

Title: Septum Reconstruction from Tool Motion Data

Team: Rohit Joshi, Manyu Sharma

Mentors: Dr. Swaroop Vedula, Molly O'Brien, Dr. Ayushi Sinha, Dr. Anand Malpani

Introduction

Septoplasty is a medical procedure conducted to correct deviated septums, and is considered one of the most basic surgeries an ENT surgeon has to master. In fact, this surgical procedure is the third most common ear, nose and throat surgery performed [1]. The most important and complicated step in septoplasty surgery is the mucosal flap elevation step, in which the surgeon uses a surgical tool called the cottle elevator to elevate the mucosal lining from the underlying cartilage in order to correct the septum deviation. Research has been done to take unstructured tool motion data from septoplasty and extract relevant information, such as classifying the data into strokes as done by Ahmidi et al [2]. However, there currently is no way to visualize the geometry of a patient's septum without CT imaging.

Problem

Due to the size and position of the septum, septoplasty surgeries are performed in a "black box," where no one is able to see exactly what is happening during the surgery. In addition, there is currently no way to visualize the geometry of a patient's septum without CT imaging. Because of this, a way of visualizing a patient's septum geometry without having to conduct imaging could enable a host of downstream applications, including surgical training, skill assessment, and enhanced patient care.

Experimental Approach

An ideal solution to this problem would be a deformable registration algorithm that makes use of an atlas. Our project focused on moving towards that end goal, while also being achievable in a semester. To this end, we designed model septum shapes and used them to create a simulation dataset of tracked surgical tool tracings. We also developed a method to take raw tool motion data and extract relevant information to create visualizations depicting the geometry of the motions made. We were able to register the measurements made from these shapes to one-another using deformable coherent point drift. These types of transformations could be used in a future atlas based registration approach, such as the one being developed by Dr. Ayushi Sinha at

Johns Hopkins. Ultimately, these methods could help create a way of visualizing a patient's septum geometry without any imaging. The sections below detail the specifics of the experimental approach we took in this project.

Data Collection Setup

While a clinical dataset was available, our mentors advised us that this data was unprocessed and would be extremely difficult to work with. Thus the first step in our experimental procedure was to set up a data collection station to collect simulation data that we could use in our project. To do so, we attached a tracked surgical tool (cottle elevator) to an Aurora EM tracking system. To collect data we proceeded to use a custom tracking software UI used by our mentors in the Malone Center for Engineering in Healthcare. We finished our setup by writing some baseline code to process the Aurora data format into usable data for our purpose. After experimenting with several libraries, we decided to use Plot.ly for our visualization engine and Python as our programming language of choice.



Figure 1. The image above depicts the data collection station we used to collect data to build our dataset

Calibration and baseline data

After setting up our data collection station, we decided to collect some calibration and baseline data. To start, we conducted a basic pivot calibration. Figure 2 below shows the pivot calibration holder we developed and printed to hold the cottle elevator while conducting this pivot calibration. Figure 3 shows the results of our pivot calibration.



Figure 2. The image above depicts the pivot calibration holder we used to conduct our pivot calibration. The multiple holes were printed as we did not originally know the best size for our tool.

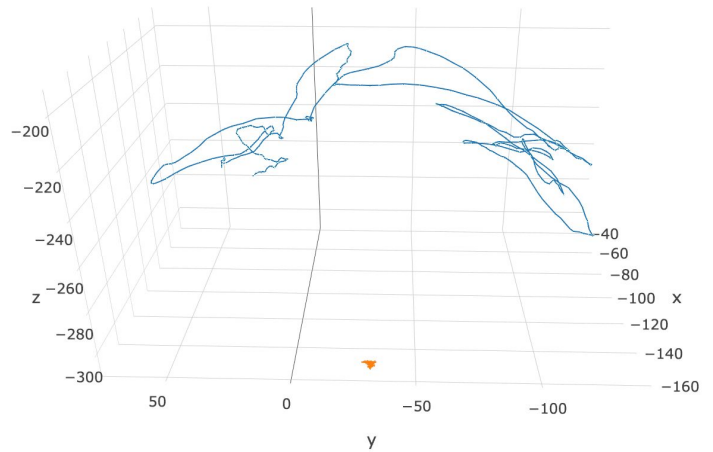


Figure 3. This image represents the output of our pivot calibration.

After this, we proceeded to conduct two simple tests of our data collection system. To do so, we first traced a piece of paper to obtain data representing a flat plane. This is represented in Figure 4 below. Then, we attempted to trace a name on the paper. That data is represented in Figure 5.

