface tracking and reconstruction for MIS have also been widely studied and different methods have been introduced to overcome the difficulties of tissue deformations and low texture [2, 37, 49, 56]. More details on the optical surface reconstruction and tissue surface tracking methods for MIS are available in [63, 76].

One goal of this study is to design efficient algorithms that can detect robust and repeatable MIS image features across different viewpoints and different lighting conditions. It is desirable to develop a feature
detector that will turn the drawbacks of the in vivo environment to advantages. We notice that blood vessels are abundant on the surface of tissue organs, such as the abdominal wall, stomach, small intestines, and colons. The explicit extraction of blood vessels can provide a large number of new types of features for MIS image analysis.

Blood vessel detection is one of the fundamental research tasks in image guided surgeries and have many medical applications. For example, in neurosurgeries, Ding et al. estimated the cortical displacement based on blood vessel detection, and overcame the problem of brain shift and deformation caused by the pressure after the open of dura [168, 169]. In simultaneous localization and mapping (SLAM) system, blood vessels can be represented as curves and used to estimate camera motion. It has been known that curves are more robust than points in camera motion estimation [170]. Since vessels are attached on the tissue surfaces and deform with them, the detection of vessels is crucial to recover the tissue deformations [169]. In retinal image analysis, the vessel detection and segmentation can provide important information for the diagnose purpose.

3.3 Vessel Feature Detection

3.3.1 Method Overview

Two types of blood vessel features are defined here: branching points and branching segments. Bifurcations and crossing points are defined as branching points. We consider a blood vessel segment that has branching points at both ends as a branching segment. A blood vessel segment that has only one branching point is called a half branching segment. An example image with two branching points and one branching segment is shown in Fig. 3.1. Note that branching segments are essentially curve segments and a pair of branch segment correspondence can generate tens of pairs of point correspondences. In this chapter, a new way of blood vessel enhancement is proposed based on a new ridgeness measure (Section 3.3.4), which provides more accurate vessel localizations. Based on the ridgeness representation, more robust methods of vessel feature detection are presented.

A novel branching point detector, ridgeness-based circle test (RBCT), and a novel branching segment detector, ridgeness-based branching segment detection (RBSD), are introduced in this dissertation. The overview of our proposed vessel feature detection is shown in Fig. 3.2. First, image pre-processing, such as
Figure 3.1: Illustration of vessel features: branching point (detected by RBCT) and branching segment (detected by RBSD).

![Diagram of vessel features](image)

Figure 3.2: Overview of the branching point and branching segment detection.

specular reflection removal, is applied on the input image. Then, Hessian matrix is calculated for each pixel, based on which Frangi vesselness and ridgeness are computed. Next, circle tests are performed to detect branching points. Last, the vessel tracing technique is introduced to detect branching segments.

### 3.3.2 Detecting Candidate Branching Points

It is known that among all three channels in the RGB fundus image, the green channel provides the best contrast between vessels and the background [18]. Our experiments show that this also applies to MIS images and, hence, only the green channel is used in our method. Another special property of MIS images is the abundant specular reflections that are view-dependent and, therefore, can cause error to endoscope tracking if they are picked up as feature points. Similar to [39, 171], the specular reflections are detected as
Table 3.1: Eigenvalues analysis towards vessel and branching points ($0 < \lambda_1 < \lambda_2$). L: low, M: middle, H: high

<table>
<thead>
<tr>
<th>$\lambda_1$</th>
<th>$\lambda_2$</th>
<th>pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>L</td>
<td>background noise</td>
</tr>
<tr>
<td>L</td>
<td>H</td>
<td>dark tubular structure</td>
</tr>
<tr>
<td>M</td>
<td>H</td>
<td>branching point and spur</td>
</tr>
<tr>
<td>H</td>
<td>H</td>
<td>blob, specular reflection</td>
</tr>
</tbody>
</table>

the pixels whose intensities are larger than a global threshold. In addition, their 3-by-3 neighbors are also marked as specular reflections.

The scale-space representation of an image $I$ (green channel) is $L(x, y, \sigma) = G(x, y; \sigma) \ast I(x, y)$, where $G(\cdot; \sigma)$ is a 2D Gaussian function with standard deviation $\sigma$, and $\ast$ represents convolution operation. A Hessian matrix is calculated for each pixel in each image level of the scale space, as shown in Equation 3.1. Note that, here, the scale space is only used during the calculation of Frangi vesselness [172] and ridgeness (Section 3.3.4) and all the remaining calculation is based on the single Frangi vesselness image or the ridgeness image.

$$H = \begin{bmatrix}
\frac{\partial^2 L}{\partial x^2} & \frac{\partial^2 L}{\partial x \partial y} \\
\frac{\partial^2 L}{\partial y \partial x} & \frac{\partial^2 L}{\partial y^2}
\end{bmatrix}. \quad (3.1)$$

The eigenvalues of the Hessian matrix are denoted as $\lambda_1, \lambda_2$ and eigenvectors $V_1, V_2$. Negative eigenvalues indicate bright tubular structures and positive eigenvalues represent dark tubular structures [172]. In this work, since the vessels are dark on MIS images, the negative eigenvalues are removed and the eigenvalues are sorted so that $0 < \lambda_1 < \lambda_2$. It is known that the absolute values of the two eigenvalues represent the intensity variances of two orthogonal directions. The tubular structure has a small $\lambda_1$ because the variance along the vessel direction is small. At the endpoint of a vessel, the intensity variance is large along the vessel. The branching point can be considered as the connection of three or four vessel segments and, hence, it has a larger $\lambda_1$ than other points on the vessels. Blob has a large intensity variance in almost every direction, therefore, it has the largest $\lambda_1$. Similar to [172], the relationship of eigenvalues and the pixel type is summarized in Table 3.1.
To detect bifurcations, Baboiu and Hamarneh presented three measures with similar performance: $\lambda_1$, $\lambda_1 \cdot \lambda_2$ and $1 - \exp(-2 \cdot (\lambda_1/\lambda_2)^2)$ [173]. The feature detector with the second measure is actually a variant of the above Hessian-Affine detector. Those measures are sensitive to noise and have a very high false positive detection rate, because many other structures also have high responses to those measures, such as blobs, specular reflections, and spurs. Therefore, it is difficult to distinguish branching points from other structures with those measures. In this study, the candidates of branching points are defined as: $\lambda_1 > \lambda_{1\text{min}}$ and Ridgeness > $R_{\text{min}}$ for a ridgeness image that we introduce later in Section 3.3.4. As an example, the $\lambda_1$ image and $\lambda_2$ image are shown respectively in Fig. 3.3 b) and c).

### 3.3.3 Blood Vessel Enhancement: Vesseness

Frangi vesseness (referred to as vesseness) was proposed in [172] and has become a popular method for tubular-structure enhancement. Similar as second-derivative-based feature point detection methods in the previous chapter, Vesseness is also based on the Hessian matrix. It first sorts the eigen-values of the
Hessian matrix based on their absolute values ($|\lambda_1| < |\lambda_2|$). Since eigen-vectors in the Hessian matrix represent the main direction of intensity change. Frangi et al. [172] observed that, for tubular structure, the eigen-vector corresponding to $\lambda_1$ was along the tube and the other eigen-vector was across the tube. Based on this observation, Vesselsness was proposed to give high response to pixels whose $\lambda_1$ is small and $\lambda_2$ is large. The definition of the Frangi vesselness is as follows:

$$V(\sigma) = \exp\left(\frac{\lambda_2^2}{2\beta^2}\right) \cdot \left(1 - \exp\left(-\frac{(\lambda_1^2 + \lambda_2^2)}{(2\epsilon^2)}\right)\right),$$

(3.2)

where $V$ stands for vesselness, $\beta$ and $\epsilon$ are soft thresholds from [172]. One example of vesselness image is shown in Fig. 3.4.

### 3.3.4 Blood Vessel Enhancement: Ridgeness

In this section, a new blood vessel enhancement technique is introduced, which is referred to as “ridgeness” in this dissertation. Different from the thick representation of vessels in the Frangi vesselness, we look for ridge pixels that achieve single-pixel width. Here, the width of ridges is defined as the number of pixels in the direction of the eigenvector $V_2$ (across the vessel).

The ridge in a 2D image is a good approximation of the vessel center-line and has been extracted for vessel segmentation [18]. Compared with the vessels in the vesselness image, the ridges are thinner and
clearer. In [18], ridges are defined as pixels where the first derivative of the raw image intensity changes sign in the direction of the eigenvector $V_2$ (across the vessel). One example of the detected binary ridges based on the above definition is shown in Fig. 3.5 a). Since a small amount of intensity change might flip the sign of the first derivative, the above definition tends to detect massive “ridges” with many false positives, which include tiny vessels and background noise, as shown in Fig. 3.5 a). The width of the detected ridge is two pixels under this definition. Since the goal of our method is to robustly and repeatedly detect vessel features, the “false” ridges from the background need to be filtered out. As shown in Table 3.1, both eigenvalues of pixels from background noise are small. Therefore, instead of using the binary ridges directly, the pixels of the ridges are first weighted by their corresponding vesselness values. The obtained measures for ridges are called:

\[
\text{Ridgeness}(x, y, \sigma) = \text{Vesselness}(x, y, \sigma)
\times \text{abs}\{\text{sign}(\nabla I(x + \epsilon u_2, y + \epsilon v_2, \sigma) \diamond (u_2, v_2)^T) 
- \text{sign}(\nabla I(x - \epsilon u_2, y - \epsilon v_2, \sigma) \diamond (u_2, v_2)^T)\}/2,
\]

where $\nabla$ is the gradient operator, $\times$ represents multiplication, $\diamond$ indicates dot product of two vectors, $(u_2, v_2)^T = V_2$, and $\epsilon = 1.0$ pixel. Up to now, the width of the detected ridge is mostly two pixels. To obtain more accurate single-pixel width ridges, each ridge pixel is further required to be the local maximum in the direction of the eigenvector $V_2$. The final definition of our new ridgeness measure is shown in Equation 3.4:

\[
R(\cdot) = \begin{cases} 
R(\cdot) & \text{if } R(x \pm \epsilon u_2, y \pm \epsilon v_2, \sigma) < R(\cdot) \\
0 & \text{otherwise}
\end{cases}
\]

where $R$ stands for Ridgeness and $R(\cdot) = \text{Ridgeness}(x, y, \sigma)$.

As an example, the binary ridge image and the ridgeness image are shown in Fig. 3.5 a) and b) respectively. In the ridgeness image, the background noise has been greatly reduced and the ridges now have a single-pixel width. However, in both the binary ridge and ridgeness images, many vessels are broken at branching points, referred to as “broken branching points” for clarity. Therefore, many segmentation-based
methods are not able to detect broken branching points. As we will show in Section 3.3.5, our branching point detection methods are based on checking pixels along the circle around the candidate points. Therefore, our method is still able to detect the broken branching points. The ridgeness value on the circle is shown in Fig. 3.7 b).

### 3.3.5 Branching Point Detection (RBCT)

Similar to [173], the detected candidates of branching points might contain blobs, specular reflections, branching points, and spurs. This section focuses on how to further distinguish branching points from the others. The major differences are their local structure patterns. One distinctive characteristic of branching points is that they have three or four connecting vessels. Many vessel segmentation methods have been proposed [174] and the branching points can be identified after the vessels are successfully segmented. Compare with those methods, the methods proposed in this dissertation have the advantage that they do not rely on any image segmentation techniques. Therefore, the proposed method does not need to solve optimization problems required by many image segmentation methods, such as [175]. Inspired by FAST feature point detector [20], we propose to place a circle centered at each candidate point on the ridgeness image and examine the ridgeness value and intensity of each point along the circle to determine whether it is a branching point or not. For clarity, this process of using a circle is termed as “circle test.”
Figure 3.6: Typical example of a circle test at a branching point on a) raw image, b) vesselness image, c) binary ridge image, and d) ridgeness image. As shown, binary ridge image has too much noise. Vessels are thinner on the ridgeness image than on the vesselness image.

illustrates the idea of the circle test at a branching point. A new method, RBCT, is introduced in this section to detect branching points by performing circle tests on the ridgeness image.

When a circle is placed at the branching point on a ridgeness image, the circle will intersect with the vessels and result in a special “white and black” pattern. Typically, for a bifurcation point, the intersections are three bright points or segments. Note that even though the ridges are single-pixel-width, the intersecting segment of a ridge and a circle might still have more than one pixel. If the intersecting segment is only one pixel, the pixel is defined as a peak. Otherwise, the point with the largest ridgeness in the intersecting segment is defined as a peak. The circle tests on binary ridge image and ridgeness image are shown in Fig. 3.6 c) and d). As an example, the ridgeness value of the pixels along the circle is shown in Fig. 3.7. Similar to VBCT [3], multiple tests are employed on each pixel p on the circle: 1) p should be bright on the ridgeness image \( R(p) > R_{\text{peak}} \); 2) p should have similar intensity with the center pixel \( |I(p) - I(\text{center})| < I_{\text{similar}} \); and 3) the middle point \( p_m \) of two peaks should be black \( R(p_m) = 0 \). 4) the number of peaks should be three or four. Note that bifurcations and crossing points have three and four peaks, respectively. Among those four tests, as long as one test is failed, the algorithm will exit early to save computation.
Because vessels have different widths, to detect as many branching points as possible, multiple circle tests with different radii are employed in RBCT. An example of candidate branching points before and after RBCT is shown in Fig. 3.8.

