

smiths / aorta Private

generated from smiths/capTemplate

[Code](#)
[Issues](#) 8
 [Pull requests](#)
[Discussions](#)
[Actions](#)
[Projects](#) 1
 [Wiki](#)
[Security](#)

...

01425ec886 ▾

...

aorta / README.md

JovieL25 Update README.md ✓

[History](#)

2 contributors

Aorta Geometry Reconstruction

Developer Names: Jingyi Lin

add credit to Kailyn

Date of project start: 2022/09/12 ← update

This project ~~aims to~~ provides a software tool to semi-automatically segment out a 3D aorta geometry from a chest CT scan. It uses the ~~3D Slicer software~~, and provides a 3D Slicer extension in the source folder (SlicerExtension/AortaGeometryReconstructor) to accomplish the goal.

User must first install the 3D Slicer software, import the extension in the source folder, and load the CT scans in DICOM file format to 3D Slicer.

This is the ~~link to the Slicer Welcome tutorial~~, a description of

~~This link described the 3D Slicer user-interface~~, information on the

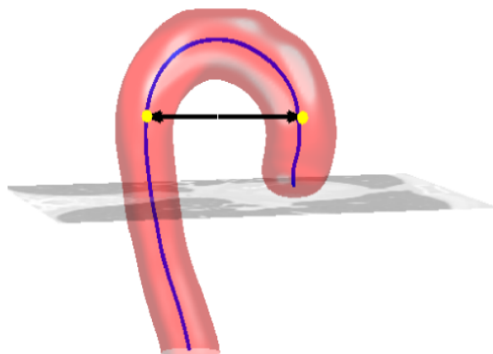
~~About Volume rendering module~~, and the

~~The 3D Slicer's developer guide~~.

Overview

Automatic aorta segmentation in thoracic computed tomography (CT) scans is important for aortic calcification quantification and to guide the segmentation of other central vessels. The work to manually segmented region of interest can be time-consuming and repetitive, and there are many automatic aorta segmentation algorithms posted.

This project implemented one of the aorta segmentation algorithm as 3D Slicer extension module. The algorithm first need user to crop the volume of interest (VOI) by using 3D Slicer volume rendering module. Next, user needs to provide two important seeds, the centre point of descending Aorta and the centre point of the ascending Aorta, as shown on the image below.



by the yellow dots

A simple diagram about the workflow:



About 3D Slicer

3D Slicer is a open-sourced Desktop software that solves advanced image computing challenges with a focus on clinical and biomedical applications. 3D Slicer software provides a clean user-interface and a clear guide to develop highly customizable extension and modules.

The folders and files for this project are as follows:

docs - Documentation for the project

refs - Reference material used for the project, including papers

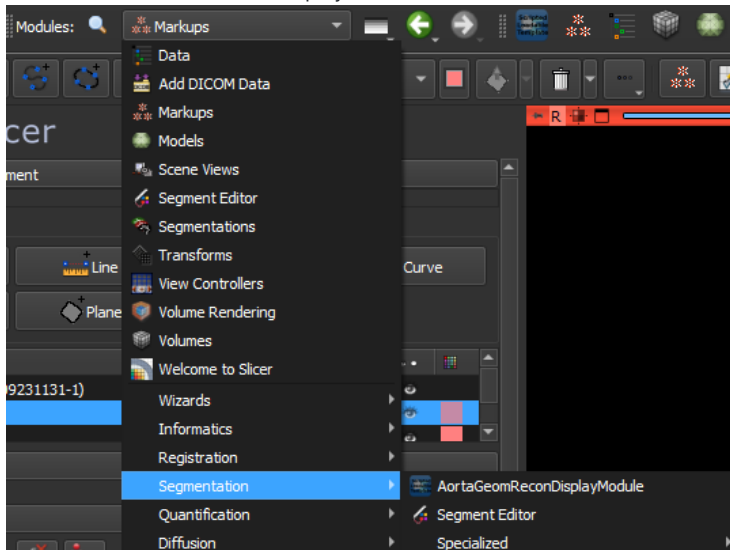
src - Source code

test - Test cases

etc.