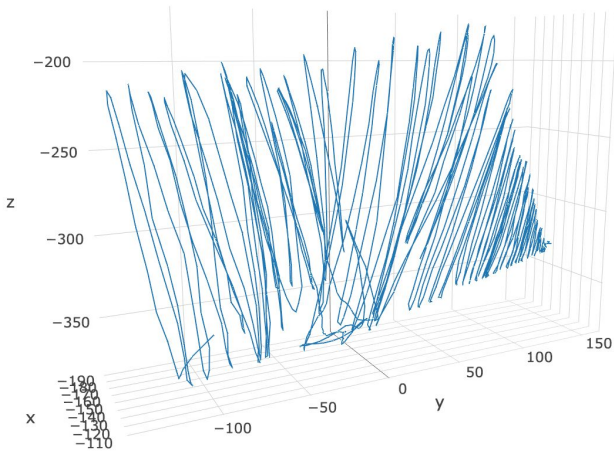


Figure 4. This represents the visualization of a tracing of a flat plane.

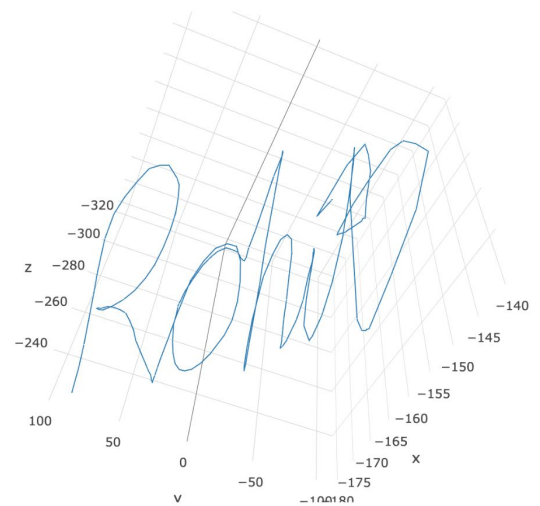


Figure 5. Here we tested the resolution and noise of tool motions by tracing out Rohit's name

Development of ideal septum models and ideal state data collection

After discussion with medical doctors, we discovered that there are three main types of septum deviations: a C shaped septum curved to the right, a C shaped septum curved to the left, and an S shaped septum. Based on this knowledge, we developed phantom models for a flat plane, a curved C shape, and a curved S shape. The flat plane represented the ideal septum shape. Because the two C shaped deviations could be formed in a model by flipping one into the other, we did not develop two separate septum models to simulate them. These models can be seen in Figure 6 below. They were developed on Solidworks and printed using 3D printers in the JHU Biomedical Engineering Design Studio. A more detailed description of how the models were built along with their specifications can be found in the technical documentation at the end of this report.

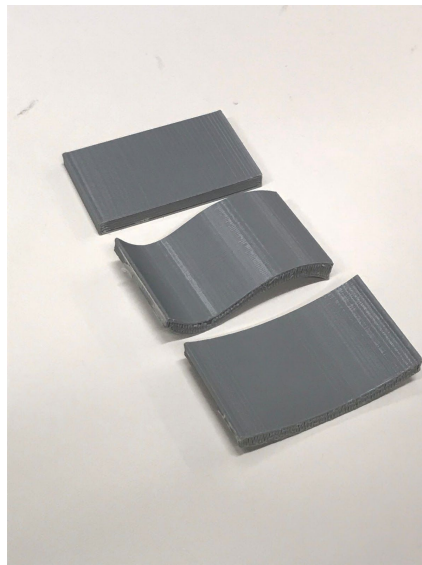


Figure 6. This image shows the three septum models we created. From left to right we have the flat plane model, the S shaped model, and the C shaped model

We then proceeded to use the tracked cottle elevator to trace the surface of both sides of each model. To hold the models down, we used a bench clamp. This represented our ideal state data collection, as it allowed us to get a sense of how accurately we could see a visualization of a septum model that had no extraneous motions. Figures 7-9 show examples of these visualizations, and clearly show that this method resulted in well defined visualizations.

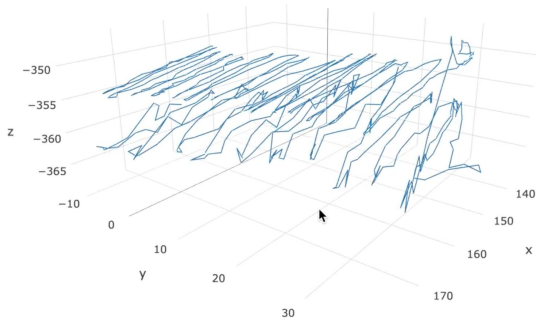


Figure 7. This image shows the output of visualizing the tool motion data from the ideal data for the flat plane model

