Software Design Specification

for

Electromagnetic Tracking of Endovascular Catheters

Huilin (Lin) Xu
Fangjie (FJ) Li
Shanelle Cao

Under the Supervision of:
Dr. Ali Uneri
Dr. Fernando Gonzalez

Laboratory for Computational Sensing + Robotics

I-STAR

Table of Contents

I. Introduction

1. Motivation

For endovascular techniques, fluoroscopy, and CT angiograms are used to help surgeons visualize the position of their catheter in order to navigate the catheter to the site of the aneurysm. However, this process exposes both the patient and surgeon to hundreds of mGy of X-ray radiation. Thus, there is currently a need for methods of tracking catheter path inside the patient without relying on X-ray imaging techniques, which will significantly reduce the amount of radiation that patients and surgeons are exposed to.

2. Software Goal

With our developed extension in 3DSlicer, we are able to provide image guidance to surgeons while navigating the endovascular catheter. This endovascular catheter will be electromagnetically tracked, and after performing registration, we are able to overlay the position of the catheter on top of a pre-operative CT or X-ray scan of the patient, providing image guidance to surgeons. This way, we can eliminate the need for fluoroscopy during the procedure, greatly reducing the amount of radiation surgeons are exposed to.

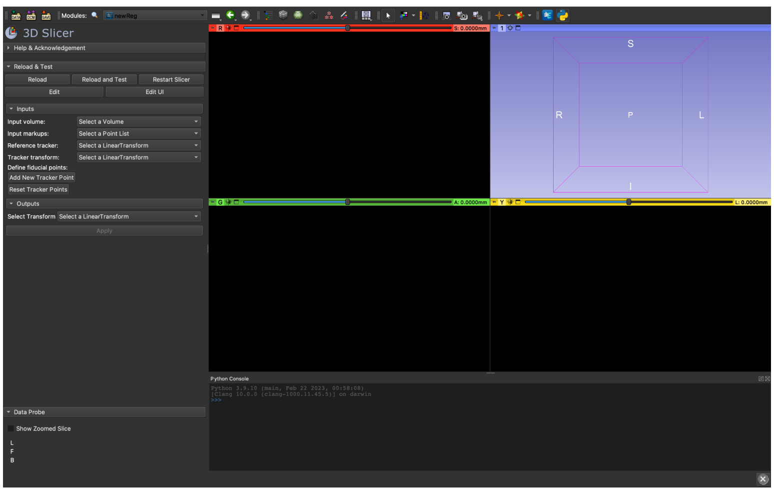
II. Design Overview

In our extension named EM_Catheter, there are three separate modules, named newReg, pathReg, and EM_tracker. The module newReg performs fiducial point registration between the tracker and CT frame. The module pathReg performs path-based registration between the tracker and CT frame. The module EM_tracker provides a way of visualizing the catheter position.

