Assignment #3: 3D Scalar Field Visualization
COSC 6344 Visualization
Fall 2021
University of Houston

Raunak Sarbajna
1956665

October 7, 2021

Exercise 1

1.1 Potential issues during the computation of Marching Cubes

Many possible topological issues during construction of a surface using Marching Cubes:

- If two triangles intersect, then something has gone wrong during the cube construction
- All lines need to be closed manifolds, but in the *quad* situation, the way we split it into two triangles may create one valid and one invalid line which we have to correctly choose between.
- There are some configurations where marching cubes is not robust and must be avoided.

1.2 Fundamental difference between Raycasting and Splatting

- Raycasting is in image order, whereas Splatting goes in object order
- Raycasting asks each data sample how much it contributes per pixel, while Splatting does the reverse and asks how much each voxel contributes to the image.
- Raycasting sends rays into the volume and samples at points, while Splatting samples the volume using a kernel and projects onto a flat cut-plane.

1.3 Limitations of Raycasting. Improvements to Raycasting

Limitations:

- Accuracy is heavily dependent on sampling intervals.
- As there is only a single ray per pixel, no secondary reflections and shadows can be shown.
- Cannot distinguish between object edges or intersection

Possible improvements:

- Improve ray templates to work is perspective views as well.
- Introduce techniques from raytracing - add diffuse light scattering, trace backwards from object to camera/light source, have multiple light rays and large number of reflections
- Add textures on surfaces and allow shadow mapping
- Allow z-buffer information to aid opacity calculations

1.4 Challenges of designing a proper transfer function for DVR

- The transfer function does not have access to any positional (x, y, z) information.
- Hard to distinguish between boundaries of different materials, due to the difficulty of finding edges without positional information.
- Material boundaries are not definite but fuzzy due to errors, noise, etc. Silhouettes are hazy.

Exercise 2

The assignment was done using

1. Python *3.8.3*
2. vtk *9.0.3*
3. PyQt5 *5.15.4*
4. VS Code *1.60.2*

2.1 3D Slicing

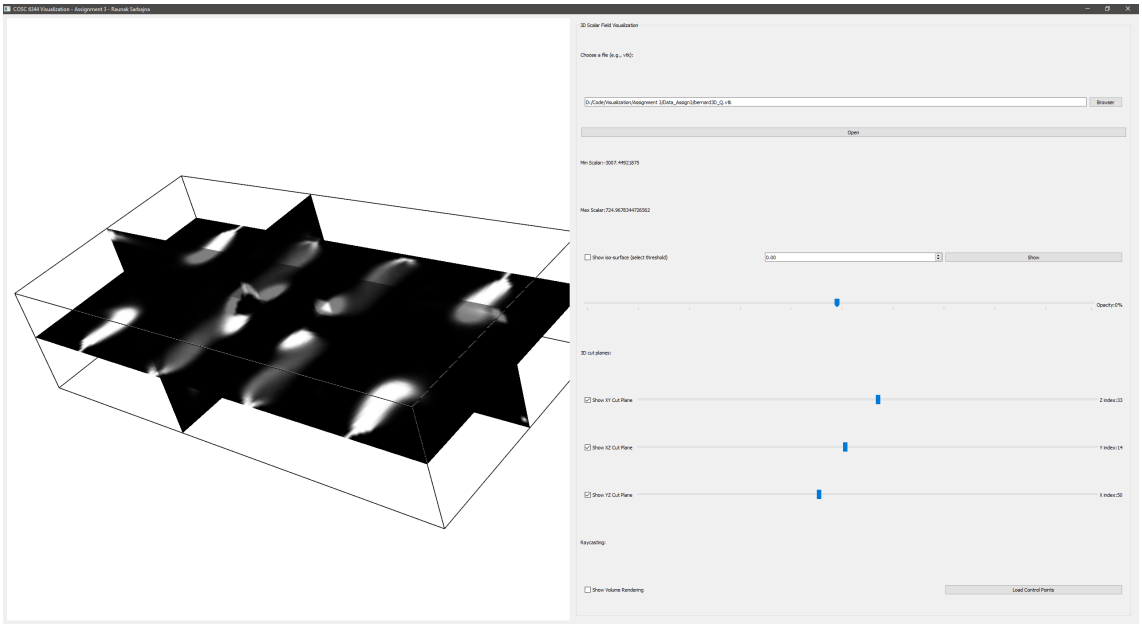

Figure 1: **3D cut planes of the Bernard 3D data, with** $Z - index = 33, Y - index = 14, X - index = 50$