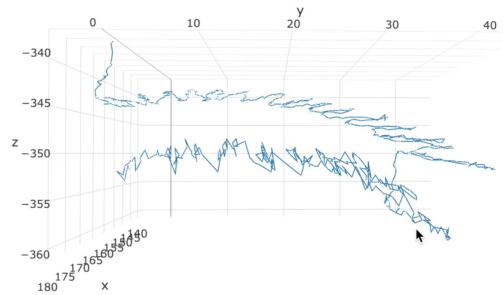


Figure 8. This image shows the output of visualizing the tool motion data from the ideal data for the C shaped model

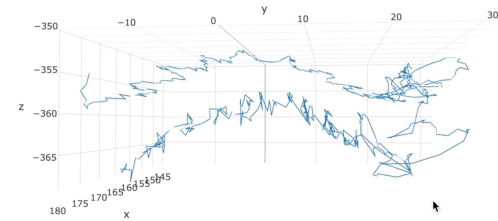


Figure 9. This image shows the output of visualizing the tool motion data from the ideal data for the S shaped model

Development of gelatin peeling model and data collection

To more accurately mimic septoplasty surgery, we decided to collect tool motion data from peeling off an adherent surface from the surface of our phantom septum models. This step would mimic the important mucosal flap elevation step in normal septoplasty surgery. Originally, we tried peeling off stickers. However, we quickly found that these were too adherent and were not working as an adequate simulation. Thus, we decided to use a gelatin layer to simulate this mucosal flap. Figures 10 and 11 show what these layers looked like and how we proceeded to peel these off. In addition to data we simulated and collected, we used this experimental method to collect data from a current clinician, Dr. Masaru Ishii, which allowed us to gain more accurate data within our dataset. The specifics of our gelatin experimental procedure is detailed in the documentation found at the end of this paper.

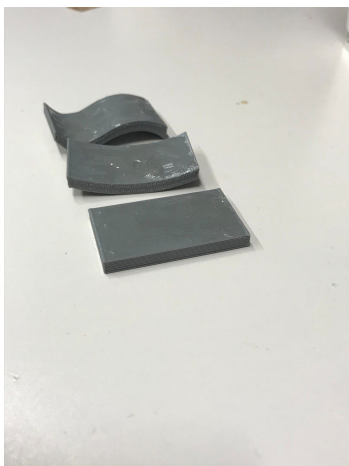


Figure 10. This represents our models with the gelatin layer applied and hardened on

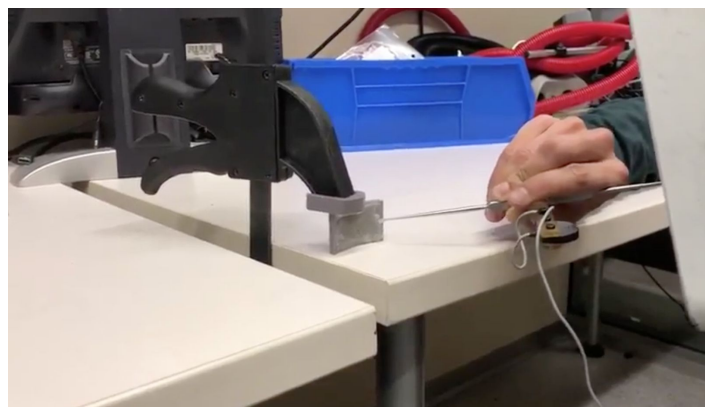


Figure 11. This image shows how the gelatin layer was peeled off using a looping motion with the cottle elevator, similar to in septoplasty itself

Figures 12-14 show visualizations of this data. Because the mucosal flap elevation step is conducted with repeated looping motions, the cottle elevator is often off the surface of the septum itself. This makes it very different from the ideal experiment we conducted earlier and this can clearly be seen in the more noisy data below. This is something we tried to correct for by isolating the on-surface points later in the project.

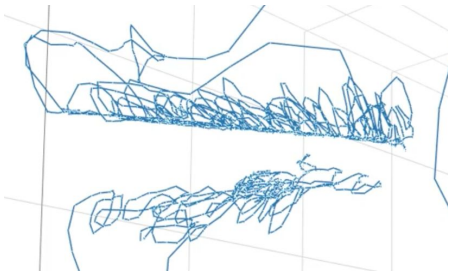


Figure 12. This image shows the output of visualizing the tool motion data from the gelatin peeling data for the flat plane model

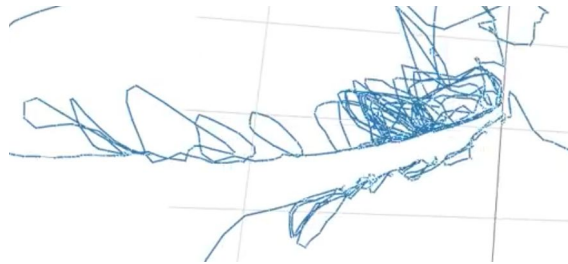


Figure 13. This image shows the output of visualizing the tool motion data from the gelatin peeling data for the C shaped model