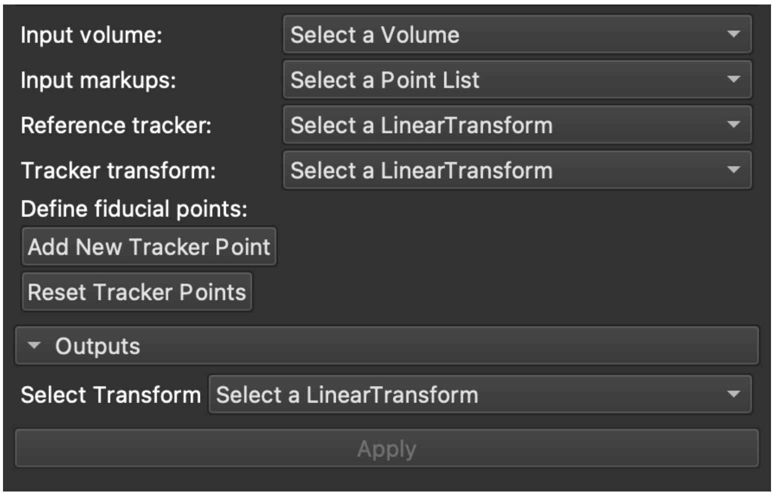
1. newReg: Fiducial Registration

1.1 Inputs

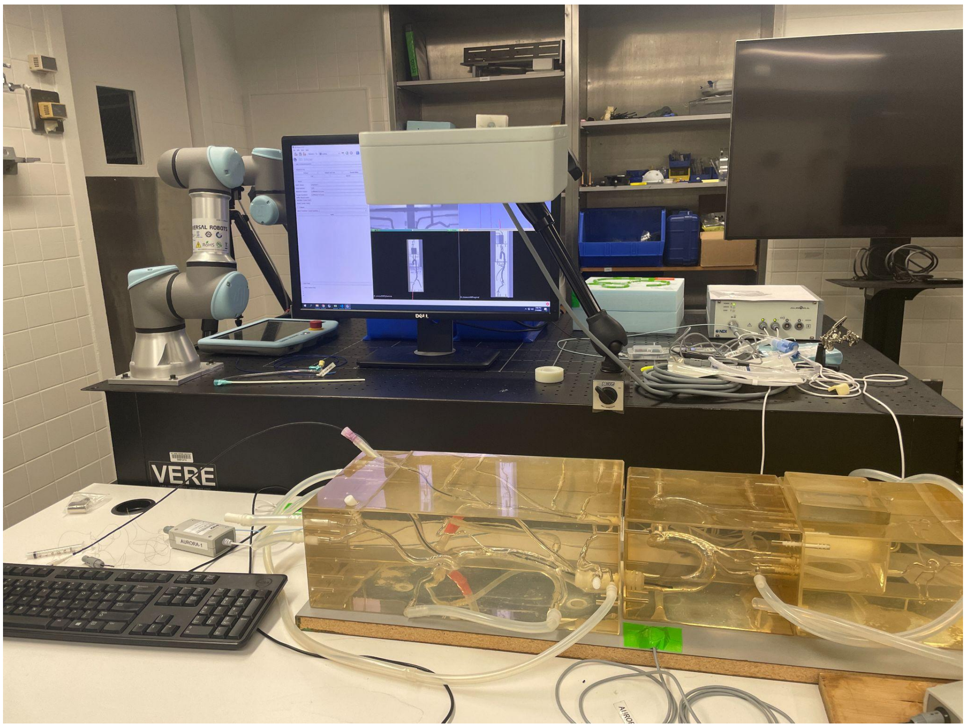
When a user first opens the module, this is the GUI that they see.



On the left, we have the input and output selections. Input volume refers to the CT image that they want to register to. Input markups refers to a 3DSlicer Markup Point List of fiducial points on the CT. If a user has yet to create such a list, they can upload their CT image as data, and navigate to the Markups module to create and save such a list.



When we developed this module, we were testing our concept on a phantom set up as shown below.



Since the EM field generated by the field generator wasn't sufficient to cover the entire phantom, we included a reference point, which is shown in the image taped down by green tape. This way, we don't have to re-perform registration when we move the field generator to cover the entire phantom. Thus, the Reference tracker in the input list refers to the LinearTransform of this reference marker. Finally, the tracker transform refers to the LinearTransform of the EM sensor that is attached to the tip of the catheter. For output, the computed transformation between the tracker and CT frame is saved to a LinearTransfrom Node. Thus, users have to select the LinearTransform node that they would like to store it in.

1.2 Work Flow

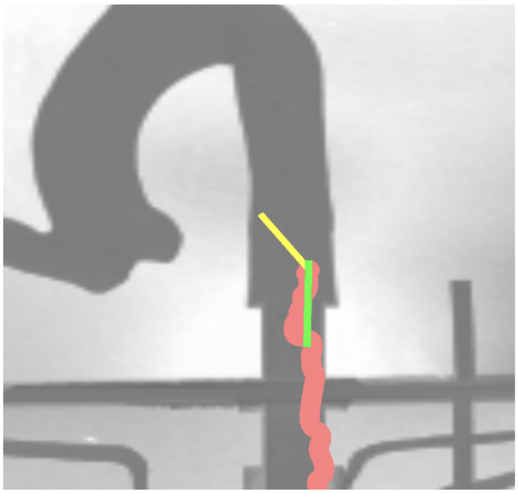
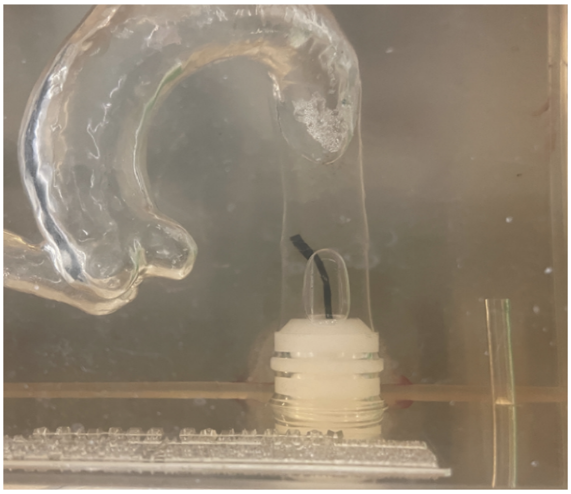
Once all the inputs are complete, the user can then perform registration. Following the sequence of the fiducial points stored in the Markup Point List, the user can touch the tip of the catheter to the corresponding position on the patient/patient, and click Add New Tracker Point. If an error was made, such as touching the wrong fiducial point, the user can click Reset Tracker Points and start over. Once all the points are added, the user can click Apply, and the transformation will be computed and stored in the given LinearTransform Node. In addition, this LinearTransform will automatically be applied to the catheter position, so there is no need for the user to rearrange the hierarchy manually in 3DSlicer.

