X-Ray Vision Project

University of California, San Diego
CSE 145/237D Winter 2018 Final Project
Nelson Ho, Xiangyu Bi, Canzhen Zhou, Vincent Tierra, Trevor McCleery, Jessy Ayala

Overview

Augmented Reality (AR) presents many opportunities for advancement in the medical research community for patient specific treatment. One of the many clinical applications of AR is in Minimally Invasive Surgery (MIS). AR enables the possibility of superimposing preoperative imaging onto the surgical environment in vivo in order to provide guidance to the surgeon during surgery. In order to advance research in this area, we need “ground truth 3D data”. This ground truth data will allow developers to have a ground truth point to point mapping in the tissue surface from frame to frame. This will encourage further development of AR algorithms in this space, because developers will be able to evaluate the accuracy of their implementation relative to a “gold standard”.

Introduction

Almost 50 million surgical procedures are performed each year, as of 2009. Surgical errors occur more than 4000 times per year. Augmented reality presents a real benefit to surgeries by offering the ability to present surgeons with more information about the state of the surgical site by overlaying pre-operative imaging onto the live surgical environment in real-time. There are several technical challenges that prevent AR from being used in its full capacity in surgical environments. Our project aims to promote and ease development of these algorithms to enable the use of AR in real-time surgical scenarios. The algorithms that are needed to enable real-time AR guidance in surgery are difficult computational problems, and without a baseline to evaluate potential solutions, we cannot tell whether a new proposed solution is better or worse than the existing state-of-art.

The target procedure guiding our project is liver tumor removal. Without accurate pre-operative information and real-time guidance, the surgeon must do a lot of guesswork to determine where the areas of interest are during the procedure. If the surgeon cuts through a large artery in the liver during the procedure, there will be uncontrolled bleeding. If the surgeon doesn't have a clear idea where the malignancy is, a more than necessary amount of healthy tissue will be removed from the impacted area. AR has the potential to improve surgical accuracy and patient recovery time.

Our work focuses on developing a baseline to evaluate methods and algorithms related to surgical image guidance. Because there exist no similar benchmarks or baselines in the current literature, we hope that our contribution will promote and simplify development in the field of surgical AR. We are a long way off from clinically usable technology, so we hope that our contribution will accelerate the path that brings us to a future with better surgical outcomes and patient care through AR assistance.

Approach

The computational pipeline that is necessary to provide AR to the surgeon is depicted below. In order to provide an AR overlay of the preoperative imaging onto the surgical site in real time, a series of steps and associated computations need to be performed. First the depth map from the surgical site needs to be acquired or computed. We assume, with the current rate of progression of 3D depth map acquisition technology, the technology will have matured enough for this application in the next few years. We then have to compute the surface transform field in the initial step from the preoperative 3D surface to the surface captured from the surgical site. In subsequent steps we perform a frame to frame registration of the surface deformation. Because this surface registration is just a small part of the AR pipeline that needs to be able to compute at frame rate, it needs to be sped up significantly.

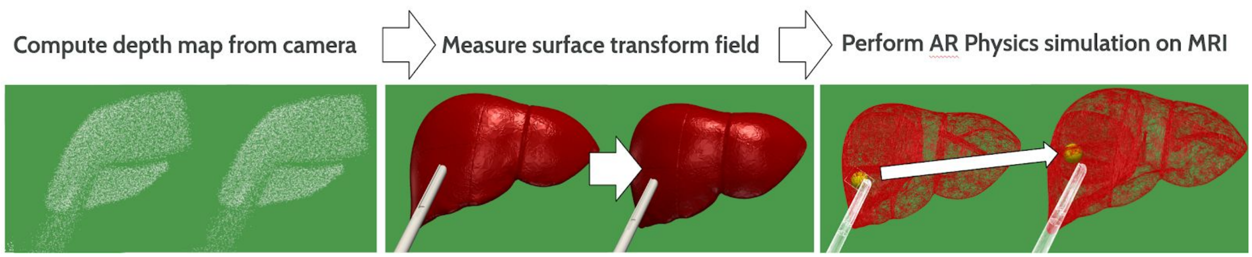

Figure 1.1 The pipeline necessary for AR assisted surgery - our focus is on the surface transform “registration” step

There is little existing research in this area because most of the algorithms developed for 3D nonrigid registration are for different applications, like pose estimation. Additionally, these other applications do not have real-time requirements. To accelerate the existing algorithms, we need a measure by which to evaluate the accuracy of our implementation with respect to the existing ones, to be able to characterize the tradeoffs we made in our design and to concretely describe the differences and tradeoffs of one implementation versus another. Currently, there exists no ground truth dataset for this sort of algorithmic development, which makes improvements in this arena difficult to quantify. By creating a ground truth dataset, development and evaluation of algorithms in this space will be much easier.

