# Project Final Report

## Project 5: EchoSure

### Detecting Blood-Clots Post-Operatively In Blood Vessel Anastomoses

Students: Michael Ketcha (mketcha3) and Alessandro Asoni (aasoni1)

Mentors: Dr. Jerry Prince, Dr. Emad Boctor, Dr. Devin Coon, Dr. Nathanael Kuo, Dr. Chen Lei

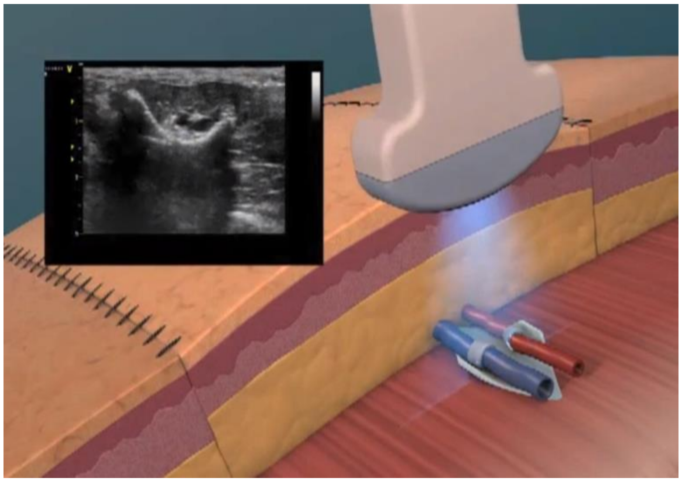
## Relevance and Importance:

In skin flap transplant surgeries, the flap of skin is large enough to need its own blood supply; therefore a blood vessel anastomosis is required. However, approximately 8-15% of these anastomoses will form a blood clot in the few days that follow the surgery. If the clot is caught in time, the patient will return to surgery where the surgeon can clear the clot and save the skin flap. However, approximately half of the time, the clot is not detected fast enough and the flap of skin undergoes necrosis. [1][2] The current methods for detecting the clots rely on examining pulse oximetry in the flap of skin. This is an inherently delayed detection of the clot. Therefore our approach aims to detect the clot directly.

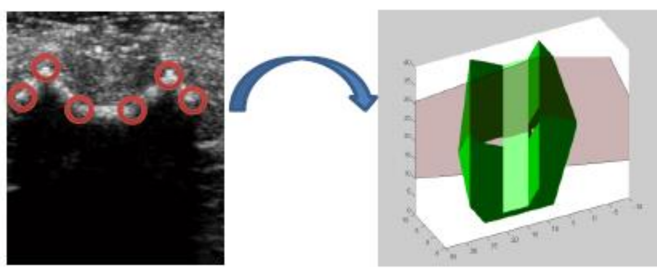
In terms of its origin, this project started off as a CBID master's project in the Biomedical Engineering department. During which time, much of the market value and proof of concept was established leading to the reception of many grants for funding and a provisional patent on the technology.

## Technical Summary of Approach:

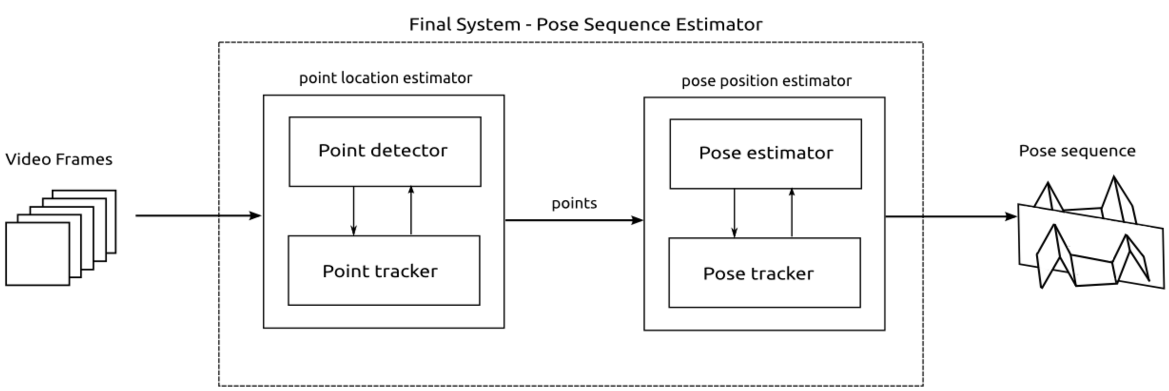
Our approach is to use ultrasound Doppler imaging to track the velocity of blood flow at the anastomosis site. The location of the anastomoses will be tracked using a biodegradable PLGA fiducial which will be placed under the vessel during surgery. A nurse will return every hour following surgery to monitor the change in velocity over time. The project for the semester then involves creating an intuitive and accurate guidance system that ensures the nurse returns to the correct location each time. Specifically, we will focus on developing an algorithm that processes an ultrasound video file and returns the key points for the fiducial pose evaluation. This includes analysis and implementation of tracking algorithms for video processing.