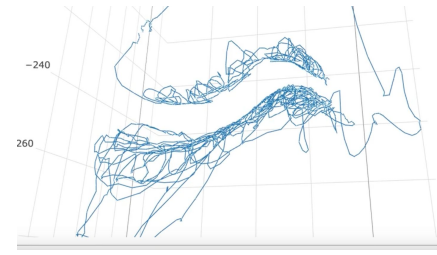


Figure 14. This image shows the output of visualizing the tool motion data from the gelatin peeling data for the S shaped model

Coherent Point Drift algorithm

In order to find the transformation needed to register one tool tracing to another, we decided to use the Coherent Point Drift algorithm. CPD works by assigning one point cloud's points to be Gaussian mixture model centroids, and the other point cloud is used as data for the GMM. This occurs using an EM approach: in the E step, probabilities of each point in one point cloud being drawn from all the points in the other are found and used to weight the results of the M step. In the M step, the optimal transformation is found. For the deformable case, this transformation uses a few special matrices that are based on the underlying assumption of 'coherent drift', specifically, a stiffness matrix is used to bound the ways in which points are allowed to drift. This process iterates until convergence is met. In general, we found that we needed deformable registration as rigid transformations were not enough to encompass the types of deformations present in our models. In order to compare the results of the CPD, we used two metrics. First, we used a proximity metric, which told how many points in the transformed point cloud were within some distance of a point in the target point cloud. For our second metric, we used Hausdorff distance on the suggestion of our mentors. Our results sections has some sample registrations and the metrics discussed here as well.

Isolation of on-surface points using SVM

After playing with our gelatin peeling data, we quickly verified that registrations between them was simply worse than the ideal data, as we had hypothesized. In order to improve these registrations, we decided to try and isolate the on-surface points from the rest of the motions. After looking at the data for some time, we decided to try and use Support Vector Machines to help us find the boundary between the left and right sides of our phantoms. Although there was plenty of noise outside the model where the peeling motion was occurring, we felt the margin of the actual model was large enough to be easily used in an SVM approach. Once the SVM was trained, we felt we could simply use the support vectors as the on-surface points. After trying this out, we found that the density of support vectors was not nearly enough to define the surfaces of the models. So, we instead found the decision boundary of the SVM, and used the closest points to the boundary as our on-surface points. This worked well for the flat model, as shown below.

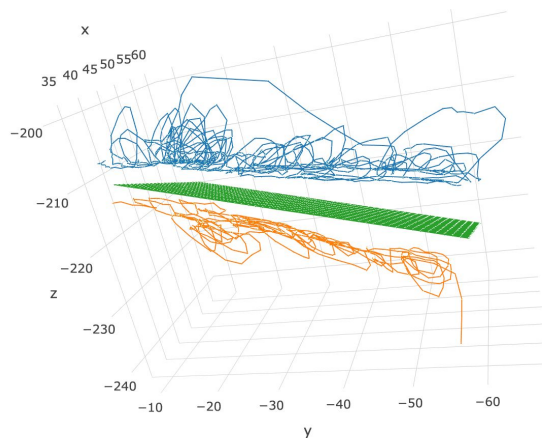


Figure 15: Gelatin peeling data. The blue here represents the left side of our flat model, and the orange is the right side. The green plane is the decision boundary of our SVM, from which we were able to extract the on-surface points.

Results

To summarize our results, we found deformable CPD to work great on our ideal data, and not so well on our gelatin data. We found affine CPD to work alright. Below are some results of deformable CPD registrations.

