Interactive Visual Data Analysis

Assignment 8 – *Probes and Nicer Particles*

In this assignment, we will improve our particle-based flow visualization by fancier rendering and by allowing the user to specify where particles should spawn.

8.1. Particle Probes

In the previous assignment, particles were spawned uniformly in the whole flow domain. However, it is often more interesting to spawn particles only in a specific region, e.g. directly in front of an obstacle. To allow the user to do this, we introduce *particle probes*, which are axis-aligned boxes inside which particles spawn.

- Allow the user to define and adjust multiple probes. Each probe should have its own set of parameters, such as particle count, lifetime, color, etc.
- It should be possible to adjust the probe position and size (in each dimension) using the mouse in a "drag&drop" fashion.
- Enhance your particle advection and rendering code so that it handles the particles from all probes. You can create separate buffers for each probe and process the probes one-by-one.

8.2. Improved Particle Rendering

Last week's point primitives do not look very impressive. Let's replace them with spheres.

- Write a geometry shader that creates a quad for each particle to serve as proxy geometry for the sphere. The quad should be perpendicular to the view vector (i.e. the vector from the particle position to the camera), and should be in front of the sphere we want to render.
- Write a pixel shader that computes the ray-sphere intersection point (if any). If there is an intersection, compute the surface normal and perform Phong lighting. Also compute the depth at the intersection and write it to `SV_Depth`.
- Adjust each particle's color based on its velocity, e.g. brighter = faster. Alternatively, you can use the transfer function editor to map from velocity magnitude to color.
- **Optional:** Create more advanced shapes for the particles, e.g. an "arrow" made out of a cone and a cylinder. In this way, you can also show the direction of each particle.

8.3. Visualize!

Like every week, take a few screenshots of some pretty renderings generated by your tool and commit them to a folder called `screenshots/assignment8/` outside of your solution directory.

The working solution must be committed till **December 11, 09:00am**. If anything is not working as described here or if you want a specific SVN revision to be rated, explain yourself in the `readme.txt` file within your `solution` directory.