To install 3D Slicer and import an extension:

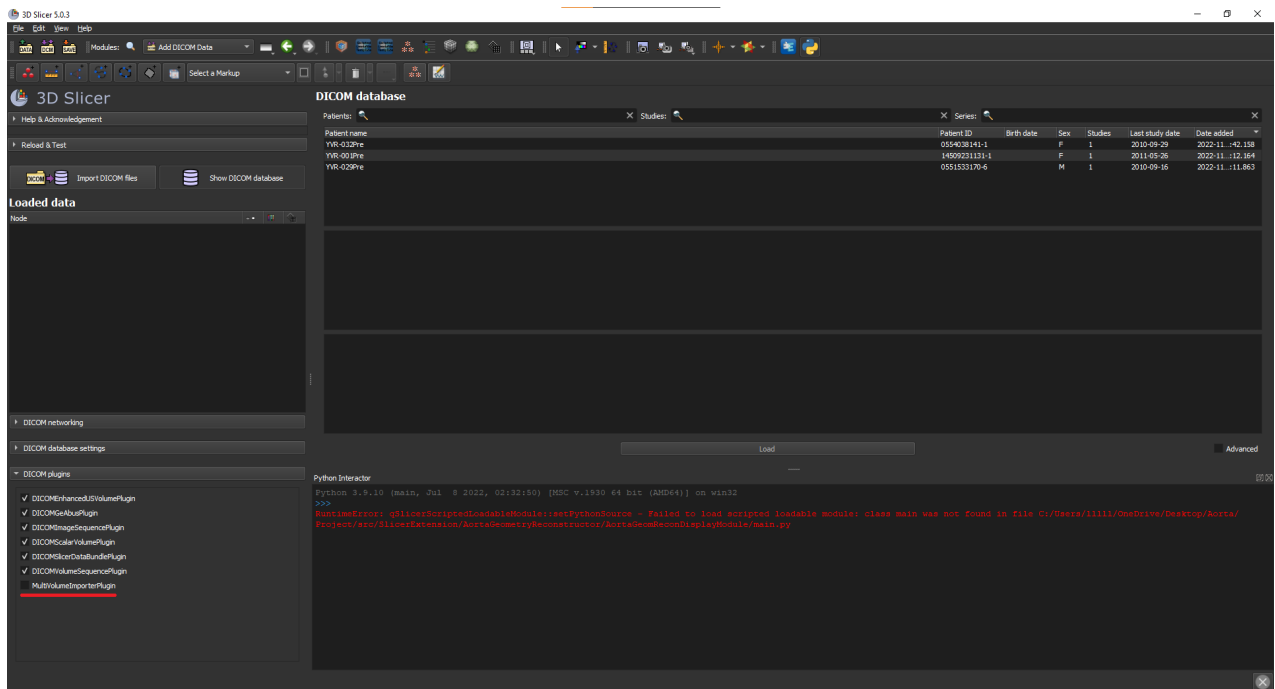
1. Download 3D Slicer from [here](#). Based the stable release, version 5.0.3.
2. Open Slicer application, select "Edit" from the bar menu, then "Application Settings" from the drop-down menu.
 - Select "Modules", and add the "AortaGeomReconDisplayModule" folder in the additional module paths. This folder is located in the [display-module](#) branch, in "src\SlicerExtension\AortaGeometryReconstructor\AortaGeomReconDisplayModule".
 - Restart the application as indicated to take into account the new module paths.
 - Find the "AortaGeomReconDisplayModule" module as shown below:



3. (Optional) In "Application Settings", and "Modules", drag "AortaGeomReconDisplayModule" to your favorite modules to access this module quickly from UI. Restart the application for the changes to the UI to appear.

88 lines (64 sloc) | 7.58 KB

1. Select "File" from the bar menu, then "Add DICOM Data".
2. Notice that the uploaded volume must be a scalar volume object to be view in Volume Rendering module. Therefore, to make sure that we are uploading volume as a scalar volume, in "Add DICOM data" module, unselected the plugin MultiVolumeImporterPlugin as shown below,



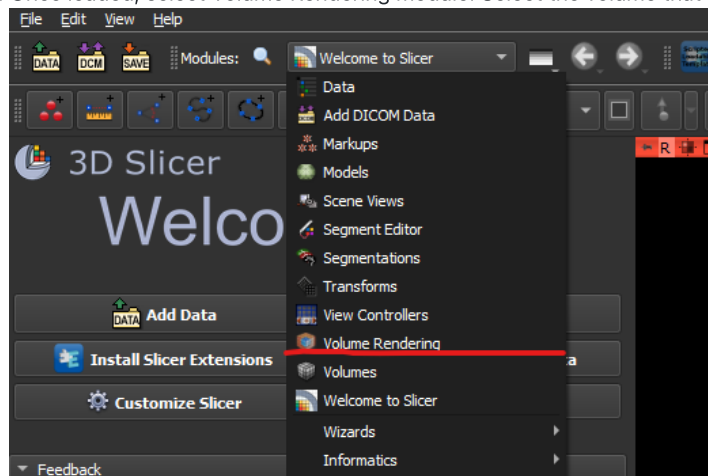
3. Select "Import DICOM files", and select one of the patient's folders.