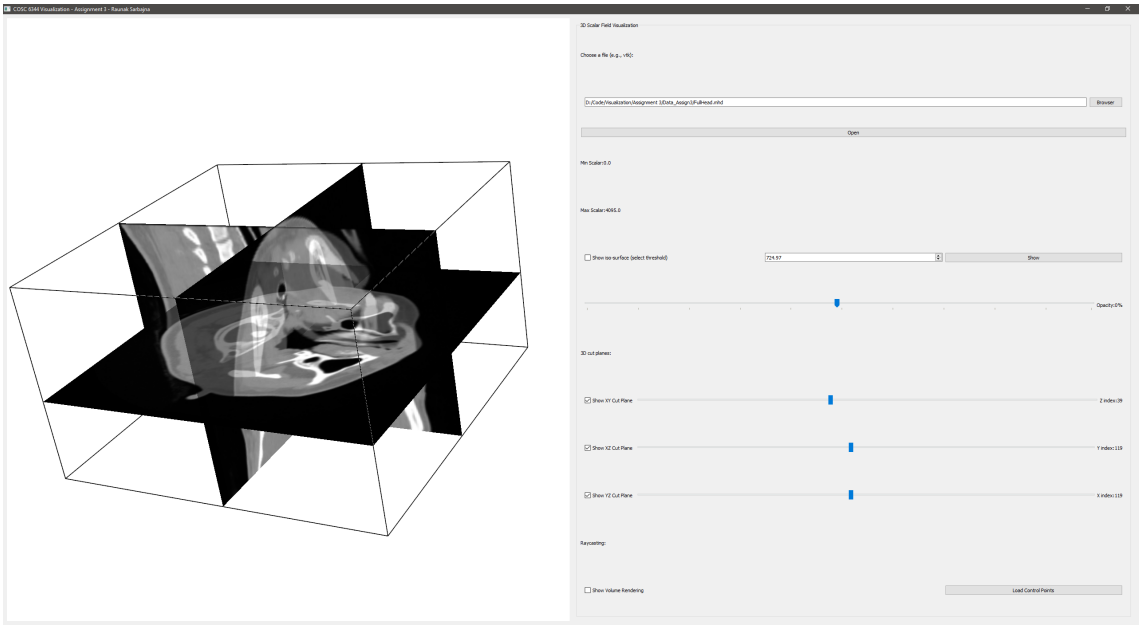

Figure 2: **3D cut planes of the FullHead data, with** $Z - index = 39, Y - index = 119, X - index = 119$

