UNIVERSITY OF CALIFORNIA SAN DIEGO

# Nonrigid Registration Techniques and Evaluation for Augmented Reality in Robotic Assisted Minimally Invasive Surgery

A thesis submitted in partial satisfaction of the
requirements for the degree
Master of Science

in

Computer Science

by

Nelson Ho

Committee in charge:

Professor Ryan Kastner, Chair
Professor Sicun Gao
Professor Rajesh Gupta

2018

The thesis of Nelson Ho is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____

Chair

University of California San Diego

2018

DEDICATION

To my beloved parents, for putting up with me, and for their endless support.

# EPIGRAPH

*Augmented reality will take some time to get right,*

*but I do think that it's profound.*

—Tim Cook

TABLE OF CONTENTS


Signature Page . . . iii

Dedication . . . iv

Epigraph . . . v

Table of Contents . . . vi

List of Figures . . . viii

List of Tables . . . ix

Acknowledgements . . . x

Vita . . . xi

Abstract of the Thesis . . . xii

Chapter 1 Introduction . . . 1
1.1 Background . . . 2
1.2 Pipeline for AR Overlay . . . 4

Chapter 2 Related Work . . . 8
2.1 Datasets . . . 8
2.2 Surgical AR . . . 11
2.3 Nonrigid Registration Techniques and Evaluation . . . 13

Chapter 3 Deformable Datasets . . . 15
3.1 Stop motion Dataset Capture . . . 15
3.1.1 Motivation . . . 16
3.1.2 Data Capture Apparatus . . . 16
3.1.3 Modeling Clinically Relevant Procedures . . . 18
3.1.4 Capturing the Physical Properties of the Tissue . . . 20
3.1.5 Simulating da Vinci Camera Arm Movement . . . 22
3.1.6 Evaluation . . . 23
3.2 Real-time Dataset Capture . . . 25
3.2.1 Motivation . . . 25
3.2.2 RealSense Camera . . . 26
3.2.3 Quality and Evaluation . . . 26
3.3 Surface Reconstruction . . . 28
3.4 Summary . . . 29

Chapter 4 Segmentation . . . . . 31
4.1 Segmenting area of interest from background . . . . . 31
4.2 Segmenting Surgical Tools from Soft Tissue . . . . . 33
4.3 Summary . . . . . 35

Chapter 5 3D Registration . . . . . 36
5.1 Early Evaluation of Registration Algorithms . . . . . 36
5.2 Global Registration Baseline Model . . . . . 38
5.3 Evaluating TRW-S Geodesic Distance based Registration . . . . . 39
5.4 Naive attempt at accelerating TRW-S Registration . . . . . 40
5.5 SIRGn Evaluation Metric . . . . . 41
5.6 Experiments using SIRGn . . . . . 45
5.7 Summary . . . . . 47

Chapter 6 Conclusion . . . . . 48
6.1 Future Work . . . . . 49

Bibliography . . . . . 51

LIST OF FIGURES


Figure 1.1: A photo of the Intuitive Surgical da Vinci surgical system. . . . . 2
Figure 1.2: This image illustrates the computational pipeline necessary to provide Augmented Reality guidance in robotic assisted surgery. . . . . 5

Figure 3.1: Camera Dolly with 2DOF designed to capture movement of the da Vinci camera. . . . . 16
Figure 3.2: Surgical Tool stand with 3DOF and attachment point for a variety of surgical tool tips. . . . . 17
Figure 3.3: A graph showing the deflection of the tool stand apparatus applying force at varying angles. . . . . 18
Figure 3.4: The full setup used for the NextEngine laser scanner and tool stand. . . . . 19
Figure 3.5: Porcine liver retraction. . . . . 19
Figure 3.6: Porcine liver resection using a 45mm surgical stapler. . . . . 20
Figure 3.7: An image of the Phidgets bridge/amplifier and the Futek load cell we used for our experiments. . . . . 21
Figure 3.8: Mean shift tracking and smoothing result for da Vinci camera arm tracking. 23
Figure 3.9: The setup used to determine the camera arm path of the da Vinci. . . . . 24
Figure 3.10: A frame of the stop-motion video dataset, collected by the NextEngine Scanner. 25
Figure 3.11: This is a photo of the setup used to scan some liver on the same calibration board that was used with the NextEngine scanner. . . . . 27
Figure 3.12: A frame from the 3D video scans taken from the RealSense. . . . . 28
Figure 3.13: The result of a liver scan after performing a Poisson surface reconstruction. 29

Figure 4.1: This figure shows the result of segmenting the background from the dataset we collected using the NextEngine scanner. . . . . 32
Figure 4.2: The result of performing the tool segmentation on the dataset we collected using the NextEngine scanner. . . . . 33

Figure 5.1: These two graphs depict the tradeoff between speed and accuracy inherent between rigid and nonrigid registrations. . . . . 37
Figure 5.2: This image depicts the difference between a single registration and a global registration. . . . . 38
Figure 5.3: This series of photos shows the output of running SIRGn on our dataset. . . 44
Figure 5.4: This series of charts show the output of SIRGn run on two different nonrigid registration algorithms. The top row shows the results for a geodesic distance based registration. The bottom row shows the results for an ICP based registration. . . . . 46

LIST OF TABLES

Table 5.1: A table comparing best and worst features of several common registration techniques on our dataset. . . . . 37
Table 5.2: This table compares the advantages and disadvantages of several available toolchains and devices for developing an FPGA accelerator. . . . . 41

ACKNOWLEDGEMENTS

I would like to thank my advisor, Professor Ryan Kastner, for the opportunity to work with him through my time here at UCSD, and for all the guidance and support he has given me.

I would also like to thank my parents for the unconditional support and love they give me. This achievement would not have been possible without them.

Chapter 3, in part, is being prepared for submission for publication of the material. Michael Barrow, Shanglei Liu, Nelson Ho, Sonia Ramamoorthy, Santiago Horgan, Quentin Gautier, Peter Tueller, Ryan Kastner. "3D Video MISAR: Minimally Invasive Surgery Augmented Reality Ground Truth Data Set". The thesis author was a co-author of this paper.

Chapter 5, section 1, in part, contains material as it appears in a poster presented at the Scientific Session, Society of American Gastrointestinal and Endoscopic Surgeons, 2017. Michael Barrow, Shanglei Liu, Nelson Ho, Xinyi Yang, Brendon Chen, Santiago Horgan and Sonia Ramamoorthy. "A ground truth 3D video dataset for Augmented Reality Robotic MIS algorithms". The thesis author was a co-author of this paper.

Chapter 5, sections 5 and 6, in part, have been submitted for publication of the material as it may appear in Medical Image Computing and Computer Assisted Intervention, 2018. Michael Barrow, Nelson Ho, Alric Althoff, Ryan Kastner. "Benchmarking Video With The Surgical Image Registration Generator (SIRGn) Baseline". The thesis author was a co-author of this paper.

## VITA

| | |
|---|---|
| 2016 | B. S. in Computer Science, University of California San Diego |
| 2017-2018 | Graduate Teaching Assistant, University of California San Diego |
| 2018 | M. S. in Computer Science, University of California San Diego |

## PUBLICATIONS

Michael Barrow, Shanglei Liu, Nelson Ho, Xinyi Yang, Brendon Chen, Santiago Horgan and Sonia Ramamoorthy. "A ground truth 3D video dataset for Augmented Reality Robotic MIS algorithms". Poster presented at: Scientific Session, Society of American Gastrointestinal and Endoscopic Surgeons, 2017; 2017 March 22-25; Houston, TX.

Michael Barrow, Nelson Ho, Alric Althoff, Ryan Kastner. "Benchmarking Video With The Surgical Image Registration Generator (SIRGn) Baseline".

ABSTRACT OF THE THESIS

**Nonrigid Registration Techniques and Evaluation for Augmented Reality in Robotic Assisted Minimally Invasive Surgery**

by

Nelson Ho

Master of Science in Computer Science

University of California San Diego, 2018

Professor Ryan Kastner, Chair

Augmented Reality (AR) presents many new research opportunities in the medical field to improve physician training, provide new forms of patient rehabilitation, and advance patient specific treatment, among many other possibilities. This work focuses on the application of AR to robotic assisted Minimally Invasive Surgery (MIS) and its challenges. Robotic MIS improves patient care and recovery time because of reduced trauma to the patient during the procedured, but the loss of sensory feedback from the surgical site resulting from the indirection of using a robot presents a real challenge to the surgeon. AR makes it possible to replace some of the lost sensory information by overlaying pre-operative imaging in the AR field onto the surgical scene

in vivo. This will enable the surgeon to see the structures under the surface of the organ, which are visible in penetrative imaging but invisible to the surgeon's eye. AR also enables an opportunity to present additional information to the surgeon depending on what is currently in scene. To perform this overlay, the captured surgical scene needs to be segmented, and the results registered to the pre-operative imaging. There are existing non-rigid registration algorithms developed for other applications that may be used. However, the algorithms are slow, and there exists no way of determining the accuracy of such methods on a surgical application. Accurate datasets for the evaluation of such applications are difficult to obtain. This work addresses each of these issues by providing methods for creating a clinically relevant dataset and an evaluation framework for algorithms run on this data. We show how existing AR algorithms can be run on our dataset, and how our evaluation framework can be used to comprehend those results.

# Chapter 1

# Introduction

Minimally Invasive Surgery (MIS) promises better surgical outcomes as well as improved patient safety and recovery by reducing tissue trauma, blood loss, and risk of infection. [12][19] This results in patients spending less time in the hospital, which can also make MIS more economical compared to traditional open surgeries. Robotic assistance for minimally invasive surgery carries the same benefits as MIS, but offers greater visualization benefits through stereo cameras driving a 3D display as well as increased dexterity through the robot arms and miniaturized tools. [1] Robotic operation requires, in the case of the Intuitive Surgical da Vinci, the use of additional hardware in the operating room: the robot, the operating console, and a dedicated compute cabinet. The 3D visualization capability of the system provides opportunity to augment the current 3D surface visualization with pre-operative information. Tracking the movement of the surgical scene and providing the surgeon with more information through overlaid visual cues derived from pre-operative knowledge can help to further improve surgical accuracy and outcomes.

**Figure 1.1**: A photo of the Intuitive Surgical da Vinci surgical system. The surgeon operates the robot from the console, shown in the left hand side of the image. The robot in the center of the image is the surgical robot, with each arm of the robot manipulating a different surgical tool, each of which enters the patient through a small incision.

## 1.1 Background

Robotic assisted MIS is state of art in the surgical community and presents substantial patient benefits like faster recovery time and decreased trauma to healthy tissue. Robotic assistance in MIS is growing steadily since the technology received FDA approval [5], but there exist drawbacks to the technology that make it less desirable than conventional surgical techniques in some regards.

A major drawback is the loss of tactile feedback available to the surgeon from the surgical site. The haptic information from touching the patient's tissue and organs that is normally available to the surgeon in an open surgery is not conveyed as precisely when translated through the robot. The surgeon uses this feedback for guidance during the surgery to navigate the area of interest. As an example a surgeon performing a liver tumor removal may use the tactile

information gained from handling the liver to search for sub-surface landmarks that disclose the location of the tumor. This is much more difficult to do in a minimally invasive setting, and results in the surgeon having to do some guesswork to determine the location of the tumor. This limits the accuracy of the surgeon in such a procedure, and may result in conversion to open surgery, limiting the potential clinical benefit of an MIS approach.

Augmented Reality (AR) can alleviate some of the issues resulting from the loss of haptic feedback that robotic assistance introduces. AR is a technology that can provide its user a composite view of both the real world and computational augmentations of the real world. During pre-surgical planning phases, the areas of interest are imaged comprehensively to be able to come up with a surgical plan. Medical imaging is now advanced enough to generate a 3D model that can be manipulated by surgeons and radiologists on a computer screen, providing penetrative views of the tissue from any desired angle. However, this imaging is less useful when the surgeon is in surgery, and consulting the imaging requires the surgeon to switch tasks. By replacing the tactile information normally conveyed in an open surgery with penetrative pre-operative information overlaid into the surgeon's field of view, the need for guesswork and workflow disruption decreases and the overall accuracy of the procedure can be improved. In addition to alleviating the problems that come with robotic assistance, AR can provide a level of guidance not previously available in the operating room.

For example, when cutting through a vascular opaque organ such as the liver, surgeons need to be careful they do not sever any large vessels in the process. The cauterizing tool they use to cut can seal off small vessels in the organ, but severing a large vessel means uncontrolled bleeding. Traditionally these are avoided by consulting pre-operative imaging available as film or as a 3D model on a computer screen. Consulting this imaging requires the surgeon to switch tasks - to stop operating and look at the imaging. Augmented reality enables the possibility of providing this information without the surgeon having to lift their head, as part of a "see-through" presentation.

The extra information available to the surgeon in real time can have implications toward the standard of accuracy that surgery in this realm is evaluated by. The state of AR development in this application domain is somewhat stagnant, suppressed by many hurdles that developers in this area of research face. The algorithms needed to drive this AR application are not fast enough to make real-time AR surgical guidance possible as described. These algorithms need to be both faster and as accurate as possible, and we would like to develop faster more accurate algorithms. Before development is possible, some groundwork needs to be done in order to effectively evaluate algorithmic contributions.

