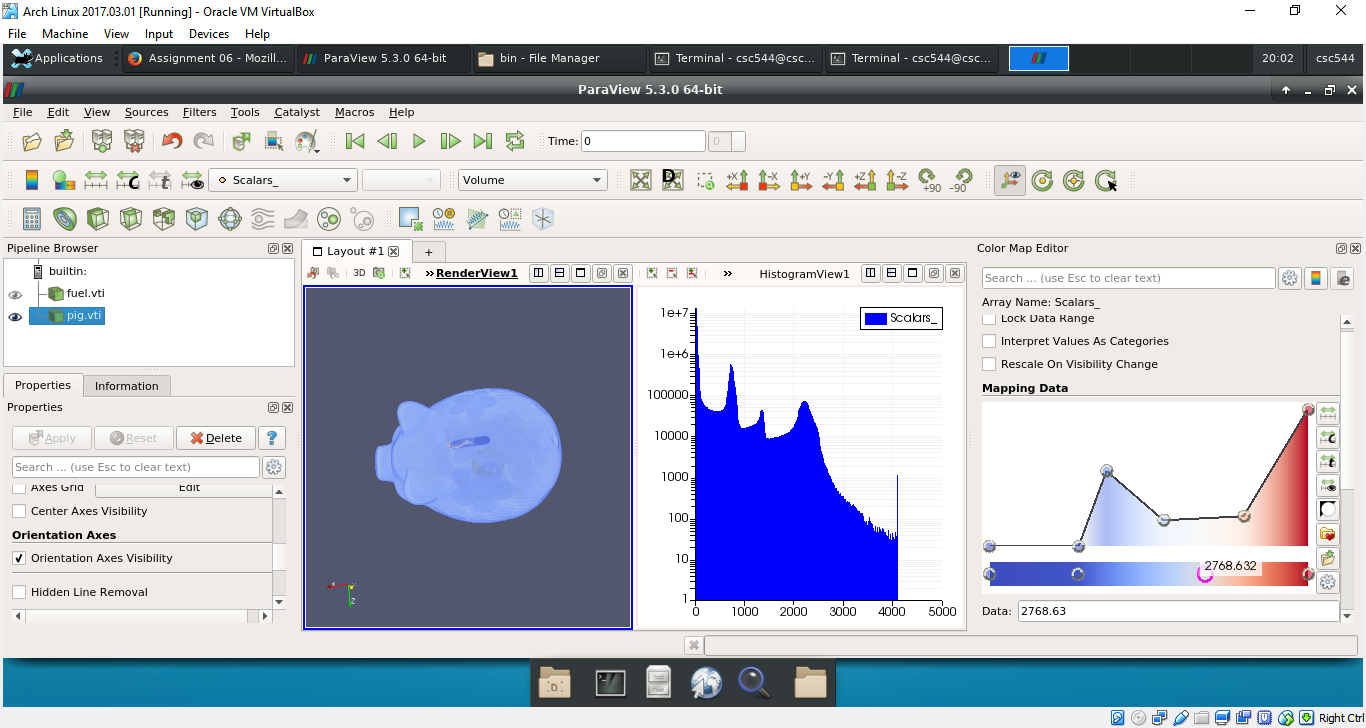
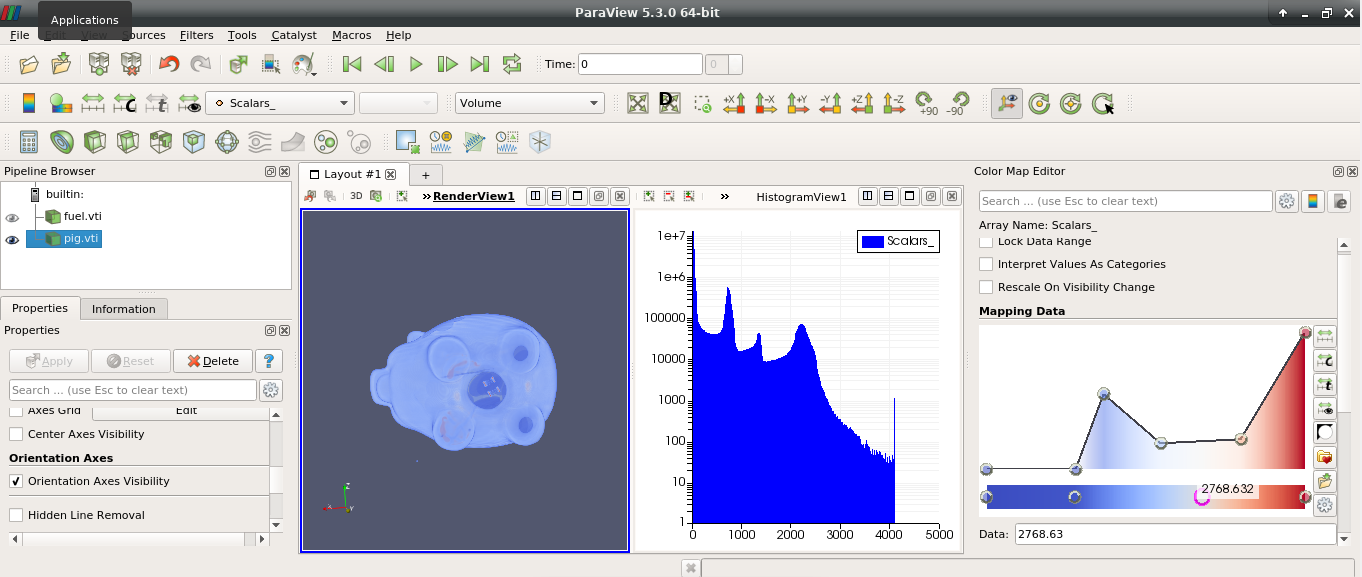
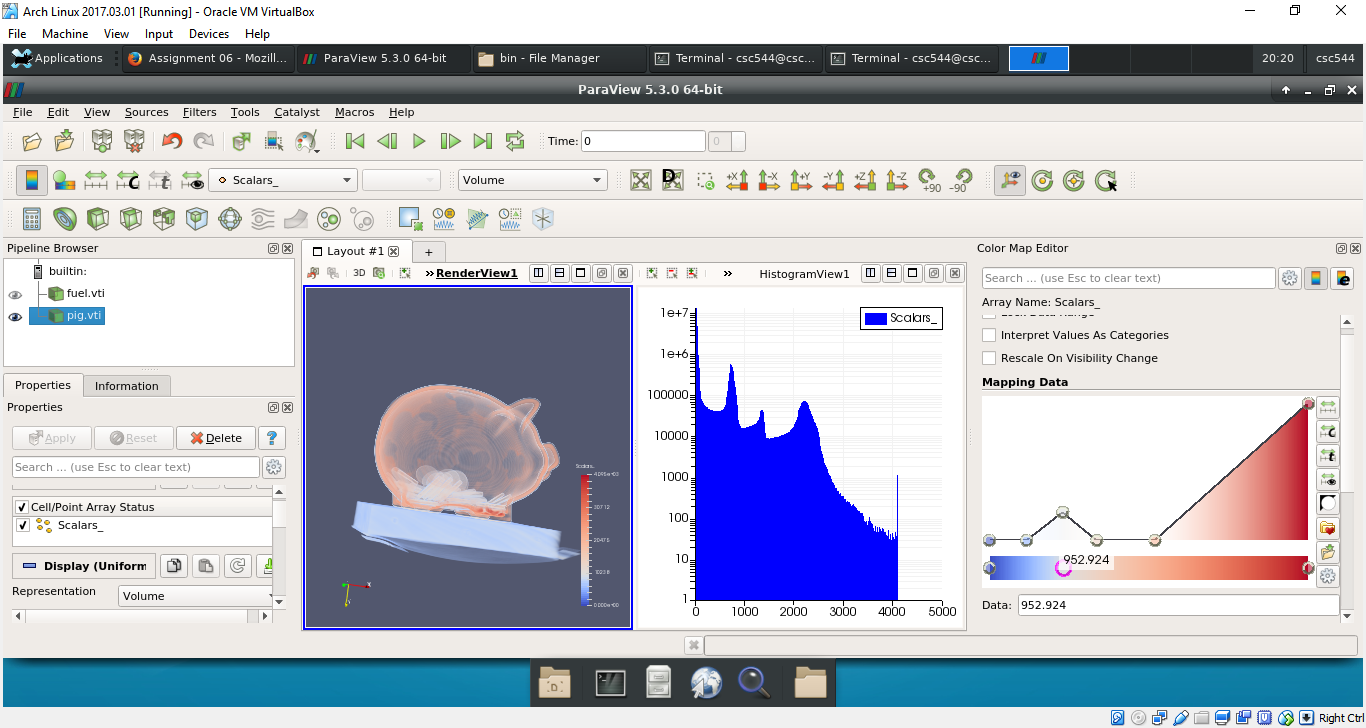
**A06P01 Report**