Animation by David A. Rini



**System Overview:**




We have implemented a set of MATLAB routines to process a sequence of frames of US video. The diagram above shows how the components of the system interact. This is what our approach was:

- We developed a point location estimator system. This has one component that uses image processing and computer vision algorithms to analyze a single frame of the video and identify interest points. These are corners of the projection on a plane slice of the fiducial model. Since processing one image at a time is time consuming we also plan to develop a point tracker system to allow for real time processing. This uses information from previous frames to facilitate the job of the point detector. This is done by reducing the size of the region of interest (ROI) by determining how the points are moving and where they are expected to be in the following frame.
- The points that are determined in this way are fed to the pose position estimator. This system uses the points, and knowledge of the 3D model, to determine what slicing plane they came from. The pose estimator system also uses a form of tracking, in that the previous frame's transformation is used to initialize the optimization in the following frame, in order to speed up the process and ensure that the correct local minima is found. This goes under the assumption that the pose won't change significantly from one frame to the next, and that the ultrasound probe is moved smoothly.

## Dependencies:

Our first dependency were access to a 3D printer to rapid prototype our fiducial design. We had access to the 3D printer in the basement of Wyman with a budget code already setup.

The second dependency was access to an ultrasound machine for gathering test data with the rapid-protoyped fiducial. Dr. Boctor's MUSIIC Lab contains ultrasound machines that we can have access to.

The final dependency was access to computers to develop and test our algorithms. Both team members had personal laptops. We also had access to Dr. Prince's servers if we needed to process large data sets.

## Deliverables:

Expected:

- A point location estimator system that processes ultrasound video data and returns interest points for the fiducial. This system includes pure detection and point tracking. These will interact to allow for faster point detection in sequential frames.
  - Status: Ongoing but full prototype created. Both detector and tracker created. Accuracy for both can still be improved. Details described in later section.
- Rough estimation of confidence in each detected pose.
  - Status: Complete. Trace of covariance calculated following projection onto principle components.

Max:

- Statistically rigorous frameworks to track fiducial shape.
  - Status: Complete. Kalman Filter Tracking incorporated.
- Both point location estimator system and pose estimator system.
  - Status: Complete. Mentor, Dr. Nathanael Kuo, developed these subroutines.

Min:

- Same as expected, slower run time (not real time processing).
    - Status: Complete

## Timeline:

**20-27 Feb:** On Time

- **Alessandro:** Build a skeleton of the estimator system with main functions that do simple (if any) processing.
- **Michael:** Object Detection - finding the set of points or lines to which the search of key points should be limited to.

**28 Feb – 6 Mar:** On Time

- **Michael + Alessandro:** Test out RANSAC techniques. Start stringing together tracking and detection.

**7-13 Mar:** On Time

- **Alessandro:** Start testing out Tracking algorithms (e.g. Optical Flow, Kernel Based, Kalman Filters) .
- **Michael:** Start incorporating tracking info (region of interest) into detection.

**14-20 Spring Break**

**21-27 Mar:** On Time

- **Michael + Alessandro:** Improve system. Try new algorithms and optimize code.

**3 Apr (MIN/EXPECTED):** Min On Time

- Working prototype (MIN vs. EXPECTED is dependent on speed and accuracy).

