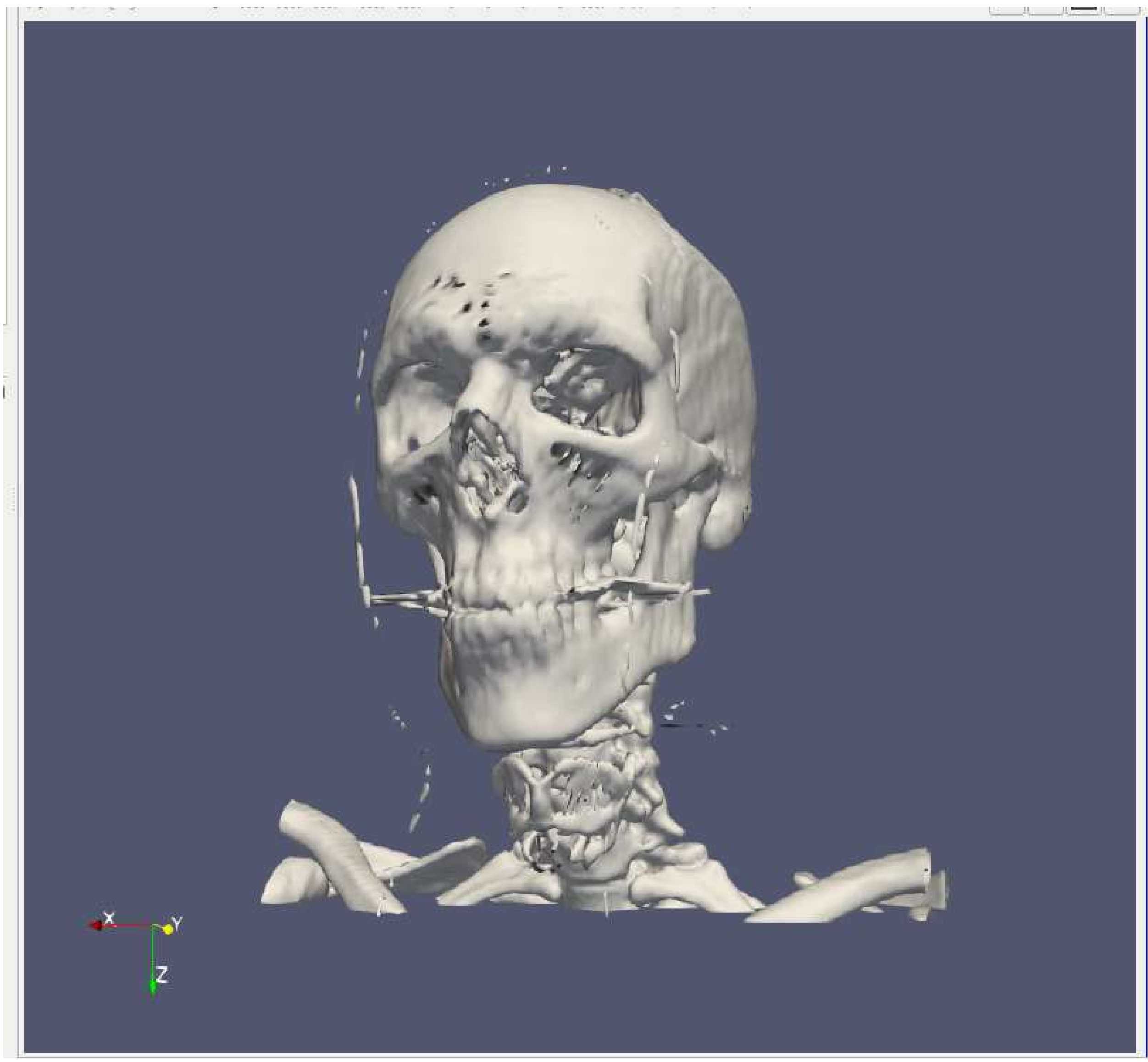
Assignment 9

Group Members

Abdul Wahab Madni, Hamza Badar, Aleksandr Semenikhin

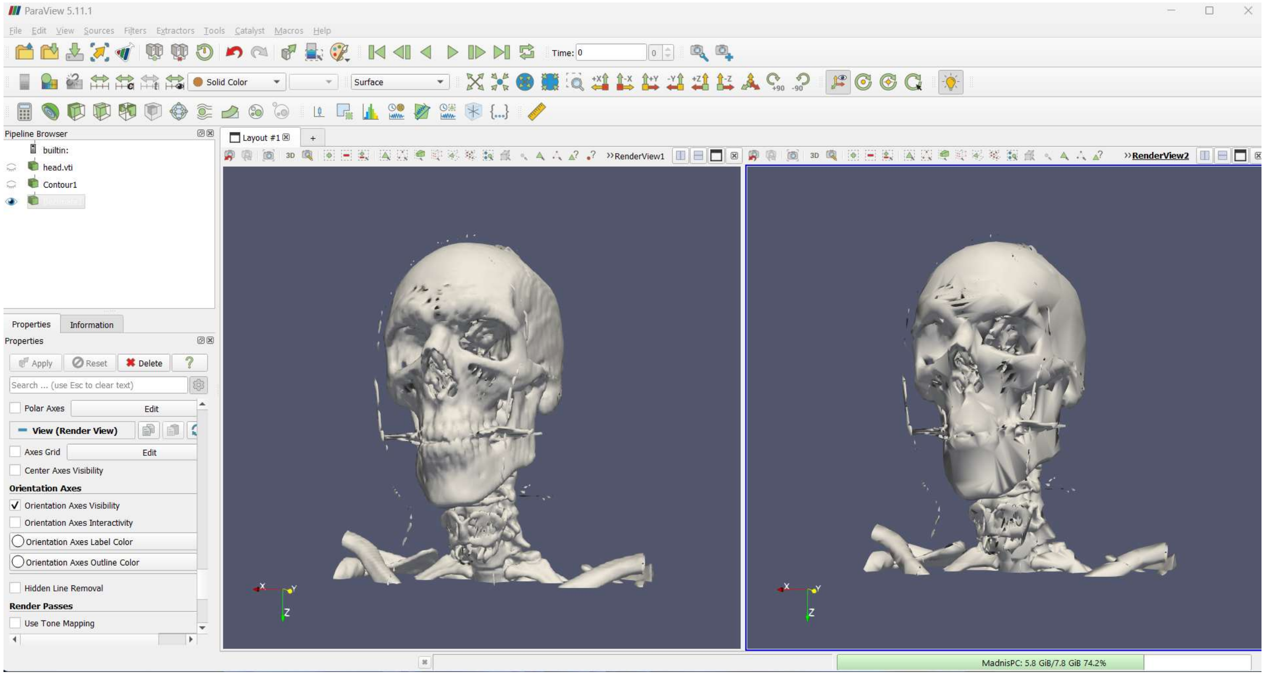
Ex.2: Mesh Decimation in ParaView

a)



It contains 185,697 triangles.

b)



c)

The mesh reduction filter preserve 18,569 triangles with default settings. If we force it to preserve topology, and do not permit boundary vertex deletion then it contains 92,413 triangles.

Ex.3: The Two Stages of Transfer Functions

a)

When adding a shading model to direct volume rendering, the second stage of mapping probabilities to color and opacity is affected. This is because the shading model introduces the concept of lighting and surface properties, which modify the color and opacity of the materials. The shading model calculates the interaction of light with the surface of the rendered objects, taking into account factors such as surface normals, light sources, and material properties like reflectance and transparency. Therefore, the color and opacity assignments in the second stage are adjusted based on the shading calculations.

b)

Associated colors are used in the second stage of mapping probabilities to color and opacity. In this stage, each material is assigned a color (Ci) and an opacity (αi) based on the probabilities obtained in the first stage. Associated colors refer to the colors (Ci) of the materials being multiplied by their respective opacities (αi) before being combined. This multiplication ensures that the color information of transparent regions does not leak into the final interpolation, preserving the intended colors of the materials. By associating colors with opacities, the final color and opacity assignments properly account for the transparency of the materials.

c)

The clustering algorithm is performed in the first stage of mapping data values to probabilities of the presence of materials. In cluster-based transfer function design, the aim is to group voxels in the feature space (transfer function domain) based on their similarity or other criteria. The clustering algorithm identifies clusters of voxels that share similar characteristics or properties. These clusters can represent different materials or regions of interest within the volume data. By performing clustering in the first stage, the transfer function is designed to assign probabilities of material presence based on the cluster assignments, allowing for more control and customization of the visualization based on the desired segmentation and classification of the data.