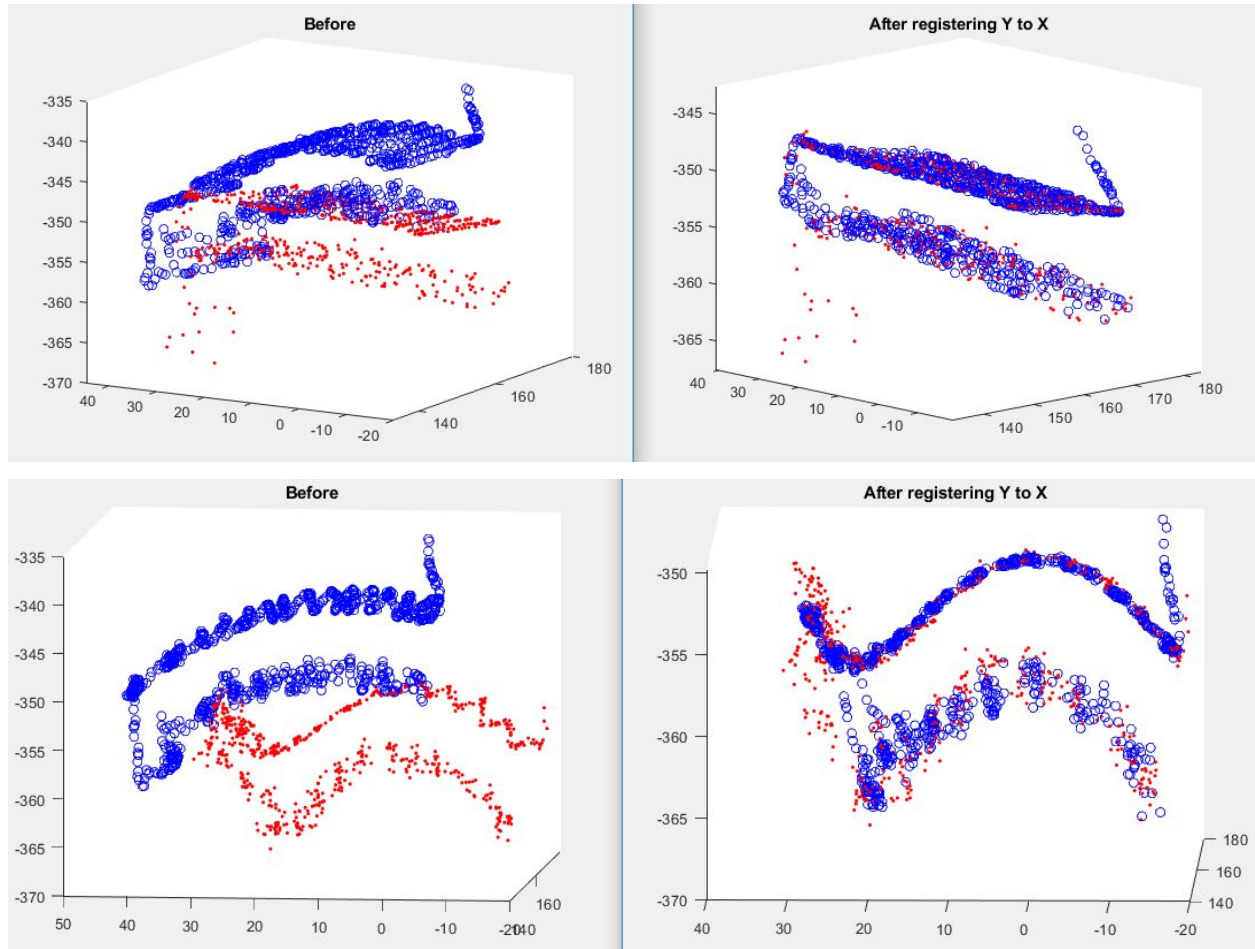
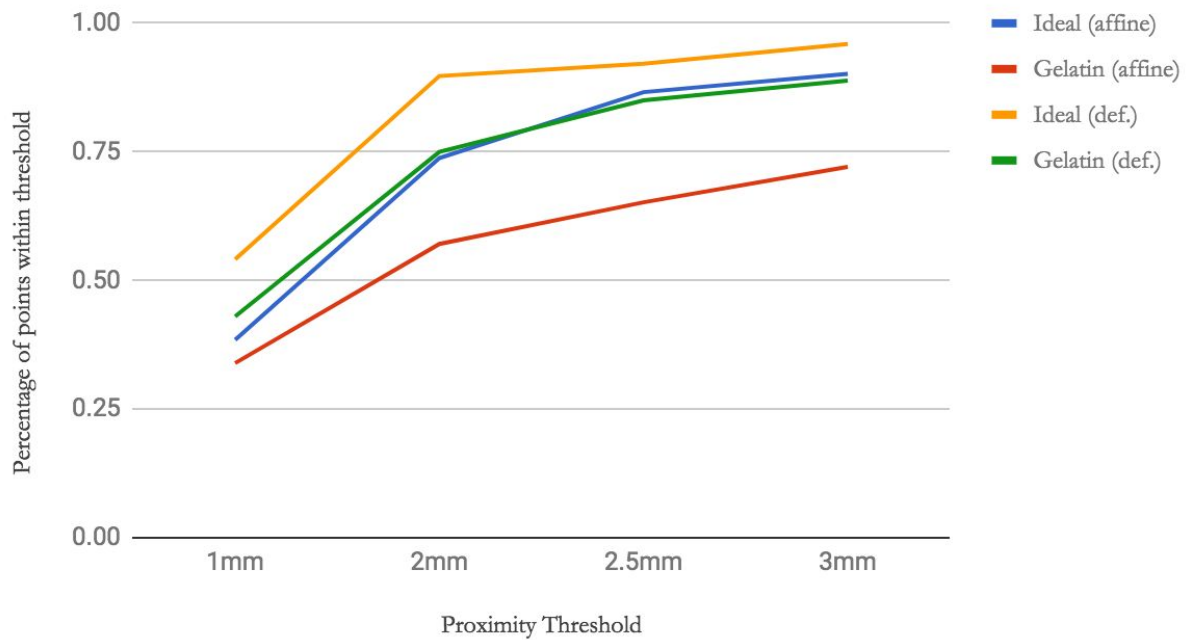