### 3.3.6 Connected Component Labeling and Non-maximal Suppression

Points that pass the circle test are not the final branching points yet. Depending on the viewing conditions and image resolutions, blood vessels have various widths and it is difficult to mathematically define a unique branching point. The circle test has the locality property that the neighboring pixels have a similar probability of passing the test. Therefore, those points that pass the circle test are grouped into different con-
nected components and each component actually represents one branching point. The 8-neighbor definition is used here to label the connected components with the two-pass algorithm [176]. After each connected component has been identified, its center is defined as the location of the branching point.

Different from corners that can be very close to each other, branching points are much more sparse and are usually far from each other. We further require that the distance between any two branching points should be larger than or equal to a predefined minimum distance. The minimum distance of branching points is determined by multiple factors, such as the resolutions of the images, the tissue-to-camera distances, and so on. Based on the collected datasets as discussed in Section 3.4.1, this minimum distance is set to be 11 pixels ($distance_{min} = 11$) in this dissertation. The minimum distance is ensured by non-maximal suppression with a 23-by-23 window. Since a large connected component is more robust than a small one, the number of points in each connected component is chosen as the score of the corresponding branching point and used in the non-maximal suppression process.

### 3.3.7 Branching Segment Detection (RBSD)

In this section, we describe the procedure of vessel tracing contained in RBSD. Since branching segment detection starts and ends at branching points, our algorithm starts from each branching point and initiates a vessel tracing process for each of its corresponding vessels. The vessel tracing process is the core of the branching segment detection, and our algorithm is based on the binary mask of vessels, which is obtained by thresholding the ridgeness image and is referred to as “ridge mask”. The ridge mask has a single-pixel width in most areas, except the specular reflections. The following discussion is based on the binary ridge mask. The vessel tracing process is recursive and stops under two conditions. First, another branching point is within a five-pixel radius ($radius_{BS} = 5$), which means a branching segment has been detected. Second, there are no unvisited ridge pixels, which results in a half branching segment.

There are three key points to determine in the vessel tracing process: the starting point, the next point, and the ending point. First, the detected position of a branching point is not directly used as the starting point, because the broken branching point may not be on the vessel. The three or four peaks from the circle test of each branching point are, therefore, chosen as the starting points for tracing.
To determine the next point and the ending point, some special points on the ridge mask have to be defined for clarity. A “forwarding point” is a point that is white on the ridge mask and has at least one white unvisited neighbor (under 8-neighbor). If a point, \( P \), has a neighbor that is a forwarding point, this neighbor is called “forwarding neighbor” of point \( P \). One example of forwarding point and forwarding neighbor is given in Fig. 3.9 b). The key observation of our vessel tracing is as follows: given the current tracing point \( P \), after marking \( P \)’s neighbors as visited, if \( P \) still has forwarding neighbors, we conclude that all these forwarding neighbors are along the vessel and in front of \( P \).

Based on this observation, the following two-pass tracing algorithm is applied in each iteration. The first pass is to collect all unvisited white neighbors of the current tracing point and mark them as visited. The second pass is to find all forwarding neighbors. If at least one forwarding neighbor is found, the next point can be chosen as either one of them; otherwise, this is the end of the current vessel tracing process. Regarding the ending point, if no branching point is found at the end of the tracing, the last point of the vessel tracing process is chosen as the ending point; otherwise, another branching point is found and is chosen as the ending point. The process of two-pass vessel tracing is illustrated in Fig. 3.9. The detected branching segments (green) and half branching segments (blue) are shown in Fig. 3.10 b) as an example.
Figure 3.10: Illustration of branching segment detection. Branching segments are shown as green, and their associated branching points are shown as cyan dots. Half branching segments are shown as blue, and their associated branching points are shown as red crosses.

3.3.8 Computational Analysis and Run Time Results

Since RBCT is a part of RBSD, this section focuses on the computational analysis of RBSD. There are three main components in the calculation of RBSD: ridgeness, circle test, and vessel tracing. Firstly, based on Equation 3.3 and 3.4, it can be seen that, the calculation of vesselness accounts for the main computation of ridgeness. Note that 2D version of Frangi vesselness with three image levels is used in this dissertation. Its calculation contains three Gaussian convolutions for each image level to obtain Hessian matrix and the calculation of eigenvalues in the 2-by-2 Hessian matrix for each pixel. Those calculations have been used and analyzed in [23, 24, 27] and can achieve real-time speed with proper implementation. Secondly, circle tests are performed on each candidate branching point, which is usually less than 1% of the total number of pixels. Typically, one circle test scans 64 pixels along the circle (11-pixel radius) and less than 6 computer instructions are executed to scan one pixel. Thirdly, the proposed vessel tracing algorithm visits each ridge point one time, at most, to detect all branching segments and half branching segments.

The run time tests of different steps in RBSD are performed on a Intel core i5 CPU 650 (3.20GHz) with 4.00GB RAM. The test data contains images with resolution 640*480 from an in vivo MIS dataset (scene7 from the public Hamlyn dataset [8] as introduced in Section 3.4.1). The proposed methods have been implemented using MATLAB. The average run time of different steps for one image are reported in Table 3.2. The results with unoptimized MATLAB implementation in Table 3.2 show that the proposed
Table 3.2: Run time of different steps in RBSD. “Labeling” represents connected component labeling. “Suppression” represents Non-maximum suppression. “S” stands for second.

<table>
<thead>
<tr>
<th>Method</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circle test</td>
<td>0.562</td>
</tr>
<tr>
<td>Labeling</td>
<td>0.027</td>
</tr>
<tr>
<td>Suppression</td>
<td>0.088</td>
</tr>
<tr>
<td>Vessel tracing</td>
<td>0.062</td>
</tr>
</tbody>
</table>

methods are fast and have the potential to achieve real-time speed with proper C/C++ implementation. Note that the main time consumption lies in circle test, which is independently executed at each candidate point. Therefore, the proposed methods can be further speeded up by graphics processing unit (GPU). The above computational analysis and run time results validate that the proposed methods are efficient and are good for real-time applications.

3.4 Experiments and Results

In this section, in vivo experiments were designed to evaluate the performances of the proposed vessel feature detectors. Many state-of-the-art branching point detectors [177], vessel detection methods [174,178], general feature point detectors have been proposed in the community. Among them, the following feature detectors were chosen based on the reports in [20, 26] and the availability of codes: VBCT [3], AFD [26], DoG [28], Hessian-Affine [27,173], FAST [179], and likelihood ratio vesselness (referred to as “Sofka” here) [178]. Note that VBCT, RBCT, and Sofka were branching point detectors. RBSD was a branching segment detector. The rest of them were not specifically designed for vessel images and were referred to as “general feature point detectors” here. Those general feature point detectors extracted different information from a given image: intensity (FAST), first derivatives (AFD), and second derivatives (Hessian-Affine, DoG). The implementations of Hessian-Affine and DoG from VLFeat library [180] were adopted here. The parameters of the above methods were chosen based on the datasets used in this chapter and the suggestions from the corresponding papers. Those parameters were shown in Table 3.3 and were fixed in all the experiments.

The parameters used in RBCT and RBSD were selected based on their performance on the datasets (Section 3.4.1) used here. During the detection of the branching point candidates, the following threshold values produced reasonable amount of candidates and were able to detect most branching points: \( \lambda_{1\text{min}} = 0.05 \), \( R_{\text{min}} = 0.01 \). In the calculation of the Vesselness, the following values were adopted based on the suggestions of [172]: \( \sigma = \{3, 4, 5\} \), \( \beta = 0.5 \), and \( c = 15 \). In the process of circle test, the threshold values
Table 3.3: Parameters of state-of-the-art feature detectors used in this chapter.

<table>
<thead>
<tr>
<th>Procedure</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFD</td>
<td>sigma = 1.5, cornerness threshold=0.2, step=1.44, and number of scales=7</td>
</tr>
<tr>
<td>DoG</td>
<td>peak threshold = 0.008, edge threshold = 10</td>
</tr>
<tr>
<td>Hessian-Affine</td>
<td>peak threshold = 0.0008, edge threshold = 10</td>
</tr>
<tr>
<td>FAST</td>
<td>intensity threshold = 14 (range [0 255])</td>
</tr>
<tr>
<td>Sofka</td>
<td>likelihood ratio threshold = 1.0</td>
</tr>
</tbody>
</table>

were set as: $R_{peak} = 0.01$, $I_{similar} = 0.03(intensity \ range \ [0 \ 1])$, which filtered out high percentage of outliers and kept most branching points. Two circle tests were shown enough to detect most branching points in our datasets and their radii were 7 and 5 pixels, respectively.

The objective of the experiments is to evaluate, in MIS images, how distinctive vessel features detected by RBCT and RBSD are compared with general feature points (corners and blobs). Note that RBCT is a part of RBSD. They are treated as a unit and are compared with the others in all the experiments. Branching points, branching segments, corners and blobs are different types of features and they have different densities in the images. The minimum distances of branching points, branching segments, and general feature points were defined as 11 pixels, 0 pixel, and 1 pixel, respectively. To have a consistent comparison, we applied non-maximum suppression with 11-pixel radius for all the methods, including branching segments, so that the minimum distance between any two feature points was at least 11 pixels. To apply non-maximum suppression, scores indicating the significance of feature points, such as cornerness scores, should be provided. Note that no scores were provided for points from DoG, to run the non-maximum suppression, all points were considered to be equally important by assigning the same cornerness score.

3.4.1 In Vivo Datasets and Ground Truths

Our datasets contained seven in vivo video clips representing different imaging conditions in different surgeries. Sample images were shown in Fig. 3.11. Those videos were taken during colon surgeries of three different patients. They were named in the order from scene1 to scene7. In scene1, scene2, scene5 and scene6, the laparoscope faced towards the lower part of the abdominal wall and was moved horizontally. Because the abdominal wall was insufflated in MIS, those areas of the abdominal wall were approximately flat. Scene3 and scene4 were small flat tissue surfaces in the pelvic area, where the uterus was removed. In
Table 3.4: Summary of the adopted datasets. “Planar” means the scene is planar. “Rotation” represents that the camera motion is rotation only. “Frame num.” indicates how many frames are kept after sampling. “Homo. error” denotes the average errors of the ground truth homography mappings. The unit of Homographic error is pixel.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Source</th>
<th>Homo. type</th>
<th>Length</th>
<th>Frame num.</th>
<th>Resolution</th>
<th>Homo. error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scene1</td>
<td>patient1</td>
<td>planar</td>
<td>3s</td>
<td>50</td>
<td>$1280 \times 720$</td>
<td>1.6 ± 0.8</td>
</tr>
<tr>
<td>Scene2</td>
<td>patient1</td>
<td>planar</td>
<td>3s</td>
<td>50</td>
<td>$1280 \times 720$</td>
<td>1.5 ± 0.7</td>
</tr>
<tr>
<td>Scene3</td>
<td>patient2</td>
<td>planar</td>
<td>5s</td>
<td>53</td>
<td>$1240 \times 800$</td>
<td>2.0 ± 1.0</td>
</tr>
<tr>
<td>Scene4</td>
<td>patient2</td>
<td>planar</td>
<td>3s</td>
<td>45</td>
<td>$1240 \times 800$</td>
<td>2.0 ± 1.0</td>
</tr>
<tr>
<td>Scene5</td>
<td>patient3</td>
<td>planar</td>
<td>8s</td>
<td>22</td>
<td>$720 \times 480$</td>
<td>1.7 ± 0.9</td>
</tr>
<tr>
<td>Scene6</td>
<td>patient3</td>
<td>planar</td>
<td>8s</td>
<td>14</td>
<td>$720 \times 480$</td>
<td>1.6 ± 0.7</td>
</tr>
<tr>
<td>Scene7</td>
<td>[8]</td>
<td>rotation</td>
<td>23s</td>
<td>49</td>
<td>$640 \times 480$</td>
<td>1.9 ± 0.9</td>
</tr>
</tbody>
</table>

those two video clips, the stereoscope rotated and zoomed in and out on top of the scenes. Scene7 was from the public Hamlyn dataset [8].

One important property of a feature point detector is that the same scene points can be detected repeatedly from different viewpoints. Homography mappings have been widely used in the literature to measure this property. To have global homography mappings, it is required that either the scenes are mostly planar or the camera is rotated around its center. The above datasets have been specially chosen in order to have global homography mappings: the first six scenes are mostly planar and in the seventh scene, the camera is mainly rotated around its center.

The ground truth homography mappings for each pair of images in each scene were obtained by manually selecting point correspondences between the image pairs. In each scene, one frame was chosen as the
reference image and the planar area of the scene was selected as the region of interest. Each image is coupled with the reference image to form image pairs. In each pair of images, twenty well-distributed point correspondences between the reference image and the others were manually selected by experienced observers. One example of the manually selected ground truth is shown in Fig. 3.12. The selected correspondences were later used to calculate the ground truth homography mappings following the methods used in [20, 27]. Due to the similarity between successive frames and the large efforts required by the manual point selection, those video clips were uniformly sampled to reduce the manual work. The number of selected frames for each video clip is shown in the fifth column of Table 3.4. The errors of homography mappings for each scene are reported in the last column of Table 3.4. Note that the errors of homography mappings indicate whether it is valid or not to use those mappings for generating ground-truth feature point positions.

