E0 271: Graphics and Visualization Assignment #2

Weightage: 30%

Due: Oct 25, 2024

Learning objectives

- Learn Visualization pipeline
- Learn applications of GLSL/Shaders to Visualization
- Learn Paraview and its use in Visualization

Tasks

The broad aim of this assignment is to develop multiple visualization methods, particularly related to scalar and vector fields. Reuse components of the first assignment wherever applicable and use Paraview (https://www.paraview.org/) for Task 3. The Red Sea dataset is provided to evaluate your code. It contains two scalar fields – temperature and salinity – and one velocity vector field consisting of three components. You may create smaller synthetic datasets to test your code during the development process. **[N]** denotes the weight for each task out of **30**.

1. **[10] Scalar field visualization (OpenGL): Slicing**
   - **[4]** Use an axis parallel plane to visualize the data via slicing. Display the bounding box of the domain to provide a context.
   - **[4]** Repeat the above task on GPU. Use 3D textures within shaders to represent the scalar field. Use 1D texture in a shader to implement a colormap.
   - **[2]** Perform a comprehensive comparison of the CPU-based and GPU-based approaches with respect to image quality and performance, and report your observations.
2. **[12] Scalar field visualization (OpenGL): Isosurfaces**
   - **[5]** Implement the marching cubes algorithm to extract the isosurface of the scalar field on the CPU. Color vertices of the isosurface using an appropriate colormap that represents the order in which the cubes are traversed by the algorithm.
   - **[5]** Use a geometry shader for computing the isosurface.
   - **[2]** Compare the CPU-based and GPU-based approaches in terms of quality, performance, and pre-processing effort, and report your observations.
3. **[8] Vector field visualization (Paraview): glyphs, LIC, streamlines**
   - **[2]** Use the components provided to compute the vector field.
   - **[2]** Visualize the vector field using glyphs. (a) Scale the glyphs using the magnitude of velocity. (b) Use uniform scale along with a colormap. Choose seeds points appropriately to avoid clutter. Which of the two leads to less clutter?
   - **[2]** Visualize the vector field using streamlines. Again choose seeds points and region of interest appropriately to avoid clutter. How do the choice of integration method, number of seed points, and other parameters affect the output?
   - **[2]** Choose appropriate slice, visualize the vector field restricted to the slice using LIC, analyse the effect of the number of steps and step-size.

Submission

1. Prepare a .zip file of the code (with makefile, readme, etc) and a text file with a description of the methods used in slicing and detailed analysis related to the slicing experiment.
2. Submit the .zip file before **23:59, Oct 25th**.
3. Any submission beyond the deadline will be considered as a **late submission**.

Notes

1. The files are provided in ascii vtk format which can be parsed by writing a simple parser. The details of the format will be explained during the tutorial.
2. Avoid hard-coding values in intermediate instructions of code. Define constants/values in the global section so that changing and testing the code is easier. You may be asked to do so during the interactive demo.
3. To compute scalar values at points that aren't grid vertices, use trilinear interpolation.