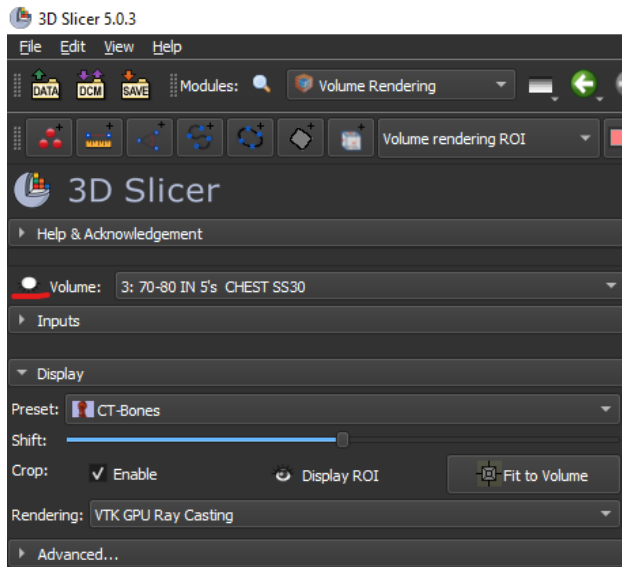
- Once loaded, 3D Slicer will keep the patient record in the DICOM database, and the user can easily reload DICOM data from the DICOM database.

To use Volume Rendering to crop a VOI:

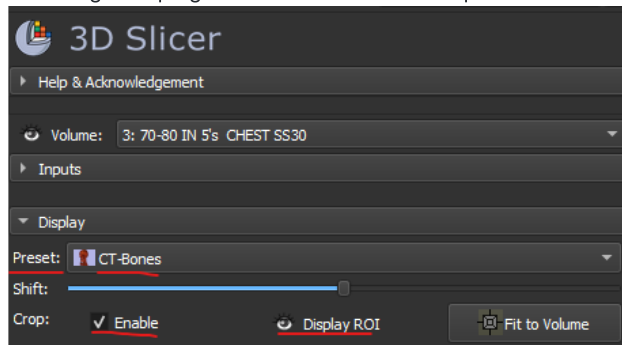
For this step we use the tutorial about use Volume Rendering module. extension for 3D Slicer.

- Load the dicom files as described in the section To load DICOM data.
- Once loaded, select Volume Rendering module. Select the volume that you want to crop, and click on the eye icon on the left.