Note that the common accuracy of homography mappings used for general feature point detectors is 1.5 pixels [20, 24], which is more accurate than that of our homography mappings as shown in Table 3.4. This is because the constraints in the MIS environment makes it extremely difficult to obtain the exact homography mapping between two MIS images. As shown in the last column of Table 3.4, the homography errors are smaller than 3 pixels. With inaccurate homography mapping, each point in the first image is mapped to a 3-pixel-radius disk in the second image. Since those disks should not overlap in the second image, the minimal distance of any two feature points should be larger than 6 pixels. Our feature points have minimal distance of 11 pixels, which are larger than 6 pixels. Therefore the obtained homography mappings are accurate enough.
3.4.2 Repeatability and Number of Points

The repeatability and total number of detected points are two widely-used measures to evaluate the performance of feature detectors [20, 24]. To this end, repeatability is defined as the portion of points that are detected in images from different viewpoints. Because there is noise in the detected positions of the feature points, two points $x_1, x_2$ are defined to correspond to each other if $|x_2 - H \cdot x_1| < \delta$, where $H$ is the homography mapping and $\delta$ is a predefined threshold. Based on the accuracies of the ground truth homography mappings shown in Table 3.4, $\delta$ is set to be 3.5 pixels for all feature point detectors. Note that this is different from [20, 24, 27], where the $\delta$ is about 1.5 pixels. In this dissertation, the repeatability is defined as:

$$\text{repeatability} = \frac{m}{\min\{n_1, n_2\}},$$

(3.5)

where $m$ is the number of points detected in both images, $n_1$ is the number of points detected in the first image, and $n_2$ is the number of points detected in the second image, whose corresponding points are also visible in the first image. We note that the repeatability measure might be biased against those detectors that detect more points.

The repeatability of different feature detectors on the seven scenes are shown in Fig. 3.13. The overall performance for each method is displayed at the right in Fig. 3.13. As shown, vessel feature detectors outperform the general feature point detectors having the highest average repeatability scores under camera translation and rotation. The major reason is that the general feature points are small in scale and less distinctive under large camera motions. On the other hand, vessels are among the most distinctive structures in MIS images and are resistant to large camera motions. Note that FAST does not perform as well as expected. One possible reason is that comparing with general images, the endoscopic images are more noisy due to the special camera sensor and imaging environment in MIS. As pointed out in [20], FAST is designed to compare as few pixels as possible and is therefore less robust towards the noises in MIS images. In scene7, the camera is mainly rotated around the optical axis and the camera does not change viewpoint. The images in scene7 have small perspective changes and therefore all feature detectors have better repeatability scores.
Figure 3.13: Repeatability scores of different methods in the seven scenes. The overall performance is shown on the right.

Table 3.5: The total number of points in all branching segments detected by RBSD. “S” stands for scene.

<table>
<thead>
<tr>
<th>Scene</th>
<th>S1</th>
<th>S2</th>
<th>S3</th>
<th>S4</th>
<th>S5</th>
<th>S6</th>
<th>S7</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>5661</td>
<td>6296</td>
<td>5749</td>
<td>5430</td>
<td>2204</td>
<td>1098</td>
<td>1177</td>
<td>3945</td>
</tr>
</tbody>
</table>

The number of points detected by the state-of-the-art feature point detectors are shown in Fig. 3.14. The average number of points detected by each method is displayed on the right in Fig. 3.14. Due to the sparsity of branching points, the number of branching points is the fewest among all methods. On the other hand, RBSD is essentially curve segment detector, which detects not only the branching points but also the points along the vessel segment. Therefore, RBSD usually detects much more points than general point detectors. We notice that it is unfair to compare the number of points detected by different types of feature detectors. Therefore, the number of points detected by RBSD is separately given in Table 3.5. The analysis of number of points detected by VBCT/RBCT and RBSD is to verify that the small number of available branching points can be compensated by the large number of pixels detected along the vessel provided by RBSD. In addition, vessel features can be combined with general feature points to extract more rich information from the images.

It is useful to see how many human-recognizable branching points can be detected by the proposed branching point detector. Four representative images are selected from each of the seven scenes and the human-recognizable branching points are manually selected by experienced human subjects. For each im-
Figure 3.14: The number of feature points detected by different feature point detectors in the seven scenes. The overall performance is shown on the right. Note that RBSD is a branching segment detector, which is different from general point detector and usually detects much more points as shown in Table 3.5.

The sets of manually-selected branching points and the automatically detected ones are denoted as $S_1$ and $S_2$, respectively. The coverage is defined as the percent of ground truth branching points that is automatically detected: $|S_1 \cap S_2|/|S_1|$, where $\cap$ represents intersection of two point sets. The average coverage for RBCT is 70%, which means, in average, 70% out of human-recognizable branching points are automatically detected.

3.4.3 Patch Matching Correctness

One target application of branching point detection is endoscope localization and mapping, in which both feature point detection and matching are crucial. In this section, a correlation-matching-based patch search process from the Parallel Tracking And Mapping (PTAM), called “fixed range image search” [109], is adopted, and different feature point detectors are applied and compared. Since the goal here is to compare different feature detectors only, to be fair, the same feature matching method needs to be used for all feature detectors. Note that SIFT descriptor is not used for DoG during the point matching process. In the patch search process, a high matching correctness indicates that the feature point detector has high repeatability and the image patches extracted are distinctive for matching purpose. The patch search procedure takes the first image as the current frame and assumes the second image as the frame that has been saved in the
endoscope localization system. The feature points of the second image are treated as known to the system and are called “map points” [109]. The goal is to identify those map points in the current frame.

The patch search process contains a couple of steps. First, since only the patch search process from PTAM is employed rather than the whole system, the temporal information from previous frames is preserved by defining a search area in the current frame for each map point. For each map point P in the saved frame, its corresponding point Q in the current frame as ground truth can be obtained through homography mapping. The search area of P in the current frame is a disk centered at Q, whose radius is 1/20 of the image width. Second, the feature points in the current frame that are within this fixed range are chosen and referred to as the nearby points. The 21-by-21 local patch of each nearby point is compared with the same size local patch of P. Third, similar to [109], an affine warping, which is obtained from the ground truth homography mapping, is applied to the patch of P to take care of the viewpoint change. After warping, zero-mean sum of squared distance (ZSSD) is calculated for each pair of patches. Finally, a nearby point is considered as a match if its ZSSD value is the minimum and the value is smaller than the pre-defined threshold of 0.02 in this chapter for the normalized intensity range [0 1]. A match Q’ is defined to be correct if |Q’ − Q| < 3.5 pixels. The correctness of the SSD matching is defined to be the ratio of the number of correct matches over the number of total matches.

The patch matching results are given in Fig. 3.15. Vessel features perform better than the general feature again, which further verifies the distinctiveness of vessel features. In patch matching, VBCT and RBCT have higher matching correctness scores than RBSD. This is because the patterns of branching points are more distinctive than the patterns of vessel points (except the endpoints) in branching segments. Note that in scene4, the patch matching correctness of AFD is significantly better than all the others. One reason is that the specular reflections are strong and abundant in scene4. Since our homography mappings are not accurate enough, the movements of the points on the specular reflections can not be captured. Therefore specular reflections are mis-treated as fixed textures and those points on the boundaries of specular reflections are mis-classified as correct matches. One example of patch matching result with RBCT is shown in Fig. 3.16.
Figure 3.15: Patch matching correctness of different methods in the seven scenes. The overall performance is shown on the right.

Figure 3.16: One example of correlation-based patch matching using “fixed range image search” with feature detector RBCT. Matches with error less than 3.5 pixels are classified as correct matches (cyan). The others are classified as incorrect matches (purple). The images are from the seventh scene.
CHAPTER 4
VESSEL-FEATURE-BASED 3D RECONSTRUCTION

In the previous chapter, RBCT and RBSD have been introduced to detect vessel features in MIS images. Based on those vessel features, in this chapter, a novel vessel feature matching method is proposed for a pair of MIS images. For stereo images taken in a MIS environment, the prior information of the abdominal scene, such as the range of depths, can be used to assist the matching of vessel features. After matching, 3D positions of vessels can be recovered provided the stereo cameras are calibrated. Additionally, 3D vessels from different views can be integrated together to obtain a large area 3D reconstruction.

4.1 Traditional Stereo Matching

Before introducing the proposed vessel feature matching method, traditional stereo matching methods are briefly summarized here as background knowledge. The goal of stereo matching is to find pixel correspondences between the left image and the right image. During the search, each pixel is represented by an image patch centered at that pixel. The size of the patch affects the matching results. The sizes of image patches are usually 11-by-11 or 15-by-15. Most stereo matching methods are based on the fact that the image patch in the left image is very similar to the image patch of its corresponding pixel in the right image. Multiple methods to measure the similarity of image patches have been widely used, such as SSD, ZSSD, and NSSD. Theoretically, each pixel in the right image can be a potential match for a pixel in the left image, which is a slow 2D search problem. Fortunately, stereo cameras satisfy the well-known epipolar geometry that each pixel in the left image is mapped to a line in the right image. Therefore, the search can be conducted along the epipolar line, which is a 1D search problem. Specifically, for each pixel in the left image, the pixel in the right image that is on the corresponding epipolar line and achieves the minimal ZSSD value is chosen as the corresponding pixel.
Epipolar geometry maps each pixel in the left image to a line in the right image. To take advantage of this property, the left and right images can be rectified so that their epipolar lines are collinear with the horizontal axis. The above process is normally called “rectification” [110]. The rectification of stereo images requires the knowledge of intrinsic and extrinsic parameters of stereo cameras. Those parameters can be calculated through stereo-camera calibration and they are fixed during the image capturing process. Different rectification methods are available, the method used here is from [181]. Essentially, rectification is trying to obtain a pair of homography mappings, which map the left and right images to the same plane. One example of stereo images before and after rectification is shown in Fig. 4.1.

4.2 Vessel Feature Stereo Matching

For each pixel in the left image, the above stereo matching process requires to compare all pixels on the epipolar line in the right image, which amounts to about 500 image patch comparisons. The direct application of stereo matching method on vessel features is not fast or robust enough. The method proposed here is based on the special structure of vessel features: vessel segments start and end at branching points. We noticed that the local patterns of branching points are more distinctive than the local patterns of other points on the branching segments. Therefore, branching points are much easier to be matched using local image patches. Our vessel feature method contains two steps. First, branching points are matched based
on local image patterns. Noticed that after branching points are successfully matched, their associated branching segments are paired with each other too. Second, those pixels on the branching segment pairs are matched based on the epipolar lines.

It is not desirable to apply the general stereo matching directly on vessel features, because some special issues need to be carefully addressed in order to achieve efficient and robust matching results. First of all, as reported in [167], branching points are large-scale features and their location accuracy is $1.6 \pm 0.7$ pixels [167]. Additionally, branching points are sparsely distributed across the whole image. Typically, about 100 to 200 branching points are available in a single image. Moreover, during the experiments, we observe that the rectification is usually not accurate and the errors are about 0 to 4 pixels. Those errors are mainly from the imperfect synchronization of stereo cameras, the rapid motion of the stereoscope, and image distortions.

In the following sections, a novel vessel-feature matching method will be introduced, which has been designed to overcome the large-location-error problem and the inaccurate image rectification. In addition, the method is able to exploit the sparsity of branching points and the distinctive patterns of branching segments.

### 4.2.1 Branching Point Matching

For each branching point in the left image, we first find those branching points in the right image that are on the corresponding “epipolar band”. Here, “epipolar band” is defined as the rectangle area that is centered at the corresponding epipolar line. The width of the epipolar band used in this chapter is 5 pixels for branching points. One example of epipolar band is illustrated in Fig. 4.2. Only those branching points that are within the epipolar band are chosen as candidate points. To further distinguish them, we compare the branching directions, ridgeness values, and intensity values of their local image patches.

For those candidate branching points in the right image, we first require their branching directions to be similar with the source branching point in the left image. To do so, we treat each vessel segment as a vector that starts from the branching point and ends at the intersection of the circle and the vessel segment. Those vectors are referred to as “vessel vectors” here and are illustrated in Fig. 4.3. Each branching point are associated with three or four vessel vectors. While comparing a pair of branching points, each vessel vector
Figure 4.2: The detected branching points are shown as cyan dots. The epipolar band is highlighted in blue and those candidate branching points are highlighted by red circles.

Figure 4.3: Illustration of vessel vectors. a) Original vesselness image of a branching point. b) A branching point and its associated three vessel vectors highlighted in red.

associated with the first point is paired to its closest vector associated with the second point. The similarity of two branching points is defined as the sum of angles between those vessel vector pairs.