Figure 16: The top image shows the results of registering the ideal curved data to our flat data. The bottom image shows the results of registering the ideal curved data to our S-shaped data.

We also developed a custom proximity metric, where we measured the amount of points that fell within different threshold levels. Below is a table and graph of the proximity metric data:

Threshold	Ideal (affine)	Gelatin (affine)	Ideal (def.)	Gelatin (def.)
1mm	38.44%	33.94%	54.12%	43.92%
2mm	73.80%	57.11%	89.74%	75.09%
2.5mm	86.61%	65.20%	92.62%	85.14%
3mm	90.16%	72.11%	96.35%	88.87%

Custom Proximity metric



Based on this proximity graph, we can see that ideal always performs better than gelatin peeling data, and that deformable ICP is preferred over affine. The only reason we had this comparison between affine and deformable was that for a while, deformable was not working properly. Eventually, we were able to get a functional form, which is clearly superior to the affine variety.

We also checked the efficacy of our deformable CPD using the Hausdorff distance metric, which is another useful way to check the agreement between two point clouds, especially since the Aurora has innate noise when it is making measurements.

We also created a dataset of tool tracings at the request of our mentors, in the case someone else needs access to it in the future.

	Curved ideal to flat ideal	Curved ideal to S-shape ideal	Gelatin S to gelatin flat
Hausdorff Distance:	7.76	13.36	169.30

It is clear again that the gelatin peeling data is much harder to properly register.

Conclusion

Significance

Our goal was moving towards an atlas based registration algorithm, similar to the work of Dr. Ayushi Sinha on sinuses. Through this project, we have registered measurements made from phantom models we designed to one-another.

The transformations we found are of the same type that will be used in an atlas based approach. Our work also provided a novel way to find on-surface points, by trying to find the points that defined the margin of the phantom septum, and calling those points the on-surface points. This was done by using an SVM trained to classify the points on the left of the phantom from the points on the right of the phantom. In practice, we realized there is really no 'margin' to be found in surgical data, so using SVMs probably would not be the idea way for this.

One additional deliverable was the dataset we developed, which can be used in future for other experiments related to tool tracings and registrations. In addition, after further iterations we envision our phantom models and gelatin peeling experimental protocol being valuable for further applications to other projects.

Extensions/Limitations

The main limitations of our project come from the fact that surgical data is much more convoluted than any of our simulation data was. We envisioned creating algorithms that worked well with simulated examples, which we could make robust using the surgical dataset. The reality we found was that our simulation examples were a completely different type of data when compared to the surgical data, and our methods we used relied on some unique properties that only applied to the simulation data (such as the clear gap to use SVMs to find on-surface points).

Acknowledgements

We would like to acknowledge Dr. Russell Taylor and Ehsan Azimi for leading us through the CIS II course. We would also like to acknowledge Dr. Swaroop Vedula, Molly O'Brien, Dr. Ayushi Sinha, Dr. Anand Malpani, Dr. Gregory Hager, and Dr. Masaru Ishii for mentoring us and helping throughout our project.

Management Summary

Who did what?

Both team members worked together on most of the core hardware and software. Overall, Manyu took primary ownership of the raw data processing, implementation of

coherent point drift and SVM's. Rohit took primary ownership of the development of the phantom septum models, experimental data protocol, and simulation data collection.

Accomplished versus planned

Figure 17 below shows our planned deliverables for the project. Our original deliverables were nearly identical, the only real change was the addition of a few extra deliverables to the minimum (phantom models and building a dataset were added). These new deliverables were added as we were working towards the others, and realized that they would be needed.