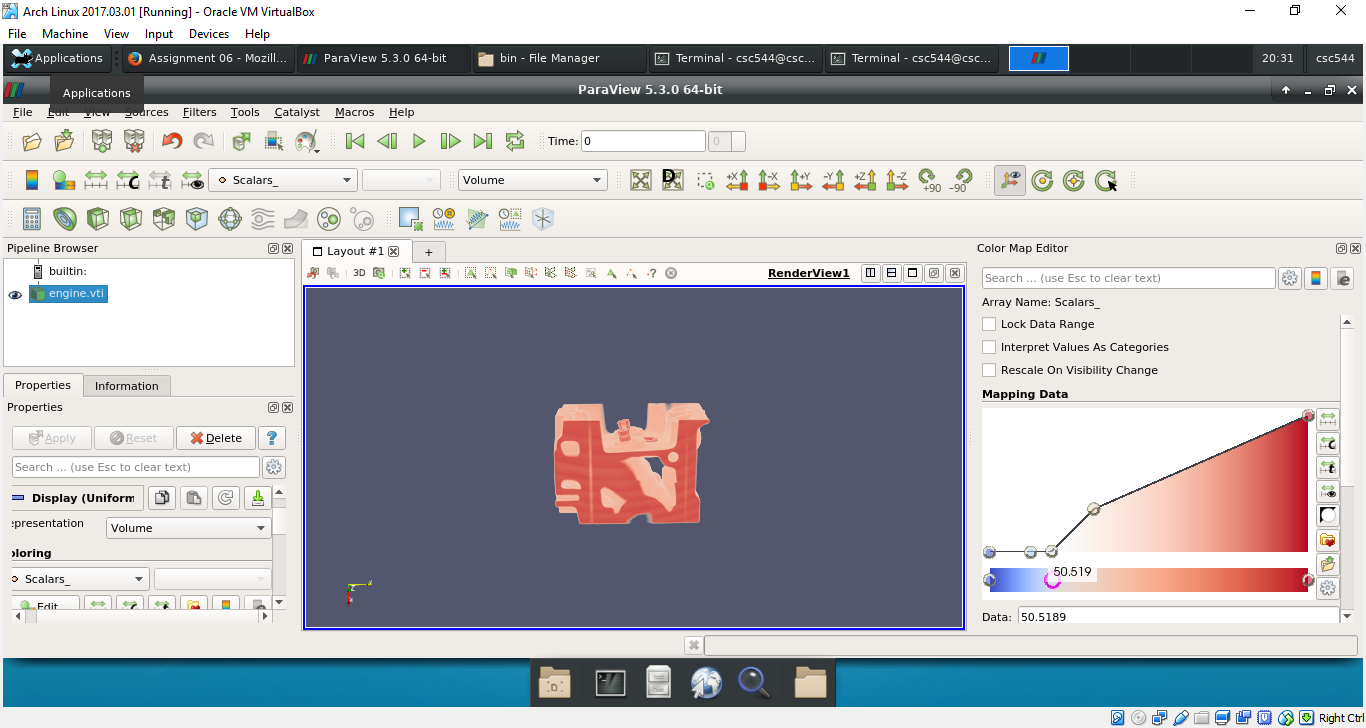
**Exploring Paraview**- I chose pig.vti, engine.vti and tooth.vti to explore in paraview. I loaded the dataset and selected volume rendering option. Then, I converted the histogram in a log scale since the data was concentrated in a very small region on the histogram. I used the same three datasets to explore in my tool so that I could draw a parallel between the functionality of Paraview and my tool. The best part about Paraview was its easy to use functionalities. The color and opacity transfer function widget was natural intuitive to use. No extra information was necessary. Also, it has multiple color maps to choose from and shows the data value for each point that we are assigning color and opacity to. However, the most important thing, in the setting color and opacity, the reset button was placed at the bottom most part of the left side window. So to reset both color and opacity (which was very often), I had to scroll all the way to the bottom every time. In my widget, I would like to give the reset button right next to the widget. Also, both color and opacity had the same reset button, even If I simply want to reset color I have to reset opacity. In my widget, I would give separate reset buttons for each.