We choose the ZSSD value to measure the intensity similarity between two image patches. Meanwhile, ZSSD is also applied on the ridgeness image to calculate the ridgeness similarity of two image patches. Similar to the branching directions, the ridgeness similarity is used to filter out those candidates that are significantly different. Among the remaining candidates, the one with the smallest ZSSD intensity value is chosen as the match.

To take care of the large-location-error problem mentioned above, during the calculation of ZSSD value, we also try to locally adjust the position of the branching point in the right image so that it matches better to the left image. Let the candidate branching point be $P$, it is possible that one of $P$’s neighboring pixels is actually a better match. The ZSSD value is calculated for $P$ and its 8 neighbors and the position of the
Figure 4.4: Illustration of local adjusting for candidate branching points. The branching point and its associated image patch are shown as red cell and red square respectively. Its eight neighbors are shown in green. The image patch of the top left neighbor is represented by the green square.

Figure 4.5: Illustration of branching point matching results. Dots represent the matched branching points.

pixel with minimal ZSSD value is chosen as the adjusted position for \( P \). This process is illustrated in Fig. 4.4. The results of branching point matching are illustrated in Fig. 4.5.

### 4.2.2 Branching Segment Matching

After branching points are matched, one remaining task is to match each pixel on those branching segments. Since branching segments are associated with branching points, one natural and naive method will first pair branching segments in the left image and the right image. Then, for each pixel on the left branching segment, its corresponding pixel is the intersecting point of its corresponding epipolar line and the right branching segment. The above method is simple and effective for those well-paired branching segments. However, the number of well-paired branching segments is usually not large, because the detection or definition of branching segments in left and right images is independent and those small and blurry vessels might not be treated consistently.
To overcome the above problem, we observe that even though the branching segments might not be consistent in a pair of images, the ridgeness representation [167] of vessels are actually very robust and consistent under different views. Therefore, instead of detecting branching segments separately, our method proposes to detect and match branching segments jointly in both images. The process of our vessel segments matching is referred to as “joint vessel matching”. The matched vessel pixels are consistent in both left image and right image and are therefore called “joint branching segments”.

On a ridgeness image, our method first detects branching segments in the first image following the traditional method in [167]. For each pixel, $P$, in the left branching segment, our method tries to find its corresponding pixel, $P'$, in the right image based on the “ridge mask” [167]. We observe that vessel segments are continuous and their depth changes are usually not large. In addition, essentially, branching points are just points on vessel segments. Therefore instead of searching along the whole epipolar line, the proposed method uses the associated branching point pair, $B$ and $B'$ to narrow down the search range. In other words, the disparity of the associated branching point can be used as an initial estimate for other pixels on the same vessel segment. The estimated position of $P'$ is denoted as $E$. $P'$ can be at the left or right of $E$ depending on its real disparity value. The search range along the epipolar line centered at $E$ is chosen as 10 pixels, based on the employed datasets.

As mentioned earlier, images are not rectified accurately. Therefore, instead of restricting the search to be on the exact epipolar line, the search is performed in an epipolar band. The height of the epipolar band is chosen to be 3 pixels here based on the experiments on the adopted datasets. The epipolar band is illustrated in Fig. 4.6. On the other hand, since $P$ is a ridge pixel, $P'$ has to be a ridge pixel too. Therefore, all ridge pixels within the above epipolar search rectangle are candidates. As shown in Fig. 4.6, only those white pixels that lie in the epipolar band are candidate corresponding points. To distinguish those candidates, an image patch centered at each candidate is compared with the image patch of $P$. The candidate pixel with the smallest ZSSD value is chosen to be the position of $P$. The above vessel pixel matching process is illustrated in Fig. 4.7.

Fig. 4.8 shows one challenging scenario that the ZSSD matching picks a wrong pixel. Since the vessel segments are continuous, the variance of disparity values of pixels on a single vessel segment is expected to be small. To remove obvious outliers, we assume that the distribution of those disparity values is Gaussian.
Figure 4.6: Illustration of vessel pixel search range (epipolar band) in the right image. The predicted position of corresponding point in the right image is shown as yellow dot and the actual matched point is shown as red dot. The pair of cyan dots are corresponding branching points.

Figure 4.7: One example of the vessel-pixel matching results. For each vessel pixel in the left image (red rectangle), its predicted corresponding point is shown as cyan circle and its final matched point is shown as red rectangle. Purple triangles show the left bounds of epipolar search range and green triangles show the right bounds.
with mean $\mu$ and standard variance $\sigma$. According to the well-known “three-sigma rule” [182], those pixels whose disparity values are not within $[\mu - 3\sigma, \mu + 3\sigma]$ are treated as outliers. One example of matched branching segments in a pair of stereo images is shown in Fig. 4.9. Since the stereo cameras have been calibrated, with stereo triangulation [110], the 3D positions of those matched vessel pixels can be computed. One result of recovered 3D vessels is shown in Fig. 4.10.

4.3 Large-area Dense 3D Reconstruction

The method introduced above is able to recover the 3D vessels seen in one frame of stereo images. To achieve large-area 3D reconstruction, 3D vessels recovered from different frames need to be integrated together. First of all, the stereo visual SLAM method [171] is applied to localize the stereoscope. The stereo
cameras have been calibrated and hence the overall scale of the reconstruction is known. The initial pose of the stereoscope is chosen as the reference coordinate. The outputs of the stereo visual SLAM include a number of selected keyframes, whose poses are known. We rely on those keyframes to build the 3D vessel network.

To have large-area 3D reconstruction, one naive method is to create 3D vessels for each keyframe and add them altogether. One obvious problem is that there might be duplicate copies of 3D points when the camera-pose estimation is not perfect. To avoid this problem, for each frame, we first project the visible global 3D vessel points back to that frame and obtain a mask, which indicates whether the scene has been explored before or not. Based on the mask, new 3D vessels are added to the global 3D vessel network. To obtain dense reconstruction, the normal of each point is first estimated based on its neighboring points in the 3D vessel network. Then, Poisson surface reconstruction [7] is applied to estimate the 3D positions of those points that do not belong to the 3D vessel network. After Poisson surface reconstruction, a dense 3D surface model can be obtained, as shown in Fig. 4.13.

4.4 Experiments and Results

In this section, \textit{in vivo} experiments were designed to evaluate the performance of the proposed methods and compare them with other state-of-the-art stereo reconstruction methods. To measure the accuracy of the proposed vessel stereo matching methods, multiple \textit{in vivo} datasets were collected and the ground truth point correspondences between left and right images are manually selected.
4.4.1 In Vivo Datasets

Five in vivo stereoscopic-video datasets were collected for the experiments and their representative images were shown in Fig. 4.11. The surgical scenes of those five datasets were mainly static during the imaging process. The first two datasets were from the Hamlyn Centre [8] and their camera motions mainly contain translation and zooming respectively. The remaining datasets were from different surgical scenes of a patient. The summary of those datasets is available in Table 4.1.

4.4.2 Vessel Feature Stereo Matching Accuracy

To measure the accuracy of the proposed vessel feature stereo matching, experienced human subjects were instructed to manually select corresponding points in the left and right images, which were used as the ground-truth data. Ten ground-truth point correspondences, \( P_i \) \( \rightarrow \) \( Q_i \), were selected for each pair of stereo images. \( P_i \) was selected from the detected vessel feature points and the experienced human subjects were asked to select its corresponding point in the right image, \( Q'_i \). The accuracy of the vessel matching was
Table 4.2: The disparity errors and 3D position errors of the propose vessel feature matching method across the six scenes. Unit of disparity error is pixel and unit of 3D error is millimeter.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Scene1</th>
<th>Scene2</th>
<th>Scene3</th>
<th>Scene4</th>
<th>Scene5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disparity error</td>
<td>1.53</td>
<td>1.26</td>
<td>1.53</td>
<td>1.53</td>
<td>1.4</td>
</tr>
<tr>
<td>3D error</td>
<td>1.95</td>
<td>1.77</td>
<td>2.45</td>
<td>2.72</td>
<td>3.7</td>
</tr>
</tbody>
</table>

Figure 4.12: The recovered 3D vessel network by integrating 3D vessels from different views.

represented by the disparity error and the 3D position error. The disparity errors and 3D position errors of the proposed methods were presented in Table 4.2.

4.4.3 Large-area 3D Reconstruction Results

The large-area dense 3D reconstruction methods introduced in Section 4.3 was applied on the first dataset and the results were presented here. The 3D vessel network obtained by integrating partial 3D vessels from different views was shown in Fig. 4.12. The large-area dense 3D reconstruction result was shown in Fig. 4.13. Note that the dense model was interpolated based on the information from 3D vessel structures only. As shown in Fig. 4.13, the recovered model provides good estimate of the surgical scenes. Note that the texture mapping is not perfect and there are some artifacts. One drawback of the proposed method is that it can not handle the surgical instrument occlusion properly. However, this can be overcame by explicitly detect and track surgical instruments [11, 15].
Figure 4.13: The untextured and textured 3D models that are recovered based on the obtained 3D vessel network. Texture is obtained by selecting representative views re-projecting their images back to the model.
CHAPTER 5
SHADOW-CASTING-BASED 3D RECONSTRUCTION

5.1 Note to Reader

This chapter was published in the IEEE Transactions on Biomedical Engineering (TBME) [2]. Permission to reproduce the work in this dissertation is included in Appendix A.

5.2 Introduction

Feature-based 3D reconstruction methods do not perform well when the scene does not have enough distinguishable texture. To recover a surface with low texture, researchers aimed to actively project patterns on the tissue surfaces. Wu et al. developed imaging systems which projected grid patterns [90] or laser strips [183] to reconstruct the 3D structure of cervix and assist the diagnosis of cervical cancer. Fuchs’s team [184] designed and implemented a miniature projector that projected structured stripe patterns on the abdominal organs. However, the stripe patterns are distractive, and the projector requires special insertion ports, which would not only complicate the surgeries, but also increase the surgery time. Instead of relying on extravagant stripe patterns, we observed that during the surgical process, shadows generated by a surgical tool could provide a weak but structured pattern, which gives a cue to generate distinguishable image features.

It is worth noting that, not surprisingly, both in MIS and computer visualization, researchers have noticed that shadows can significantly improve depth perception [81,82,185]. The study of how to generate optimum shadows in terms of contrast and location of shadow-casting illumination by using a second endoscope was introduced in [81]. A secondary light source was also used in [82] to carefully cast an “invisible shadow,” which was digitally detected and enhanced to provide a depth cue. It should be noted that in order for the cameras to capture the shadows cast by surgical tools, the cameras and light source should be separated. Specifically, the cameras and the single-point light source are separated here.
To the best of our knowledge, the first work to use actively-cast shadows to recover low texture surface was introduced in [186] with a method called “weakly structured light” [186]. However, that method required a calibrated light source and placed two perpendicular planes in the scene. These two requirements are difficult to be satisfied, as the space is very limited in an MIS environment. Here, we remove these two requirements by using stereo cameras and a separated light source. Our method first extracts the shadow borders and interpolates them with epipolar lines to generate disparity maps. Other than being able to achieve dense and accurate reconstruction results, this approach does not require stereo matching, which is much more computing-intense than shadow extraction in the proposed method. Therefore, we expect that this method could be much more efficient than the traditional stereo-matching-based approaches with an optimized implementation. Another advantage of our method is that only stereo cameras and a separated light source are required, since surgical tools are part of a standard MIS setup and surgeons wave surgical tools in front of organs already.

It should be noted that our method only recovers a relatively small area of tissue surface at one time due to the narrow field-of-view in MIS as noted in [56]. To overcome this limitation, as proposed in [60], camera localization using SLAM technique can be integrated to combine small tissue surface patches recovered at different time and obtain a larger recovered 3D tissue surface. The SLAM technique is beyond the scope of this chapter and will not be discussed here. We have evaluated the proposed approach on different phantoms and ex vivo organs and report the accuracies of reconstructed surfaces in comparison with state-of-the-art algorithms.

5.3 Shadow-Scanning-Based 3D Reconstruction

5.3.1 System Overview

We propose to use weakly structured light to recover the dense 3D surfaces of internal organs with stereo cameras. Our method does not require a projector or laser stripe. Instead, similar to [186], we actively cast shadows on the object as a cue to establish semi-dense stereo correspondences. There are four major steps involved: shadow curve extraction, intersection of curves and epipolar lines, field surface interpolation (FSI), and 3D reconstruction.
First, a series of images containing shadows is obtained. The shadow boundaries are extracted and used as shadow curve correspondences between the two corresponding images. Then, epipolar lines are calculated and used to intersect with the shadow curves to efficiently generate precise point correspondences along the curve pair from two images. The accuracy of point correspondences is further improved to sub-pixel accuracy by proper interpolation. Finally, we develop a novel FSI approach to estimate the points that are between two shadow curves by exploiting both the spatial and stereo calibration information to generate dense correspondences between two images, which are used to recover the organ surfaces. The overall scheme of our approach is illustrated in Fig. 5.1.