Exercise 3

3.1 Iso-surface Extraction and Visualization

3.1.1 Bernard 3D Dataset

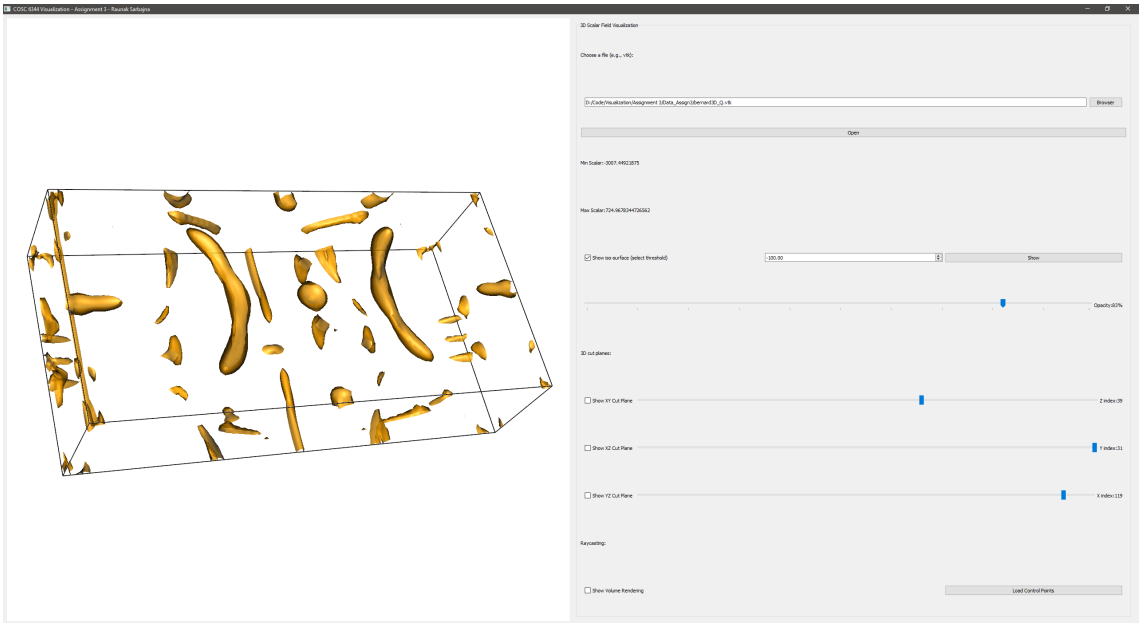

Figure 3: **Iso-surfaces of the Bernard 3D data, with** $Threshold = -100, Opacity = 0.83$

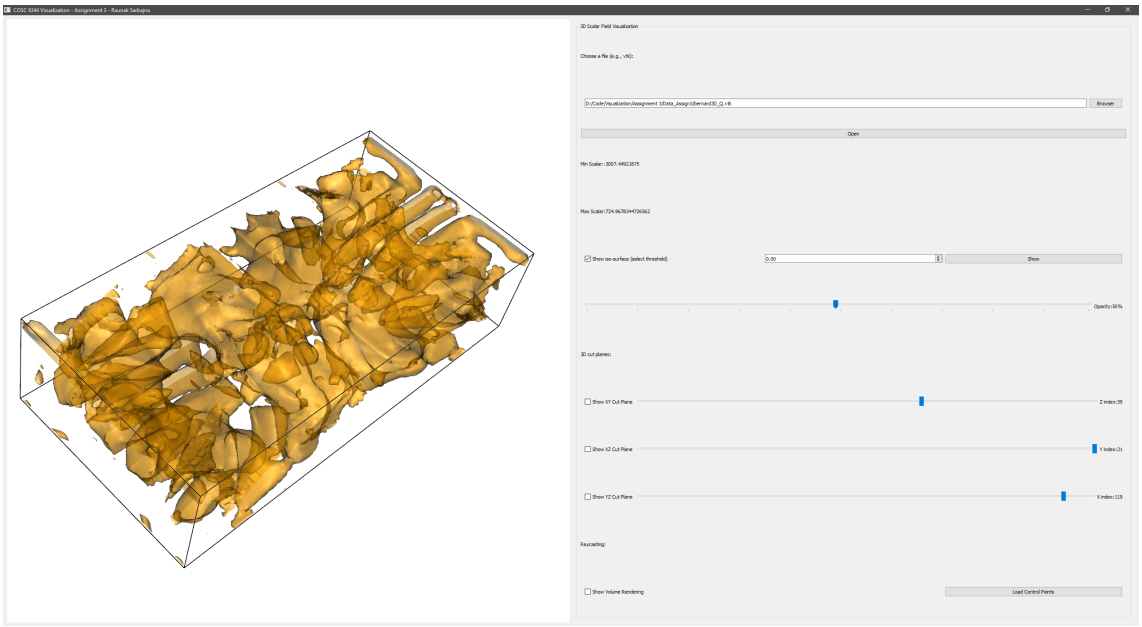

Figure 4: **Iso-surfaces of the Bernard 3D data, with** $Threshold = 0, Opacity = 0.5$