For a ground truth dataset, like the one we are proposing, to be effective and useful, it must be clinically relevant to the procedures (the application) that our technology targets. In collaborating with surgeons from the medical school, we have acquired a list of video sequences that are of both clinical interest and present challenging engineering scenarios. We then collect the dataset using procedures described below, and process them to reveal a set of ground truth registrations. Prior to visiting the surgery room, we tested the recording apparatus in the lab to get familiar with the tools and software. In the surgery room, we attached the recording apparatus to the surgical lights using the stage ninja action clamp to get a better angle and lighting of the liver. We did a total of 13 sets of recordings in one day with different angles, with and without the surgical tool to give us options to find the best recording to segment in the next step. The result includes a global 3D registration mesh that represents all the scans in a single video dataset registered to each other. We also provide a correspondence from each point in each original frame to a point in the global model - this is the ground truth correspondence. Having a ground truth correspondence allows this video dataset to be registered using different algorithms, which may output different correspondences from the ground truth. This difference can be used to quantify the accuracy of the algorithm.

Objectives and Deliverables

There will be three sets of deliverables. The first will be a set of 3D recorded liver procedures. The second will be a copy of the collected dataset with the rigid and nonrigid portions segmented from each other. The third will be a global registration for each the non-rigid and rigid segmented portions of the scans. This will enable AR researchers in the field to use the data we collected, run their own registration algorithms on the data, and evaluate the quality of their registration using the global baseline registration we provide.

Constraints, Risk, and Feasibility

We have never recorded liver data with a Realsense camera before. We have never used a Realsense camera before. We do not know if the collected data will be too noisy for the current algorithms we have. We do not know if the collected data will be of high enough resolution for our current workflow. We do not know that our segmentation and global registration workflow will work well for all types of collected data (for example, angle of the camera, location of the liver, whether the liver is dissected out or still in the host, etc).

Details of Development

During practice runs of the data collection where we scanned store-bought liver, we realized that there is a lot of noise with the realsense camera because of the light refracting across the wet glossy surface of the liver. We experimented with a variety of camera angles and determined that it was best to position the camera at a normal to the scan target. This reduced the amount of surface refraction and reduced the noise in the collected mesh.

Figure 2.1 This is the setup that during the practice runs of data collection where we scanned a store-bought liver, as you can see in the image above the setup looks pretty simple and mobile, but the mount is not a perfect fit.

In practice, when we went to the Center for the Future of Surgery (CFS), this technique worked out quite well. The quality of the scan surface was consistent and there were few holes in the collected point cloud. The surface is a little bit noisy in terms of depth, but we are hoping some post-processing can rectify this.

The setup in the Center for the Future of Surgery was pretty simple, however, due to the limitation of the tripod that was used in the practice run, we had to find another tripod to use at the CSF to accomplish this task. We utilized the tool shown in figure 2.2 (below), Stage Ninja Action Clamp, and utilized this feature and clamped the camera to the surgical lights to get a better angle and lighting of the pig's porcelain.

Figure 2.2 Stage Ninja Action was a perfect fit to get the best angle/position necessary to achieve the best recordings at CFS (image from amazon.com)

Due to the strict policy at the CFS, we were not able to get a photo of the setup, however, the same steps from the README (https://docs.google.com/document/d/1TiBEM7tzKxobayh0WAOByzgRmVQutL2lt0g98VVYyQk/edit **must have a ucsd email to view it**) still follows, and the only adjustment made was mounting the camera on one of the surgical lights.

In terms of segmenting the newly collected dataset, we did not need to use the segmentation codebase that was developed earlier this quarter. Because the scanner was oriented at a fixed angle and location, and the line of sight of the scanner was in the same axis as the normal, only a plane cut was necessary to properly segment the collected data. The portion below the area of interest was segmented with a plane cut in the Z axis, and thrown away. Because the tool was positioned just above the area of interest, a plane cut just above the area of interest was also enough to segment the tool from the nonrigid liver. This technique worked for this particular dataset, but is not guaranteed to work with other datasets we might collect later this quarter.