## 1.2 Pipeline for AR Overlay

The computational pipeline required to overlay pre-operative imaging onto a surgeon's live camera view of the operating site is very expensive. The pipeline can be broken down into three logical portions, as shown in Figure 1.2. 3D surface depth map capture technologies like laser scanning are improving at a fast pace. Depth map acquisition time is decreasing, scanners are getting smaller, and resolution of scans is increasing. There is significant progress being made toward the first stage of the pipeline, but it is outside the scope of this work. The acquired depth map of the surface will be in the format of a mesh in 3D cartesian space. Medical imaging techniques like MRI can be used to generate a penetrative pre-operative 3D model of the organ. The pre-operative model can also be represented as a mesh figure in 3D cartesian space. Segmentation of the area of interest from the captured surface depth map is the next step of the pipeline. Real time automated segmentation is also an open research area, but is out of the scope of this work. Once the organ is segmented from the rest of the patient, the surface is *registered* to the pre-operative model.

Surface registration performs a point to point matching of the live 3D organ surface capture to the pre-operative penetrative model. There exists several different methods of performing

this registration: some are locally weighted nearest neighbor searches, and some take a global point transform cost minimization approach to the matching. Once the surface transform field is computed, the surface of the model generated from the pre-operative imaging matches the surface of the surgical site in the captured coordinate space. The final step of the pipeline is the physics simulation that determines how the inside of the organ moved based on its observed surface deformation.

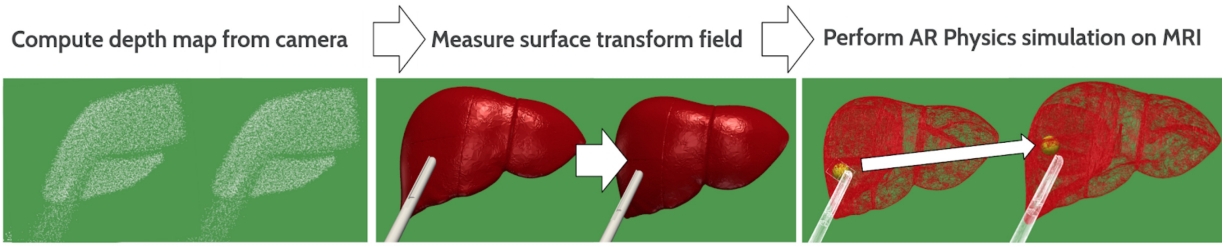



**Figure 1.2**: This image illustrates the computational pipeline necessary to provide Augmented Reality guidance in robotic assisted surgery. The first step is to compute the 3D depth map of the surface of the tissue in the area of interest. The next step is to calculate the surface transform from one video frame to another - this surface transform is called registration. Once the surface transform is known, that information can be used to perform a physics simulation to see how the internal structure has changed.

Pre-operative penetrative imaging like MRI generates a 3D model that can be exported in DICOM format which can be easily converted to other accepted 3D data encodings.[14] Pre-operative imaging allows surgeons to see the internal structures of the organ. To provide surgical AR guidance using this preoperative imaging, we must be able to deform the 3D model to match the orientation of the organ during surgery, so that desirable or useful portions of the model can be properly overlaid in the surgeon's field of view. To accurately deform the 3D model to fit the surgeon's field of view is a difficult problem. To properly simulate the movement of the internal structure of the organ as it is being deformed requires knowledge of the amount of strain applied to the surface and how the internal structures react to this strain. We cannot place strain gauges inside the organ during surgery, so we need some other indirect way of measuring the applied force on the tissue. By measuring the amount of strain applied to the surface of the organ and observing the surface deformation that results, we can build a model to correlate these two

measurements. Once we have that model, surface deformation can be measured in vivo via depth map of the surface, and the amount of strain induced on the surface can be interpolated from the amount of deformation exhibited. We can use that information to perform a physics simulation on the tissue to calculate how the internal structure has changed as a result of the applied surface strain.

These steps of the pipeline describe the computationally limiting factors of using AR for the described application. The depth map acquisition technology that attaches on the front end of the pipeline and the AR visualization devices that the pipeline feeds into are currently being used in industry and are seeing adoption and further development. Frame rate laser scanners like the Intel RealSense and AR display devices like the Microsoft HoloLens demonstrate maturity of technology on either end of this pipeline. There is work being done to explore the many uses of AR in medicine, as well as explicit stated interest from the School of Medicine at UCSD for the specific application described here.

This project alleviates some of the problems that exist with researching AR methods for surgical guidance. As previously mentioned, development in this area is difficult because of the lack of data available for testing, and the lack of an evaluation metric to be able to compare implementations and heuristics. I present solutions for both of those problems, in the hope that easing the development process will promote research in this area.

Chapter 2 is a literature review of the field, and frames the originality of this research. Earlier research does not satisfactorily address the problems for which we propose solutions. In Chapter 3, I will discuss methods to collect deformable data sets, used as input representative to our problem domain to run segmentation and registration algorithms. The dataset must be representative of a real-world problem, otherwise evaluations made of the AR algorithms will not be indicative of the actual performance. I will describe a segmentation method used for our collected data set in Chapter 4. As mentioned before, real-time automated segmentation of medical imaging is an open research problem that will not be addressed in this document.

The segmentation described is simply an intermediate processing step of the framework that we present, and is mostly applicable to the datasets we used. In Chapter 5 I will give an evaluation of nonrigid registration algorithms on the dataset collected using the described method. I will present a general method for quality evaluation of nonrigid registration algorithms.

# Chapter 2

# Related Work

This chapter is a literature review of the field and identifies the relevant previous work, while also making more clear the novelty of our contribution. This chapter is split into three sections - datasets for surgical AR, the progress of surgical AR research, and techniques and evaluation methods for nonrigid 3D registration. There exists work that is similar to our work, but none of the prior work meets our needs.

## 2.1 Datasets

We need to create our own dataset because existing datasets are insufficient. We need a dataset that accurately represents the challenges that exist during an actual procedure. In order to estimate applied force based on surface deformation, we need surface strain measurements to go along with the dataset. We also need the dataset to involve a series of frames over time, to represent a video feed. Each frame is a 3D depth map of the scene. The goal is real-time AR guidance; we need to be able to register one frame to the next frame 25 times per second to keep up with the 25 frame per second video stream. Ideally, the 3D dataset would capture and represent accurately surface deformation over time at a rate of 25 scans per second. No datasets exist that fit all of these requirements, but there are several that satisfy some of those requirements.

Offline pre-surgical planning relies heavily on pre-operative penetrative imaging - such as MRI, CT, and ultrasound, among others. This imaging is also used during the surgery so the surgeon can reference it while they are operating. Segmentation of the imaging marks and separates important anatomical structures. This is useful to the surgeon to more clearly visualize where certain structures are located; for example, to mark the boundaries of a tumor embedded in the patient's liver, or to identify the organ we want to provide the visual overlay on. This is also a necessary step in the AR pipeline - a preprocessing step before registration can be performed. [17]

There are many existing datasets available that we evaluated for our project before deciding to create our own. The Visible Human Project [3] provides 3D data for the entire body. This data can be used for segmentation and other image guidance development. However, the scans are of non-diseased tissue, and the images are static and do not capture how live tissue deforms. The dataset was created from a cadaver, sliced into many thin cross sections. During surgery, the surgeon likely does not encounter a complete cross sectional view of the patient. There are some more targeted datasets published after the Visible Human Project that depict diseases in a specialized area, such as the brain or liver, that are useful for disease segmentation. But these are static images and, again, do not reflect the way that live tissue deforms during a procedure. The static views also do not exhibit the characteristics of frame-rate deformation of the tissue that we aim to take advantage of in our processing pipeline. They do not provide depth data, nor do they provide surface strain data. [11] [26]

Some existing datasets are suitable for some registration work, but they do not address the issues of the missing physical measurements that are required for AR development. Most critically, these datasets do not have measurements for the amount of force that the surgical tool is applying to the tissue, nor does it provide a biomechanical model of the tissue. As a result, we are unable to correlate the deformation exhibited in the surface of an organ, to the force that was applied to that surface. Without this information, combined with the biomechanical model of the

tissue, we are unable to estimate how the internal structure of the organ has shifted in response to this external manipulation.

The AME Surgical Video Database [25] provides surgical video of actual open and laparoscopic procedures. This dataset is a great contribution to the public domain because videos of surgeries are not readily obtainable in general. These are specifically uploaded because they comply with all the strict federal guidelines of how patient data may be used. None of these videos contain patient identifiable information. This dataset is useful for frame to frame video segmentation of the organs in view of the surgeon. However, this dataset is only of 2D video, which means only 2D segmentation techniques may be applied to this dataset. Since our target procedure is for viscoelastic organs like the liver or prostate, this dataset has been useful to us to see how those organs are manipulated during a minimally invasive laparoscopic procedure. However, the dataset is not useful for AR algorithm development.

Another dataset that showed promise is the SUrgical Procedure PORTal [18]. However, this dataset suffers from the same shortcomings as the previous one. This dataset was mostly intended as education for students of surgery, and provided us the same benefit of education that [25] did. Because it does not provide depth or surface strain measurements, it was not particularly useful for AR development.

An educational dataset called GIBLIB [2] is also available with plenty of surgical video that may, again, be used for 2D segmentation of various structures during procedures of interest, but prove to be of no use for AR development. There is a section of their database dedicated to "360VR Video" that grabbed our attention initially. However, the virtual reality 3D data it provides is of the operating room during the surgery, not intraoperative subdermal 3D data of the patient.

The Hamlyn dataset came closer to meeting our needs, as it provided stereo video of a laparoscopic surgery. While depth information is not provided with the dataset, depth can be reconstructed from the stereo video. The procedures included in this database are also procedures

of our interest. This dataset could be used for segmentation and development of 3D registration algorithms, as it satisfies both necessities of exhibiting nonrigid surface deformation and clinical relevance. The dataset even provides camera lens calibration parameters. However, the resolution is quite low, introducing doubt that a stereo reconstruction algorithm could accurately reconstruct the 3D surface. Additionally, this dataset fails to deliver the surface strain measurements that correspond with the deformation exhibited in the video, which makes it unsuitable for AR development.

Synthetic datasets can be created to alleviate some of the issues present with current surgical video data, because the biomechanical model of a fabricated organ would be readily available and 3D video can be taken of the model. They can be manufactured to a certain mechanical specification, so we know the mechanical properties of the material. However, these synthetic models are time consuming to create, and the color and environment generally do not accurately represent a real surgical environment.

## 2.2 Surgical AR

The application of Augmented Reality and Virtual Reality to the surgical field has generated much interest in the world of medicine. Physicians and researchers are interested in methods to use this new technology to enhance the surgical experience and patient care. [16] Surgery has been identified as one of the areas of healthcare that could most readily benefit from this technology. Instead of a traditional wall mounted "head-up" display that shows more detailed information about the visual scene, AR enables a "see-through" superimposition of that data. The idea of a superimposed display, what we call Augmented Reality today, was first introduced by Ivan Sutherland to the military in 1965. [20] So it is evident the idea is not new, but the technology has advanced to a point where this is a feasible idea today.

There is existing research today that shows the application of Augmented Reality in the

fashion we propose - the superimposition of sub-surface structures into the surgeon's field of view in real time. However, this has only been applied to rigid structure surgery, such as maxillofacial surgery. Unique computational challenges exist in trying to do the same thing during surgery of non-rigid organs and structures.

Wang et al. demonstrate the feasibility and effectiveness of using Augmented Reality to assist surgeons during procedures that involve rigid structures like the jaw. [23] Using AR, they are able to provide the surgeon an overlay of the bone structure that is hidden under the gums. The goal of their work is real-time guidance for these structures. The works they cite are either unable to provide real-time guidance because of required computation time, or are inaccurate because they perform a 2D-3D matching from the 2D visual feed to the 3D imaging obtained before the surgery. Many of the methods they propose take around 30 seconds to register each frame - that is far from what is required for frame rate. The method they propose can accurately register pre-operative 3D skeletal models into the surgeons field of view at 30 frames per second. The results, obtained by evaluation on non-patient volunteers and on video obtained during an actual maxillofacial procedure, demonstrated accurate registration and promise in a real clinical setting. The method requires no fiducial markers or trackers, which is greatly preferable to the alternative. This research demonstrates the viability of this method of guidance. The problem we are tackling arises when attempting to perform the same type of guidance with soft organs, instead of the rigid structures tht exist in oral and maxillofacial surgeries.

The registration method used by [23] is a rigid registration. This registration is necessary to align the offline generated model to the patient's orientation during surgery. Since bone does not deform, a rigid registration algorithm like Iterative Closest Point can be used to align the model. Correction and registration failure recovery are achieved through course alignment by a learning framework. On a GPU, this pipeline can be optimized, as demonstrated, to achieve real-time performance. However, once the requirement of tracking non-rigid deformation comes into the picture, this performance cannot be achieved.