5.3.2 Extracting Shadow Curves

Since the accuracy of shadow extraction directly affects the accuracy of the recovered surface, it is important to extract the shadow borders in both images as precisely as possible. Our shadow extraction method is based on two assumptions. Firstly, the scene is stationary during the shadow casting process. The static scene is also required in [186], which processes shadows on the temporal domain. Secondly, in order for the surfaces to clearly display the shadow boundaries, we assume the surfaces are locally smooth. It should be noted that this is a relatively weak assumption and most tissue surfaces are locally smooth.
In fact, locally smooth surface is also necessary for structured light based 3D reconstruction methods to project clear patterns. Besides these assumptions, it is worth noting that our method can not extract shadow boundaries from self-shadowed areas, because the intensity changes are very small in those areas. This is an inherent limitation of methods using shadow for 3D reconstruction, such as [186].

Even though the temporal shadow edge has been used to estimate the shadow time for each pixel and has been shown to be very accurate [186], some of its limitations prevent it from being used as it is. For example, that method has difficulty in the self-shadow area. Also, it assumes that the shadow moves forward only, specifically from left to right. This is an unreasonable requirement, because human hands may, at times, be shaky, which makes the shadow move back and forth and causes the algorithm to become unstable. Our method is designed to overcome these problems.

In [187], Agrawal introduced a way to detect depth edges and shadow edges with multi-flash light sources. It has been shown that the method is effective for handling self-shadows. Here, a sequence of images with a moving shadow rather than a fixed shadow is used. Similar to [187], a shadow-free image is generated by taking the maximum of intensity value at every pixel from the sequence of images, which is called the reference image $I_{ref}$. A difference image is defined as:

$$I_{diff} = I_{ref} - I,$$

(5.1)

where $I_{diff}$ reflects the intensity changes of the pixels with and without shadows, which is exactly the main property of the shadow area. One example of a difference image is shown in Fig. 5.2(a). Based on the difference image, adaptive thresholds are set for different rows to discriminate shadow areas from one another. Similar to [186], we calculate maximum and minimum intensity for pixels along each row in each image, and the mean value is used as the threshold for each row. The shadow mask is defined in the following equation:

$$I_{mask} = I_{diff} > \text{threshold}.$$  

(5.2)

The above method naturally marks the shadow area white and the other area black, as shown in Fig. 5.2(b), as the intensity change of the shadowed area is much larger than the other areas. Due to the existence of
In practice, only one shadow scan is enough for 3D reconstruction and the shadow is scanned along one direction. We observe that during the shadow scan, the shadowed area increases gradually and stably on the locally smooth surfaces. Therefore, we propose to accumulate the shadow area and extract the rightmost border as the shadow curve. The binary accumulated shadow image is initialized as a black image. Its formal definition is given iteratively as in the following equation, where the operation is pixel-wise.

\[ I_{acc} = \text{Max}(I_{acc}, I_{mask}). \]  

(5.3)

It is worth noting that only the newly-generated shadow areas are processed and the backward shadows will be ignored. This makes shadow extraction more robust and solves the potential shaky hand problem. One example of an accumulated shadow mask image is shown in Fig. 5.3.

Intuitively, the shadow curve is defined along the vertical direction as the rightmost border of the accumulated region. For each row, the rightmost column of shadow is recorded. Due to the camera speed, resolution and the tool motion, the shadow boundaries in the image might be blurry. For the blurry shadow boundary, the shadow curve is not unique and depends on the threshold value. The shadow curve after thresholding is typically zigzagging due to the discretization nature of image. In addition, the curve is
highly sensitive to image noises. As a result, the shadow curve in the left image does not correspond to the curve in the right. Because the surface is assumed to be locally smooth, the shadow curve is expected to be locally smooth. We apply LWR to smooth the zigzagging curve locally, which makes the left and right curves more consistent and robust towards the image noise. Since the curve might contain multiple segments, the locality is extended to 2D image space so that each segment can be smoothed separately. After LWR, the coordinates of curve pixels reach sub-pixel accuracy. A shadow border before and after LWR is shown in Fig. 5.4.

5.3.3 Intersecting Shadow Curves with Epipolar Lines

After the shadow curves are obtained, for points along a curve in one image, we find their corresponding points in the other image by using the intersection between the shadow curves and epipolar lines. When the images are not rectified, the epipolar lines can be calculated using fundamental matrix from the calibration results. When the images are rectified, the epipolar lines are just the image rows. For each point in the left image, different from the traditional stereo matching method that searches along the epipolar line in the right image, our approach directly calculates the intersections between the shadow curves and epipolar lines, as proposed in [188]. Since the corresponding point in the right image should lie on both the shadow curve and the epipolar line, their intersection point is exactly the corresponding point. This is illustrated
in Fig. 5.5. To simplify the problem, we arrange the two cameras perpendicular to the tool so that the epipolar lines are perpendicular to most of the casted shadows and there will be a unique one intersection. However, there are extreme and rare cases, in which there might be more than one intersection from zigzags by discretization. For those cases, we use the order of the intersections on the epipolar line to define the matching. On the other hand, the shadow curves in the self-shadow areas are marked as invalid and there will be no intersections, which is the reason why no corresponding points can be found in the self-shadow areas. For computation efficiency, even considering the overhead of the shadow boundary extraction, direct calculation of the intersection point should be much more efficient than stereo matching, which simply requires an extra 1D search for each pixel pair.

5.3.4 Field Surface Interpolation

The 3D coordinates of the pixels on the shadow curves can be directly calculated by the traditional triangulation method [110] or from the disparity values. Those shadow curves divide the image into small regions. For pixels inside of those regions, their 3D coordinates can be interpolated by nearby pixels whose depths have been calculated. The interpolation method used here should exploit two constraints: spatial constraint and stereo constraint. The spatial constraint is based on the fact that the pixel is between two shadow curves. The stereo constraint comes from the stereo calibration. Bouget [186] proposed to estimate the shadow time for each pixel, which could not take into consideration of stereo information. In the surface

Figure 5.4: Shadow curves before and after LWR. Blue dotted line represents the original shadow curve. Red solid line represents the curve after LWR.
reconstruction community, an interpolation in 3D space is always used, such as Delaunay triangulation, does not consider the stereo information. Here, we propose a novel FSI method, which incorporates both spatial information and stereo calibration information.

First, consider only a single pair of curves on two images. It is known that a pair of lines, one from each image, can define a mapping between the coordinates of the two images [189]. The difference of a curve and a line here is that each point on a line has the same normal, while different points on a curve may have different normal directions that might intersect with one another, which makes the mapping between two images not bijective. To avoid the intersection of normals, epipolar lines provide a natural alternative, which are guaranteed to have no intersection. Specifically, each point on the curve is attached to a direction that is along the corresponding epipolar line. Now, each pixel on the curve has two coordinates: one is along the epipolar line and the other is along the curve itself. The mapping is illustrated in Fig. 5.6.

Curve $AB$ in the first image corresponds to curve $A'B'$ in the second image. For each point $X$ in the first image, its epipolar line intersects with curve $AB$ at $M$. The corresponding epipolar line in the second image intersects with curve $A'B'$ at $M'$. For $X$, its coordinate along $MX$ is defined as $v$, which is calculated as follows:

$$v = \frac{|MX|}{|AB|},$$

(5.4)
where $|\overline{AB}|$ represents the arc length of curve $AB$. In the second image, the same $v$ is used as the coordinate along $M'X'$ to find $X'$. That is,

$$|M'X'| = v \cdot |\overline{A'B'}|. \quad (5.5)$$

After the above steps, for each point in the first image, a unique point in the second image is found. Also, each point in the second image corresponds to a unique one in the first image. This gives a bijective mapping.

The mapping defined above is only for the special case with one pair of curves. In practice, a large number of pairs of shadow curves are available. For general case, those curves might intersect with each other and divide the image into small regions. Instead of taking a global mapping, a local mapping is defined for each region. Even though a region might be surrounded by multiple curves, for simplicity only the mapping for the region surrounded by two curves is explained, as illustrated in Fig. 5.7. For the region surrounded by curve $AB$ and curve $CD$ in the first image, each point $X$ lies on one epipolar line that intersects with curve $AB$ at $M$, curve $CD$ at $N$. The coordinate of $X$ along segment $MN$ is defined as:

$$v = \frac{|MX|}{|MN|}. \quad (5.6)$$

As in the single pair case, the corresponding point $X'$ is defined as the point on $M'N'$, which has coordinate $v$:

$$|M'X'| = v \cdot |M'N'|. \quad (5.7)$$
It is worth noting that the mapping for all pixels we defined here is consistent with the mapping for pixels on curve boundaries.

### 5.3.5 3D Reconstruction

The derived mapping gives dense correspondences between the two images. The proposed method establishes dense correspondences and depends only on the information of the shadow curves and the epipolar lines. This means that no texture on the object surface is used. The 3D reconstruction can be performed with or without image rectification. Most stereo reconstruction methods perform rectification before stereo matching, which simplifies the 2D correspondence matching into a 1D search task. After rectification, the 3D reconstruction is equivalent as building the disparity map and disparity values have been chosen as the standard for the comparison of different stereo matching algorithms [190]. Following the same framework for the comparison purpose, we also perform rectification and build disparity map. For accuracy and efficiency, we adopt the rectification from [181]. It is worth noting that our method is not limited to rectified images.

### 5.4 Experiments and Results

To take advantage of our approach, it is necessary to have stereo cameras and a separated light source. In a regular MIS, a stereoscope can be used along with a light source through a separate port for the generation of natural shadows, as in [81,191]. It is also possible to use a new shadow telescope [191] with light delivered
through a separate illumination cannula. This approach naturally fits with our novel wireless camera setup, as introduced in [192].

The experiment setup for this chapter is illustrated in Fig. 5.8, which contains a rigid shell with an insufflated abdomen (Chamberlain Group, MA, USA). The cameras we used are micro wireless CCTV cameras ($10\text{mm}$ diameter), with $640 \times 480$ resolution and $30$ fps speed. The cameras were synchronized by a SENSORAY frame grabber. The light source was built from a Cree XLamp XM-L LED with a footprint of $5\text{mm} \times 5\text{mm}$. This single LED can deliver up to 1000 lumens. The abdomen has size of about $39\text{cm} \times 34\text{cm} \times 21\text{cm}$ ($\text{length} \times \text{width} \times \text{height}$). The camera-to-target distance ranges from $11\text{cm}$ to $15\text{cm}$. Each camera’s field of view covers area of size about $10\text{cm} \times 9\text{cm}$ and their overlap field of view has size of about $7\text{cm} \times 9\text{cm}$. The distance between the stereo camera and the single-point light source is about $6\text{cm}$. The surgical tool has diameter of $5\text{mm}$ and length of $34\text{cm}$. To cast the shadow, the surgical tool is inserted in the abdomen and horizontally rotated in front of the light. The perpendicular distance of the tool to the light source is about $7\text{cm}$ and $8\text{cm}$ to the cameras. During the shadow casting process, the distance of the tip of the tool to the object is within $4\text{cm}-7\text{cm}$. With only about $30$ degrees of surgical-tool waving, the casted shadow is able to cover both cameras’ fields of view. Since the waving movement is small, the motion can be achieved in most abdominal MIS surgeries. The video of the shadow casting process and the videos captured by stereo cameras are all available online (http://rpal.cse.usf.edu/project1/index.html).

To better illustrate the setup we used, a diagram is presented in Fig. 5.9. As shown in the figure, a stereo rig and a single-point light source were both mounted using needles [192, 193] on the abdominal wall. However, our setup is flexible, especially the placements of the light and cameras. This flexibility allows the light and cameras to be mounted at different positions for different surgeries.

### 5.4.1 Phantom and \textit{ex vivo} Images

To validate the proposed method, we used the above setup to capture images and tested the algorithm on four phantoms with different types of material: a flat textured paper, an intestine, a lung, and a heart. The flat textured paper was placed on a flat board. The intestine and lung are plastic and the heart is made of silicon. To be clear, the heart phantom was used only as an example for its life-like surface. We do not claim that our current approach can be used in cardiac surgery since the real heart has fast and complex
Figure 5.8: Images of the experiment setup. a) Experiment setup with a stereo camera, a single-point light source, and a surgical tool. b) Illustration of how the shadow is casted by waving the tool in front of the light.

Figure 5.9: Diagram of the experiment setup.
motion [43]. Examples of the original images are shown in Fig. 5.10. It can be seen from the images that they all have the specular reflection problem. Because specular reflection is perspective-dependent, the specular reflection areas of the two cameras are different, which means correspondences based on the specular reflection texture will not be correct. Meanwhile, the texture on the images tends to be uniform and not distinctive enough, which makes it difficult to establish correspondences.

To show the performance of our method on ex vivo images, we tested the algorithm on images taken from a porcine liver. Because the porcine liver was wet, specular reflection and inter-reflection became more severe and caused a larger error in shadow extraction. The numerical results of both the phantoms and ex vivo images are available in Section 5.4.4.

5.4.2 Disparity Maps

To illustrate the benefits of using shadow information for 3D reconstruction for MIS, we have compared our approach with traditional stereo algorithms, in which stereo cameras were calibrated and images are rectified [194]. The rectification we adopted here is from [181] due to its accuracy and efficiency. After rectification, the focal length of the two cameras was 798.40 and the baseline was 11.16 mm. The valid disparity range for our setup is [60 130], which is used in stereo matching algorithms as a priori. The
The proposed method is compared with three popular stereo matching algorithms. The first one is considered to be the state-of-the-art stereo matching algorithm applied in MIS [56], which is referred as seed propagation (SP). The second one is referred to as believe propagation (BP) [195]. Following the notation in [56], the last one is abbreviated as RT [196].