1.3 Algorithms

The algorithm that our module uses to compute the transformation is Arun's method.

1.4 Pros and Cons

The pros of this module, specifically, the pros of fiducial registration is that it is quick and accurate. As we can see below, the tracked position of the catheter matches up closely with the real position of the catheter in the phantom.

However, the con of this registration is that in reality, when we try to register to patients instead of phantoms, it is very hard to find distinct fiducial points on the patient body. Thus, this motivated us to develop the next module, pathReg, which performs path-based registration.

2. pathReg: Path-Based Registration

2.1 Inputs

The user will first need to push the catheter forward for some distance inside the phantom. Then we will map the catheter's path to the vessel's center line using the algorithm described below. The input of our algorithm is the pre-existing data of the center-line model and the path the user created initially. The initial path will be saved in 3D slicer MarkupCurves type and the algorithm will read it automatically.

2.2 Work Flow

Once all the inputs are complete, the user can then perform registration. After inserting the catheter into the body with sufficient length, it is possible to process the apply button of the module.Tthe transformation will be computed and stored in the given LinearTransform Node. In addition, this LinearTransform will automatically be applied to the catheter position, so there is no need for the user to rearrange the hierarchy manually in 3DSlicer.

2.3 Algorithms

We implemented a gradient descent based rigid registration method with the following approach.

1. We prepared the centerline model of the arteries from the CTA data, by
   a. segmenting the 3D artery model from the CT scan
   b. using prebuilt modules in 3D slicer to acquire the centerline model from the 3D artery model.
2. We also acquired a sufficient long portion of the catheter trajectory using the markup curve generated by the EM_tracker module.
3. We provide an initial crude estimation that places the catheter trajectory close to the trajectory model. This can speed up the algorithm significantly.
4. Afterwards, the gradient descent algorithm is applied. The trained parameters are the 6 variables that define the linear transform. The learning metric used is the mean square error between each data point of the trajectory data and the closed point on the artery centerline model.
5. After the iteration count has reached, or the algorithm achieves < 3mm error, the gradient descent algorithm terminates, and the resultant transform is returned to 3D Slicer. The transform is also automatically applied to the tracker transform data.

2.4 Pros and Cons

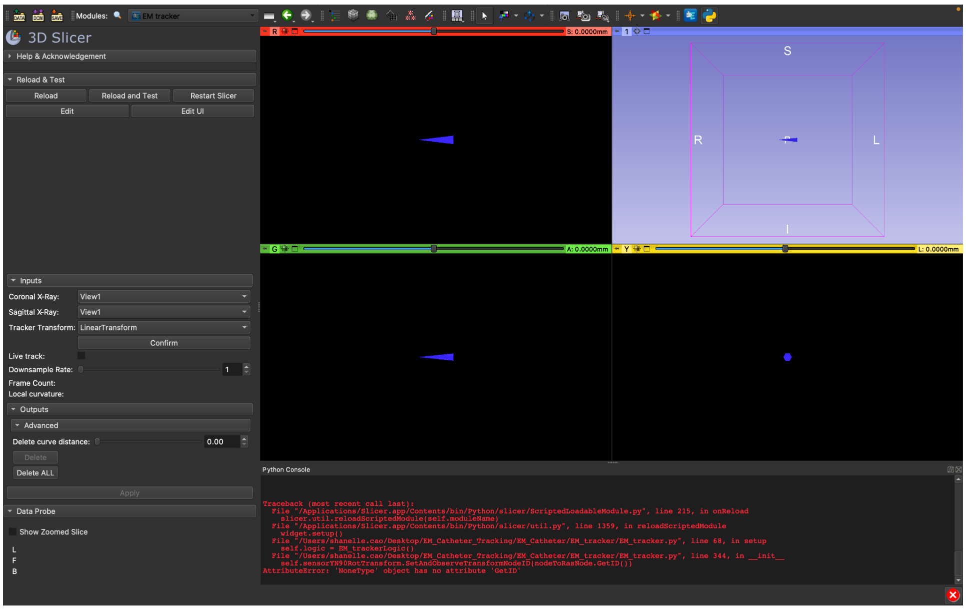
The most significant pro of this approach is the ability to perform registration without any registration operation or definition of fiducial points, both of which can be hard when operating on live patients without well defined external geometries. Registration may be possible during the initial insertion process. It promises a convenient and relatively fast registration approach that minimally disrupts the normal catheter workflow.