3.1.2 FullHead Dataset

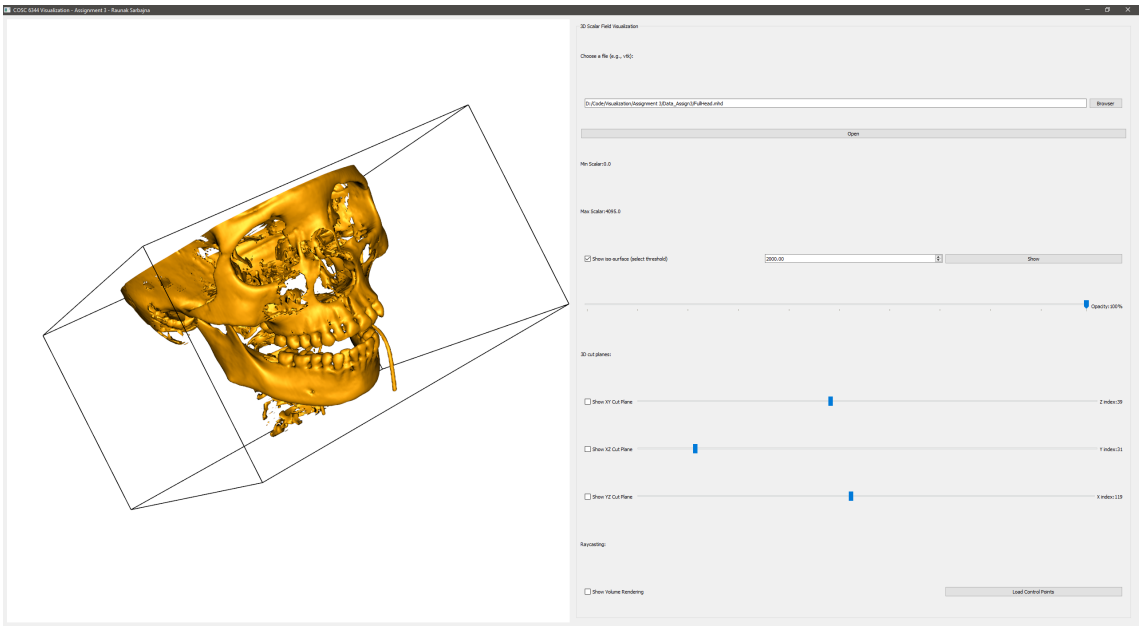

Figure 5: **Iso-surfaces of the FullHead data, with** $Threshold = 2000, Opacity = 1.0$