- **Minimum:** (Expected by 4/8/18)
 1. Develop phantom models for simulation data collection **COMPLETED**
 2. Build a dataset using phantom models and Aurora tracking device **COMPLETED**
 3. Visualizations from using the coherent point drift on "ideal" simulation data **COMPLETED**
 4. Comparison of our CPD model with the actual model **COMPLETED**
- **Expected:** (Expected by 4/22/18)
 1. Enhance the model to simulate mucosal flap elevation using gelatin peeling **COMPLETED**
 2. Visualizations from using deformable ICP on gelatin peeling simulation data
 3. Visualizations from using CPD on gelatin peeling simulation data **COMPLETED**
 4. Comparison of the two algorithms with the expected output and with each other **SEMI-COMPLETED**
- **Maximum:** (Expected by 5/7/18)
 1. Processed surgical data from the R21 dataset with accompanying visualizations of deviated septums created from deformable ICP and CPD
 2. Metric measuring accuracy of the outputs of these models, most likely computed from comparing the two with each other **COMPLETED**

Figure 17. This shows our planned deliverables for the project

As you can see, we were able to complete almost all our deliverables. The main deliverable that we were not able to fully accomplish that we had originally planned to was a fully working implementation of deformable ICP. Because this algorithm involves statistics, including PCA modes, it was significantly more work than we had planned. In the end, we were not able to fully implement deformable ICP, which resulted in one deliverable not being completed, and another missing about half of what it needed. Since the main purpose of our project was to prove the concept of any deformable registration working with septoplasty tool motion data in order to move to an atlas based registration algorithm in the future, we feel that using only deformable coherent point drift still achieves our main goals.

We also did not achieve part of our maximum deliverable. While we began working with the clinical data, after spending some time trying to see if we could use any of our methods with the surgical data, we found that it would be nearly impossible. The

surgical data was much harder to use than we initially expected. Most of the cases we looked at had data clouds with no underlying structure at all, solely strokes leaving and entering the nose. The inherent problems such as the non-rigidity of the actual septum, the possibility of patient head motion (although occasionally corrected for with head sensors), and missing frames, were all together too many confounding factors for us to be able to make any sense of before the project deadline.

To summarize, we accomplished all of our minimum deliverables, missed one of our expected deliverables, and missed one of our maximum deliverables.

Next steps:

The primary next step is an extension of our project to an atlas based registration approach using CT scans. The project here has laid the groundwork for a more intricate approach.

Next, creation of more accurate phantom models based on CT scans could potentially create a more accurate dataset as well. We believe it should be possible to create very accurate models of septums, probably by using some non-rigid material as the septum, and building it using CT scans of septums which are also available to the research group we worked with. While more of a hardware problem, there could be many uses for these model septums besides data collection, such as the potential for medical training.

Lastly, applying our approach to a clinical dataset instead of just a simulation dataset will be important for achieving the final goal.

Because we will be graduating after this semester and will not be able to continue our work, we have developed extensive documentation of our code and experimental protocol for future use. In addition, we will have passed on our dataset to our mentors to be used when an atlas based registration approach is begun.

Lessons Learned:

We learned many lessons throughout this project. One lesson was that actual surgical data is much more complex than any simulation data. One of our original maximum deliverables was applying registration algorithms to a clinical dataset, however as we began to try this we found out that the data was much noisier than we had expected due to circumstances that we had not modeled in our simulation data, such as the movement of the septum and the patient.

In addition, we learned how important regular meetings with mentors are to successful project progression. At many stages in our project, we were struggling with finding the best way to proceed and long discussions with our mentors allowed us to find the optimal solutions to our problems.

Lastly, we learned the importance of prioritizing deliverables. As our project progressed it became clear that some of our original deliverables were too ambitious to be achieved in one semester and many important challenges that we had not predicted arose. Thus, we had to develop the skills to deal with these changes and continue to modify our project plan as needed.

References

[1] Bugten V, Nilsen AH, THorstensen WM, Moxness MHS, Amundsen MF, Nordgard S. Quality of life and symptoms before and after nasal septoplasty compared with healthy individuals. *BMC Ear, Nose, and Throat Disorders*. 2016

[2] Ahmidi, N., Poddar, P., Jones, J.D. et al. (2015) Automated objective surgical skill assessment in the operating room from unstructured tool motion in septoplasty. *International Journal of Computer Assisted Radiology and Surgery* (2015) 10: 981. <https://doi.org/10.1007/s11548-015-1194-1>

Berger, Tagliasacchi & Seversky. (2014). State of the Art in Surface Reconstruction from Point Clouds. The Eurographics Association.

Bernard, Salamanca, Thunberg. (2017). Shape-aware surface reconstruction from sparse 3d point-clouds. *Medical Image Analysis*.

Myronenko, Song, Carreira-Perpiñán. (2006). Non-rigid point set registration: Coherent Point Drift. *Conference on Neural Information Processing Systems*.

Poddar, Piyush & Ahmidi, Narges & Swaroop Vedula, S & Ishii, Lisa & Hager, Gregory & Ishii, Masaru. (2014). Automated Objective Surgical Skill Assessment in the Operating Room Using Unstructured Tool Motion. *International journal of computer assisted radiology and surgery*.