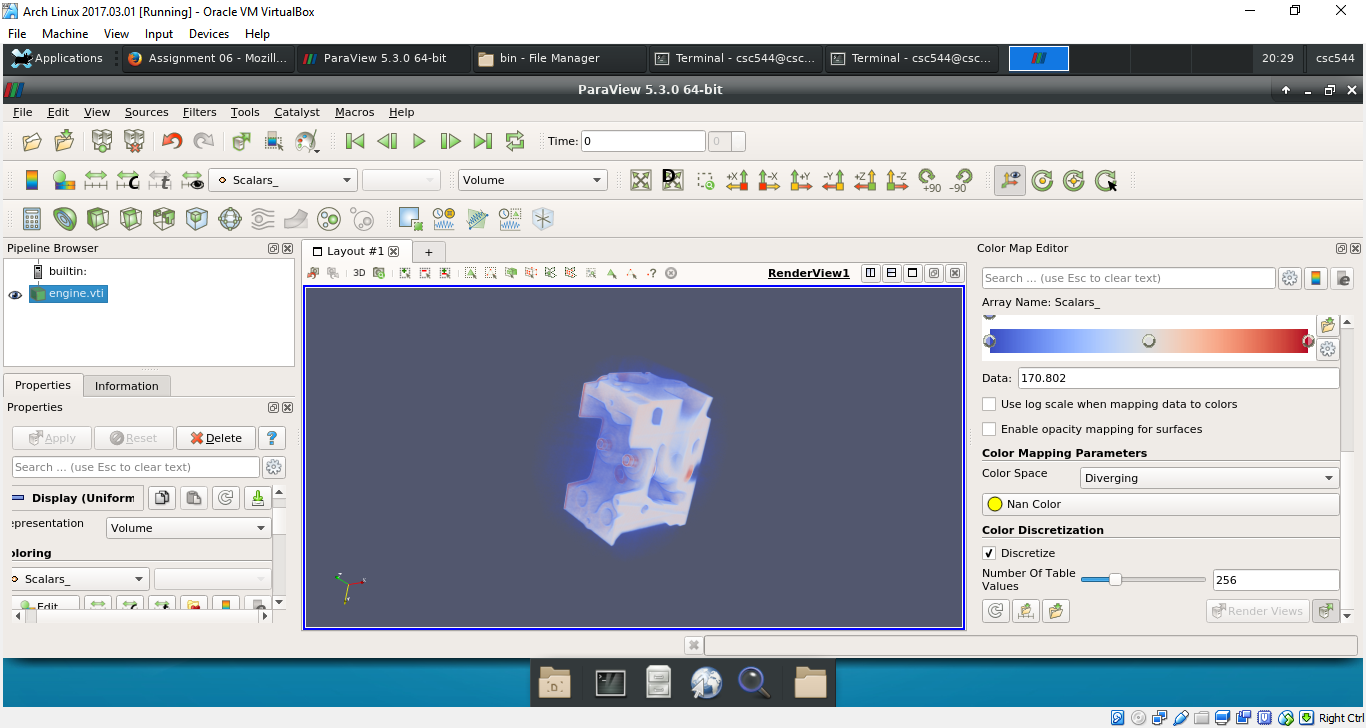
**Pig.vti**-





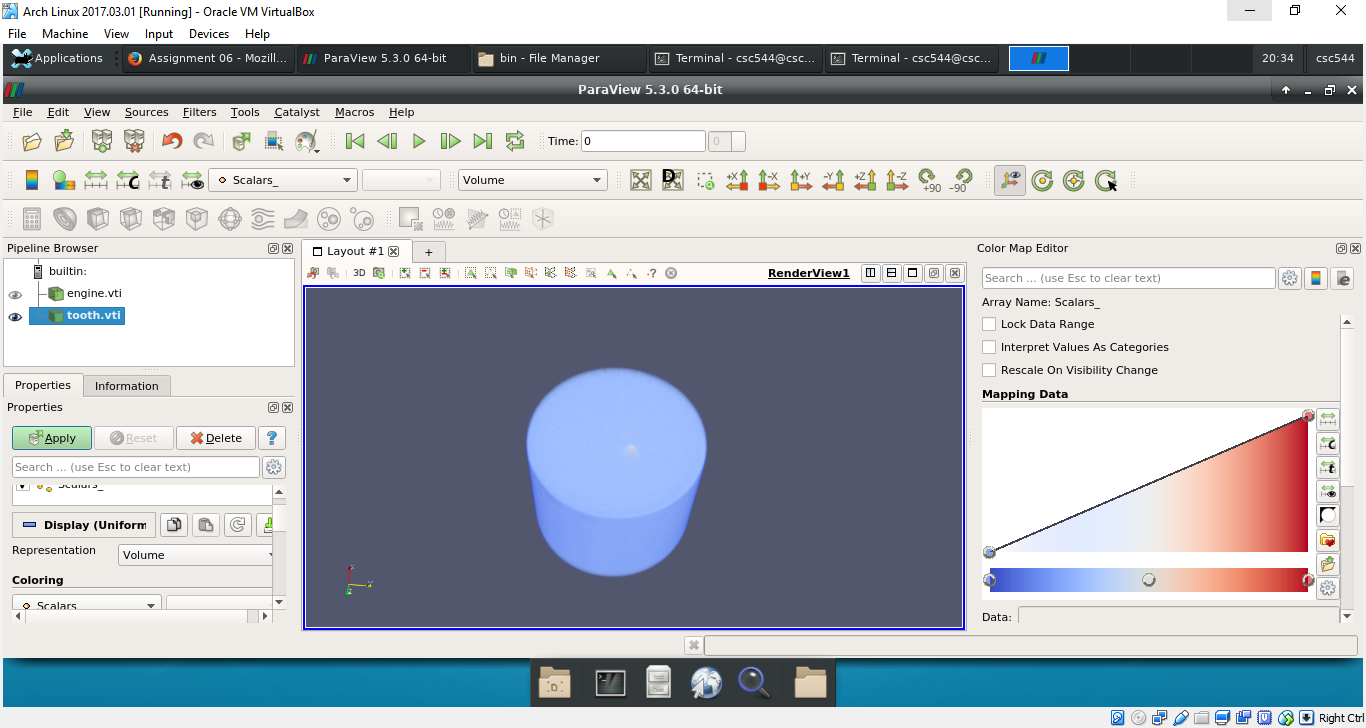