## 2.3 Nonrigid Registration Techniques and Evaluation

The initial registration that must be done to enable AR guidance is to register the 3D model created from pre-operative penetrative imaging to the current frame in the surgeon's visual feed. Once the initial registration has been performed, the surface of the 3D pre-operative model matches the surface of the organ in vivo. The 3D organ model captured by MRI can now be overlaid onto the surface of the organ during the surgery. As the surgeon continues to manipulate the organ during surgery, the registration must keep up with how the orientation of the organ changes from frame to frame. Each frame needs to be surface matched to the next so we can keep track of how the surface is changing, so we can continue to provide guidance. If the pre-operative model is not transformed as the organ is manipulated during surgery, then we lose the benefit of AR guidance.

There exist several nonrigid registration techniques in the literature that may be suited for use in the application of surgical AR guidance. As mentioned before, an algorithm that does not use fiducial markers or external trackers is preferable. Amberg et al. present a nonrigid variant of ICP in *Optimal step nonrigid ICP algorithms for surface registration* [4]. This is a dense registration method that uses ICP iteratively with a stiffness parameter that limits the search space of the point matching. The target application of their development was facial registration. We took interest in this algorithm because it does not require the use of fiducial markers or trackers. We present an evaluation of this algorithm for our application in Chapter 5.

A different method of nonrigid registration is based on geodesic distances rather than ICP. The method presented by Chen et al. [10] uses this method of geodesic distance matching. Geodesic distances are the shortest distances between the vertices of the mesh. These distances are then mapped to Euclidian space. Chen's method is an optimization method where the minimization of Euclidian distance distortion is the objective function. The algorithm was developed for human pose estimation. Like nonrigid ICP, it does not depend on fiducial markers

or external trackers, making it a viable approach for surgical guidance. This registration method also reported the best accuracy of the methods that we found, though the preprocessing cost of obtaining the geodesic distances is higher. We present an attempt at acceleration of this algorithm and an evaluation of its quality in Chapter 5. There are other nonrigid registration methods based on geodesic distances published in the literature. One notable one, also cited by Chen, is the method proposed by Wang et al. in [22]. This method is based on the classic maximal flow/maximal bipartite matching problem - the edge weights in the bipartite graph are relative to the geodesic distances between those points.

We previously addressed the issue of having a representative dataset for evaluation of our methods. We provide a survey of several approaches to nonrigid registration without the use of fiducial markers. We face another issue - now that we have several different registration methods, we need a way to effectively evaluate the quality of a given registration algorithm, or more generally a given AR guidance algorithm, on a particular dataset. Most existing evaluation frameworks are for rigid registration. [13] There exists work that describes quality metrics for nonrigid registration, but only works on methods and datasets that use fiducial markers. [24] We need a method that works on information from the tissue surface only, as it is hard to place markers on tissue during an in-vivo operation. Our dataset does not contain fiducial markers, though they can be placed, because we are concerned with evaluation of algorithms with respect to online “real-time” surgical scenarios.

# Chapter 3

# Deformable Datasets

Deformable datasets for evaluation of AR algorithms in surgery are scarce, and sources of data are protected by privacy regulations. This makes it very difficult to obtain an accurate dataset to use as test data for algorithm development. Because the algorithms we want to evaluate are of high computational complexity and use heuristics to improve runtime, accurate data is necessary to accurately evaluate the efficacy of a given heuristic. The problem does not get easier when evaluating a given algorithm for multiple surgical scenarios, as each scenario needs a different dataset(s) in order to be evaluated. Shortcomings of existing solutions were described in Chapter 2.

## 3.1 Stop motion Dataset Capture

Because of regulatory concerns, collecting data from an actual surgery to use for development is an unlikely endeavor. To capture the mechanical properties accurately, we had to use real tissue. Our stop-motion dataset capture method is used to capture surface deformation in a store-bought porcine liver.

### 3.1.1 Motivation

A 3D frame-to-frame nonrigid registration is an essential part of the pipeline necessary to support AR in the surgical guidance context. We found existing published nonrigid registration algorithms and wanted to see how well they would work for registering surgical video. The registration algorithms we found were mostly developed for other applications. We began looking for 3D surgical video datasets to evaluate this on, but we could not find any. So we made our own.

### 3.1.2 Data Capture Apparatus

We desired to build an apparatus that can mimic the motion of the surgical robot we want to simulate. The laser scanner we chose that met our needs in terms of accuracy was the NextEngine 3D laser scanner. We built a carriage for the laser scanner that will allow the scanner acquire depth maps from a top-down vantage point, similar to the surgical robot. The carriage sits on rails to simulate side to side panning of the camera, shown in Figure 3.1 and can rotate as well to mirror the viewing angle of the da Vinci camera arm.

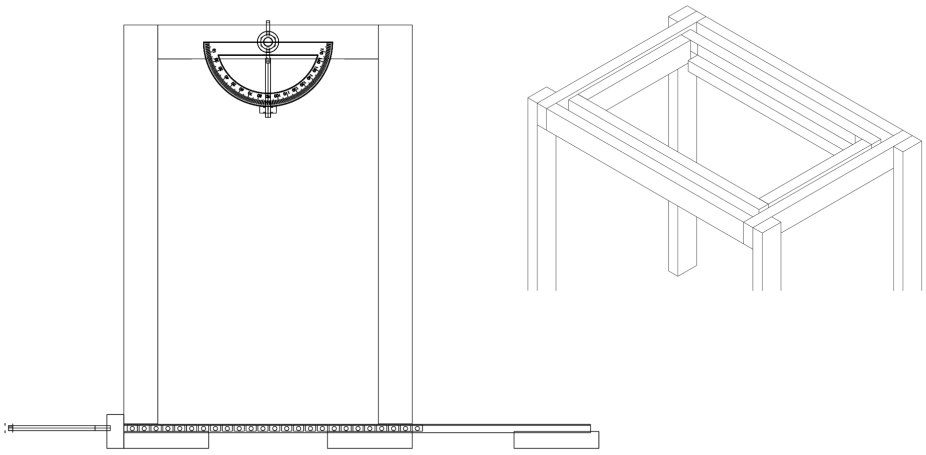


**Figure 3.1**: Camera Dolly with 2DOF designed to capture movement of the da Vinci camera. The top tray can rotate, with the angle of rotation of the camera marked by a protractor. The entire dolly can slide on rails to allow a panning motion of the camera.

A separate apparatus was built to simulate motion of the tool arm attached to the surgical robot with 3 degrees of freedom and the ability to attach a variety of tools to the end. There is also a load cell inline with the tool, to measure the force that the tool is applying to the surface of the liver.

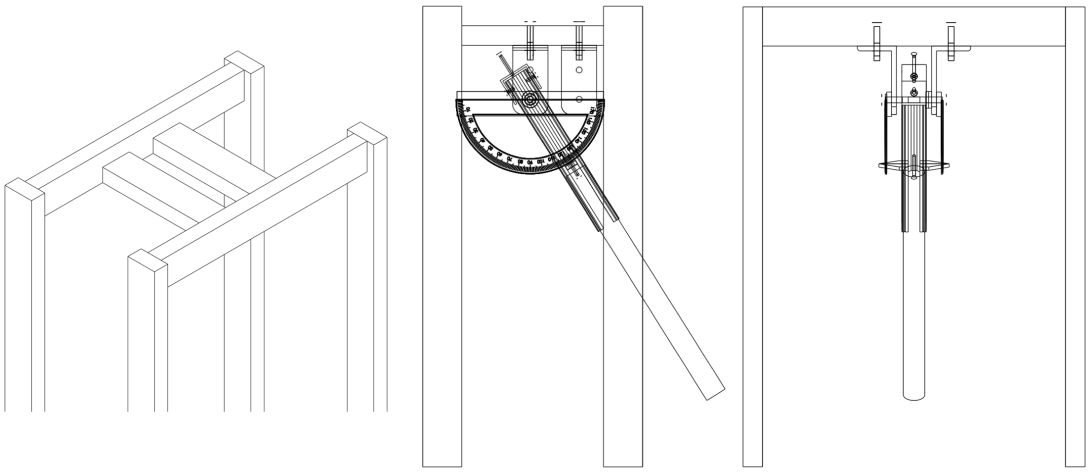


**Figure 3.2**: Surgical Tool stand with 3DOF and attachment point for a variety of surgical tool tips. The angle and length of the tool holder is adjustable. The tool tip is mounted on a telescopic rod to enable the tool to depress the surface of the organ. The stand itself can also be horizontally displaced.

The tool mount has a telescoping rod, adjustable by a threaded rod running through the center of the rod. The rod sits in a machined aluminum billet, and slides back and forth through a channel machined into the aluminum mount. We realized that because of the loose tolerances we used, there may be some varying deflection of the tool depending on the angle and displacement. We attempted to characterize the deflection, and performed a series of experiments with the tool applying force on a surface at varying angles. We found the maximum deflection was $1^{\circ}$. We reason this is an acceptable amount of deflection because it is small enough to be used for small-angle approximation in calculating the force vector. The results of the experiment are shown in Figure 3.3.

The camera dolly and the tool stand come together with the camera calibration board to complete the full setup, shown in Figure 3.4.

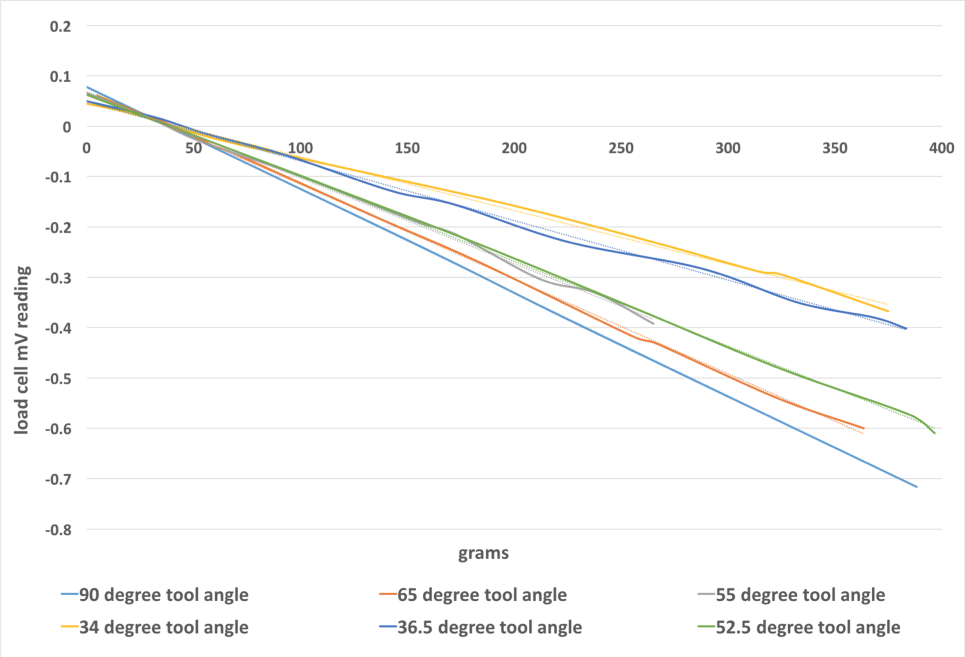



**Figure 3.3**: A graph showing the deflection of the tool stand apparatus applying force at varying angles.

### 3.1.3 Modeling Clinically Relevant Procedures

For this dataset to be useful, it should model procedures and manipulation of tissue that is clinically relevant to a MIS procedure. We chose key moments of a MIS hepatic resection to inspire the manipulation we show in our dataset. The viscoelastic property of liver tissue combined with the difficulty of MIS hepatic surgery makes this an interesting procedure to target. We chose several common manipulations of the liver to include in our video dataset: camera pan with tissue palpation, organ retraction, and organ resection.

#### Camera pan with tissue palpation

This is a simple dataset, to provide a simple test case for early development. Liver resection is clinically challenging. Having a simple base case of just the camera moving, then a round prod slowly depressing the surface, provides a simple example to perform initial evaluation of algorithms on.