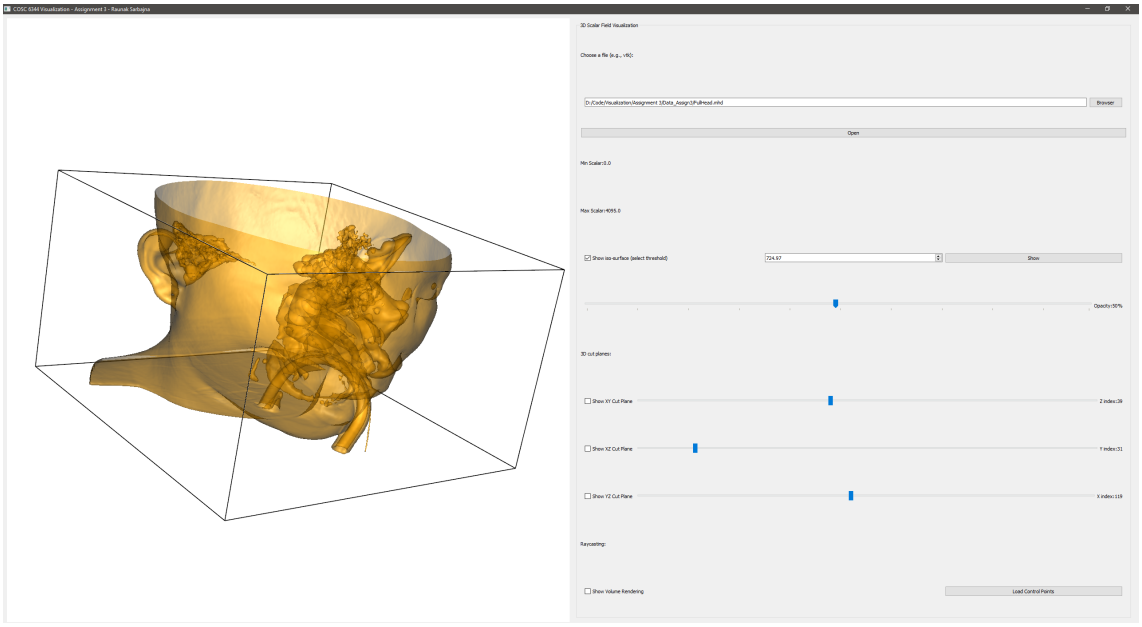

Figure 6: **Iso-surfaces of the FullHead data, with** $Threshold = 725, Opacity = 0.5$

Exercise 4

4.1 DVR-Raycasting

The *vtkSmartVolumeMapper*() was used for raycasting.

4.1.1 Bernard 3D Dataset

The selected control points of the Bernard Dataset can be seen as follows:

```
#Color
    -3000, 0.0, 0.1, 0.1
    -100, 0.0, 0.5, 0.5
    0, 0.5, 0.2, 0.3
    10, 1.0, 0.0, 0.0
    100, 0.5, 0.1, 0.1
    700, 0.0, 0.0, 0.0

#Opacity
    -3000, 0.1
    -100, 0.15
    0, 0.15
    10, 0.95
    100, 0.15
    700, 0.15
```

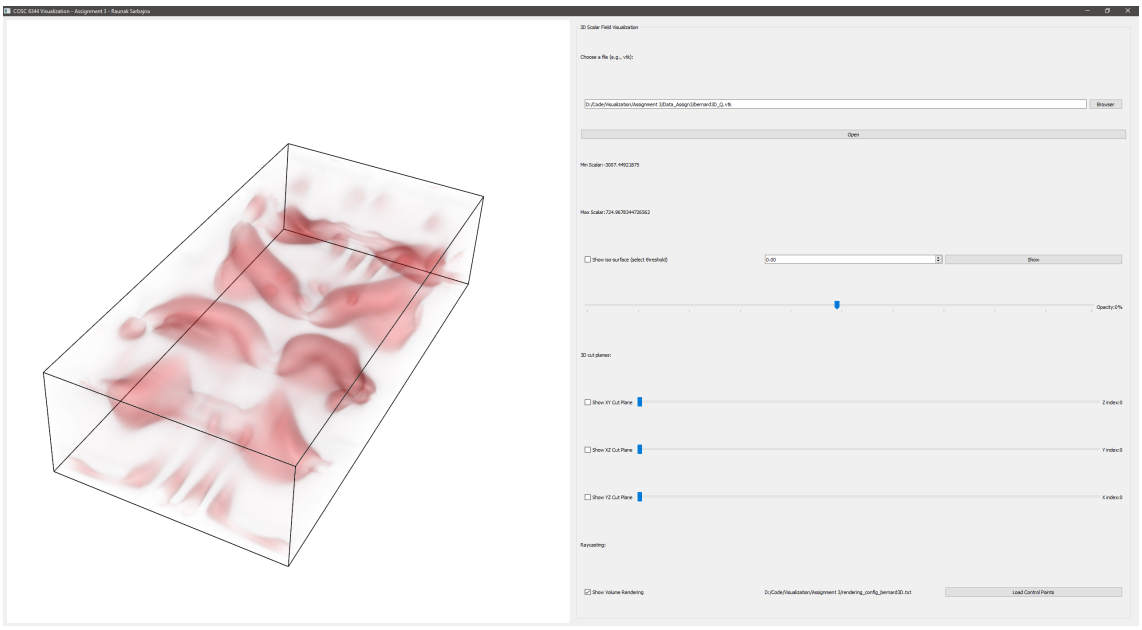

Figure 7: **Raycasted DV renders of the Bernard 3D data, showing external structure**