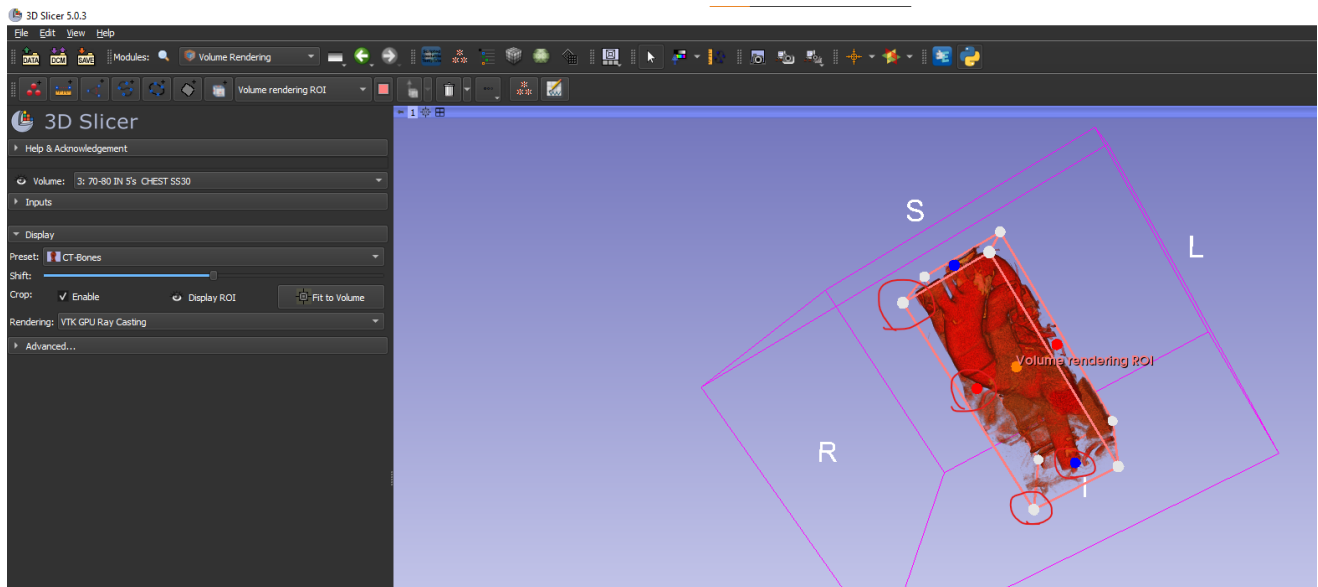




3. Within the module UI, under Display, enable Crop Display ROI, ^{drag} then select CT-Bones for preset. You should be able to see the 3D rendering on top right corner. You can use the points ^{control the} controls to crop.

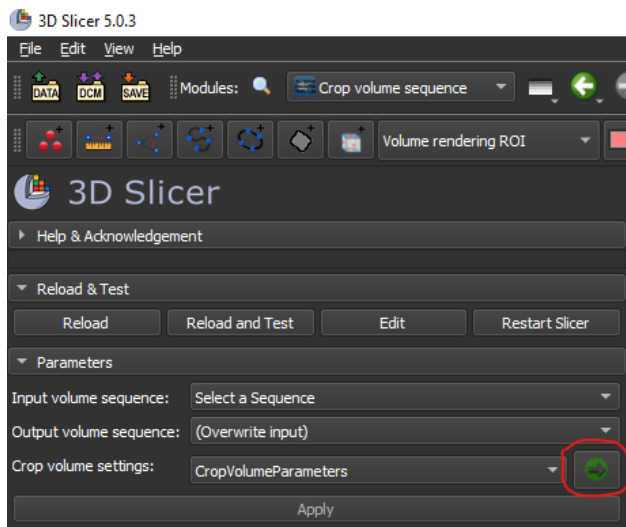
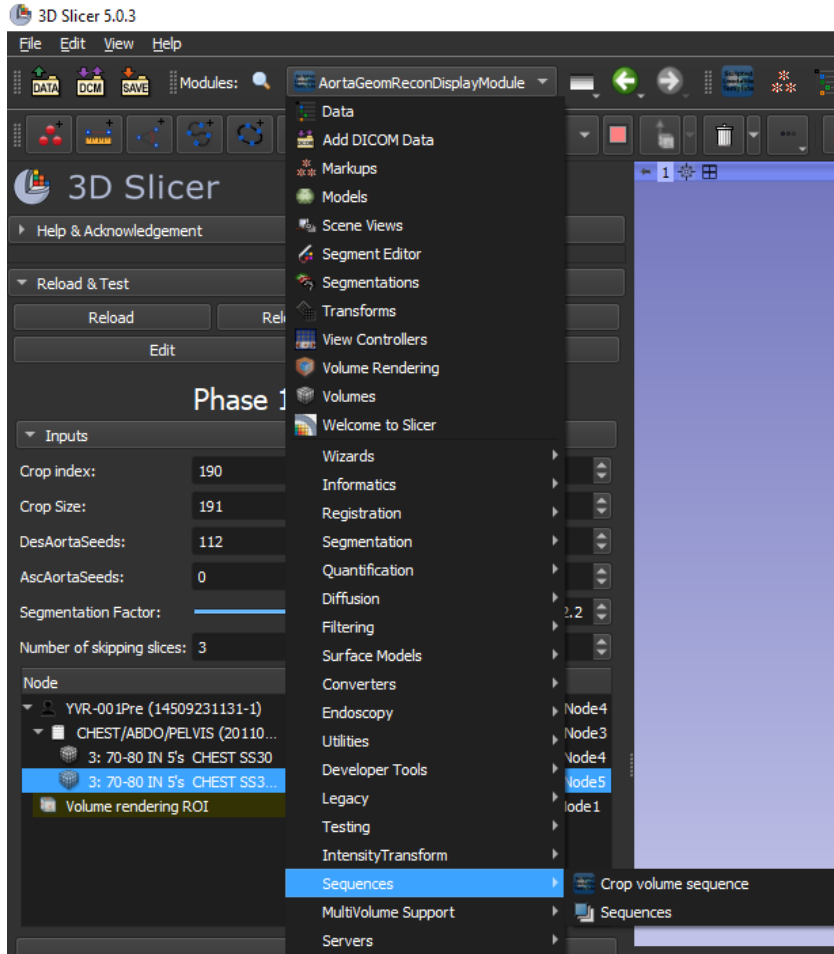