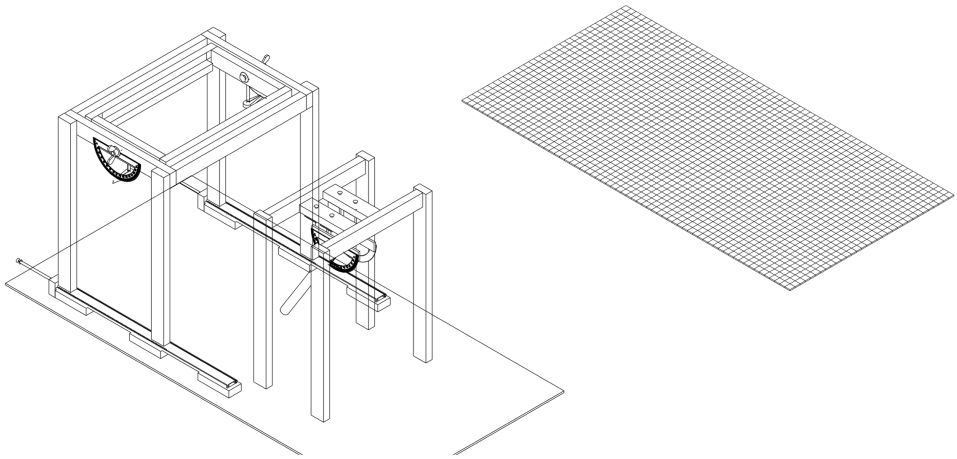
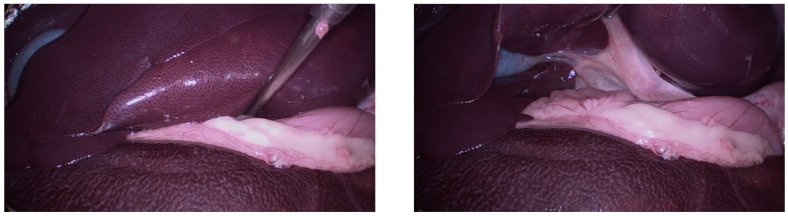


**Figure 3.4**: The full setup used for the NextEngine laser scanner and tool stand. The checkerboard is used for camera calibration, and has mounting holes to fix the location of the stands.

### Organ retraction

Retraction of the liver is a common step in hepatic resection. This is done to provide access to the underside of the liver, and in general easier access to the various surfaces of the organ in an insufflated abdomen. Figure 3.5 depicts this step.



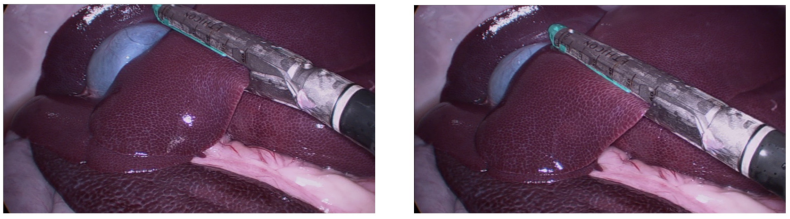
*(a)* ***Placing the retractor*** *(b)* ***Raising the liver***

**Figure 3.5**: Porcine liver retraction. The retraction is performed using a Nathanson retractor.

Retraction is a simple step, but presents an interesting case in terms of registration. As the liver is lifted, a different portion of the liver is visible in each frame. This procedure does not exhibit permanent plastic or tearing deformation.

**Organ resection**

Liver resection is often done with a surgical stapler, to remove the afflicted tissue once the malignancy is identified and dissected. The stapler we use is an Echelon Flex powered endopath stapler, capable of resecting a 45mm length of tissue. Figure 3.6 shows the deformation caused by closing the stapler on the organ.



*(a)* ***Stapler open*** *(b)* ***Stapler closed***

**Figure 3.6**: Porcine liver resection using a 45mm surgical stapler.

This is probably the most difficult procedure to model and perform the computation for. Permanent plastic deformation and tearing of the tissue is exhibited in this procedure. None of the registration algorithms we evaluate can handle the deformation exhibited in this procedure.

### 3.1.4 Capturing the Physical Properties of the Tissue

As mentioned above, the apparatus we used to apply a surface deformation to the tissue we were recording has a load cell inlined into the tool so we can record how much pressure we apply to the tissue per frame. The load cell we used was an S-beam style load cell - specifically a Futek LSB200. It has a capacity of 2lbs and has accuracy $\pm$0.5mN. We had to use a brige with a signal amplifier to read the voltage from the load cell. We used a Phidgets RB-Phi-107 to serve this need. The load cell and bridge are shown in Figure 3.7

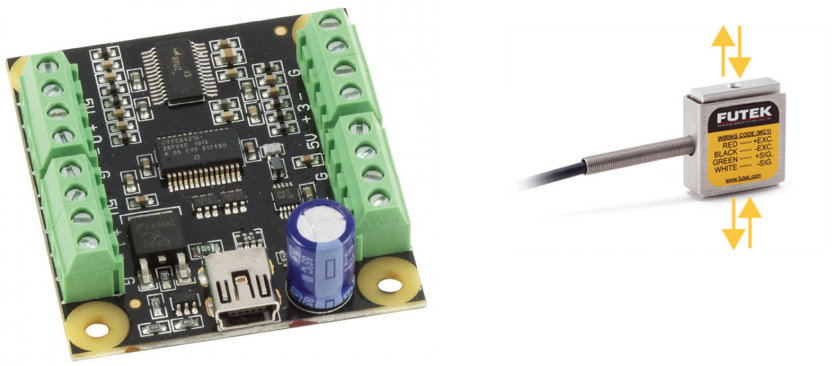



**Figure 3.7**: An image of the Phidgets bridge/amplifier and the Futek load cell we used for our experiments. The bridge is shown on the left, and the load cell shown on the right.

### Calibrating the load cell

The load cell was calibrated by the factory less than two weeks before we began our experiments. After integrating the PhidgetBridge and the load cell, we had to perform our own calibration. We found the PhidgetBridge amplifier is linear, if just a little bit noisy. We used this calibration information to relate the output voltage of the load cell to a force measurement.

### Simple Biomechanical model

A biomechanical model relating tissue movement to force is provided with each video data set to help verify AR methods. This model can be used to reconcile between tissue surface deformation and the corresponding force measurements. This knowledge is necessary to simulate how internal organ structures move over time in AR image guidance. This model combined with the recorded force applied for each frame of the dataset gives developers a comprehensive model to relate amount of surface deformation to applied force. Force readings obtained while monitoring the stress decay at a constant strain in each of the data collection experiments were used to calculate th biomechanical properties. Using the force reading and other known physical dimensions of the liver, we substitute in the canonical derivation of Young's modulus to obtain the modulus of elasticity. We also were able to give a stress relaxation model of the tissue in a

two term Maxwell-Wiechert model by recording the stress decay as measure by the load cell.

### 3.1.5 Simulating da Vinci Camera Arm Movement

The camera dolly and tool mount were designed with the intention of being able to accurately capture the movement of the da Vinci camera arm. Once the equipment was constructed, we had to carefully plan its adjustment from frame to frame in order to ensure that the movement of the camera accurately represents the viewpoint of the da Vinci camera. We tried to use an IMU to perform this camera path planning, but there was too much drift in the collected sensor data to be able to use the information. Because we used an off the shelf solution (TI sensortag) we did not have the correct calibration settings to account for this particular sensor's drift due to gravity and inherent sensor inaccuracies. There were available complete solutions with software appropriately calibrated to the device's drift, but they were expensive.

We settled on a cheaper solution - we recorded a video of the camera arm of the da Vinci while it was being driven by a skilled surgeon, performing a clinically relevant motion. We tracked the location of the camera lens by placing a marker on the end of the camera arm, and we postprocessed the video we took. Using a mean shift tracking algorithm, we tracked the movement of the marker on the camera end and emulated that movement with our scanner carriage. Because the output from the tracking algorithm was a bit choppy, we used a 3Hz Butterworth filter to smooth the path data. The result is shown in Figure 3.8.

With the fiducial marker in combination with the mean shift tracking algorithm, we were able to track the movement of the camera through space. However, we realized that having only that one marker made it impossible to calculate the angle of the camera, only its lateral displacement. Our scanner carriage was designed to be able to emulate both lateral movement as well as camera axis rotation. We had to find a way to calculate the camera angle from the video we took.

In each frame of the video, the camera arm meets with a fixed horizontal to form an easily

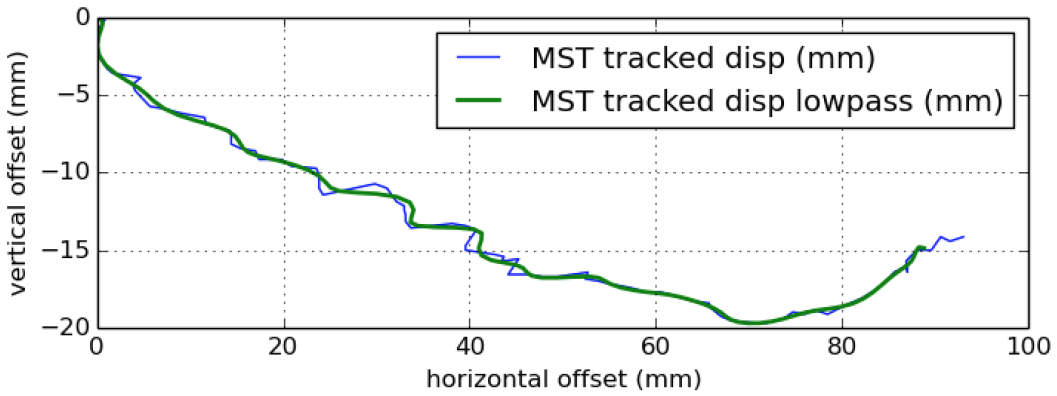



**Figure 3.8**: Mean shift tracking and smoothing result for da Vinci camera arm tracking. The blue line is the camera position we ascertained from the video and feature tracking. The green line is the filtered path.

measured angle. The horizontal is the same in each frame, and its levelness was ensured at the beginning of the video, shown in figure 3.9. The camera was mounted at the same level as the ruler as well. Using the fixed point and the slope, we were able to describe the angle of the camera by a linear equation in a 2D Cartesian plane with the fixed horizontal as the x-axis and the vertical as the y-axis. After doing this for each frame, we have an overdetermined system of linear equations. We calculate the least squares solution for this system by $A^TAX = A^TB$. The solution is plugged back into the functions that were derived from the tracked point and the slope of the camera arm, which we then used to determine the angle that the camera carriage should be set at in each frame.

### 3.1.6 Evaluation

The dataset that we were able to acquire using this procedure turned out to be quite good for our registration work. The collected scans were of high quality, even without using talc to treat the shiny surface of the porcine liver. Figure 3.10 shows an example. There were a few holes in the surface from light refraction, but a Poisson reconstruction in Meshlab fixed this and made the surface watertight. We were able to evaluate several registration algorithms using this dataset.

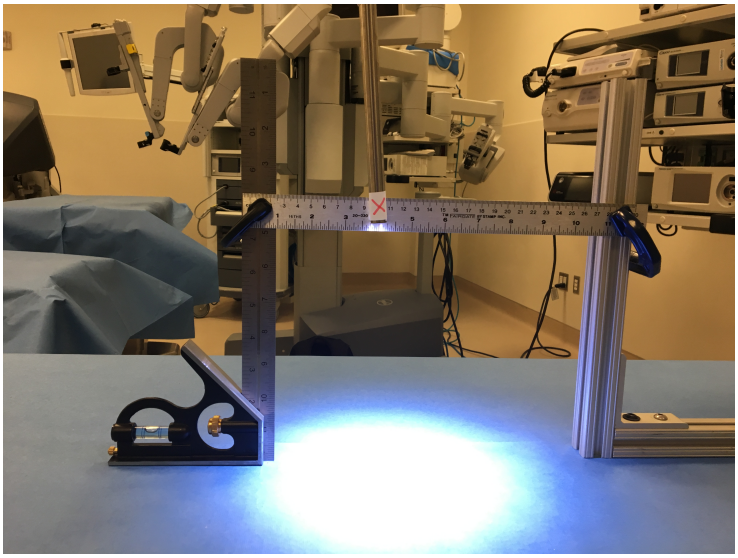


**Figure 3.9**: The setup used to determine the camera arm path of the da Vinci. The camera was mounted at the same height as the ruler, and the ruler was mounted using a line level.

Shortcomings center around the amount of time that this process takes. Each scan we collect corresponds to one frame in the resulting video dataset. At 25 frames per second, 75 scans needed to be collected for just 3 seconds of video. This took a considerable amount of time, because the acquisition time for the scanner is about 5 minutes per frame, with some staging adjustments in between each frame. This meant that 3 seconds of video took a whole day of work to collect. Manpower and the tediousness of the task were real limiting factors. Additionally, due to the long acquisition times, the liver began to dry out and change temperature, which did mildly affect its mechanical properties. We attempted to characterize the mechanical property of the liver before and after the recording session, but were not able to capture those changes in an accurate way. We also used a spray bottle to keep the liver from drying out. While this procedure did produce good results in terms of the dataset, it was not scalable and quickly became infeasible as it took an incredible amount of manual effort and time to accomplish.