The disparity maps obtained by different algorithms are shown in Fig. 5.11. The first column (Fig. 5.11a) shows the rectified left images with a cast shadow. The second column (Fig. 5.11b) illustrates the results from our proposed approach. The rest of the figure gives the results from SP (Fig. 5.11c), BP (Fig. 5.11d), and RT (Fig. 5.11e) stereo matching algorithms. Those disparity images are all color coded, by which white (255 intensity value) corresponds to the maximum disparity value (130). Since the proposed method relies on shadow information rather than texture, to make a relatively fair comparison, the shadow is kept during the stereo matching procedure. In all the experiments, even though the surfaces do contain texture, the texture is not discriminative enough to establish correspondences. As shown in Fig. 5.11, all three stereo matching algorithms have difficulty in propagating the correspondences. This is most likely because the low texture surface gives only very sparse feature correspondences, which are not enough to propagate a dense and accurate disparity map. Overall, it is clear that with the addition of shadows, the proposed method significantly outperforms the other three stereo algorithms.

5.4.3 3D Reconstruction Results

The above disparity maps are further processed to get 3D reconstruction results. For the proposed method, the recovered 3D surfaces with and without texture are given in Fig. 5.12. Those surfaces are displayed using MeshLab, and the snapshots are shown in the figure. Those images in Fig. 5.12 show that the proposed method is able to recover the 3D surface to a certain degree. For example, in the intestine surface, the deep slopes are nicely recovered. However, it should be noted that errors do occur. For instance, in both the plane and heart examples, the specular reflections cause holes. In the ex vivo experiments, the markers themselves have a certain size, thus making the shadow extraction inaccurate when the shadow goes across the markers. In addition, stripes can also be observed in Fig. 5.12 and they can be reduced if more shadow images are processed. As comparison, 3D reconstruction results of other three methods are also provided in Fig. 5.13. We recommend to zoom in the figure to have a better understanding of the
Figure 5.11: Comparison of disparity maps of different methods. a) Rectified left image with shadow and detected border. b) Disparity maps derived by our proposed method. c) Disparity maps by SP [56]. d) Disparity maps by BP [195]. e) Disparity maps by RT [196].
reconstruction results. Comparing Fig. 5.12 with Fig. 5.13, it is clear that the proposed method has great advantage in both accuracy and coverage.

5.4.4 Numerical Comparison

To get the ground truth point correspondences for quantitative error analysis, markers are put on the surface and later selected manually from the images, as shown in Fig. 5.10. Those marker points, \((P_l, P_r)\), selected from left and right images, serve as ground truth point correspondences. For each point \(P_l\) in the left image, \(P'_r\) is the calculated corresponding point in the right image. One example of \(P_r\) and \(P'_r\) in the right image is shown in Fig. 5.14. The 2D Euclidean distance between \(P_r\) and \(P'_r\) is named as disparity error and used to reflect the accuracy of disparity maps. In addition, the 3D positions of \((P_l, P_r)\) and \((P_l, P'_r)\) are also computed using triangulation, and the distance between them, named as 3D position error, serves as a measure for the accuracy of the recovered surface, even though the calibration error is inherited in the calculation of the 3D positions. Both disparity error and 3D position error are calculated to compare among the four different methods. Since the disparity maps are sparse and some markers might have no values, for a fair comparison, the nearest valid disparity values (within range [60 130]) are chosen to represent those markers.

The disparity error results of the four methods over the five experiments are given in Table 5.1. The 3D position error comparison results are displayed in Table 5.2. First of all, compare BP with SP and RT, it appears that BP has very low disparity error and 3D position error. In fact, based on our observation, this is most likely because BP method explicitly detects and matches some markers on the image. Both SP and RT do not show such obvious operations. However, even stereo matching methods might get better results because of markers, as show in those tables, the proposed method still significantly outperforms the others. For instance, in the phantom experiments, the disparity errors of the proposed method are within 1.04 pixel, and the 3D position errors of the proposed method are within 0.7mm. In addition, it is worth to note that in \textit{ex vivo} experiments, both the disparity error and 3D position error are larger than the phantom ones in all four methods. This is probably caused by the wet surface of the porcine liver, which causes more specular reflections. However, even with the higher complexity in \textit{ex vivo} images, the 3D position error of our method is still within 1.2mm. On the other hand, in each disparity map, the percentage of pixels whose values are
Figure 5.12: 3D reconstruction results of flat plane, intestine phantom, lung phantom, heart phantom and porcine liver. The left column shows the recovered 3D model without texture mapping. The right column shows 3D model with texture mapping.
Figure 5.13: 3D reconstruction results of different methods over five experiments. a) SP method, b) BP method and c) RT method. The first row is experiment on a plane. The second row corresponds to experiment on an intestine phantom. The others are lung, heart and liver respectively.
Table 5.1: Disparity error of the four methods over the five experiments. All the number is in pixel.

<table>
<thead>
<tr>
<th>Experiments</th>
<th>Our method</th>
<th>SP</th>
<th>BP</th>
<th>RT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plane</td>
<td>0.5387</td>
<td>1.0796</td>
<td>4.0737</td>
<td>31.3568</td>
</tr>
<tr>
<td>Intestine</td>
<td>0.5321</td>
<td>1.7709</td>
<td>7.9680</td>
<td>15.1386</td>
</tr>
<tr>
<td>Lung</td>
<td>0.9546</td>
<td>2.2666</td>
<td>3.0994</td>
<td>20.2294</td>
</tr>
<tr>
<td>Heart</td>
<td>1.0332</td>
<td>9.8639</td>
<td>6.0335</td>
<td>22.3429</td>
</tr>
<tr>
<td>Porcine liver</td>
<td>1.3675</td>
<td>28.5919</td>
<td>4.5216</td>
<td>15.4770</td>
</tr>
</tbody>
</table>

in the range [60 130] is recorded in Table 5.3. The numerical comparison of those three tables concludes that our method performs significantly better than the other three both in accuracy and coverage. Next to our method is the SP method, which is followed by BP method. RT method ranks last, probably because it sacrifices the accuracy to achieve real time performance.

5.4.5 Robustness Analysis

Since the input of the proposed method comes from shadow curves and calibrated stereo cameras, the accuracy of the final disparity results depends on the precision of shadow extraction and stereo calibration.

Table 5.2: 3D position error of the four methods over the five experiments. All the number is in mm.

<table>
<thead>
<tr>
<th>Experiments</th>
<th>Our method</th>
<th>SP</th>
<th>BP</th>
<th>RT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plane</td>
<td>0.3823</td>
<td>1.7286</td>
<td>5.1175</td>
<td>37.7707</td>
</tr>
<tr>
<td>Intestine</td>
<td>0.5923</td>
<td>2.7590</td>
<td>10.7844</td>
<td>24.1257</td>
</tr>
<tr>
<td>Lung</td>
<td>0.6553</td>
<td>2.2382</td>
<td>2.5720</td>
<td>27.8008</td>
</tr>
<tr>
<td>Heart</td>
<td>0.5834</td>
<td>7.8710</td>
<td>4.4799</td>
<td>21.5661</td>
</tr>
<tr>
<td>Porcine liver</td>
<td>1.1406</td>
<td>22.1534</td>
<td>4.2925</td>
<td>11.6948</td>
</tr>
</tbody>
</table>
Table 5.3: The percentage of pixels in image with valid disparity value.

<table>
<thead>
<tr>
<th>Experiments</th>
<th>Our method</th>
<th>SP</th>
<th>BP</th>
<th>RT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plane</td>
<td>53.87%</td>
<td>30.06%</td>
<td>61.67%</td>
<td>23.34%</td>
</tr>
<tr>
<td>Intestine</td>
<td>51.97%</td>
<td>34.00%</td>
<td>60.58%</td>
<td>32.16%</td>
</tr>
<tr>
<td>Lung</td>
<td>55.16%</td>
<td>32.58%</td>
<td>65.09%</td>
<td>45.50%</td>
</tr>
<tr>
<td>Heart</td>
<td>46.25%</td>
<td>25.25%</td>
<td>53.70%</td>
<td>33.35%</td>
</tr>
<tr>
<td>Porcine liver</td>
<td>55.80%</td>
<td>09.74%</td>
<td>46.66%</td>
<td>40.90%</td>
</tr>
</tbody>
</table>

There are a couple of contributing factors to the shadow border extraction error. The first one is the intensity contrast, that is, a dark shadow and a light background can give better extraction results. Second, the sharpness of the shadow edge directly affects the accuracy of the shadow border. In addition, the synchronization between the two cameras is also an important issue, because only properly-synchronized cameras can guarantee that the left and right shadows correspond to each other. Finally, strong specular reflection can lighten the shadowed area and may disturb the shadow extraction. On the other hand, inaccurate stereo calibration causes error in epipolar lines estimation. Because epipolar lines are used to intersect with the shadow border and establish the point correspondences, the error of stereo calibration will introduce the horizontal error in point correspondences.

Following the same notation in previous section, we denote $(d_x, d_y) = P'_r - P_r$. For the proposed method, $d_x$ is mainly caused by the shadow extraction error and $d_y$ is the result of the epipolar line calculation error. To analyze how robust the final results are towards the accuracy of shadow border extraction and epipolar lines, uniform pixel noises in different ranges are added to the calculated marker coordinates in the right image, and the corresponding shadow extraction error and 3D position error are recorded. Uniform pixel noises in the range of $[-3, 3]$ are added for shadow border extraction, and uniform noises in the range of $[-10, 10]$ are added for epipolar line computation. In Fig. 5.15, the 3D position error as a function of the shadow extraction error is displayed, revealing that 3D reconstruction accuracy is linearly proportional to the accuracy of the shadow extraction. This means that the proposed method is robust without significant error propagation towards the accuracy of shadow border extraction. In Fig. 5.15, it shows that even when the epipolar line error is as large as 10 pixels, the 3D position error is still within $1.5\, mm$. The comparison of the two figures in Fig. 5.15 empirically indicates that the 3D reconstruction is more sensitive towards the accuracy of the shadow border extraction than the accuracy of epipolar lines.
Figure 5.15: Robustness analysis. The left figure shows the 3D position error v.s. shadow extraction error. The right figure shows 3D position error v.s. epipolar line error.
CHAPTER 6

VISUAL SLAM IN A DEFORMING ENVIRONMENT

6.1 Note to Reader

This chapter was published in the Lecture Notes in Computer Science by Springer [171]. Permission to reproduce the work in this dissertation is included in Appendix A.

6.2 Introduction

Real-time on-site simultaneous endoscope localization and 3D structure recovery are important tasks for Minimally Invasive Surgery (MIS). First of all, based on the endoscope localization result, the recovered tissue structures from past and current endoscope locations can be merged together to obtain a larger field of view. Additionally, most current registration methods of intra-operative and pre-operative data in abdominal MIS are global and static and therefore the registration becomes inaccurate when tissue organs shift and deform. Endoscope localization and tissue structure recovery based on the intra-operative video can be used to refine the global registration and reduce the errors from organ movements. To achieve the benefits mentioned above, general tissue deformations, which can be caused by tool interaction as well as patients’ respiration and heartbeats, should be carefully taken care of during the endoscope localization and structure recovery procedure.

Many existing endoscope localization methods in different anatomical settings typically assume a static scene. For example, in the monocular Simultaneous Tracking and Mapping (SLAM) system introduced in [95, 119] for sinus surgery, the endoscope’s pose was estimated by two successive frames based on the static assumption. Mountney et al. [128] applied and extended the monocular Extended Kalman Filter SLAM (EKF-SLAM) framework from Davison [6] to stereoscope in MIS environment. Combining the stereo EKF-SLAM framework and Stoyanov et al.’s semi-dense reconstruction [56], Totz et al. [60] presented a
method to recover a large and dense abdominal tissue surface. For periodic liver deformation, Mountney and Yang [155] proposed to learn the parameters of the periodic motion first and then use it to improve the SLAM estimation. For non-periodic deformation, Giannarou and Yang [197] presented a work to detect deforming points using monocular Structure From Motion (SFM) framework, whose speed is unclear and doesn’t seem to be fast.

6.2.1 PTAM

PTAM was originally designed for monocular cameras. Instead of updating the 3D map in each frame as EKF-SLAM does, tracking and mapping have been separated into two parallel threads and the mapping thread has much lower priority. The two threads run in parallel and communicate with each other through the 3D map. The tracking thread estimates the camera pose based on the 3D map generated from mapping thread. The mapping thread receives new well-tracked frames from the tracking thread and updates the 3D map accordingly. There are three major steps in tracking. First, a decaying velocity motion model is used to predict the current camera’s pose. Second, 3D points are reprojected on the current frame and a fixed-range search is applied to find the reprojected points. Third, the identified 3D points and their stereo measurements are used for pose estimates. On the other hand, mapping also requires three important steps. First, user is required to translate the camera and the obtained “stereo” pair and the tracked features are used for map initialization. Second, when exploring a new area, a frame will be saved in the map for 3D reconstruction purpose. The saved frame is called keyframe. Third, local and full bundle adjustment [110], which is a standard routine to simultaneously optimize the 3D points and camera poses, are run to refine the map. The major advantage of PTAM is its ability to recover a large number of 3D points. However conventional PTAM is difficult to be directly applied in MIS setting because a static environment is assumed.