**4 – 15 Apr :** Altered

- ~~**If** speed/accuracy of point location estimator is not good enough:~~
    - ~~Work on improving detection and tracking~~
- ~~**Else:**~~
    - ~~Start looking into pose estimation~~
- **Michael + Alessandro:** On Time
    - Work on incorporating PCA results into Kalman Filter.
    - Polish and Improve working prototype.

**<u>16 – 24 Apr :</u> On Time (But not as accurate as expected)**

- **Michael + Alessandro:**
    - Use local tracker for fast point tracking (-> normalized cross correlation)

**<u>24 Apr – 8 May:</u> On Time**

- **Michael + Alessandro:** Clean and document code. Work on poster for presentation

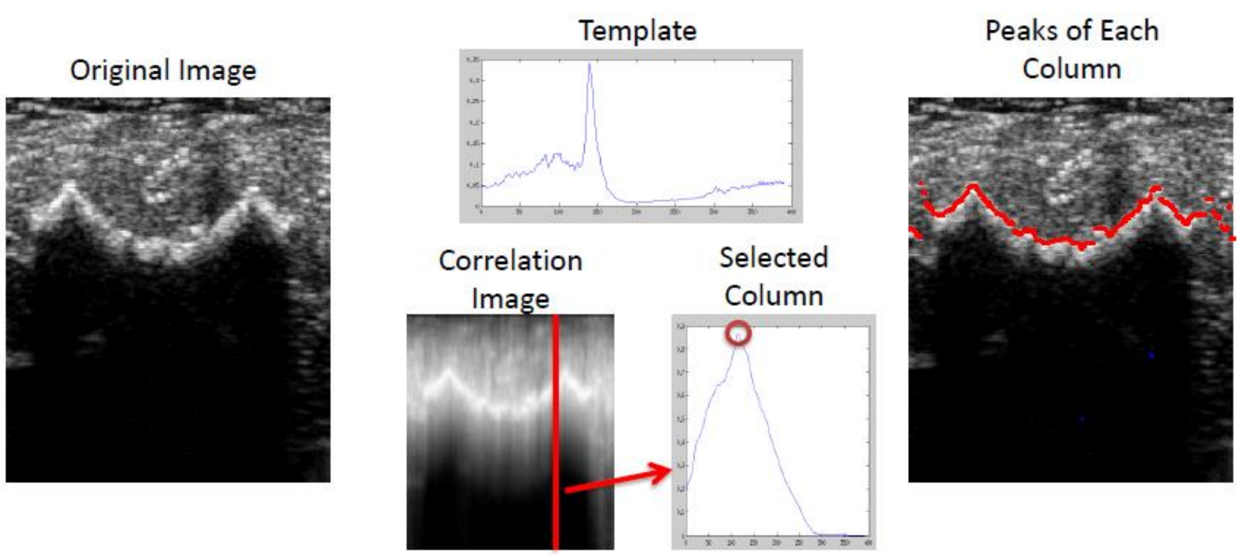
**<u>9 May</u> On Time**

- **Poster presentation / Project Final Report**