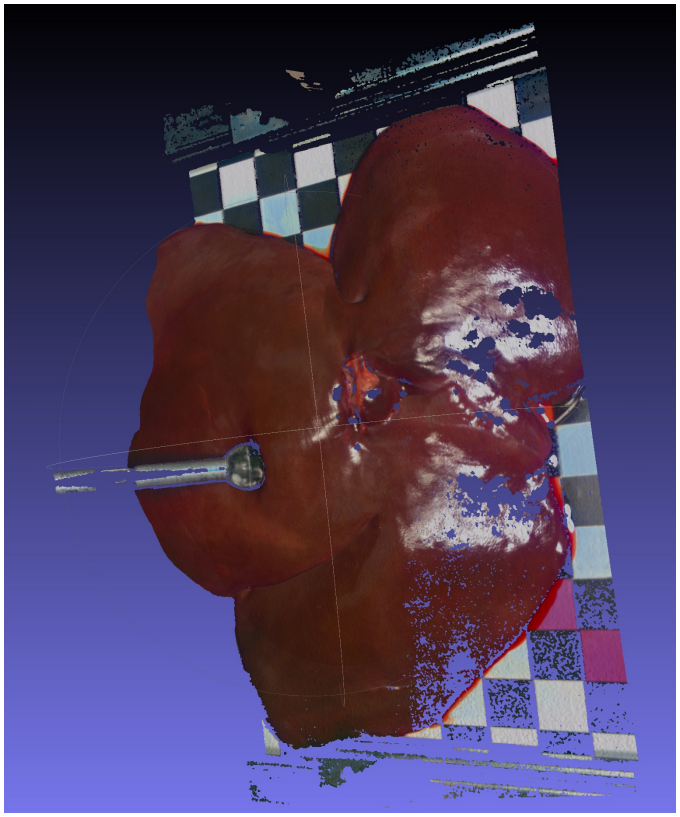


**Figure 3.10**: A frame of the stop-motion video dataset, collected by the NextEngine Scanner. As you can see, the point cloud is very dense, and has colored faces, making it possible to use that information if your algorithm depends on it.

## 3.2 Real-time Dataset Capture

The stop-motion dataset capture method produced high resolution results, but it did not capture the surface deformation at frame rate. Individual staged frames were compiled into a video offline. New technology presented the possibility of being able to capture these 3D video datasets simply by a point and shoot method.

### 3.2.1 Motivation

There were two main motivating factors to develop a new dataset capture method, both of which intend to address the shortcomings of the aforementioned method. The first is to shorten the amount of manual effort and time required. The second is to more accurately capture the physical

properties of the target - because of the long scan acquisition time of the previous method, the tissue becomes dehydrated, settles, and deforms during the long recording procedure. Being able to record accurate 3D video in real-time would solve the time issue, because recording 3 seconds of video would take 3 seconds, plus setup time. Real-time recording would also capture the deformation in the area of interest at the time the deformation was applied, instead of allowing the tissue and area of interest to settle over the scan acquisition time.

### 3.2.2 RealSense Camera

The RealSense camera we used enabled us to record depth information at "real-time", with depth information being captured at up to 90 fps and RGB frames captured at 30 fps. Compared to the NextEngine scanner, this is a tremendous improvement over the 5 minutes per frame that the NextEngine took per frame. This enables much faster and more efficient data collection, as well as higher accuracy in terms of timeliness of frame capture. However, the resolution is much lower with the RealSense camera as compared to the NextEngine scanner - in practice we achieved about 250,000 vertices per frame with the RealSense, and about twice that with the NextEngine. The NextEngine is more consistent as well, which is expected given the long scan acquisition time. Figure 3.11 shows the very simple setup required to use the RealSense camera - a simple tripod achieves the task.

### 3.2.3 Quality and Evaluation

While the RealSense camera greatly alleviates the burden of capturing new data, it is only helpful if we can actually use the data for development needs. Because of the lower uniformity of density and lack of faces in the collected scans, we had concerns that this data would not work well in existing registration algorithms, or would fail in the registration benchmarking tools we have developed. The point density is also non-uniform, with points clustering around the

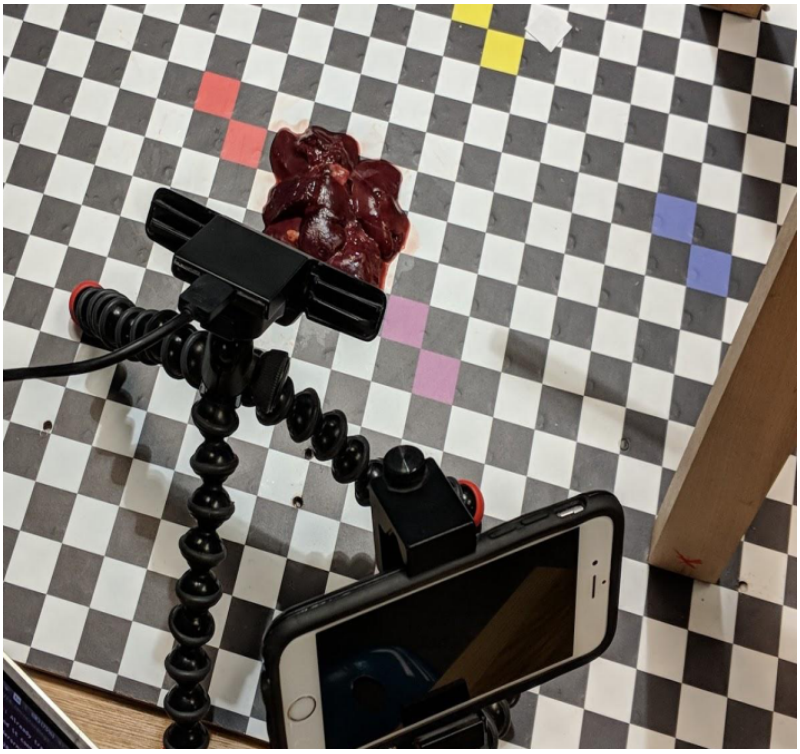


**Figure 3.11**: This is a photo of the setup used to scan some liver on the same calibration board that was used with the NextEngine scanner. As is clearly visible, the setup for this scanner is much simpler and requires much less manual effort than the NextEngine setup.

projected line from the sensor. This presents further accuracy concerns because the low point density areas may be construed by our registration benchmarking tool as areas that have exhibited a high level of deformation from frame to frame. We were able to generate a registration baseline model using this data, but we do not know if the quality of the baseline model is negatively affected by the varying point density. The baseline model is further discussed in section 5.2. As the technology evolves, the RealSense and similar types of scanners will likely become the preferred capture method, as they are more accurate in a temporal sense, and take less time to procure the data. Considering the man-hour savings, the scans we were able to capture in three seconds with the RealSense were incredible. We expect the quality of the scans that this style of scanner can collect will increase over time as well.

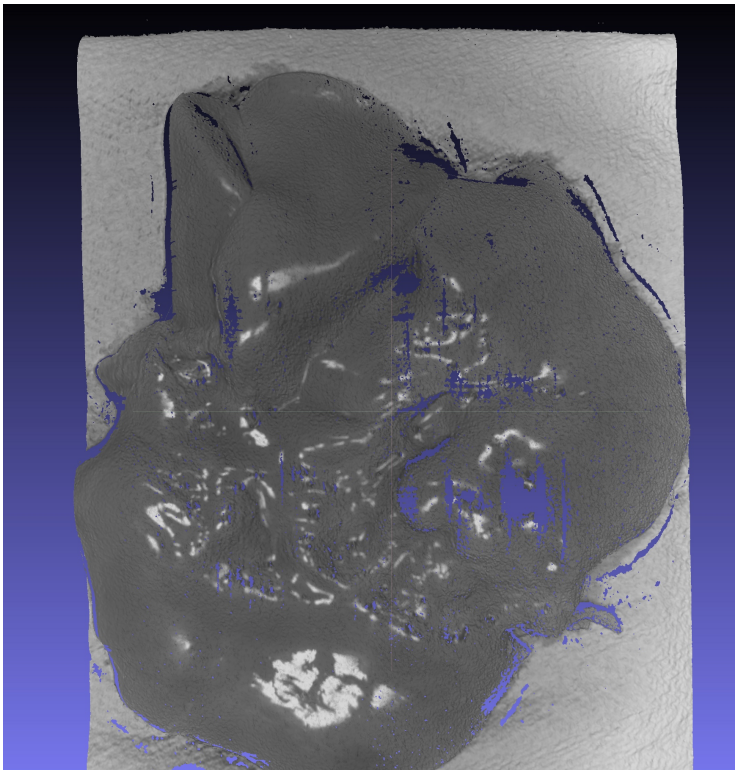


**Figure 3.12**: A frame from the 3D video scans taken from the RealSense. As is visibly demonstrable, the point density is much less uniform than the NextEngine scanner. The point density is very good in many areas, but the scanner does not handle the glossy surface as well as the NextEngine scanner. The glossy surface of the tissue causes some recorded surface noise in those glossy areas as well, of a greater amount than the NextEngine scanner does.

## 3.3 Surface Reconstruction

Because of the glossy surface of the tissue and other noise, some scans exhibit areas of missing data. Some registration algorithms may need a volumetric mesh, or at least a watertight surface mesh. We explored various options to reconstruct the surface using existing available tools. We found that the Poisson surface reconstruction in Meshlab will do a good job of reconstructing the surface. However, we were unable to quantify the accuracy of the reconstruction. An example of the reconstruction is shown in Figure 3.13.

Some versions of meshlab reconstruct the liver mesh with a large bulbous base, instead of just a flat base where the surface data ends. Reconstructing deformable surfaces with missing

data is difficult to do precisely. Surface reconstruction may be useful for development of AR algorithms which consider a volume displacement, but we would still face the same issue of unknown accuracy of reconstruction. These reconstructed meshes were registered using the algorithm presented in [10], producing visually satisfactory results.

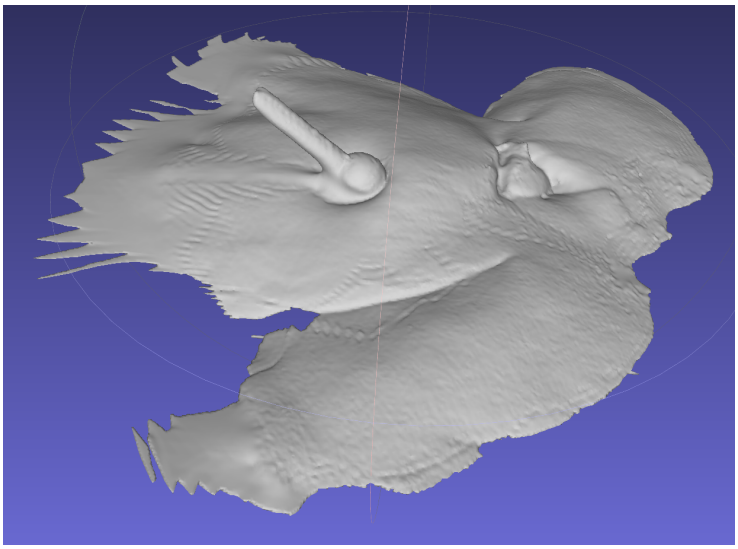


**Figure 3.13**: The result of a liver scan after performing a Poisson surface reconstruction. The result is watertight, but the accuracy of the reconstructed areas cannot be verified.

## 3.4 Summary

We present a method for creating a 3D video dataset that can be used for surgical AR algorithm development. The novelty of this contribution lies in the "3D video". No other existing datasets capture 3D surface deformation of tissue at frame rate. We improve upon our work by taking advantage of new scanner technologies that reduces the scan acquisition time, significantly decreasing the amount of time a new dataset takes to create. Datasets need to be specific to the procedure in order to support accurate development - our contribution eases the effort required to develop AR algorithms for a particular procedure. Lowering the development effort and barrier to entry will hopefully promote further work in this area.

Chapter 3, in part, is being prepared for submission for publication of the material. Michael Barrow, Shanglei Liu, Nelson Ho, Sonia Ramamoorthy, Santiago Horgan, Quentin Gautier, Peter Tueller, Ryan Kastner. "3D Video MISAR: Minimally Invasive Surgery Augmented Reality Ground Truth Data Set". The thesis author was a co-author of this paper.

# Chapter 4

# Segmentation

Many use cases of the datasets we describe in Chapter 3 require first segmenting an area of interest from the surgical scene prior to the next stage of the pipeline. For nonrigid soft tissue registration, we want to segment the area we want to register because oftentimes we do not desire to register the entire scene. A surface matching can be from frame to frame or from pre-operative model to the current frame.

## 4.1 Segmenting area of interest from background

As shown in the previous section, the scans we acquire often contain objects in the background that are part of the setup or are just areas of non-interest. These areas should be removed prior to evaluating any algorithms on the data, because the objects in scene will affect the quality of the registration. If the algorithm development is targeted toward a specific application, we want to be sure we remove portions of the scene that do not reflect the real environment of the application.

This sort of clean-up segmentation is performed by scripting Meshlab to crop areas of a series of scans. We performed a plane cut in the coordinate space to rid the scene of areas of non-interest. The results are shown below. While this method may not be relevant to in-vivo

surgical image segmentation, it enabled us to clean up our dataset to be more representative of an actual surgical scene. Figure 4.1 shows segmentation of the area of interest.