6.3 Stereoscope PTAM

In order to accurately track the scope in MIS and reconstruct the deforming surgery scene in real-time, we adopt PTAM to utilize the stereo cameras on a stereoscope and develop both stereo tracking with deforming point detection and stereo mapping in our new stereoscope PTAM. First of all, MIS images have abundant specular reflections, whose boundaries can easily be picked up as feature points, which would
Figure 6.1: Outline of our stereoscope PTAM. There are two parallel threads: stereo tracking and stereo mapping. Stereo tracking has two modes: static tracking and deforming tracking.

Stereo tracking cause large error to the pose estimation due to their view dependent property. Before further processing, specular reflections should be detected and removed. For efficiency, bright pixels with intensities larger than 180 (0 for black and 255 for white as in standard grayscale image) are simply detected as specular reflection as well as their 5-by-5 neighbors.

For stereo tracking, we design two modes: static tracking mode and deforming tracking mode. The static tracking mode is very similar as the conventional PTAM except that the 3D points are reprojected and found in both left and right images from stereoscope. In the deforming tracking mode, the system detects deforming points and only rigid points are used for pose estimation. Our tracking system does not detect deforming points in each frame due to two reasons. The first is for efficiency to get nearly real time performance. Second, not all tissue organs in abdomen have deformations all the time. For stereo mapping, the original bundle adjustment is extended for stereo images. The outline of our system is shown in Figure 6.1 and the components are detailed in the following sections respectively.

6.3.1 Deforming Point Detection

As static tracking mode is similar to PTAM, we only describe deforming tracking mode in detail. The deforming tracking mode is triggered based on two conditions: 1) whether the tracking quality is poor; and 2) whether the speed of the camera is slow, which is designed to allow the stereoscope to explore a
deforming area. The measure of poor tracking quality in [109] is adopted here. The camera pose update is a 6D vector, when the L2 norm of this vector is smaller than 0.1, the camera motion is considered as slow.

When the deforming tracking mode is triggered, both the tissue deformation and the stereoscope movement can contribute to the pixel displacement. To detect deforming points, each stored 3D map points that are visible in the current camera’s field of view is projected on the image and a square area with width of 50 pixels centered at the projected position is searched. The set of 3D map points that are found in both left and right images is called the first point set, which may contain deforming, rigid and mismatched points. To remove the mismatched points from the first point set, each pair of points found in the left and right images is further required to be a stereo correspondence, namely, their corresponding patches should be similar and their sum of square distance (SSD) should be small. After the above removal, the rest of the mismatched points, if any, will be treated as deforming ones. On the other hand, with calibrated stereo cameras, triangulation is applied to calculate the 3D coordinates of points in the first set which leads to a second set of 3D points represented in the left camera’s coordinate. The first set of 3D map points is denoted as \( \{ p_i \}_{i=1}^n \) and the second one as \( \{ p'_i \}_{i=1}^n \).

From these two point sets, we can estimate the stereoscope’s pose and identify the rigid points based on the fact that only rigid points will follow a global Euclidean transformation while deforming or mismatched points do not. We apply RANSAC to select rigid points as inliers. During each RANSAC iteration, 3 pairs of corresponding 3D points are randomly selected to calculate the Euclidean transformation, which minimizes the following objective function:

\[
\min \sum_{i=1}^n \| p_i - (R * p'_i + T) \|, \tag{6.1}
\]

where \( R \) is a rotation matrix and \( T \) denotes translation. A closed-form solution of Equation (6.1) is obtained using Horn’s absolute orientation algorithm [198]. With the derived transformation, we can identify rigid points as inliers and deforming points as outliers. The threshold of the residual error using in the RANSAC iteration is 2mm here.

The above classification of rigid and deforming points is based on a single frame. However, as claimed in [145], the most significant property of deforming points is that they will continuously deform and hence their 3D registration errors in Equation (6.1) will always be large. Therefore, the 3D registration error for each point is accumulated and the average registration error is used to classify whether a point is deforming.
or not. The average registration error contains temporal information and is therefore very robust to detect deforming points. It is worth noting that once the deforming points are detected, they will not be used in the following non-linear pose refinement and stereo bundle adjustment procedure. Therefore, the error caused by tissue deformation can be significantly reduced.

### 6.3.2 Non-linear Stereo Pose Refinement

An initial estimate of the stereoscope pose can be obtained from previous information for each tracking mode: motion updated pose from previous frame in static tracking mode and pose from RANSAC in deforming tracking mode. The initial pose estimation is further refined by minimizing the reprojection error, which is a non-linear least square optimization problem. Different parameterizations are available for this problem and we follow the $SE(3)$ parameterization used in [109], which is claimed to give better results than others [199]. Since we assume that the stereo cameras are well synchronized, the extrinsic transformation between the left camera and right camera should be fixed during the pose estimation procedure. Therefore, the stereo pose update optimization problem is given in Equation (6.2) and 6.3. The calculation of Jacobian matrices of Equation (6.3) can be found in [199].

\[
\mu' = \arg\min_{\mu} \sum_{i=1}^{n} \rho(||e_i||_2) \tag{6.2}
\]

where $e_i = [e_{1i}^T, e_{2i}^T]^T$ is the reprojection error from both cameras.

\[
\begin{align*}
\begin{cases}
\hat{e}_{1i} = \begin{pmatrix} \hat{u}_{1i} \\ \hat{v}_{1i} \end{pmatrix} - ProjCam_1(exp(\mu) \oplus E_{LW} \oplus p_j) \\
\hat{e}_{2i} = \begin{pmatrix} \hat{u}_{2i} \\ \hat{v}_{2i} \end{pmatrix} - ProjCam_2(exp(\mu) \oplus E_{RL} \oplus E_{LW} \oplus p_j)
\end{cases}
\end{align*} \tag{6.3}
\]

in which subscripts 1 and 2 represent left and right camera respectively; $(u, v)^T$ represents the measured 2D feature point location; $\rho(\cdot)$ is Tukey biweight objective function; $\mu \in SE(3)$ denotes a 6D vector parameterization of Euclidean transformation; and $exp(\cdot)$ is an exponential map, which maps a 6D vector to an element in $SE(3)$. Further, $E_{LW}, E_{RL} \in SE(3)$ and $\oplus$ are the pose-pose and pose-point compositions.
Subscript $LW$ denotes the transformation from the world coordinate to the left camera and $RL$ for the transformation from the left camera to the right camera. $\text{ProjCam}(\cdot)$ represents the camera perspective projection. Due to real-time performance requirement, only multiple iterations of re-weighted least square are applied to refine the pose.

### 6.3.3 Map Initialization

With calibrated stereoscope, no user cooperation is required for map initialization and the coordinates of 3D points are in $\text{mm}$. Since the created 3D points are used for tracking purpose, we do not perform classic stereo matching [56], which is likely to generate more 3D points but not necessarily good for tracking. Instead, similar as PTAM, we detect FAST feature points in both images and keep the ones that are easy to track. To speed up stereo matching procedure, prior information of the tissue environment is exploited. Since our target application is abdominal MIS, we accordingly set the minimum and maximum distance of stereoscope to the target as $20\text{mm}$ and $400\text{mm}$ respectively when performing epipolar search.

### 6.3.4 Stereo Bundle Adjustment

To incorporate calibrated external information to the local and full bundle adjustment, we minimize the following objective function:

$$\{\{\mu'\}, \{p'\}\} = \text{argmin}_{\{\mu\}, \{p\}} \sum_{i,j} \rho(||e_{ji}||^2)$$

(6.4)

where $e_{j,i} = [e_{1ji}^T, e_{2ji}^T]^T$ is the reprojection error of the j-th point in the i-th keyframe. $\mu$ represents the poses of keyframes and $p$ 3D map points.

### 6.4 Experimental and Results

#### 6.4.1 Tracking Accuracy

To show the tracking performance of our method, we quantitatively analyze the camera tracking accuracy using a non-deforming intestine phantom. The ground truth is obtained from OptiTrack system (NaturalPoint Inc.), whose tracking accuracy is within $0.01\text{mm}$ and tracking speed is $100\text{fps}$. In this experiment,
Figure 6.2: Phantom experiment setup. a) The intestine phantom. b) The stereo cameras attached with four optical markers. c) One example of detected deforming points shown as white.

Table 6.1: Mean error and variance of the tracking results.

<table>
<thead>
<tr>
<th></th>
<th>3D Trajectory</th>
<th>X Axis</th>
<th>Y Axis</th>
<th>Z Axis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean error (mm)</td>
<td>1.29</td>
<td>1.52</td>
<td>0.15</td>
<td>0.66</td>
</tr>
<tr>
<td>Variance (mm²)</td>
<td>0.66</td>
<td>8.37</td>
<td>0.13</td>
<td>0.65</td>
</tr>
</tbody>
</table>

the intestine phantom is shown in Figure 6.2 a) with dimension 19cm * 14cm * 6cm. The stereo cameras used in this phantom experiment has been introduced in [192, 200]. Four optical trackers are attached on the stereo system’s back, as shown in Figure 6.2 b). The stereo vision system is designed for evaluation purpose only, and therefore the system is not a miniature one. The stereo cameras are first placed at a distance of about 11cm to the phantom. They are then manually moved at a speed of about 10mm/s and held still at four locations.

It should be noted that our tracking system selects the first frame of the left camera as the world coordinate and the OptiTrack system has a different world coordinate. To enable the comparison of trajectories from these two different coordinate systems, the Euclidean transformation between them need to be calculated. To achieve this, the stereo cameras are held still for a couple of seconds at four different locations during the movement, which results in four line segments in the trajectories in Figure 6.3. These four point pairs in the two trajectories can be used to calculate the Euclidean transformation using Horn’s absolute orientation algorithm. The two 3D trajectories are represented in OptiTrack’s coordinate system and shown in Figure 6.3 a). The tracking accuracies in each dimension are also displayed in Figure 6.3 b), c), and d). The numerical tracking accuracy is available in Table 6.1. Notice that the tracking error along the X axis is much larger than the others. One contributing factor is that the stereo system’s viewing direction is mostly parallel with the X axis.
Figure 6.3: Comparison of stereo tracking accuracy with ground truth. Tracking results of our method are shown in solid red and ground truths are shown in dotted blue. The recovered 3D trajectory and ground truth are shown in a) and their projection in X b), Y c) and Z axis d).
6.4.2 Evaluation with *In Vivo* Data

We further reviewed the tracking results of our system on three *in vivo* videos. The first two stereo videos were from Hamlyn Center [8]: Dataset1, Dataset6. The last video was recorded while the surgeon was performing a colon surgery. Our system was able to run at speed of $15 \sim 20 \text{fps}$ with a desktop computer (3.2GHz×4 cores, 3.7GB memory). Since no ground truth of the trajectories of those videos is available, we only show the 3D trajectories from our methods in Figure 6.4. The number of recovered 3D points for the three datasets are about 600, 900, and 1600 respectively. As an example, the detected deforming points are shown as white in Figure 6.2 a). The typical feature points detected in a single frame among different videos are shown in Figure 6.5, where a virtual white grid is mounted at a fixed position in the real scene to indicate the tracking accuracy. In the colon surgery video, to demonstrate the stereoscope tracking accuracy, a virtual bladder was manually registered in the first frame and was successfully tracked and augmented throughout the whole video. Four frames are randomly picked to show the augmented results in Figure 6.6.
Figure 6.4: 3D trajectories of the stereoscope tracked by our method over three videos. a) Dataset1, b) Dataset6 and c) Colon surgery.
Figure 6.5: Typical 3D feature points detected in one frame (zoom in for detail). Each frame has four pyramid levels and the color of each point indicates at which level it is detected [109]. Each column shows two random frames from one experiment. a) Intestine phantom, b) Dataset1, c) Dataset6 and d) Colon surgery.

Figure 6.6: A bladder model reconstructed from Computerized Tomography (CT) was augmented in the colon surgery video.
CHAPTER 7

PERIPHERY AUGMENTATION SYSTEM AND EVALUATION

7.1 Periphery Augmentation System Design

The large-area dense 3D reconstruction based on vessel feature and the laparoscope localization in MIS environment have been introduced in Chapter 4 and Chapter 6 respectively. Based on those techniques, in this chapter, a new in vivo video visualization system is designed to enlarge the field of view of the original MIS video by augmenting the peripheral areas according to the obtained dense 3D model. The system is called “periphery augmentation system” here.

The outline of the periphery augmentation system is shown in Fig. 7.1. The key steps are summarized as below. At the beginning, to collect data of target areas, the user needs to move the laparoscope and explore the scenes. Simultaneously, Visual SLAM technique introduced in the previous chapter is run on the laparoscopic video to localize the laparoscope. Based on the localization results, a 3D vessel network is recovered and a textured large-area dense reconstruction (model) is estimated. While using the periphery augmentation system, the laparoscope is localized with respect to the recovered textured 3D model and the model is projected back to the camera to generate a virtual view. The virtual image has a larger field of view and contains peripheral information. The central area of the virtual image is from the live video, which provides real-time information of the scenes.