open the eye for ROI, and
the
You can use the sliders at the
top of each image to scroll through
the slices.



4. Change back to AortaGeomReconDisplayModule, you should also have a Volume rendering ROI, or vtkMRMLMarkupsROINode object.

5. Select a new module, Crop volume sequence, in Sequences category. Click on the green button which has an arrow pointing right.



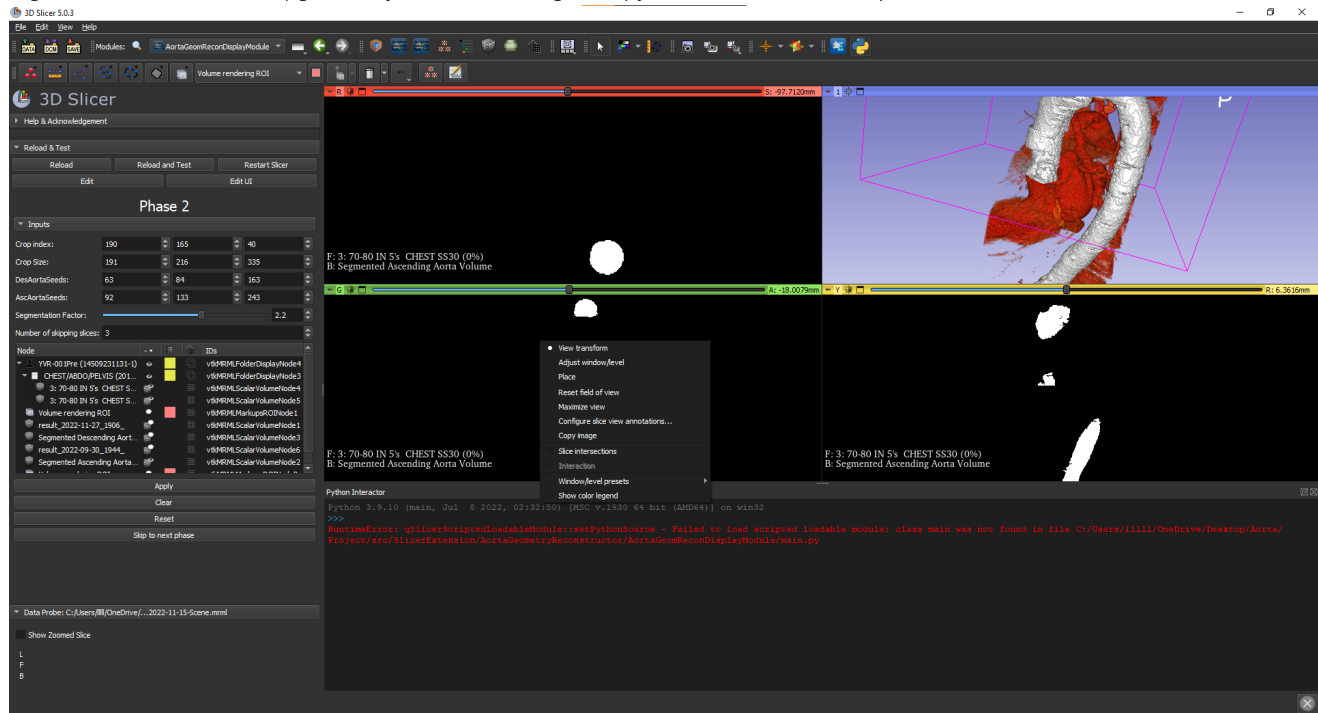
✓ this didn't appear to be an option - the input volume should be the cropped version?