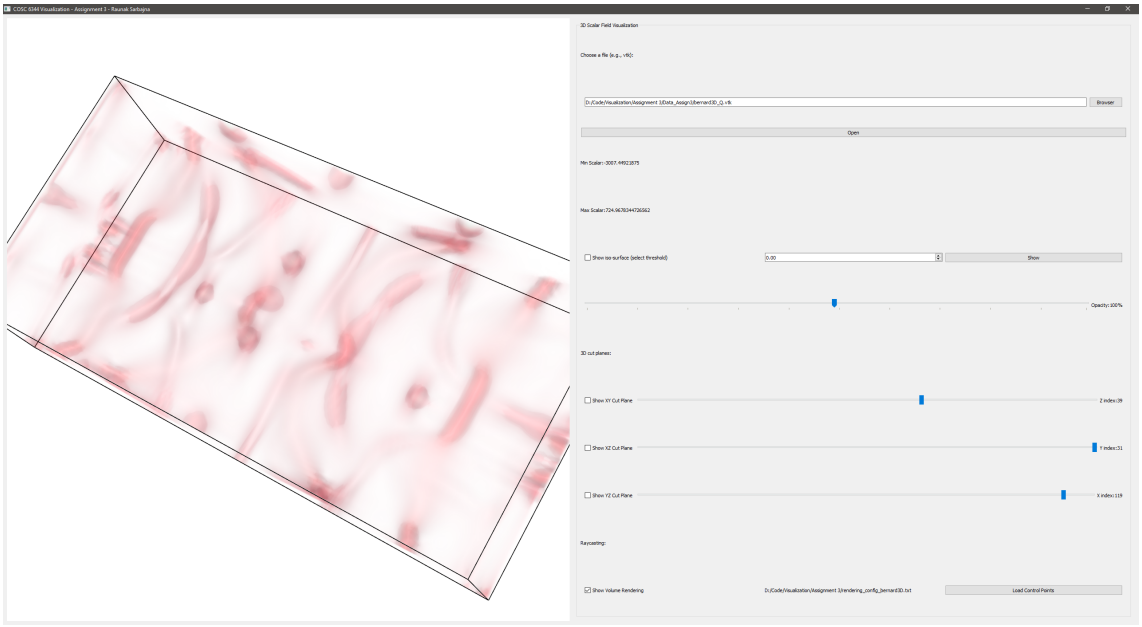

Figure 8: **Raycasted DV renders of the Bernard 3D data, showing internal structure**