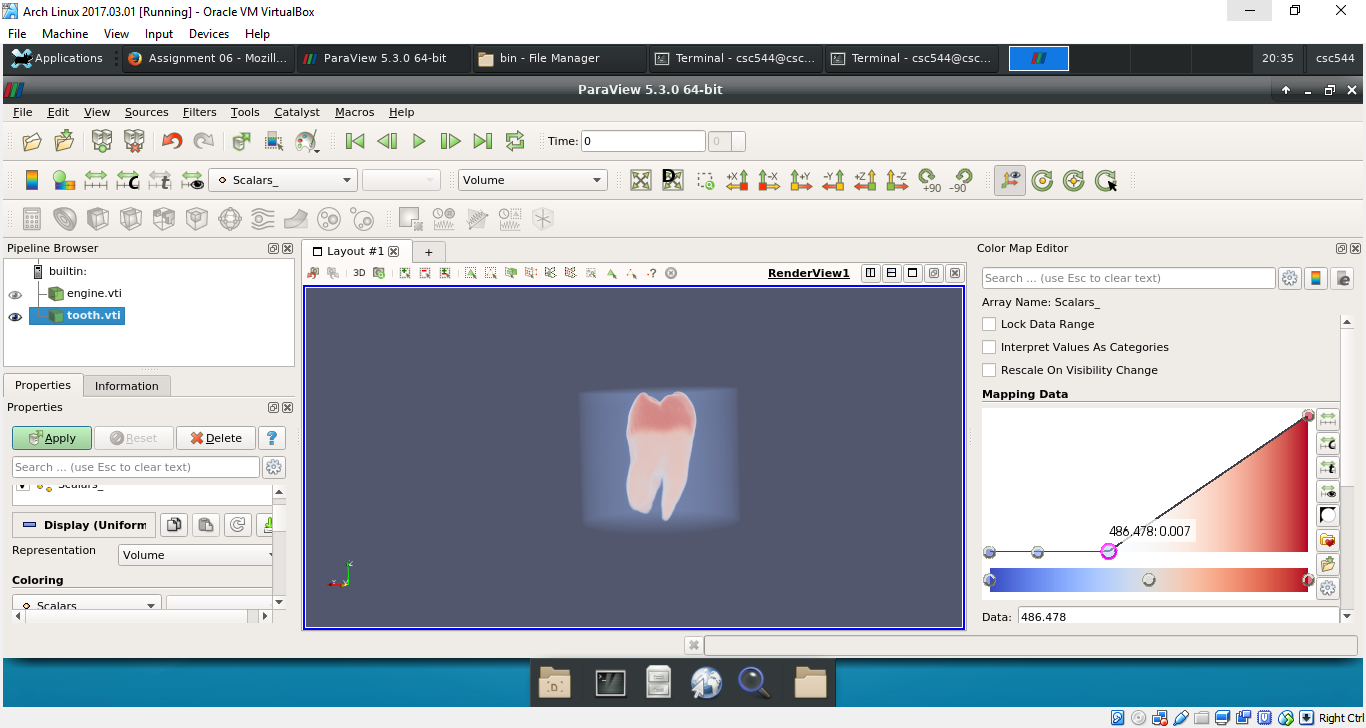
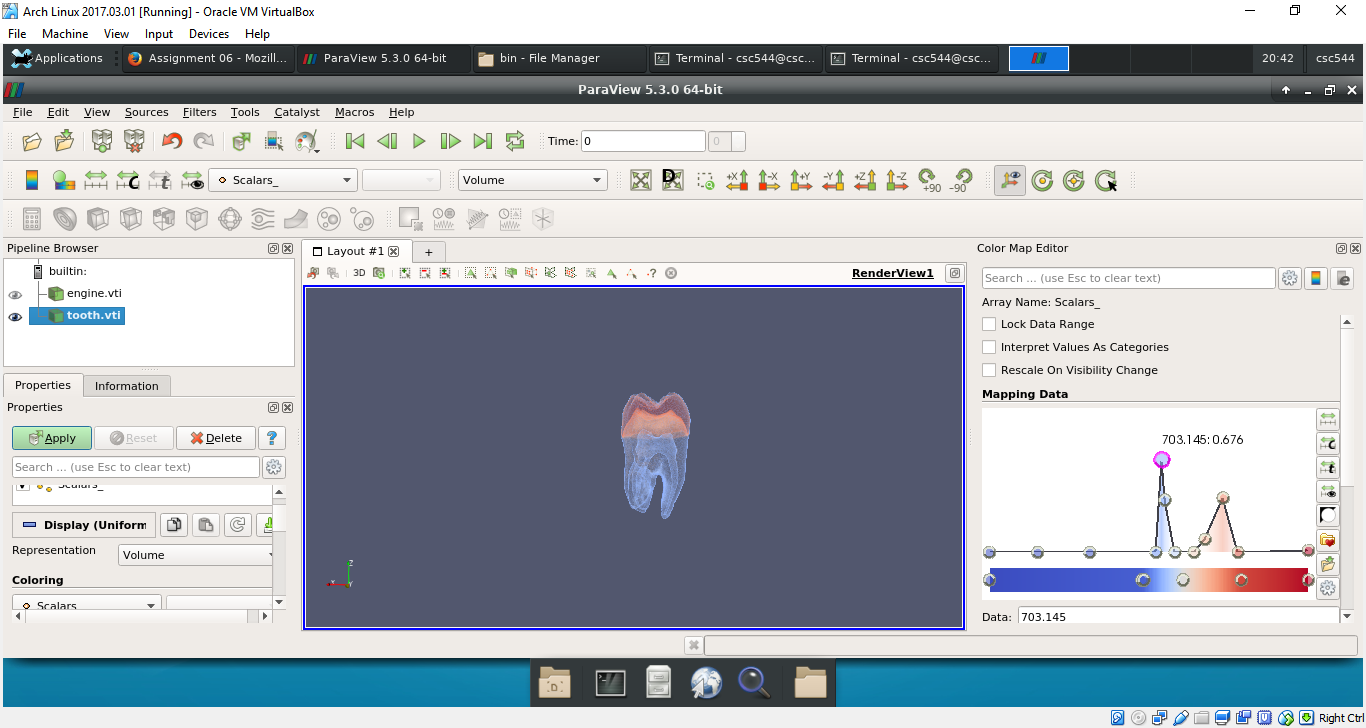
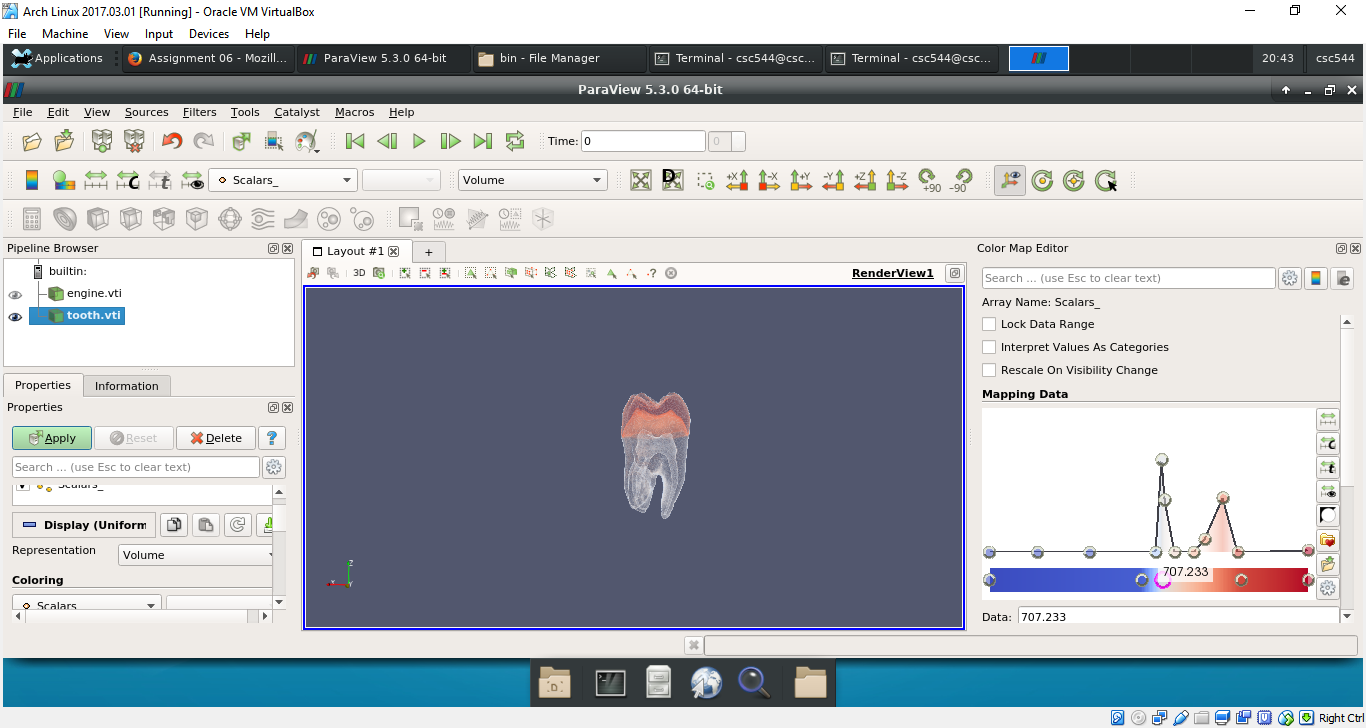
**Engine.vti-**