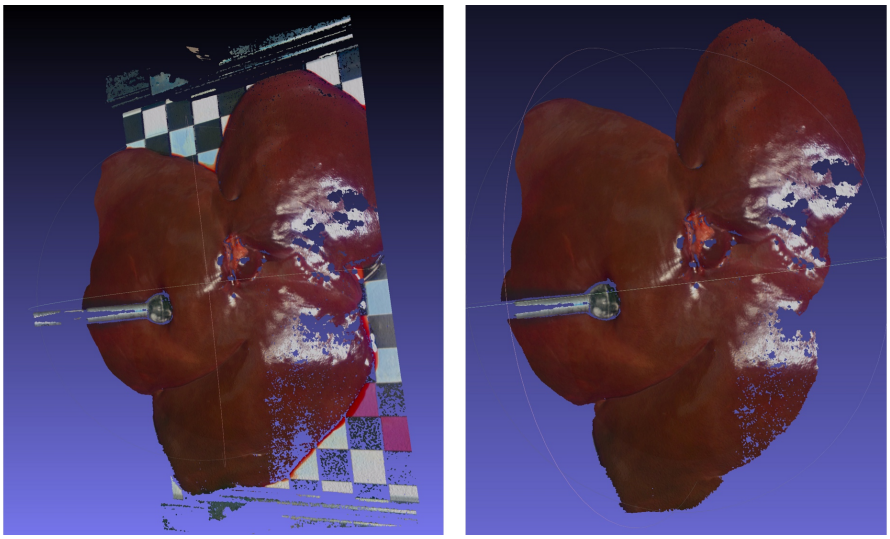


**Figure 4.1**: This figure shows the result of segmenting the background from the dataset we collected using the NextEngine scanner. As is shown in the first image, the colored calibration checkerboard and the slide rails of the scanner carriage are areas that we do not want to evaluate with respect to our algorithmic development.

We demonstrate this background segmentation on the dataset that we created using the stop-motion technique, in a non-surgical environment. This background segmentation may be necessary for other data capture techniques as well. For example, the dataset generated with the RealSense camera was done during a practice open surgery on a pig. This is a scenario which may occur during an actual surgery, though different segmentation methods would likely be employed during an actual surgery, rather than the method we mentioned here. The RealSense captured the dermis of the pig, which has different biomechanical properties from the organ tissue we are interested in, and so should be segmented from the organs. Real-time automated segmentation of soft tissue is still an open research problem, but is outside the scope of this work.

## 4.2 Segmenting Surgical Tools from Soft Tissue

The liver dataset we collected using the stop-motion method needed to be segmented prior to running a nonrigid registration on the tissue. Depending on the registration algorithm used, the rigid transformations from frame to frame would sometimes introduce compounded noise into the resulting global registration. To resolve this, we had to segment the rigid from the nonrigid portions prior to registering the dataset. The workflow diverges into two streams at this point, with the rigid portions of the scene being tracked separately from the nonrigid portions. The results will be merged back into the same scene after the surface transform is computed.

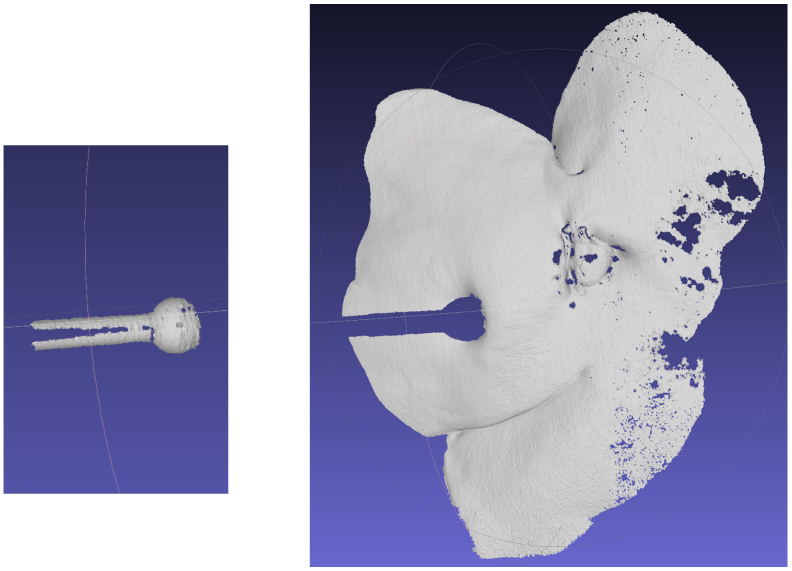


**Figure 4.2**: The result of performing the tool segmentation on the dataset we collected using the NextEngine scanner. The tool and liver are separated into rigid and nonrigid processing flows, because some nonrigid registration algorithms are sensitive to the rigid movement of the tool from frame to frame.

Using meshlab, we are able to segment out the tool from the liver in each of the frames, but that requires a lot of manual effort because it has to be done frame by frame, and would not be scalable to a real-time solution. We want to develop a way to do this automatically. This segmentation is relevant to a clinical setting, because there is a need to perform this tissue - tool segmentation in the AR pipeline; the problem with rigid transformations in nonrigid registration

is not mitigated in an actual surgical environment.

We decided to use a rigid registration algorithm to identify the tool which we want to segment from the nonrigid scene. Because the tools to be used are known prior to the start of the surgery, 3D models of the tools may be generated. A rough manual alignment of the tool model to the current surgical frame may be necessary for the sparse ICP algorithm to work. [8] This initial alignment can be done by the operator. Successive alignment is guaranteed by the previous frame, under the assumption that the tools in the scene won't move significantly from frame to frame at 25 FPS. Those models can then be registered to the image collected from the surface of the surgical site, to match the tools that are present in the scene. The registration will provide a transformation of the tool model to fit the tool as it is observed in the surgical scene. The procedure for a single frame segmentation is shown in Algorithm 1.

**Algorithm 1** Tool Segmentation

```
procedure SEGMENTTOOL(toolCloud, sceneCloud)
    pointsToRemove, set of tool points to remove from the scene
    pointsRadiusSearch, vector returned from point radius search
    PCLoctree, octree used for point radius search
    alignedToolCloud ← SPARSEICP(toolCloud, sceneCloud)
    PCLoctree.SETINPUTCLOUD(sceneCloud)
    PCLoctree.ADDPOINTSFROMINPUTCLOUD()
    radius ← 1.95f
    for each point in alignedToolCloud do
        PCLoctree.RADIUSSEARCH(point, radius, pointsRadiusSearch)
        for each neighborPoint in pointsRadiusSearch do
            pointsToRemove.INSERT(neighborPoint)
        end for
    end for
    tool, liver ← EXTRACTINDICES(sceneCloud, pointsToRemove)
end procedure
```

Following the rigid registration, performing a radius search around the points in the tool model can identify the points in the scan of the surgical site that belong to the tool. The tissue and the tool can then be separated into separate processing streams, where the tissue will undergo a nonrigid registration, and the tool will be registered in a rigid fashion. Results of the segmentation

are shown in Figure 4.2. Once the first frame is segmented against the tool model, subsequent frames can be segmented using the result of the first frame. Since the tool segmented from the previous frame is already roughly aligned in the scene and the tool does not exhibit a huge amount of displacement from frame to frame due to its relatively slow movement, the sparse ICP alignment is more likely to succeed in the succeeding frame.

## 4.3 Summary

The segmentation methods discussed in this chapter are not universal. Depending on the AR algorithm under test, the user may wish to segment the dataset in some other way to more closely approximate the real application. The methods here may provide inspiration for other offline segmentation methods for post-processing of collected datasets. This work does not discuss online segmentation, which remains an open research problem.

# Chapter 5

# 3D Registration

Registration performs a point to point transformation of a 3D model in one orientation and perspective to another orientation or perspective. It is used to specify the transformation of an object from one frame to the next in the AR pipeline. Rigid registration algorithms perform a simple transformation, applied to every point in the point cloud. Non-rigid registration algorithms apply non-uniform transformations to different points and regions in the point cloud, with the goal of deforming it to closely match the registration target.

Registration is a necessary step in the AR pipeline required for surgical guidance. Different surgeries and procedures have different characteristics, and so likely need different heuristics to perform optimally for a given scenario. The specific surgical scenario can be modeled by the dataset used for evaluation. This chapter evaluates several registration algorithms and provides a metric for objective evaluation of registration quality.

## 5.1 Early Evaluation of Registration Algorithms

After collecting the stop-motion dataset described in section 3.1, we decided to perform some preliminary evaluations of several well-known registration algorithms on our dataset. We evaluate rigid registration as well as non-rigid registration, mainly to highlight the accuracy versus

runtime tradeoffs. We evaluate three different Iterative Closest Point (ICP) based algorithms, and one Coherent Point Drift (CPD) algorithm. The results are shown in Table 5.1 and Figure 5.1.

**Table 5.1**: A table comparing best and worst features of several common registration techniques on our dataset. Sparse and Simple ICP are rigid registration algorithms.

| **Algorithm** | **Best Feature** | **Worst Feature** |
|---|---|---|
| Sparse ICP | Error Resilience | Inaccurate |
| Simple ICP | Fast | Inaccurate |
| Non-rigid ICP | Error Resilience | Slow |
| Non-rigid CPD | Accurate | Slow |

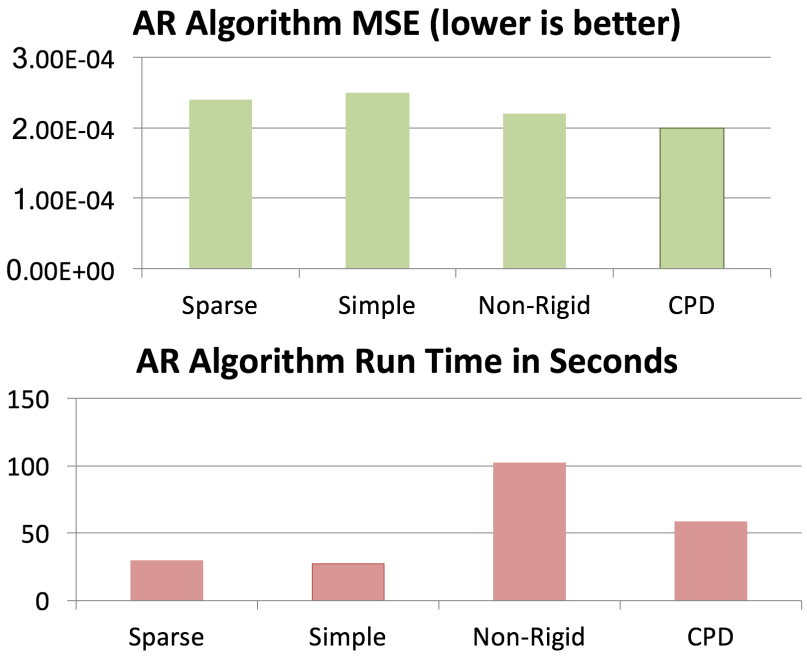


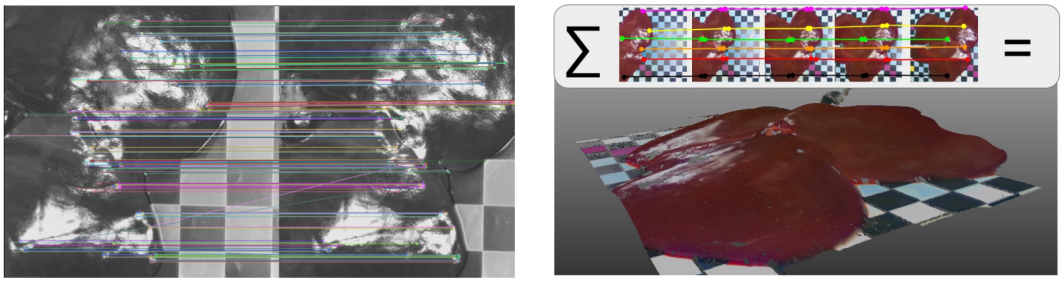

**Figure 5.1**: These two graphs depict the tradeoff between speed and accuracy inherent between rigid and nonrigid registrations. We observe runtime of rigid ICP variants is very close, but nonrigid registration methods take significantly longer. Nonrigid methods achieve higher accuracy than rigid methods, as is expected in this case.

This preliminary evaluation affirms the usefulness of the dataset we created in evaluating registration algorithms, and motivates our work on accelerating nonrigid registration algorithms. Because accuracy is an important feature in the surface transform step of the AR pipeline, an

accurate nonrigid registration is what we want to use to measure surface deformation in tissue. Speed is also an important factor, as we want to perform this registration at frame-rate in order to support real-time AR guidance.

## 5.2 Global Registration Baseline Model

We obtain a global registration model of our dataset to use as a baseline for evaluation, using the method presented in *Global Non-Rigid Alignment of 3-D Scans* [9]. A "global registration" is obtained by registering all the frames of a dataset to one another. In our particular case, we have chosen to use Thin Plate Spline (TPS) interpolation to compute this model. We use this method for its stability, and because it offers a smooth bijective mapping from the original scan to the global model. Figure 5.2 provides a graphical illustration of the difference between a single registration and a global registration.




(a) Single registration (b) Global registration

**Figure 5.2**: This image depicts the difference between a single registration and a global registration. In the single registration, one scan is registered to another. In the global registration, all the frames of the dataset are registered to one another to produce the global baseline model.

We had to modify the code provided by [9] to produce a global model for our dataset. The original code did not produce a global output of the registration result of all scans in the sequence. We took the driver code for the TPS warp and appended the transformed points of each input scan to a final output PLY file. The resulting file contains all the inputs warped to fit the surface to each other.