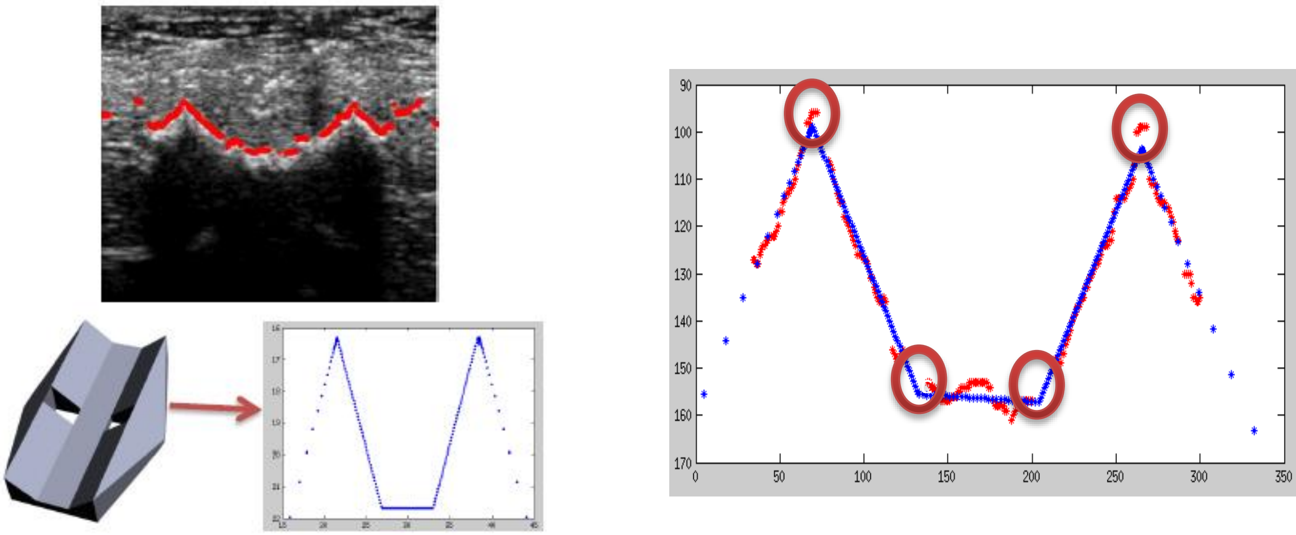
## <u>Methods:</u>

<u>Detection</u>

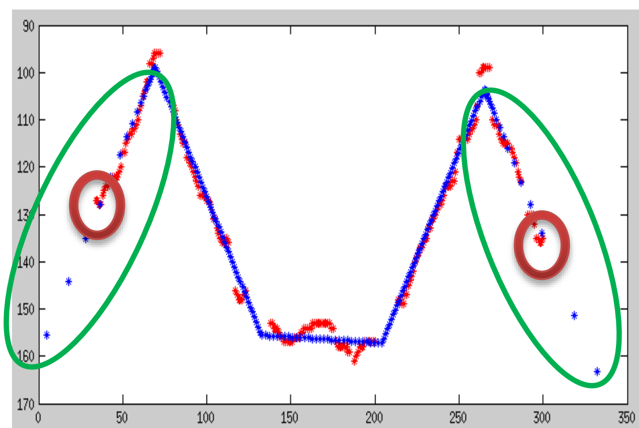
The first step of detection is template matching by column of the image using normalized cross-correlation (NCC). This method takes advantage of the large shadow that lies beneath the fiducial in the image. By taking the peak of the correlation along each column, a point set of the fiducial can be extracted.



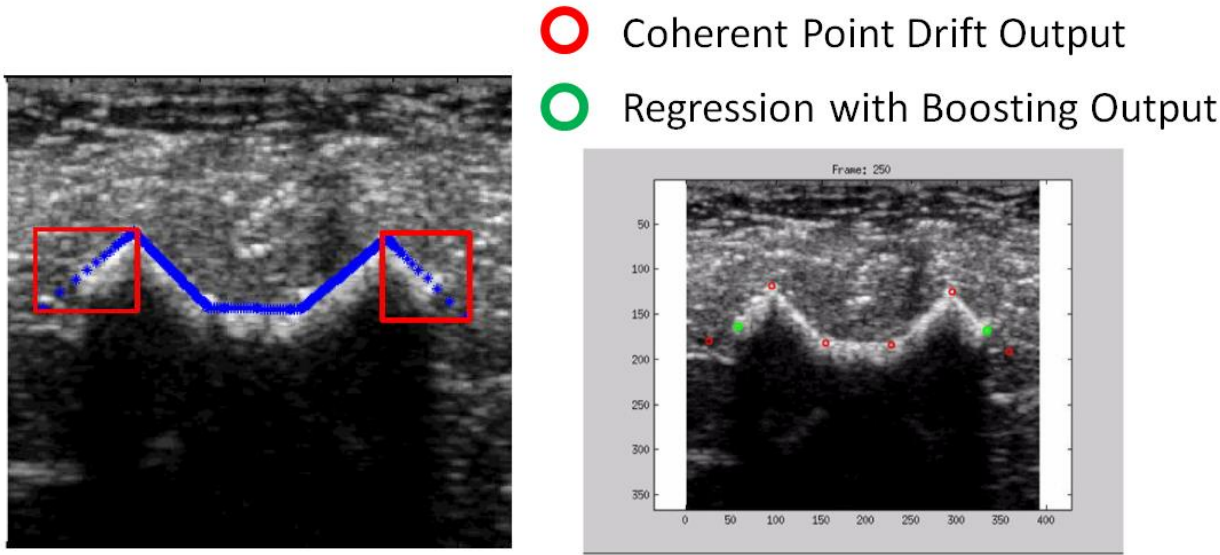

Next, affine Coherent Point Drift [3] is performed to register a point set taken from the model of the fiducial to the point set extracted from the NCC. With this registration the location of the inside four corner points can be determined robustly.