****

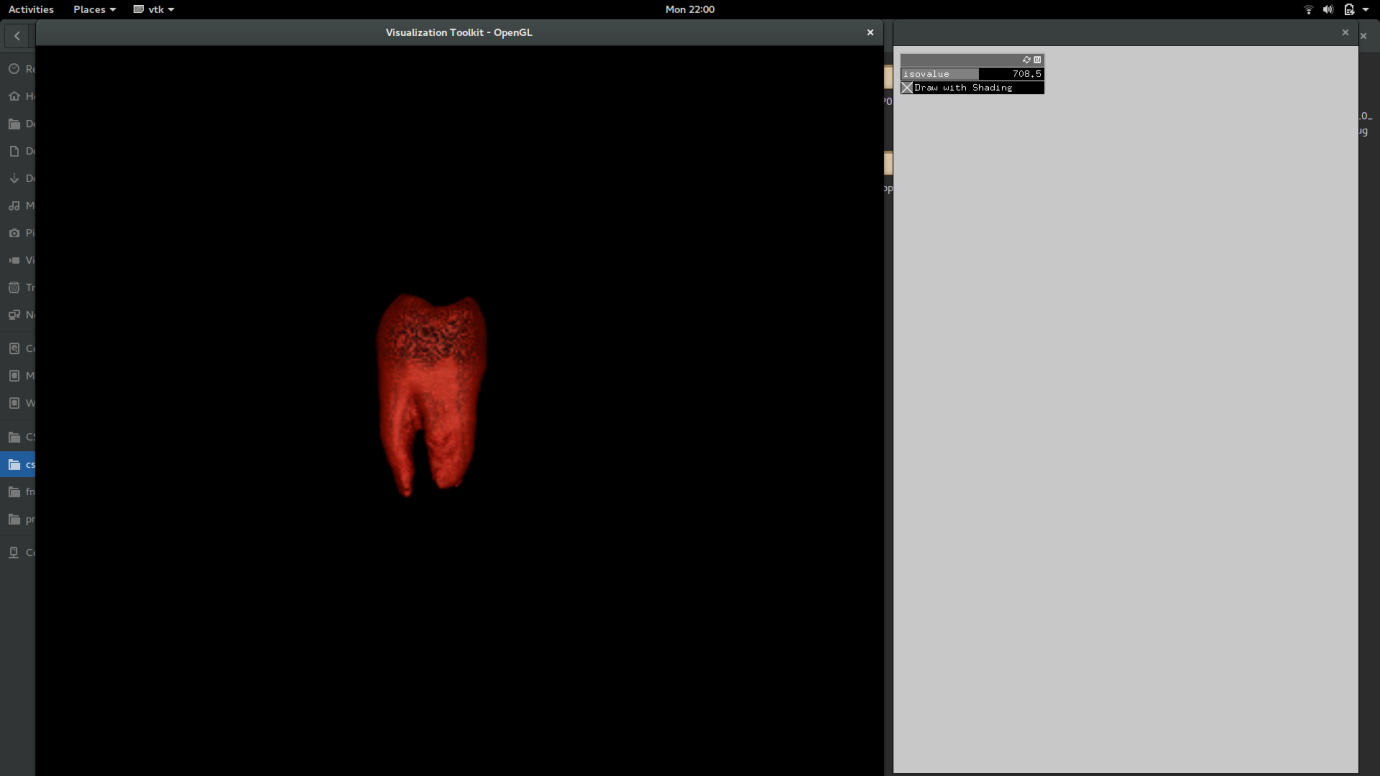
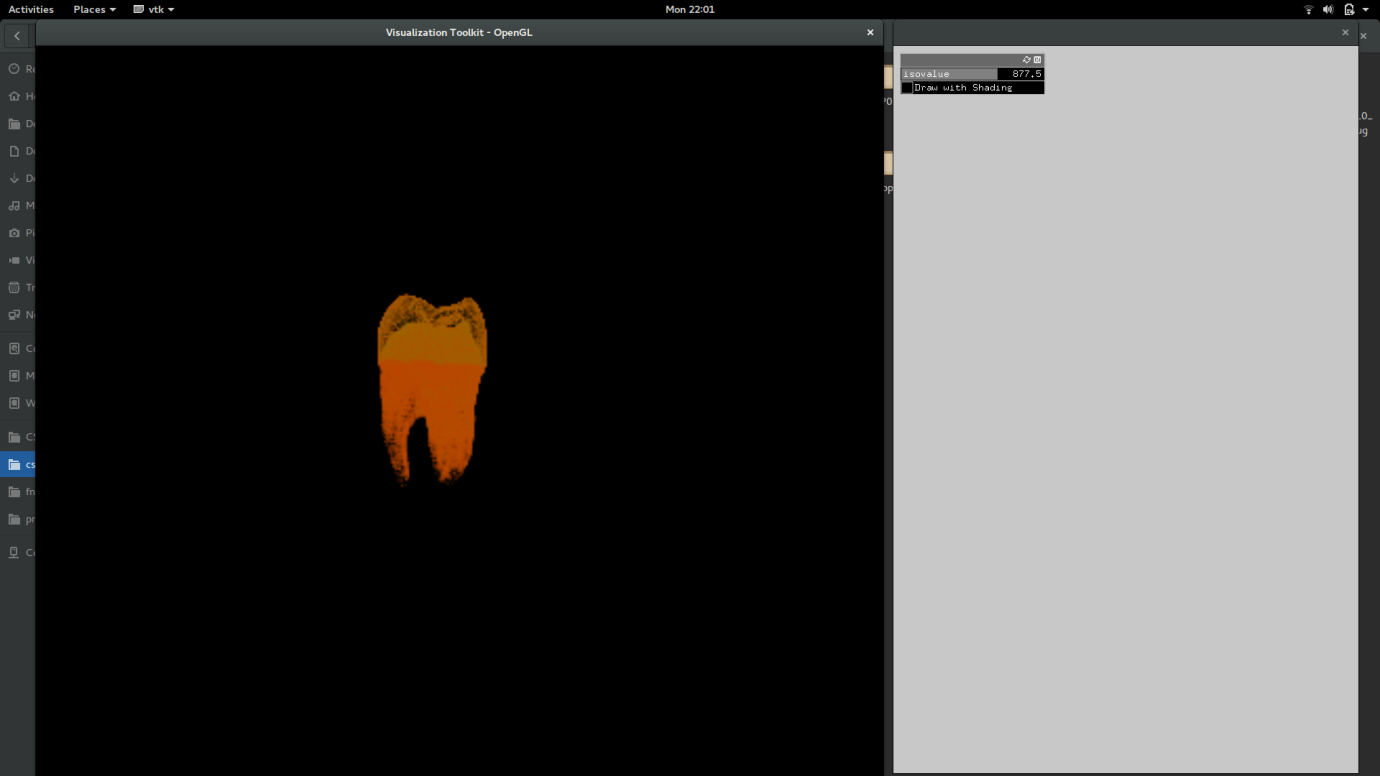
****