Additionally, we wrote utilities to identify the mapping between source scan to global model. The utility, if passed a point in the global model, will produce the source scan and the point in the source scan that the point came from. If passed a point in a source scan, it gives the point in the global model. This mapping was necessary for the evaluation metric we developed.

## 5.3 Evaluating TRW-S Geodesic Distance based Registration

We run the TRW-S based nonrigid registration presented in [10] on the datasets we collected using the method presented in section 3.1. This registration code presented many issues for our dataset, and we spent a significant amount of time debugging. Because Chen's code was written for a different dataset, specifically the FAUST dataset [7], assumptions were made for this dataset that did not hold with ours. The documentation in the code was poor, so this contributed to the lengthy amount of time spent debugging this code. Chen's registration code provides a nonrigid source to target point mapping of a subset of the input points, and does not actually perform any transformation on the points. Additionally, through the documentation in the code, we discovered that the input meshes need to be watertight volumetric meshes.

We used a Poisson reconstruction in Meshlab to reconstruct the missing portions of the surfaces of the scans. This Meshlab filter also constructed a base for the liver model, making it volumetric. However, by the time this reconstruction was done, the size of the mesh had increased significantly. Calculating the geodesic distances of the mesh consumed all the memory we had available on our development server, and never completed. We attempted to decimate the mesh, but to no avail.

The solution to the problem turned out to be the scale of the meshes; the registration code expected the input meshes to be of a much smaller scale than the meshes we collected for our dataset. In fact, the scale was off by an enormous amount - we had to scale down the dimensions of our dataset by a factor of 500 in order for this registration code to run. After we discovered

this, running the registration on non-reconstructed meshes worked as well. The runtime is $O(n^4)$, so as the meshes got denser, the runtime grew by a polynomial factor. To make the algorithm run in a tractable amount of time, we had to decimate the mesh. We found that, for the stop-motion liver pan and prod dataset, about 10000 vertices and 20000 faces was the minimum we needed to produce a visibly accurate representation of the liver surface. We do not yet know whether such a reduction in resolution will present problems further down the pipeline.

## 5.4 Naive attempt at accelerating TRW-S Registration

We made a naive attempt at accelerating the TRW-S registration code on an FPGA. We profiled the code and determined where the bottleneck is. This algorithm is state of art in non-rigid registration in terms of accuracy and speed, so we felt it was a good candidate to begin exploration of nonrigid registration acceleration.

There exists a hot loop in the TRW-S code that updated messages as part of the message passing algorithm. The code already had OpenMP pragmas surrounding this section of code. We implemented the function in OpenCL and used Altera OpenCL tools to put the design on an Altera DE5-Net. We chose this device and toolchain based on an analysis summarized by Table 5.2. While OpenCL would likely not produce an optimal design on the FPGA nor would it produce the best results on an Nvidia GPU, it would be the most time efficient way of evaluating all platforms. At the time, we had plans for an accelerator evaluation framework, but fell short of producing a usable tool.

We realized a slow-down of about 100x. We dug deeper into the code, and realized that the working set of the problem was about 8GB. The memory on the FPGA device was only 4GB - we had to enable the FPGA to share main memory for its working input. Sharing memory over the PCI-E bus was really costly, which was the main cause for the extreme slow down.

**Table 5.2**: This table compares the advantages and disadvantages of several available toolchains and devices for developing an FPGA accelerator. We decided to use the Altera FPGA toolchain.

| **Platform** | **OpenCL** | **CUDA** | **HDL** |
|---|---|---|---|
| CPU vector units | ✓ | × | × |
| Nvidia GPU | ✓ | ✓ | × |
| AMD GPU | ✓ | ✓ | × |
| Xilinx FPGA | × | × | ✓ |
| Altera FPGA | ✓ | ✓ | ✓ |

Naturally we investigated an appropriate approach for partitioning the input and batch transfer of the input to the FPGA, but we then realized that the memory access is not linear. In fact, it is highly irregular, and the input gets permuted several times in the algorithm. At this point we realized we would have to get clever, and somehow reduce the memory bandwidth requirements. The bus latency on the Intel HARP V2 platform, where the CPU and FPGA are connected via QPI represents great reduction to this cost, but this is clearly the bottleneck of the problem. The input size for the solver used in the registration is generated from the geodesic distances of the input, which is much smaller. The geodesic input is first subsampled, then a series of permutations of that subsampling is generated causing the size to grow exponentially. If there is a way to generate the input from the geodesic distances on the device side, rather than on the host CPU, the memory bottleneck may be alleviated.

## 5.5 SIRGn Evaluation Metric

The viability of Augmented Reality guidance in robotic MIS surgery is limited by the performance of the algorithms required for guidance. There are currently no metrics that exist to evaluate the quality of nonrigid registration in a surgical guidance context. We propose a metric here that works for many different types of 3D registration methods, and provides fine grained registration quality results.

Our evaluation metric does not provide a single quality evaluation for the entire registra-

tion, but rather divides up the registration into regions of interest, and highlights areas according to the quality of the registration. This is useful because there may be areas within the evaluation dataset that produce consistently poor results, which may have to do with some surface feature of that region. As an example, iterative modifications to an algorithm can be evaluated as improving or worsening registration for a particular area within the scene that exhibits some specific type of deformation, while not exhibiting regression in other regions.

The SIRGn metric requires, as input, known-good expert defined landmarks in the dataset used for evaluation. This set of landmarks is identified in every frame of the input data. SIRGn operates on the notion that points that exhibit a high level of deviation from known good points are less likely to be accurately registered. Because our target application is real-time frame-rate registration on deformable tissue, it is unlikely that a point will be transformed significantly more or differently than its spatially or temporally neighboring points between frames in a 3D video. The viscoelastic nature of many types of human tissue, and the lack of rigid plastic surface deformation during most types of manipulation of the tissue during surgery makes this a reasonable assumption.

**Global Registration and Warp Function**

The first step to use SIRGn is to generate a global registration model $M$ using the registration algorithm under test by registering the set of all the frames of the dataset, $V$, to each other. The set of expert-defined landmarks, $G$, may or may not be used by the registration algorithm under test, but must be provided along with the global registration model to SIRGn. The same landmarks are identified in each frame of the 3D video, and together make the set $G$.

SIRGn then has to calculate the warp function that defines the warp from each frame of the dataset to the global model. Let us call this function $\varphi$. We calculate $\varphi$ using Thin Plate Spline (TPS) interpolation from [9]. This method guarantees $\varphi$ to be a smooth bijective function. Using this function we can find the pairing from a point in the global model to a point in an original

input frame and vice versa.

### Landmark Distance Vectors

The next step is to generate the landmark distance vectors. We generate a set of vectors, one for each point $g \in G$ to describe its transformation to its corresponding expert defined point in $M$, called $g'$. The result is a set of vectors from each of the expert defined landmarks in each frame $v_i \in V$ to the landmarks $g' \in M$. We use these vectors to evaluate the quality of the registration of points nearby, with the assumption that the distance vector is similar for nearby unknown points. The most heavily warped points are points of interest in this region, because they are more likely to be points that were registered incorrectly due to some feature of that scan.

The boundaries of the regions of interest are defined by a Delaunay triangulation over the points $G'$, the set of landmarks $g'$ in $M$. Given $G'$, we create a triangular mesh from the points using Delaunay triangulation. Each of the triangles is now a region of interest, and SIRGn will report a registration quality result over this triangle by color coding the triangle overlay.

### Warping Distance Vectors

For each of the triangles in the mesh created from the Delaunay triangulation, SIRGn finds the most heavily warped point in the interior of the triangle using $\varphi$ and gradient ascent. A set of worst-case warps for each of the triangles for each frame of the 3D video is generated by this method, called $C$. The most heavily warped point in a triangle is identified as the point that has undergone a transformation that differs the most from the way the expert defined landmarks have transformed in that scene. A point that has been transformed differently from the expert landmarks nearby is more likely to not have been registered correctly. The expert defined landmarks are the vertices of the triangular region of interest.

### Correlation Mesh

The correlation mesh faces are given values representing the correlation between the heavily warped points and the average of the all the vectors in the region of interest. If the variance in the region of interest is high, the result may warrant a closer inspection. If the most heavily warped point covaries with the rest of the points in the region, then it is more likely a good registration. If the most heavily warped point exhibits high variance and low covariance with the mean, it is very likely the region, or some points in the region, has been registered poorly. By considering the warp vector for each of the points in the region across several 3D video frames, we achieve a dense metric. By evaluating the quality of registration by region of interest instead of over the whole scene, we have accomplished a granular metric. Together, these qualities represent a novel method of evaluating the quality of nonrigid registration.

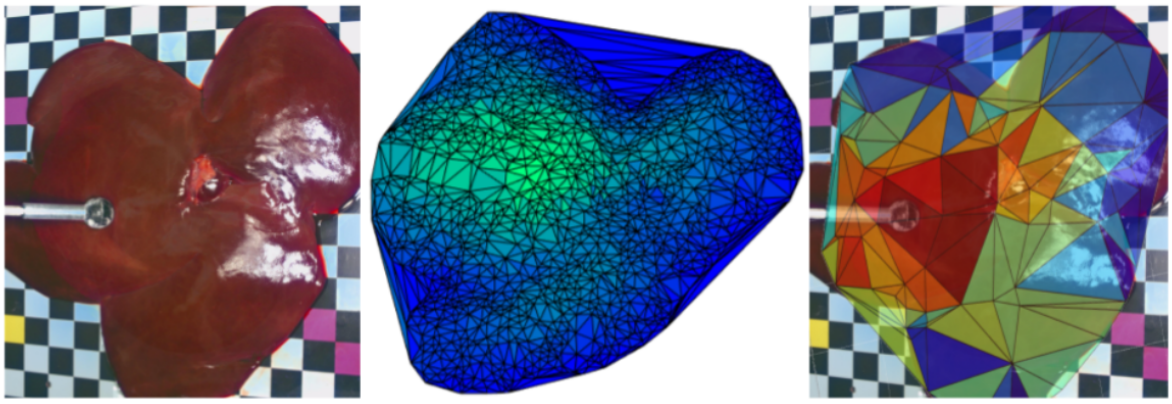



**Figure 5.3**: This series of photos shows the output of running SIRGn on our dataset. Figure 5.3a shows a frame from the video dataset we used for evaluation. Figure 5.3b shows the registration under test. Figure 5.3c shows the heat map covariance mesh overlay, where higher heat (warmer colors) indicates lower correlation and thus poorer registration quality.

## 5.6 Experiments using SIRGn

We use SIRGn to evaluate two different non-rigid registration algorithms, and we compare their SIRGn results to provide a concrete way to use this metric to evaluate registration quality. We use as input a 3D video dataset collected by the stop motion dataset collection procedure detailed in Chapter 3. The particular dataset we used was a simple 25 frame sequence of palpation of the surface of a porcine liver with a spherical tool. The two registration algorithms both take as input the series of 3D video frames and output a globally registered model. The quality overlay mesh described in the previous section is generated for each of these registrations under test, which enables us to see how each of the registration algorithms performs with certain surface deformations exhibited by the dataset. This information allows us to determine which registration algorithm may perform better in areas critical to a specific procedure.

We evaluate two different approaches to nonrigid registration using SIRGn. The first is a geodesic distance based registration, the second is an iterative closest point based registration. SIRGn reports lower variance for the geodesic distance based registration than for the nonrigid ICP. This can be observed in Figure 5.4.