6. For Input volume, select the volume to crop. For Input ROI, select the ROI we created with Volume Rendering module. For output volume, you can select create new volume or modify the original volume.
7. Click on Apply, and change back to AortaGeomReconDisplayModule, you should be able to find a new volume or the cropped volume.

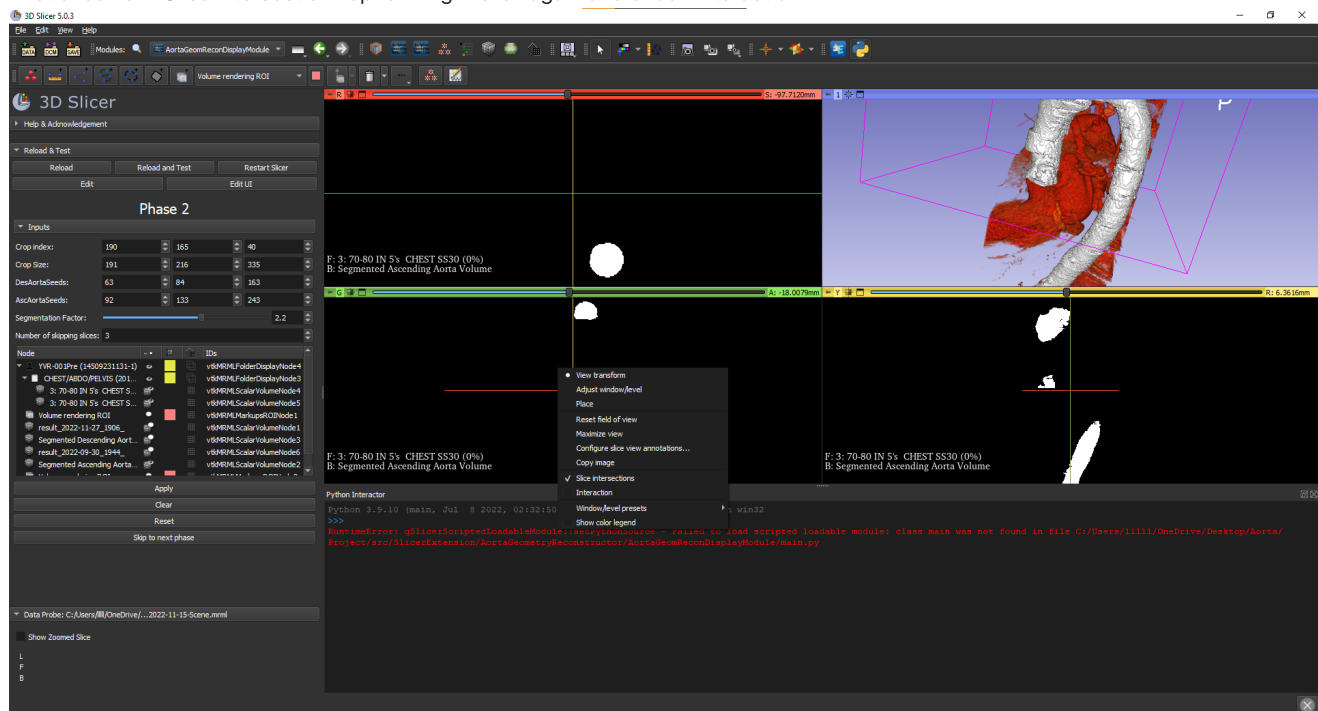
To get descending aorta seeds and ascending aorta seeds in phase 2 and phase 3

1. Make sure that you have done the cropping and have a cropped volume. If you are in phase 1 Crop Aorta, click on Skip to next phase to be at phase 2.

2. Right click on one of the red, green or yellow window image area, you should see a list of options as shown below.



3. First check on "Slice intersection" option. Right click again and check Interaction.



4. Hold on Mouse left button to drag and place the intersection point on the point of interest, as described in the image from Overview section.

I couldn't do this part

5. If in phase 2, you should see that the value of DesAortaSeeds is changing when moving the intersection point. In phase 3, AscAortaSeeds should be changing.

Additional tips to use this application:

1. The user can save and load an MRML scene object, which is used to store all types of data, including the loaded Dicom data, any inputs by the user on the UI, markups, etc.
2. We can use 3D Slicer's "Markups" module to draw control points and planes. These markup data are stored based on the Anatomical coordinate system, which can be independent of the Dicom data.

[Give feedback](#)