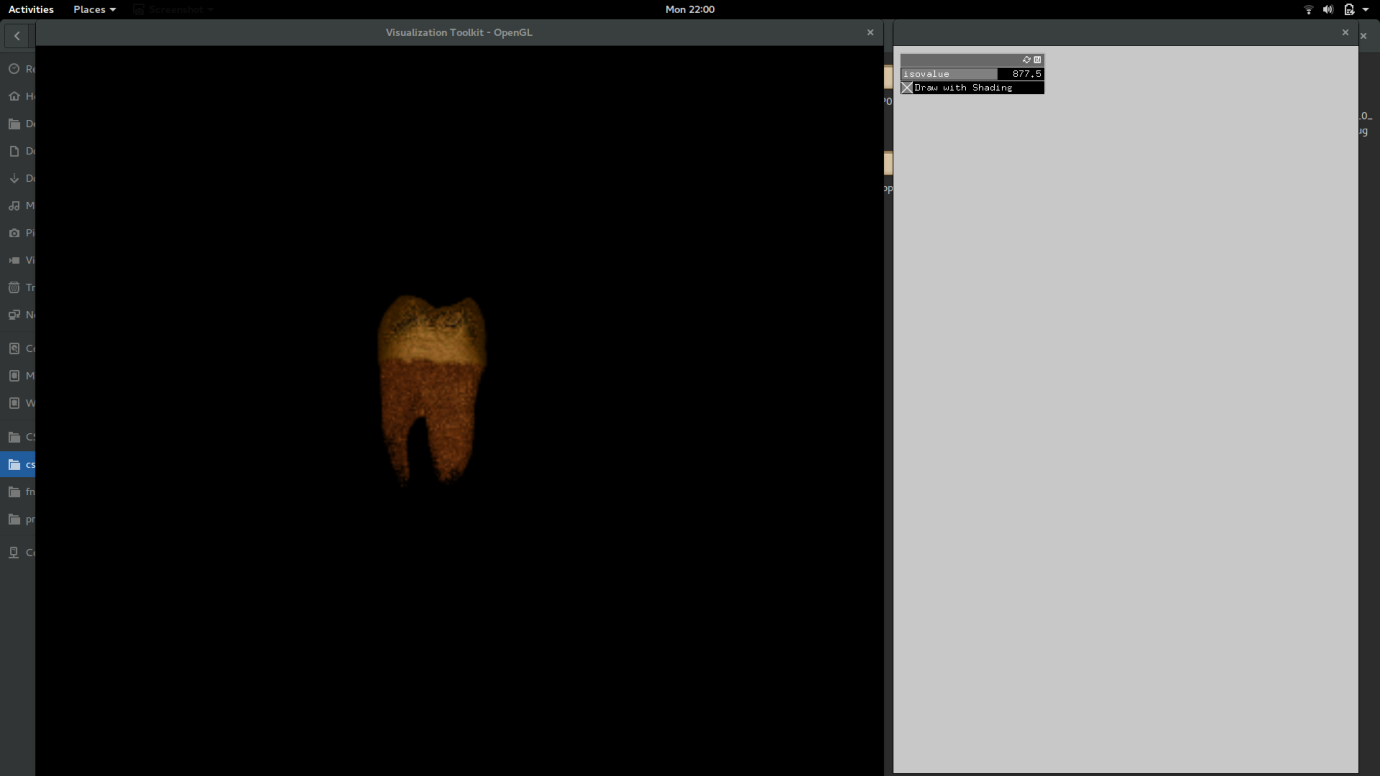
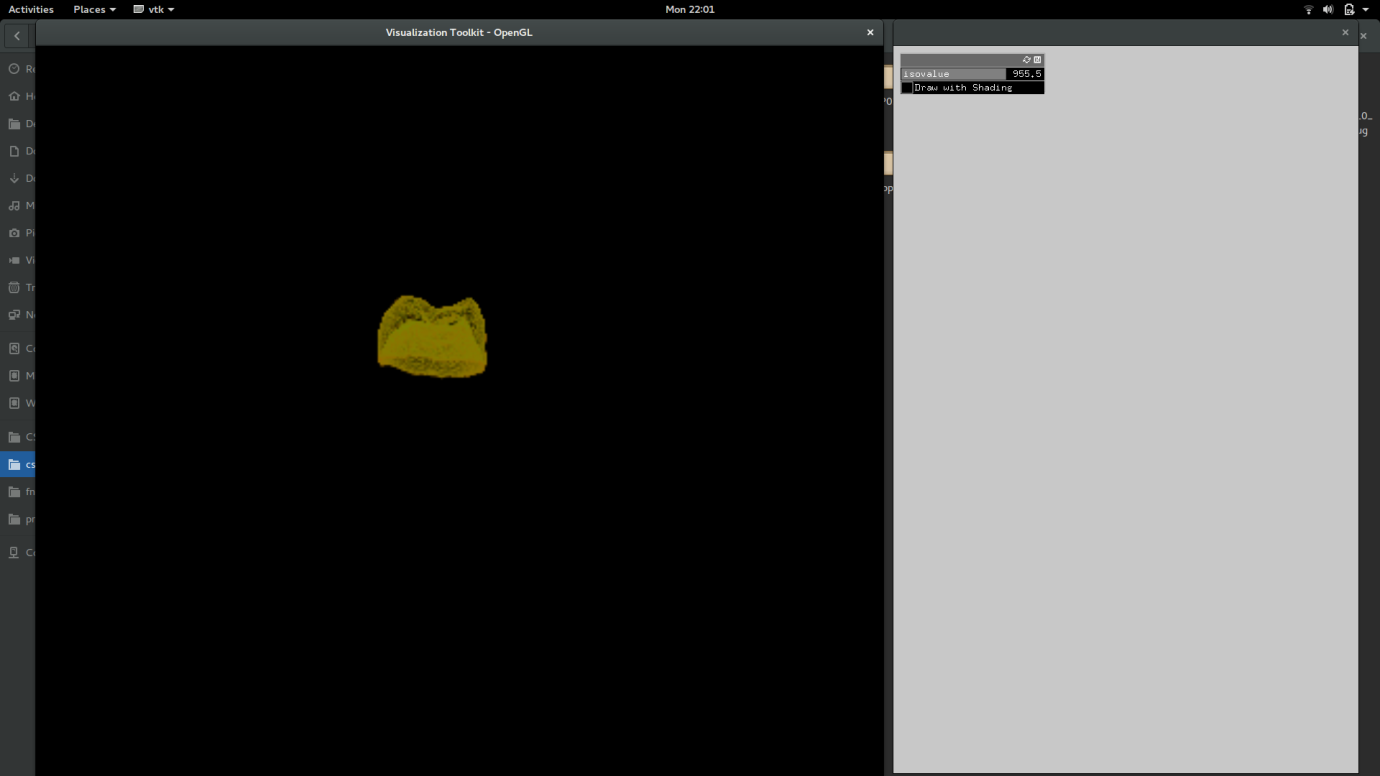
**Tooth.vti-**