4.1.2 FullHead Dataset

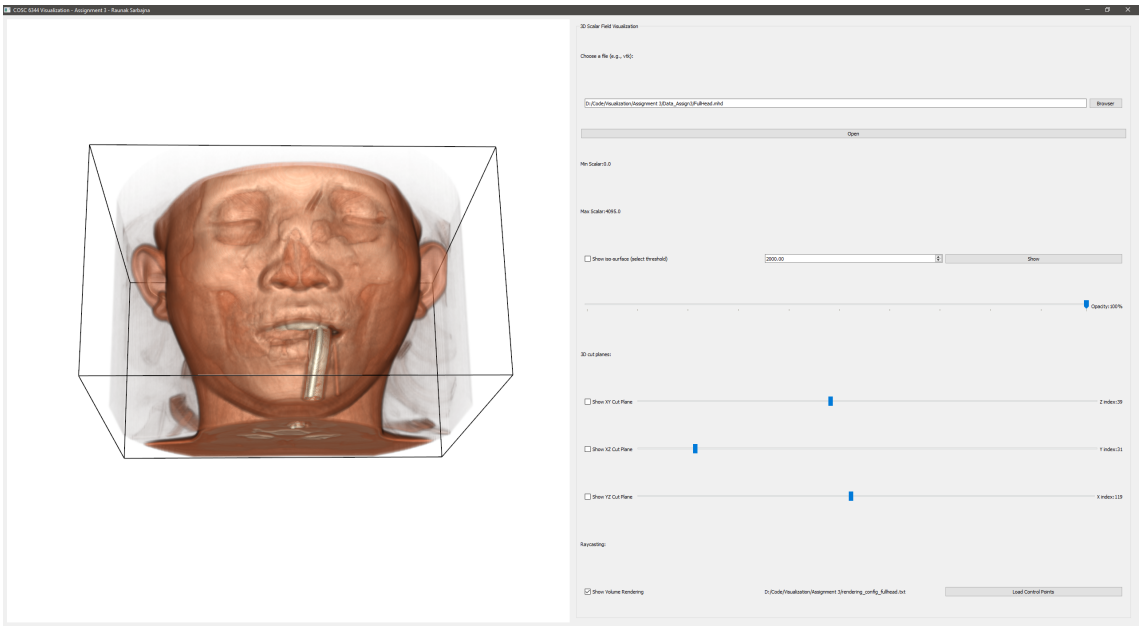

Figure 9: **Raycasted DV renders of the FullHead data, showing bottom-up perspective**

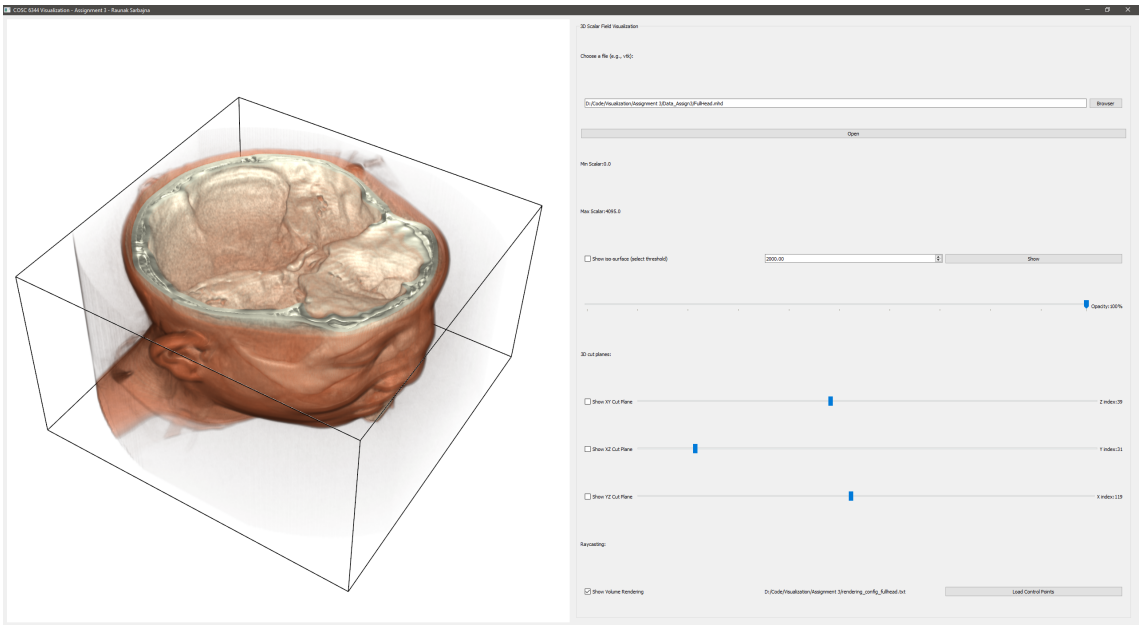

Figure 10: **Raycasted DV renders of the FullHead data, showing top-down perspective**