Following this step, the two endpoints of the fiducial need to be determined. The first method attempted was, given the above registration, taking the lines defining the model pointset locations (blue) outlined below, and select the furthest points in the NCC pointset (red) that fit these lines.




While this is fast and can be effective, a more robust approach was sought out using supervised learning with a boosted regression model. This approach extracts a region of interest around these endpoints and turns information contained at each of the pixels into a feature vector. Manual point (x,y) coordinate selections act as the labels for the training data. With the model trained, given a new feature vector, an estimate of where the end point is located can be obtained.




Tracking:

The tracking component of this project uses a Kalman Filter[4]. In practice this corresponds to a set of mathematical equations to estimate the optimal state of a linear dynamic system, governed by the following equations:

$$x_k = Ax_{k-1} + Bu_k + w_k$$

$$z_k = Hx_k + v_k$$

Where the first equation is a state space equation with the addition of zero mean Gaussian noise, and the second is a measurement equation that relates a set of measurements $z_k$ to the state $x_k$, also with the addition of Gaussian white noise. For our project the state $x_k$ that is tracked using the Kalman Filter is a 12 entry vector corresponding to the stacked $x\,y$ coordinates of the six fiducial points. The measurements are the detected locations of the points in each frame.

The interesting part of using a Kalman Filter for this application was determining what the underlying dynamics of the system were. Or, in other words, what $A$ should be set to.

The way this was done was using Principal Component Analysis[5]. A large set of acceptable poses were generated by simulating plane slices through a 3D model of the fiducial. From these slices the corresponding position of the 6 points was found. These were then stacked into a large 12xN matrix to which PCA was applied. This gave a set of 12 principal components and corresponding coefficients.
The idea at this point was to set the eigenvectors of A to be these principal components. In other words we wanted the internal dynamics of the system to follow the modes of variation that we expected in the detected points. The eigenvalues of A were then chosen to be negative and large for principal components with large coefficients, and negative and small for principal components with small coefficients.

This results in the following behavior. Given an initial pose of 6 points if we let the Kalman Filter evolve without providing measurement input, and set the state covariance to zero, the points will converge quickly towards their projection onto the principal components with highest coefficients. And then slowly towards their projection onto the components with lower coefficients, eventually converging towards the mean shape.

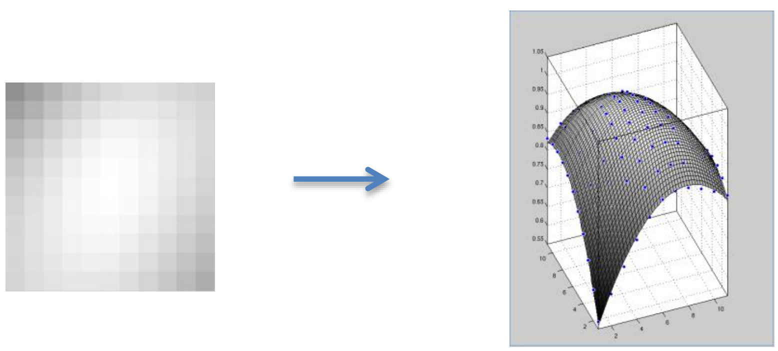
A way to look at this is by assuming that the principal components define a feasible space for the 6 detected points. The dynamics of the system should then attempt to correct any noise in the system, due to error in detection, by “pulling” the points towards the feasible space.

For tracking to be effective some form of local detection is necessary in order to quickly find where each point has moved in the next frame. We took the following approach. At each frame we extracted a template around each point. We then used Normalized Cross Correlation to search a small neighborhood of each point in the next frame, and took the maximum correlation to be the new detection.