The global registration workflow failed to generate a global model with the dataset collected. The reason it failed is likely because the dataset is a point cloud and the registration code expects a mesh. However, because of the noise in the surface of the liver, automated reconstruction may not produce desirable results. We are currently still looking for solutions to this problem.

In the meantime, development for the global registration flow is being done with previously captured data, using a scanner with higher resolution and accuracy. Scripts are being developed so that, given a feature in the global registration result, we can find the vertices that correspond to the feature in the source mesh frames. Tools for identifying points and features in the opposite direction - from an individual frame to the global model - are also being developed. Because there is varying precision in the points output from the registration as compared to the original mesh, identifying points is not as easy as originally anticipated.

After some analysis of the global registration workflow with the newly collected dataset, we discovered where the issues were arising that caused the registration to fail on the dataset. Because the registration workflow was designed for use with very dense and uniformly sampled scans, the noise present in the realsense scans presented a problem when performing the point thinning prior to registration.

The realsense camera provides advantages in being able to collect data and record deformation at frame rate, but at a cost of consistency and accuracy. There was a lot of noise in the surface of the scans, and the delineations between scan lines projected from the realsense were clearly visible in the results.

After some consideration, we felt the next logical step would be to attempt to denoise the scans, hoping this would give the registration algorithm a more consistent surface to interpolate over. We explored options to reduce the noise in the collected scans to try to improve the global registration result. We apply a statistical outlier removal (PCL) to accomplish this. The results were satisfactory, shown in the following figures below.

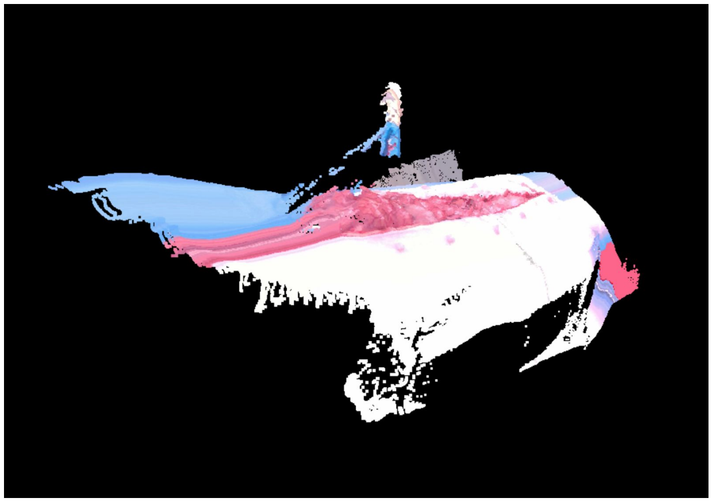

Figure 2.3 The image before applying the PLC algorithm to reduce the noise in the collected scans.

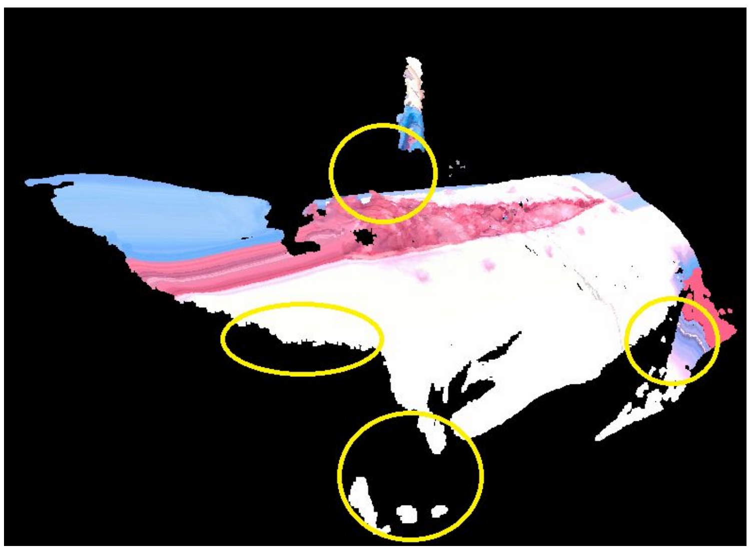

Figure 2.4 The image after applying the PLC algorithm to reduce the noise in the collected scans

Unfortunately, the denoising of the scans did not fix the problem with the registration. After running the registration, an empty mesh was still output. We went back to the drawing board, reconsidering the implications that the non-uniformity in the distribution of points in the collected point clouds, for example the concentration of points around the scan lines, might have in the registration. We found that the point thinning algorithm as a preprocessing step to the registration removed a lot of points that otherwise should have been included. This situation was exacerbated by the fact that the point density of the collected scans would vary even from frame to frame within the same “video”. After the point thinning was performed, there were little to no points left for the registration algorithm to register, and so the global registration resulted in an empty mesh.