Wu, Murtha, Jaramaz. (2015). Construction of statistical shape atlases for bone structures based on two-level framework. *International Conference on Computer Vision Theory*.

Septum Tracking Documentation

The documentation below explains both our Aurora EM Tracking setup/procedure as well as our septum model design procedure/specifications.

First we describe the procedures needed to use the Aurora EM tracking system with the septoserver custom software to collect tool motion data.

Aurora EM Tracking Procedure

1. Turn on the main power supply, mac mini, and power supply for the Aurora trackers.
2. Plug in any EM emitters you will be using to the sensor array.
3. Log in to the mac mini and create a network on the wifi.
4. Once the network is active, copy it's IP address found in network preferences.
5. Navigate to <IPaddress>/gui on safari.
6. Activate the septoserver application on the desktop.
7. You should now verify that any tools being tracked show up with a check mark when that tool is in view of the Aurora.
8. When ready, start the data collection on the safari gui. Collect data on whatever motions you are making.
9. Terminate the data collection.
10. To start a new collection, start from step 6.

Once data has been collected, your output should be stored as a file. This file has a few subfolders, all you should need is within the aurora folder, a file called aurora.txt. This text file has all of your data. This file is formatted as a series of 17 column wide outputs. Each 17 column section represents one of the tracked tools. For each of these, the first 4 columns represent meta-data such as the sensor name, and whether or not the tool was in view. The next 3 columns encode position, and the next 9 columns represent the rotation of the tracked piece, stored as the 9 entries in the rotation matrix flattened row by row.

Model Design Specifications

Here, we detail the design specifications for our septum models and the pivot calibration holder

All our models were developed using Solidworks and 3D printed in the Johns Hopkins BME Design Studio using Makerbots. We have three septum models that mimic the three main classes of septums that exist in the population: a flat plane that represents an “ideal” septum, a curved “C” shaped model that represents one type of deformation, and a “S” shaped model that represents the other type of common deformation. The dimensions are approximately 4 cm x 3 cm x .5 cm, depending on which model is used. The steps below outline roughly how each shape was created on Solidworks.

Ideal Septum Model:

1. Open new drawing on Solidworks
2. Navigate to top plane view
3. Open new sketch and create rectangular shape using the rectangular drawing tool
4. Use Smart Dimension to correctly dimension your rectangular shape
5. Extrude .5 cm (or the width you want)
6. Save as an .stl file and use to print on a 3d printer

C Shaped Septum Model

1. Open new drawing on Solidworks
2. Navigate to top plane view
3. Open new sketch
4. Select the multipoint arc drawing tool and place points in the shape of the arc you want to create
5. Copy this arc to the right of the arc and use line tool to connect the top and bottom of the surface
6. Extrude .5 cm (or width you want)
7. Save as an .stl file and use to print on a 3d printer

S Shaped Septum Model

1. Open new drawing on Solidworks
2. Navigate to top plane view
3. Open new sketch and use drawing tool to create a S shape
4. Copy this shape to the right and place points in shape of an arc you want to create
5. Extrude .5 cm (or width you want)
6. Save as an .stl file and use to print on a 3d printer

Pivot Calibration Holder

1. Open a new drawing on Solidworks
2. Navigate to top plane view

3. Open a new sketch and create rectangular shape using the rectangular drawing tool
4. Use Smart Dimension to dimension this to a 10 cm x 10 cm surface
5. Extrude out to the depth you want
6. Move to the top plane and create new sketch of a circle using circle drawing tool
7. Draw a line through the center of a circle and select and erase one half of the circle
8. Use revolve cut on the semicircle to cut out a spherical hole in the pivot calibration holder
9. Repeat these steps if you are unsure of the size of the pivot calibration hole that you want to create and want to experiment with various sizes

The Solidworks files will also be uploaded soon to give you a starting point for creating your own models.

Below we describe our protocol for adding gelatin to the models in order to mimic the mucosal layer that needs to be elevated during surgery.

Gelatin Experimental Procedure

Here we detail the steps we took to create the gelatin layer on our septum models to mimic the mucosal membrane around the septum.

1. Fill a cup with some water (around .25 cups or less)
2. Heat the water for a minute
3. Pour half a packet of gelatin into the water and stir the mixture
4. Use a knife or straw to collect the slimy parts of the mixture and spread onto surface of septum model. Avoid the more watery areas as it is not as sticky.
5. Let the septum model dry for 5-10 min
6. Once the gelatin surface seems to have hardened and is not sticky to the touch, slowly turn over the septum model and these steps on the other side. If the gelatin-water mixture is no longer warm you can recreate it.

Once the gelatin has been applied onto the septum model, it can be used to collect data by slowly elevating the layer off the surface with the tool tip in repeated circular motions.