The most significant con of this approach is the potentially lower accuracy for the gradient descent algorithm, the error of which may be dependent on the length and geometric distinctiveness of the supplied tracker geometry. Additionally, the algorithm may fail to properly converge in certain cases, leading to failed registration.

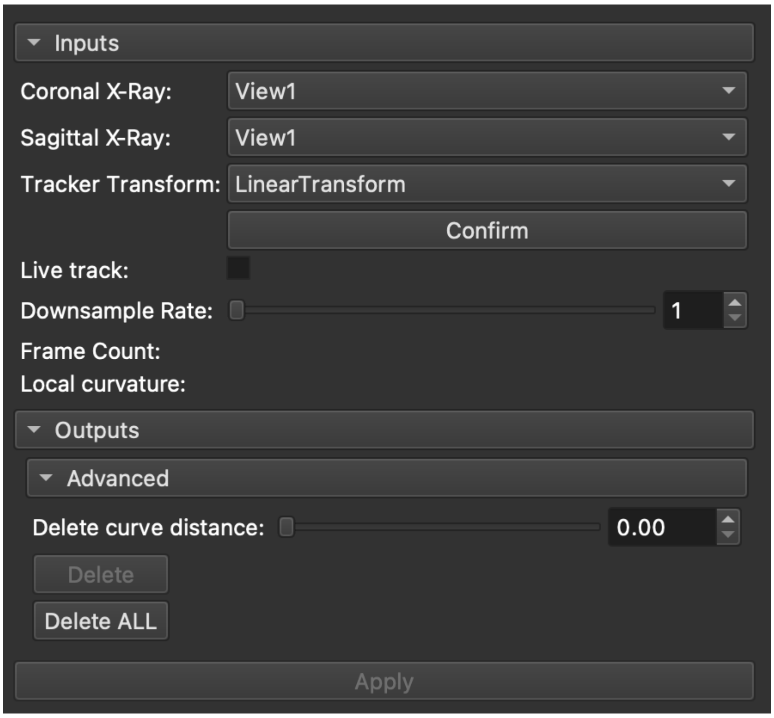
3. EM_tracker: Path Visualization

3.1 Inputs

When a user first opens the module, this is the GUI that they see.



On the left, we have the input selections. Currently there is no output for this module. There is a placeholder for outputting the curvature of the catheter path which can help surgeons to determine when they have gone the wrong way. This feature will be implemented in the future. Similarly, Downsample Rate, Frame Count, and Local curvature are also currently placeholders.