We resolved this by adjusting the threshold distance between points in the point thinning/pruning algorithm. The threshold distance between the points was initially set to 4, which worked for the denser scans that were collected using the high resolution scanner. This threshold distance had to be adjusted experimentally, and we determined that a value of between 10^{-6} and 10^{-2} resulted in an acceptable prune. As shown in figure 2.5, After doing this, we were finally able to generate a global registration using the newly collected data.

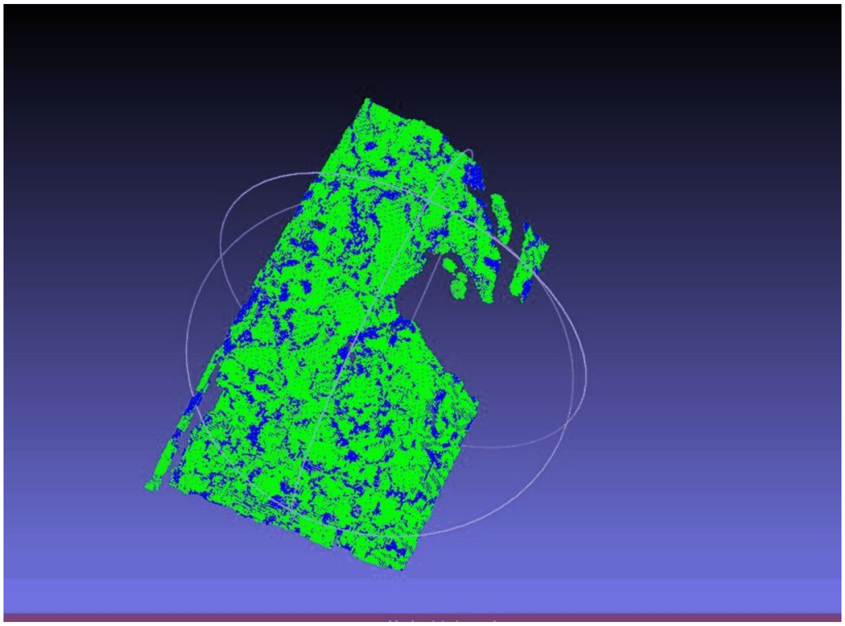

Figure 2.5 The image of the global registration after using the newly collected data by adjusting the threshold distance between points in the point trimming algorithm

Milestones

The milestones that our team accomplished are listed below. We ran into issues with many of these milestones, but we were able to find solutions for the problems that we encountered. A more detailed description of milestones are accessible in our Milestones document. We detailed the issues we encountered, and their respective resolutions, above in the Details of Development section.

1. Get Realsense camera to work with hardware provided:
    - *Deliverable:* Dummy meshes collected from camera and displayed in Meshlab.
    - *(first mesh of dummy scan):* https://github.com/jayala-29/xray_vision/blob/master/scan1trevor.png
2. Develop a specific procedure for data collection:
    - Specific operation of computer/camera interface: how scripts are run, which laptop to use, which USB ports will work with camera.
    - Detail of how camera will be mounted and positioned for recording. The goal is to create a documented, repeatable procedure that can be referenced when analyzing the quality of the data recorded.

- *Deliverable:* A document containing step-by-step instructions of the data collection process. This is important to document how data is collected for these ground truth data sets. The procedure must be repeatable and able to accommodate adjustment to individual steps as necessary.