****

****

**Running Volume Renderer-**

The volume renderer was very easy to use with just two functionalities. I explored tooth.vti using the tool. I was able to discover the internal crown of the tooth by adjusting the isovalue. I viewed the image with and without shading and found that shading adds an extra bit of context to the image, and therefore, shading was the most useful part of this tool. I have attached the screenshots below of my exploration. Of course, this tool was not as a Paraview since we cannot assign opacity and color to each scalar value. All it does is, play with lighting on the entire image and the isovalue, making it less efficient than transfer functions for volume rendering. Paraview and my tool helped me explore the same dataset better.

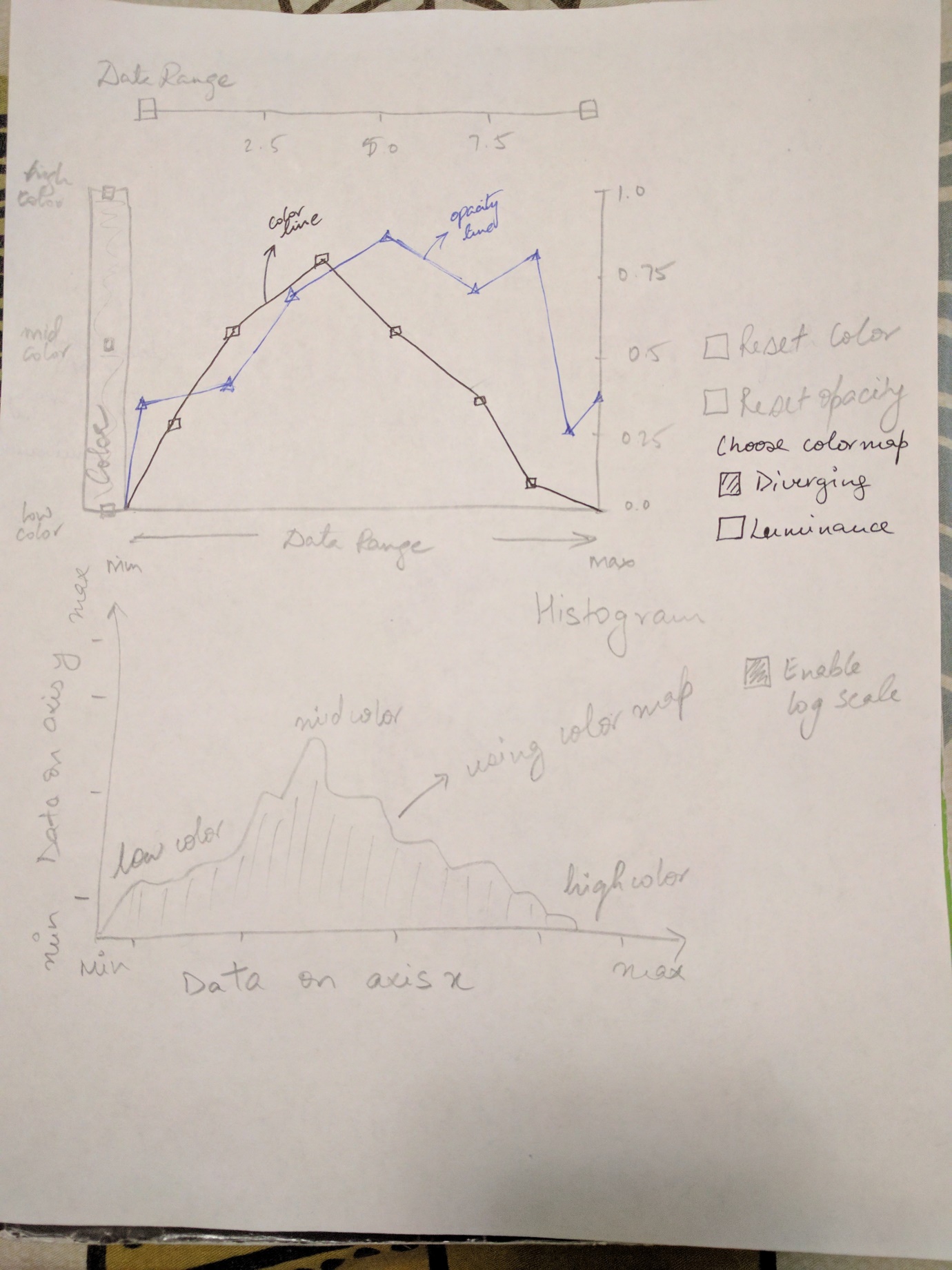
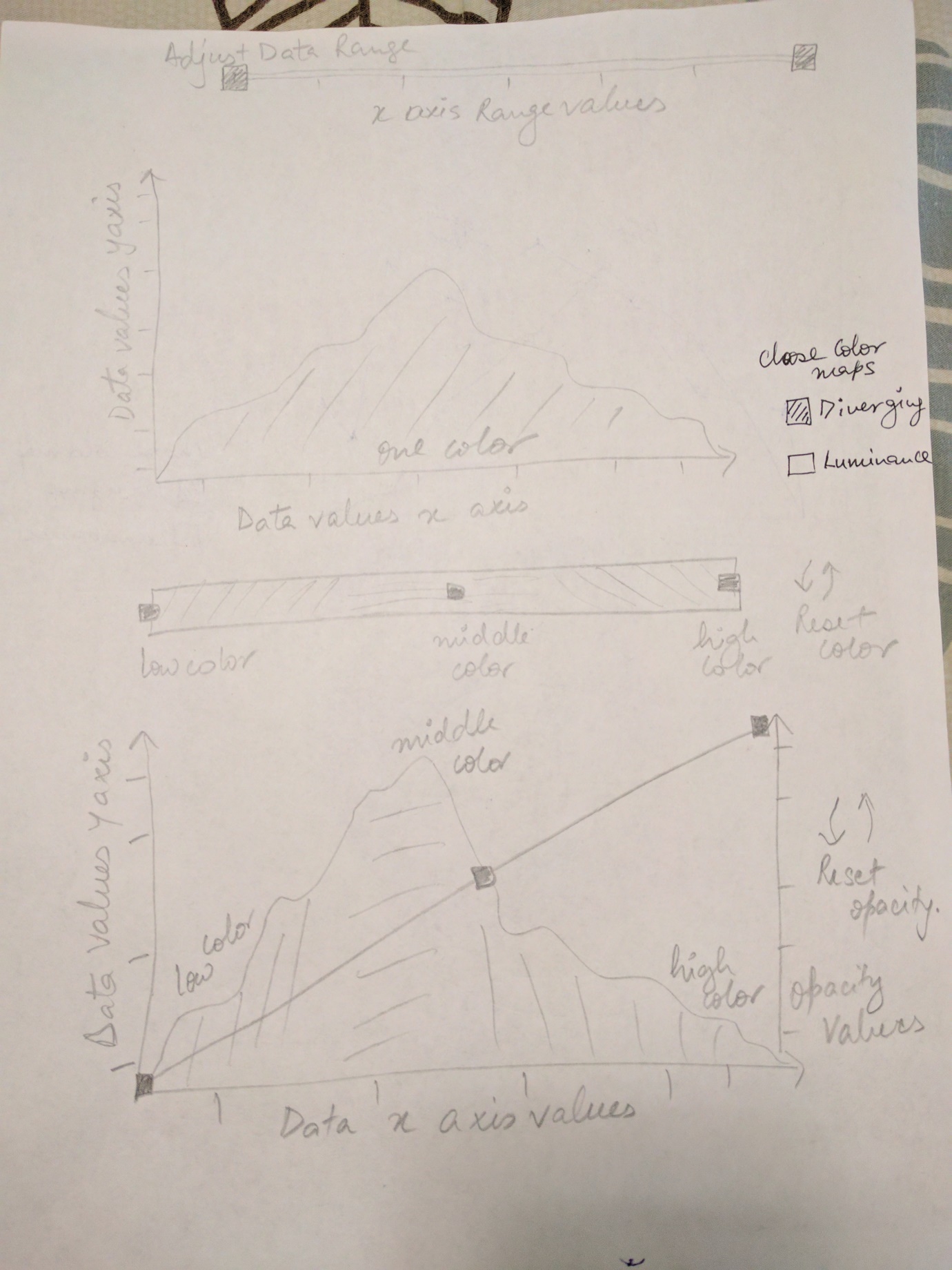