Note that the large-area reconstruction is based on the historic data and might not exactly match with the current surgical scenes if some parts have been changed. To overcome this problem, the dynamic view expansion system in [201] was designed to fade historic data to gray scale to help surgeons be better aware the distinction of live video and out-of-date data. Similar to [201], our visualization system also keeps the color live video unchanged in the center and augments the peripheral areas as gray using recovered 3D reconstruction results. The contrast of color and gray areas gives a clear distinction between the historic
information and live data. One example of the large field-of-view image with augmented peripheral areas is shown in Fig. 7.2 b).

With the recovered 3D structure and the localization results, the periphery augmentation system has multiple advantages compared to the existing surgical platform. First of all, it provides a larger field-of-view and contains more information for the surrounding areas. Additionally, the system has the potential of providing an estimation of the depth information of the region of interest to the surgeons. Moreover, the system can display the endoscope’s pose with respect to the recovered model to better assist surgeons understand the orientation of the endoscope in case they are lost.

7.2 In Vivo Evaluation

The periphery augmentation system has multiple advantages, this section focuses on the evaluation of the benefits of having a larger field-of-view with peripheral information. With extra information of surrounding areas from the larger field-of-view, the periphery augmentation system is expected to improve the surgeons’
sense of orientation and awareness of the surgical environment. In the remainder of this section, numeric measures are proposed to represent the environment awareness and experiments are designed to compute those measures.

### 7.2.1 Interface and Point Estimation Procedure

The *in vivo* dataset is chosen from [8], where the stereoscopic laparoscope is moved around to explore a static scene. The vessel-feature-based method introduced in Chapter 4 is applied to obtain a dense 3D model, as shown in Fig. 4.13. Laparoscope localization method from Chapter 6 is applied to obtain and record the pose of laparoscope for each frame. The 3D model is loaded in Unity [202] and a virtual camera is placed at the recorded poses to generate the periphery augmented images.

The user interface of the point estimation test is through mouse clicking and the whole process lasts about 10 minutes. At the beginning, subjects are randomly assigned to start either with (Fig. 7.2 a and b) or without (Fig. 7.2 c and d) periphery information. As shown in Fig. 7.2 a) and c), at some point, the system is paused for 5 seconds and a scene point is highlighted so that subjects can get familiar with the scene structure. Then the system resumes and after around 120 frames, the previous point has been outside the field of view of the original video as shown in Fig. 7.2 b) and d). At this moment, subjects are asked to estimate where the previous highlighted point would be in the current image by clicking the mouse. The above process repeats three times to sequentially test three points.

The above point tests are performed either with or without the periphery information. To compare the difference of with and without the periphery information, the whole process is repeated again for the one that has not been conducted, which completes one round of point tests. To collect as much information as possible, the point tests are performed for five rounds with different sets of three points. One example of the ground-truth points and user-estimated points for the first round is shown in Fig. 7.3.

17 subjects in total have participated the test. Among them, 4 are surgeons whose specialties are general surgeries and are very familiar with the laparoscopic videos. The other subjects are non-surgeons with no knowledge of laparoscopic videos. Those subjects are divided into two groups. The first group starts the point tests with periphery information and the second group otherwise. At the end of the task, subjects are
Figure 7.2: Illustration of the point estimation for the periphery augmentation system (top) and the original MIS video (bottom). The point is first highlighted as shown in left and then the subjects are asked to estimate where that point is in the current image as shown in the right.

Figure 7.3: One example of ground-truth points (green) and user-estimated points (blue) on the periphery augmentation image.
asked to provide a score from -5 to 5. Positive numbers indicate that the background information from the periphery augmentation system is useful and negative numbers indicate the opposite.

7.2.2 Environment Awareness Measures

Environment awareness is one kind of human sense and is difficult to be quantified. Here, the environment awareness refers to the ability of understanding and memorizing the scene by observing videos only. The numeric measures of environment awareness adopted here are based on how accurate users can estimate, in the current field of view, the locations of those scene structures that they have seen before. Denote a pre-selected scene point as \( P_1 \) and its corresponding user-estimated point as \( P_2 \). Note that when users select a pixel on the screen, a 3D ray starting from the camera center is generated and intersects with the 3D model at a 3D point. Denote the rays from the camera center to \( P_1 \) and \( P_2 \) as \( Ray_1 \) and \( Ray_2 \). Two numeric measures are chosen to represent the awareness of the environment. The first one is defined as the angle between \( Ray_1 \) and \( Ray_2 \). The pixel distance between \( P_1 \) and \( P_2 \) on the image that users estimate the points is chosen as the second measure.

7.2.3 Results and Analysis

In each round of the point estimation test, three points are selected by each subject. The mean error, defined in Section 7.2.2, of those three points is computed to represent the error of one subject in one round. Five rounds of tests have been conducted for each subject on the periphery augmented video and the original MIS video respectively. Angle errors and pixel errors for each subject and each round are shown in Fig. 7.4. The histograms of angle errors pixel errors are available in Fig. 7.5. The overall means and standard deviations with and without periphery information are shown in Table 7.1 and Table 7.2.

First of all, the left and right images in Fig. 7.4 have similar distributions of markers, and the left and right images in Fig. 7.5 look alike. Those similarities validate that the results obtained using either angle errors or pixel errors are consistent and these two kinds of errors are equivalent to each other in representing point estimation accuracy. Additionally, in Fig. 7.4, most green stars are underneath the red circles. In Fig. 7.5, the green bars are generally in the left side of the red bars. They both verify that errors from the periphery augmented system are much smaller than errors from the original MIS video. The numeric
Table 7.1: The overall mean and standard deviation of the angle errors and its Student’s t-test results. “t-value” represents the corresponding value in the normalized Student’s distribution. “Pr” is short for probability.

<table>
<thead>
<tr>
<th></th>
<th>Mean (degree)</th>
<th>Sample size</th>
<th>t-value</th>
<th>Pr(T ≤ t)</th>
<th>Significant</th>
</tr>
</thead>
<tbody>
<tr>
<td>With peri. info.</td>
<td>3.9 ± 1.6</td>
<td>85</td>
<td>9.5</td>
<td>&lt; 0.1 × 10^{-10}</td>
<td>Yes</td>
</tr>
<tr>
<td>Without peri. info.</td>
<td>5.8 ± 1.8</td>
<td>85</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 7.2: The overall mean and standard deviation of the pixel errors and its Student’s t-test results. “t-value” represents the corresponding value in the normalized Student’s distribution. “Pr” is short for probability.

<table>
<thead>
<tr>
<th></th>
<th>Mean (pixel)</th>
<th>Sample size</th>
<th>t-value</th>
<th>Pr(T ≤ t)</th>
<th>Significant</th>
</tr>
</thead>
<tbody>
<tr>
<td>With peri. info.</td>
<td>29.1 ± 12.8</td>
<td>85</td>
<td>27.7</td>
<td>&lt; 0.1 × 10^{-10}</td>
<td>Yes</td>
</tr>
<tr>
<td>Without peri. info.</td>
<td>45.2 ± 15.7</td>
<td>85</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Comparison of the overall errors from those two methods are available in the second column of Table 7.1 and Table 7.2, which further validates the above observance.

To validate that the difference between with and without periphery information doesn’t happen by chance, Student’s t-test [203] for two samples with one tail is performed on the obtained overall angle errors and pixel errors. The significance level (α) is chosen to be 0.01. The null hypothesis is that the mean angle errors (pixel errors) with and without periphery information are the same. The t-test results for angle errors and pixel errors are reported in Table 7.1 and 7.2 respectively. It is shown that the probability of t-test is smaller than 0.1 × 10^{-10}, which is smaller than the significance level (0.01) for both angle errors and pixel errors. Therefore, the null hypothesis for both angle errors and pixel errors are rejected, which concludes that the difference between with and without periphery information is statistically significant.

The obtained results validate that, compared with the original MIS video, the periphery augmented system achieves higher point estimation accuracy and improves subjects’ awareness of environment. This conclusion is further strengthened by the subjective evaluation of participated subjects. The average score of subjective evaluation for the usefulness of the peripheral information is 3.0 with range [-5 5].
Figure 7.4: The obtained errors per subject per round: a) angle errors, b) pixel errors. Green stars indicate errors from the periphery augmented video and red circles represent errors from the original MIS video.

Figure 7.5: Histograms of a) angle errors and b) pixel errors for each subject in each round of point tests. Green bars represent the periphery augmentation system and red bars stand for original MIS video.
CHAPTER 8

CONCLUSION AND FUTURE DIRECTIONS

8.1 Summary of Contributions

3D reconstruction and laparoscope localization for abdominal MIS are two crucial tasks to automatically understand the surgical scenes and better assist surgeons. The special MIS imaging environment causes multiple problems to the traditional methods and makes these tasks challenging. This dissertation has addressed these problems and advanced the state-of-the-art methods.

This dissertation first identified that distinctive image feature detection was a significant bottleneck for 3D reconstruction and laparoscope localization. In stead of relying on general feature points, this dissertation found that blood vessels are abundant on tissue surfaces, and proposed to explicitly detect those blood vessels and used them as image features. Efficient novel methods were introduced in Chapter 3 to detect those vessel features. Extensive in vivo experiments were conducted to compare the vessel features with other state-of-the-art methods and verified the distinctiveness of the vessel features.

After vessel features were detected, this dissertation focused on how to obtain 3D reconstruction based on those vessel features. In Chapter 4, novel methods were presented to match those features in stereo images. The proposed methods were specially designed for vessel features and were able to match most vessel features accurately. As a result of vessel feature matching, 3D structures of those vessel features were obtained.

Besides 3D reconstruction, this dissertation also focused on the laparoscope localization problem. To tackle the challenges of general tissue deformations, Chapter 6 proposed a new SLAM framework for the abdominal environment. The proposed method relied on RANSAC to handle general tissue deformations, and achieved accurate camera localization results. With stable laparoscope localization results, the above
vessel stereo reconstruction results were integrated together to obtain a consistent global 3D vessel network, based on which large-area dense reconstruction of surgical scenes was available.

The proposed large-area dense reconstruction method and the laparoscope localization technique can be used to assist surgeons in MIS. As one application, the periphery augmentation system (Chapter 7) provided surgeons with periphery information of the surgical environment. A user-evaluation system was designed to test whether it is useful to augment peripheral areas. Thirty subjects, including four surgeons with specialties on general surgeries, participated the test. Student’s t-test was performed on the collected data and verified that the difference between with and without periphery information was statistically significant. The results concluded that the proposed periphery augmentation system improved users’ understanding and awareness of surgical scenes.

8.2 Future Research Directions

Vessel features have been verified to be distinctive image features for 3D reconstruction and localization purposes. Vessel feature matching across stereo images has been shown to be accurate. In the future, one research direction is to design a proper image feature descriptor for branching points. The descriptors will allow matching of branching points across different viewpoints. In MIS environment, the light sources are mounted at the tip of the endoscope and move with the camera. In the experiments, it has been observed that the central areas of the images have better lighting than the others. In the future, it is desirable to improve the vessel feature detectors so that they are invariant towards the lighting conditions. The vessel features have been applied for 3D reconstruction purposes and shown promising results. In the future, it would be beneficial to apply the vessel features for localization purposes.

Compared with general scenes, the abdominal environment is much more constrained and its prior information can be discovered to assist the 3D reconstruction and laparoscope localization. The abdominal environment is composed of multiple deforming tissue organs. Generally speaking, those tissue surfaces are smooth and can be exploited to recover their 3D information. Meanwhile, the depth changes are usually much smaller in MIS environment and occlusions are less severe.

The stereoscope localization method presented in Chapter 6 can handle general tissue deformations. However it is not able to track the deforming points, which is important to learn the tissue deformation
models. In the future, one interesting research direction is to design methods to track the deforming points during the localization procedure. Additionally, it is also necessary to learn the bio-mechanical properties of those tissues, which allows better understanding of the scene and can exploited to improve the localization results.
REFERENCES


Appendix A: Permission to Reproduce Material

Below is permission for the reproduction of material in Chapter 3.

This is a License Agreement between Bingxiong Lin ("You") and Springer ("Springer"). The license consists of your order details, the terms and conditions provided by Springer, and the payment terms and conditions.

Get the printable license.

Copyright © 2015 Copyright Clearance Center, Inc. All Rights Reserved. Privacy statement | Terms and Conditions. Comments? We would like to hear from you. E-mail us at customerservice@copyright.com
Appendix A (Continued)

Below is permission for the reproduction of material in Chapter 5.

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © [Year of original publication] IEEE. 
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table. 
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author’s approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright credit notice should be placed prominently in the references: © [Year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis online. 
3) In placing the thesis on the author’s university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity’s name goes here]’s products or services. Internal or personal use of this material is permitted. If interested in reprints/publishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rightsrights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.
Appendix A (Continued)

Below is permission for the reproduction of material in Chapter 6.

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's University website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reimmersing publishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.