Based just on the graphs in Figure 5.4a and Figure 5.4d, we may come to the conclusion that nonrigid ICP performs better than the geodesic distance based registration. However, looking at the Figure B and E in the image, we get a different story of what happened. In the $G$ to $C$ projection for nonrigid ICP, we see relatively light coverage of the surface, indicating that many points may be in the exact same location. ICP's locally weighted transform pulls points into clusters, and eventually, a local singularity. Additionally, the 3D covariance mesh suggests poorer performance of the nonrigid ICP at the edges of the 3D data. We surmise that this is caused by local weighting of the ICP, with a radius that extends into empty space at the scene edge. These findings show that an ICP based nonrigid registration may not be suitable for this type of surgical guidance. We can also note that the variance of the $G$ points shown in Figure 5.1d is quite high,

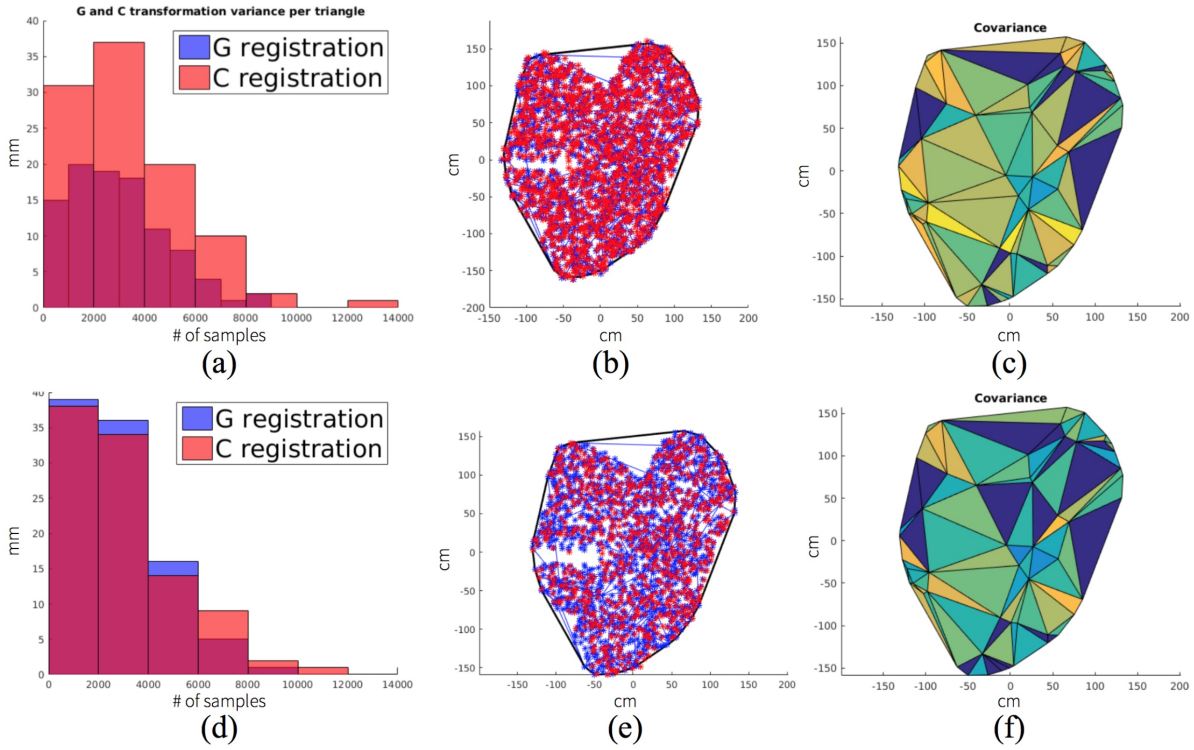



**Figure 5.4**: This series of charts show the output of SIRGn run on two different nonrigid registration algorithms. The top row shows the results for a geodesic distance based registration. The bottom row shows the results for an ICP based registration. Figure 5.4a and 5.4d show histogram plots of the variance of G and C registration. Figure 5.4b and 5.4e are 2D projections of $G$ points (blue) onto $M$. Figure 5.4c and 5.4f show the covariance mesh overlay, with higher intensity color indicating higher covariance.

indicating poor registration, because these points are supposed to be confidently tracked and should have low variance.

The SIRGn results for the geodesic distance based registration also provides some insight into the weaknesses of such an approach. As can be seen in Figure 5.4a, the variance reported for this method is fairly high when comparing the transformation of expert defined landmarks to the transformation of other points. The nonrigid ICP reports roughly equal variance for the G and C points, but this may be due to the clustering phenomenon noted above. We can conclude from the high variance that the geodesic distance based approach is more sensitive to feature selection than the nonrigid ICP, and will need to have features carefully selected to correctly represent the

movement of the surrounding area.

## 5.7 Summary

From this analysis, we conclude that all of the evaluated aspects and outputs of SIRGn combine to form a comprehensive metric. The results taken individually may not give the whole picture, but together they give a holistic interpretation of the registration quality. Existing registration evaluation metrics do not work with markerless datasets and methods such as ours. This metric is agnostic to the registration approach, and provides a quality measure that indicates registration quality by region rather than a single measure across the entire scene.

Chapter 5, section 1, in part, contains material as it appears in a poster presented at the Scientific Session, Society of American Gastrointestinal and Endoscopic Surgeons, 2017. Michael Barrow, Shanglei Liu, Nelson Ho, Xinyi Yang, Brendon Chen, Santiago Horgan and Sonia Ramamoorthy. "A ground truth 3D video dataset for Augmented Reality Robotic MIS algorithms". The thesis author was a co-author of this paper.

Chapter 5, sections 5 and 6, in part, have been submitted for publication of the material as it may appear in Medical Image Computing and Computer Assisted Intervention, 2018. Michael Barrow, Nelson Ho, Alric Althoff, Ryan Kastner. "Benchmarking Video With The Surgical Image Registration Generator (SIRGn) Baseline". The thesis author was a co-author of this paper.

# Chapter 6

# Conclusion

Augmented reality has huge potential in enhancing surgical guidance in future surgical environments. Minimally invasive surgical techniques and robotic assistance in surgery are the way of the future because of the benefits they offer in terms of patient care and recovery time. We identified several areas in the field of surgical AR guidance and made meaningful contributions to these areas.

Existing datasets are insufficient for developing real-time AR image guidance. The existing datasets lack physical information about the surgical site that is necessary to determine the applied strain to an object that correlates with its exhibited surface deformation. Without this information, we cannot estimate the movement of the internal structure of the tissue, and therefore cannot cannot guide the surgeon as they cut into the tissue.

A crucial step in the AR image guidance pipeline is to be able to quickly and accurately capture the surface deformation from one frame to the next, in order to track how the surface of the organ changes over time. This surface deformation information can be mapped to information compiled offline to determine the amount of force being applied by a given surgical tool to create that amount of deformation. Once the force acting on the surface of the organ is known, a physics simulation can be performed to determine how the structure under the surface deformed due to

the applied surface strain. Then the pre-operative imaging can be deformed to match the surgeons field of view, then overlaid into the surgeons field of view. However, if the surface deformation cannot be accurately obtained, then the force estimation will be inaccurate, and as a result the guidance will be inaccurate. Obtaining an accurate idea of how the surface deforms is crucial to the accuracy of AR guidance.

Existing evaluation frameworks do not objectively evaluate the quality of nonrigid registration algorithms. Given that our application depends so heavily on the accuracy of nonrigid registration, we had to have a way to determine the quality of a given registration or registration algorithm. We devised a novel global metric based on expert defined landmarks and point transform variance that allows us to both quantitatively and qualitatively evaluate a given nonrigid registration for a particular dataset.

We also provide a method for creating datasets that are accurate and representative of specific surgical procedures. This, combined with our registration evaluation framework, enables evaluation of various registration algorithms for any variety of target surgical procedures. Because different surgical procedures present very different computational and visual challenges and AR algorithms have to be tailored to these idiosyncrasies, having a workflow and metric that can take these factors into account and provide an objective evaluation proves invaluable to researchers that want to contribute to this field.

## 6.1 Future Work

This work does not address a crucial component in the computational pipeline required for AR surgical guidance: the physics simulation that estimates the internal deformation of an organ based on some observed amount of surface deformation. This is likely the most computationally intensive portion of the pipeline, and research exists in attempts to accelerate this computation to be suitable for a real-time application. Berkley et al. provide a real-time solution in [6], albeit

with a different target application and with computational relaxations made under the assumptions of the application. A revisitation with modern heterogeneous architectures may yield interesting results. An attempt at GPU acceleration of Finite Element Method (FEM), a common method to analyze structures and material deformation, is presented in [15][21]. Surgical simulation needs to solve biomechanical modeling problems similar to those that need to be solved in Augmented Reality. FEM is the method of choice when modeling biomechanical properties in simulation, and seems from the literature to be a good candidate for real-time approximation as well.

Augmented Reality presents a great opportunity for the advancement of surgical techniques and accuracy, leading to better patient care, better surgical results, and faster recovery times. Being able to use augmented reality for guidance in real time during surgery necessitates performing several very difficult computational problems in a small amount of time. One of those difficult computational problems is nonrigid registration. Existing techniques for nonrigid registration are not fast enough, so we need to develop algorithms that are fast enough. This turns out to be more difficult than expected because of the lack of available data to test such an algorithm on, as well as a lack of metric to be able to determine the quality of the result of said algorithm. In this thesis I have proposed solutions to each of those problems, in the context of the nonrigid registration step that is required in the AR guidance pipeline. It is my hope that this research will promote and ease continued development in the area of surgical AR guidance, because I believe it has the potential to benefit many.

# Bibliography


[1] *Benefits of Robotic Surgery*.

[2] *GIBLIB*, Accessed April 21, 2018.

[3] Michael J Ackerman. The visible human project. *Proceedings of the IEEE*, 86(3):504–511, 1998.

[4] Brian Amberg, Sami Romdhani, and Thomas Vetter. Optimal step nonrigid icp algorithms for surface registration. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.

[5] Nicholas Baer. *Advisory Board: what 2015 holds for Da Vinci*, Accessed January 3, 2017.

[6] Jeffrey Berkley, George Turkiyyah, Daniel Berg, Mark Ganter, and Suzanne Weghorst. Real-time finite element modeling for surgery simulation: An application to virtual suturing. *IEEE Transactions on visualization and computer graphics*, 10(3):314–325, 2004.

[7] Federica Bogo, Javier Romero, Matthew Loper, and Michael J Black. Faust: Dataset and evaluation for 3d mesh registration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3794–3801, 2014.

[8] Sofien Bouaziz, Andrea Tagliasacchi, and Mark Pauly. Sparse iterative closest point. In *Computer graphics forum*, volume 32, pages 113–123. Wiley Online Library, 2013.

[9] Benedict J Brown and Szymon Rusinkiewicz. Global non-rigid alignment of 3-d scans. In *ACM Transactions on Graphics (TOG)*, volume 26, page 21. ACM, 2007.

[10] Qifeng Chen and Vladlen Koltun. Robust nonrigid registration by convex optimization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2039–2047, 2015.

[11] Kenneth Clark, Bruce Vendt, Kirk Smith, John Freymann, Justin Kirby, Paul Koppel, Stephen Moore, Stanley Phillips, David Maffitt, Michael Pringle, et al. The cancer imaging archive (tcia): maintaining and operating a public information repository. *Journal of digital imaging*, 26(6):1045–1057, 2013.

[12] Wen Wei Gerard Ee, Wen Liang Joel Lau, William Yeo, Yap Von Bing, and Wai Mun Yue. Does minimally invasive surgery have a lower risk of surgical site infections compared with open spinal surgery? *Clinical Orthopaedics and Related Research®*, 472(6):1718–1724, 2014.

[13] J Michael Fitzpatrick, Jay B West, and Calvin R Maurer. Predicting error in rigid-body point-based registration. *IEEE transactions on medical imaging*, 17(5):694–702, 1998.

[14] Richard NJ Graham, RW Perriss, and Andrew F Scarsbrook. Dicom demystified: a review of digital file formats and their use in radiological practice. *Clinical radiology*, 60(11):1133–1140, 2005.

[15] Grand Roman Joldes, Adam Wittek, and Karol Miller. Real-time nonlinear finite element computations on gpu–application to neurosurgical simulation. *Computer methods in applied mechanics and engineering*, 199(49-52):3305–3314, 2010.

[16] Wee Sim Khor, Benjamin Baker, Kavit Amin, Adrian Chan, Ketan Patel, and Jason Wong. Augmented and virtual reality in surgerythe digital surgical environment: applications, limitations and legal pitfalls. *Annals of translational medicine*, 4(23), 2016.

[17] JB Antoine Maintz and Max A Viergever. A survey of medical image registration. *Medical image analysis*, 2(1):1–36, 1998.

[18] John Mayo, Nakul Valsangkar, Neha Lad, Leonidas Koniaris, Adam Wilson, et al. *Surgical Procedure PORTal: Hepatic video*, Accessed November 3, 2016.

[19] MJ Rosen and JL Ponsky. Minimally invasive surgery, 2004-2005. *Endoscopy*, 38(02):137–143, 2006.

[20] Ivan E Sutherland. The ultimate display. *Multimedia: From Wagner to virtual reality*, pages 506–508, 1965.

[21] Zeike A Taylor, Mario Cheng, and Sébastien Ourselin. High-speed nonlinear finite element analysis for surgical simulation using graphics processing units. *IEEE transactions on medical imaging*, 27(5):650–663, 2008.

[22] Chaohui Wang, Michael M Bronstein, Alexander M Bronstein, and Nikos Paragios. Discrete minimum distortion correspondence problems for non-rigid shape matching. In *International Conference on Scale Space and Variational Methods in Computer Vision*, pages 580–591. Springer, 2011.

[23] Junchen Wang, Hideyuki Suenaga, Liangjing Yang, Etsuko Kobayashi, and Ichiro Sakuma. Video see-through augmented reality for oral and maxillofacial surgery. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 13(2), 2017.

[24] Jay B West, J Michael Fitzpatrick, Steven A Toms, Calvin R Maurer Jr, and Robert J Maciunas. Fiducial point placement and the accuracy of point-based, rigid body registration. *Neurosurgery*, 48(4):810–817, 2001.

[25] Tristian Yan et al. *AME Surgical Video Database: Hepatic video*, Accessed November 3, 2016.

[26] X-R Yu, W-Y Huang, B-Y Zhang, H-Q Li, and D-Y Geng. Differentiation of infiltrative cholangiocarcinoma from benign common bile duct stricture using three-dimensional dynamic contrast-enhanced mri with mrcp. *Clinical radiology*, 69(6):567–573, 2014.