This also gave a way to estimate the measurement covariance for each detected point. If we assume that each $x, y$ coordinate, at which normalized cross correlation was computed, is sampled from a probability density function $f(x, y)$ corresponding to the score of normalized cross correlation, then the covariance is given by:

$$\frac{1}{N-1}\sum f(x,y)\left(\begin{bmatrix}x\\y\end{bmatrix}-\mu\right)\left(\begin{bmatrix}x\\y\end{bmatrix}-\mu\right)^T$$

This is explained conceptually in the following image:



On the left the normalized cross correlation scores of an example point is shown. The assumption here is that these scores represent a probability density function like the one shown on the right. With the given equation we can then estimate the covariance of each of the six marker point. This can then be given to the Kalman Filter in the form of a bigger block diagonal 12x12 covariance, corresponding to our estimate for the measurement covariance. The idea is that we want to supply the Kalman Filter with directional variance information. In other words we want to “tell it” in which direction we have confidence of the point position, and in which direction we don’t.

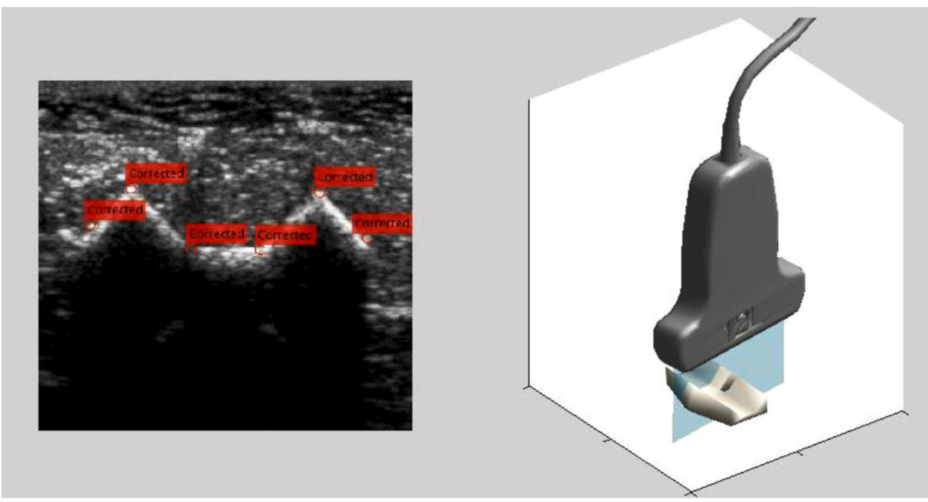
NOTE: Unfortunately this local search method is not working as accurately as we hoped. The normalized cross correlation based on intensity alone does not seem robust enough, especially with the end points of the marker. For now we have resorted to doing full frame detection for every frame, which works much better, but unfortunately does not achieve real time speed.

## Results:

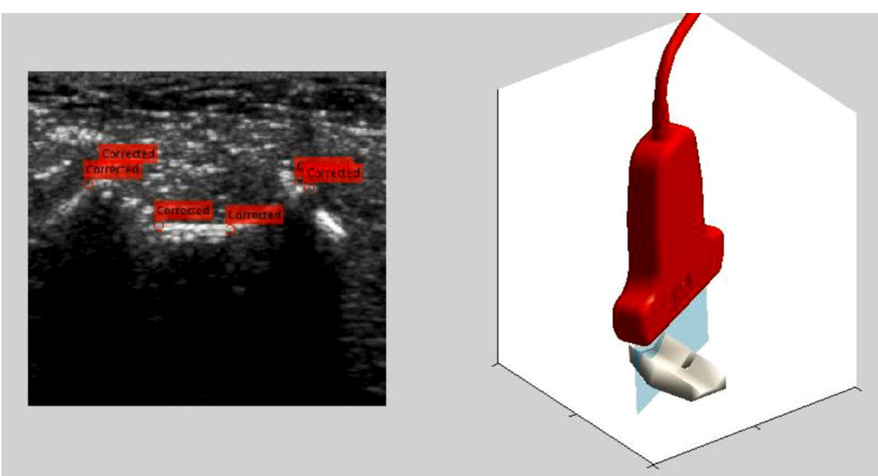
To wrap up, this semester we combined the following three components into one system: first-frame detection, tracking, and pose estimation. The first frame detection finds the six corner points of the fiducial. These points get passed to both the tracking algorithm and the pose estimation algorithm. The tracking algorithm allows the six points in the next frame to be quickly detected so that the system may perform in real time, while also ensuring that the points selected lie in the feasible space of the fiducial shape. The pose estimation takes the six points detected and determines the plane on the model that best fits the cross-section seen in the image.