3. Make an appointment at the Center for the Future Surgery (CFS) to conduct recordings and do recordings of the pig's liver at the surgery site.
    - Deliverable: A recording of the pig's liver at the CFS with various angles and lighting.
    - Google drive link to the recordings:
    - https://drive.google.com/drive/folders/1-OOP0Wo8JggQuUAvSluBppixM2xEYloo?usp=sharing (must have a ucsd email to view it)
4. Record porcine liver on site at the Center for the Future of Surgery (CFS)
    - *Deliverable:* A set of 13 recordings of the porcine liver with various lighting configurations. On the day of the recording, Michael, Trevor, and Vincent traveled to the CFS and performed a series of recordings on a living pig. The pig was sedated, and prepared for operation by the medical staff. Between surgeries, we were able to perform our recordings using the surgical lighting provided in the operating room. The lighting was adjustable by angle and position, but not by intensity. Therefore, the various lighting conditions we could provide were always full intensity, which sometimes created shadows when not pointing directly at the subject.
5. Make the script more robust and efficient to allow us to collect more data given a small window opportunity to conduct recordings.
    - Deliverable: an example of how the script has been improved and a follow up explanation of why this was done.
    - https://docs.google.com/document/d/1xqIC4PwHt0w7C4AgD_fXy52px6v7BNTQnZzmJ_-bxX0/edit?usp=sharing (must have a ucsd email to view it)
6. Develop a formal specific procedure for data analysis:
    - Debugging Meshlab/Meshmixer when necessary with mesh files
    - Running data through the pipeline of scripts developed by the other subteam of the project and the general outputs of those specific filters (documentation)
    - *Deliverable:* A document containing step-by-step instructions for data processing and a type of guide for using mesh based computer tools. Will be continuously adjusted and updated.
    - https://docs.google.com/document/d/17KMw9fFUo-SyKEaU_YaVBbyhU273UJ4z3vM2Qtg3OQ0/edit?usp=sharing
7. Establish web presence:
    - *Deliverable:* Type of online reference containing information and specifications about our project.
    - https://github.com/jayala-29/xray_vision
8. Create a reference for processed data to reflect updated algorithms:
    - Running data through the pipeline of scripts developed by the other subteam of the project (data results)

- *Deliverable:* Various displays of segmentation process.
- https://docs.google.com/document/d/1xKXWmlcxklLrOYatB7hMiOvvv2c6UjyC-1UvnGoOI4E/edit?usp=sharing

9. Create a reference for processed data to reflect updated algorithms:
    - Running data through the pipeline of scripts developed by the other subteam of the project (data results)
    - *Deliverable:* Various displays of global process.
    - https://docs.google.com/document/d/1VGprMFix1DAcU3kUJPLZ6pM3HVp81K2LIY-E7560sTk/edit?usp=sharing
10. With the collected test dataset, run through and get familiar with details of segmentation and global registration benchmark, fixing potential bugs.
    - *Some bugs in the current segmentation code.* The liver segmentation process can be divided into several steps. First we manually segment the rigid part from the first scan. Then for the next scan, we use algorithm to build match between the segmented rigid part and the current scan, which means roughly the corresponding points in the current scan are found. Next do a geometric search around those points thus all the rigid region can be included. Finally we cut it off as the rigid part to match for the next scan. Iteratively all the rigid parts in the scans can be segmented. We found the current segmentation always use the first scan as the rigid template. We fixed that to make the iterative one work.
11. Improve the performance of the algorithm.
    - Reducing computation time: The entire workflow of global registration can roughly decompose into three steps and took much running time. First find feature corresponding between each pair of scans, then use the correspondings to get global positions of points, at last warp up the scans into global registration scan with Thin-Plate Splines(TPS). Since each scan contains over 20000 vertices and it will take huge computation time if there are too much features. We found that with about 1-10% sample rates, which means 200-2000 feature points, the actual TPS alignment size is always around 30-100 after nonoptimal feature rejections. So we take a relatively small sampling rate, to select only around 30 features in each scan. It does make sense since most of nonoptimal features are removed because they are too close to each other, thus the performance of TPS alignment will be not affected. The running time is reduced incredibly. It only needs a few seconds to perform registration with 10 scans, before it took more than 2 hours.

Conclusion

We were able to develop a workflow to perform 3D global registration over scans collected in real-time using a new depth scanner. We identified the issues of using a different depth scanner than what the supporting software was designed to work with, and we fixed the problems that we encountered. The end product is a documented process to collect a frame-rate series of 3D scans of some organic tissue, and a software framework to generate a global registration from the collected scans.

This is a valuable contribution, because the previous workflow took a significant amount of time and manual involvement to obtain the 3D depth scans, and it had much less flexibility in the types of surfaces it could capture. With the old workflow, it took an entire day of manual effort to collect 3 seconds of video. With the new workflow, it takes just seconds.

To evaluate registration algorithms, we need to have a relevant dataset and a registration baseline. It greatly increases the development cycle of these algorithms for a given application if it takes great effort to collect a new dataset. This limits progress that can be made on the algorithmic development front. With the new procedure, data can be collected almost in real-time. This decrease in data collection time translates to decreases in development cycle time of AR algorithms, which we hope can enable researchers to make faster progress toward clinical readiness for this technology.