Ex.5: Style Transfer Functions

a)

αd represents uniform opacity in the transfer function, while αu represents the directional opacity. Multiplying them together allows for a combination of both opacities, resulting in a more flexible and customizable opacity for the sample. The exponent applied to αu has a qualitative effect on the opacity of a sample. If the exponent is lower than one, the opacity is enhanced, and if it is greater than one, the opacity is reduced.

b)

A simpler alternative to the procedure for style transfer function would be to use a 3D texture that stores an (r, g, b, α) tuple for every data value and normal direction. However, this approach has high storage requirements and is not practical. The authors invented the more complex procedure described in the paper to avoid storage issue, it allows for the use of a smaller amount of memory by storing the set of styles separately and using references to these styles in the transfer function lookup table.

c)

We can skip the index function texture lookup if no style is used multiple times in the style transfer function. This is because the index function texture maps the index values retrieved from the transfer function texture to locations in the style function texture, and if no style is used multiple times, the mapping is one-to-one and the index function texture lookup becomes redundant. Skipping this can save computational resources.

d)

The authors explain that style transfer functions are a bit faster than regular transfer functions in Figures 7 (a) and (d) because these cases have low number of styles(one style) and a high degree of spatial coherence(same style is used for many voxels in a row). In contrast, Figures 7 (b) and (c) have a higher number of styles and a lower degree of spatial coherence(same style is not used for many voxels in a row), resulting in less efficient caching and slower performance. In Figure 7 (b), the transfer function has a high number of styles, resulting in a large number of texture lookups. In Figure 7 (c), the transfer function has a low degree of spatial coherence.