For this section we will show some stills for reference, but please visit our wiki to watch the full video of the described system at work.

As mentioned above the normalized cross correlation for tracking is not as accurate as planned, so the videos created use the first frame detection at each frame. This results in the current state of the system not functioning in real time. On average, it takes approximately .762 seconds to make all calculations between frames




We use the trace of the state covariance from the Kalman Filter to determine how likely the current pose is. If the trace of the state covariance is high, the probe turns red, indicating that this frame's pose likely has high error.



## Significance:

While the accuracy of the system can still be improved, these results are very important for the progress of this project. First, it gives a working prototype for the full detection and pose estimation system. It also lays out a convincing proof of concept, that given the image of a cross-section of the fiducial, we can find meaningful features of the model that can be used to determine which plane the probe is looking at.

## Management:

Weekly Meetings:

With Mentors (Dr. Prince and Dr. Kuo): Thursday: 8:30 AM– 9:30 AM

Team: Thursday 9:30 AM – 1:30 PM, Sunday 11:30 AM – 2:30 PM

Version Control: Git Repository (Link on Webpage).

The work for the project was split in the following way:

- Michael: Detection
- Alessandro: Object Tracking/Video Processing (Kalman Filter)

### Accomplished Vs. Planned:

In this project, most of our planned deliverables, including the max have been completed. Our minimum deliverable, which was the complete system, including point detection and pose estimation, at below real time speeds has been completed. The first frame detection system has been implemented and works well. Furthermore the tracking method using normalized cross correlation has also been implemented. At the present moment, normalized cross correlation has trouble accurately tracking points for multiple frames due to the noisiness of the ultrasound data. Kalman Filter based tracking, along with projections of our detection onto the principle components has been fully implemented. One of our mentors, Nathanael Kuo, finished one of the max deliverables, which was developing the pose estimation system. This system determined the pose of the frame given the six detected corner locations.

### Future Work:

For the first frame detection, adjustments can still be made to improve robustness and accuracy. This includes optimization over parameter values, and an efficient method to weed out NCC points that are clearly not part of the fiducial. For tracking, future work will include developing a feature set to track that will be robust to the ultrasound speckle so that real-time point tracking will be accurate over many frames.

What We Learned:

In this project, we learned many advanced computer vision techniques both for object detecting and tracking. This was accomplished through both reading literature on existing algorithms and implementing many of them for testing purposes. The project, also gave us insight into the importance of understanding the physics behind the imaging modality in order to better understand how to take advantage of those properties. For instance, this was integral in our taking advantage of the shadow caused by the fiducial. Team management skills were also learned. This included keeping code in the repository up to date so that other members can effectively move forward. It also became very important to have a good understanding of what each of the other team members were doing so that everyone had a good vision of how the system would eventually come together.

## References:


1. Nakatsuka T et. al. (2003). Analytic Review of 2372 Free Flap Transfers for Reconstruction Following Cancer Resection. Journal of Reconstructive Microsurgery, 19(6): 363-368.

2. Bui DT et al. (2007). Free Flap Reexploration; Indications, Treatment, and Outcomes in 1,193 Free Flaps. J of Plastics and Reconstructive Surgery, 119(7): 2092-2100.

3. A. Myronenko and X.B. Song, Point-Set Registration: Coherent Point Drift, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 32, no. 12, pp. 2262-2275, Dec. 2010.

4. Welch, G., and G. Bishop (1995), An introduction to the Kalman Filter. Technical Report TR 95-041, University of North Carolina, Department of Computer Science

5. Jolliffe, Ian. *Principal component analysis*. John Wiley & Sons, Ltd, 2005.