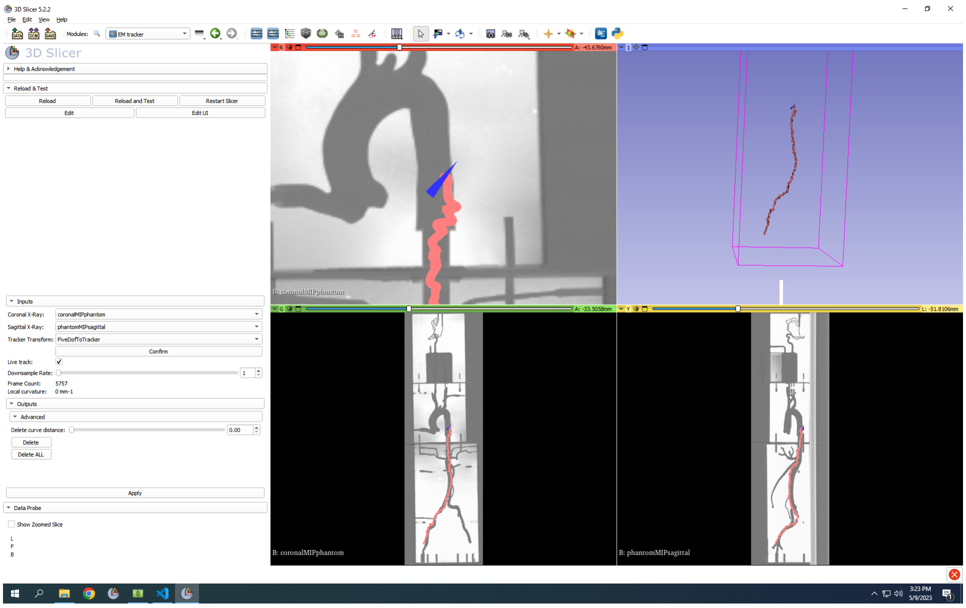


The reason that we chose X-ray images as input instead of CT volumes is that X-ray images provide clearer guidance information to surgeons compared to CT volumes. For CT volumes, the user is required to scroll through the slices to get an understanding of the context of the catheter position. With X-ray images, this information is clear with a quick scan.

Tracker Transform refers to the LinearTransform of the EM sensor that is attached to the tip of the catheter.

3.2 Work Flow

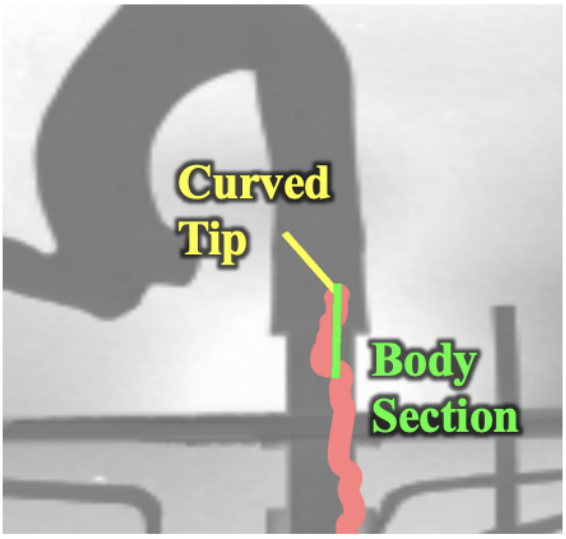
Once the user has completed all the inputs and hit Confirm, the four panels will adjust their views. The top left panel will show a zoomed in tracked view of the catheter tip. The top right panel will show the path taken by the catheter in empty space. The bottom left panel will show the whole coronal X-ray view of the patient with the tracked path overlaid. The bottom right panel will show the whole sagittal X-ray view of the patient with the tracked path overlaid.



With the view panels configured, the user can now check Live Track to track the position of the catheter. As the catheter moves, our module will plot points on the X-ray images to indicate the path taken by the catheter. If the surgeon retracts the catheter, the points will be deleted to indicate that the catheter is no longer at that position. This feature is important because we want to accurately reflect the path of the catheter so that we can provide realistic information to surgeons.

As we can see in Figure X, on the top left panel, we also show the orientation of the catheter tip. This is especially important because the tip of the catheter is curved, as shown below, to help surgeons turn the catheter into a branched vessel. We want our module to retain this information so we can provide useful image guidance that is able to replace fluoroscopy.

The single arrow indicates the direction directly obtained from the 5DOF sensor. Additionally, we developed another version of visualization that better models the catheter tip as shown below. Since the sensor is 5DOF, we computed the position of the catheter body from the catheter tip.



3.3 Algorithms

For the path deletion algorithm, we utilized code that was written previously by a master student for this project.

The body of the catheter is computed as follows. We assumed that the catheter is sufficiently rigid, such that the actual body pose will align with the trajectory formed by recent EM sensor readings in a short range. Therefore, we take the adjacent sensor readings to the current sensor reading (within 1cm range), and fit a vector to the trajectory using a PCA based method. This would give an efficient estimation of the current body orientation.

3.4 Pros and Cons

The pros of this module is that surgeons are able to clearly visualize the catheter as they navigate through the patient's body.

The cons of this module is that currently the module relies on fiducial registration. For path-based registration, it requires the catheter to be navigated blind for a first section to obtain enough data for registration. In addition, we have not tested the performance of this module with increased error from path based registration. Finally, since the path is simply plotting the tip of the catheter, as the tip spins, the path is shown as jagged, where in reality the body of the catheter is smooth.