**Designing a transfer function widget-**

1. In this design, I decided to have both opacity and color functions in the same widget. This was because when I used Paraview, I felt it was cumbersome to use different tools for opacity and color. Also, using the same tool for both provides better context to the user while choosing the appropriate values.

We have data distribution on x axis, and on the left y-axis I have the color map a chose to implement and on the right y-axis I have the opacity values. There are two selection lines on the graph of different colors for each, with different shapes as value pickers. The user can independently move these two lines while still having a full context of the selection, unlike Paraview. I have a slider above that for data range selection, different reset buttons for both color and opacity and a choice of color maps between diverging and luminance. I also have a histogram that displays the scalar values of the dataset, to give the user better understanding of the distribution of the data values, colored using the color map on the above graph. The x-axis and y-axis has data set values. The user can view the histogram in both log and normal scale.

1. In the second design, I have the data range slider, reset buttons and the color map choices similar to the previous approach. The histogram here is similar to design 1 except, the color used to draw the data distribution is one plain color. Here I have a separate color map bar with color value picker and a separate, data values versus opacity values graph at the bottom with a single line to pick opacity. The graph at the bottom has a histogram of the scalar value distribution of the dataset as in the histogram above, except it is drawn in the appropriate color map. This is so that the user can correlate between the value he/she is picking on the color/opacity picker and the corresponding scalar value. The log scale button and the data range slider are common to both the histograms. I decided to implement this design, because when I tried to implement design 1, it was difficult to move the points around and capture the selected value in the first graph, when both color and opacity values overlap. I went back and decided to change the design.
2. Design 2 is on next page and design 1 is on the page after that.



**Final implementation-**

**Few notes-** **To select the dataset of your choice please make changes in the setup function in ofApp.cpp.**

The final design was mostly same as design 2, except I could not implement more than one color maps. I choose a diverging color map to start with but could not incorporate the luminance color map. This is a weakness of my widget when compared to paraview. Also, I have reset icons next to each tool to reset that particular tool. The icon next to the bottom most graph resets opacity selection and the icon next to color bar resets color selection. The strength of my widget is that, it provides a good correlation between the two transfer functions, so that when the user selects values for both, he/she knows exactly which color/opacity is being assigned to each scalar value in the dataset.

In volume rendering, huge amounts of pre-processing on dataset occur which makes the rendering quick and easy. However, volume rendering is sensitive to noise. Also, not all datasets can be easily approximated with geometric primitives.

Observations on datasets-

1. Pig.vti- i) The pig dataset is an image of a piggy bank.

ii) The piggy bank has flowers made on it.

iii) It is kept on a surface and has a money dropper at the top and a place to take all money out at the bottom.

iv) It has money stacked inside which can be seen by careful adjustment of the opacity and color values.

1. Engine.vti- i) This dataset has a lot of noise around it.

ii) With careful adjustment of the opacity and color values we can see the 2 spindles inside the engine.

iii) Most of the values are skewed towards one color, without color adjustment.

1. Tooth.vti- i) This dataset also has a lot of noise around it.

ii) Adjusting the opacity of the crown of the tooth, we can see the internal crown of the tooth.

iii) By adjusting both color and opacity, we can see the top internal crown as well as the internal root of the tooth.

iv) The top part of the tooth, outside the gums, is colored with high color on the color map, and the part inside the gums is colored with low color on the color map.

