

CS 450/550 Project #1: Point-Cloud Solar System

Haochuan Zhang (zhanhaoc@oregonstate.edu)

October 6, 2025

What I built

I created a small solar-system scene in OpenGL/GLUT featuring a sun and eight planets. Each body is *my own geometry*, generated procedurally as a UV-sphere but rendered in a **point-cloud “dusty” style** (GL_POINTS), producing a soft, star-dust aesthetic similar to the sample image in the prompt.

How the geometry is mine

Instead of GLUT/OSU canned meshes, I compute sphere vertices from latitude/longitude:

$$x = R \sin \theta \cos \phi, \quad y = R \cos \theta, \quad z = R \sin \theta \sin \phi$$

and emit either triangles (for a fallback) or points (for the final look). I also apply a per-vertex color gradient (brighter caps, darker belt) and a small outward jitter to each normal to achieve the granular glow.

Randomness

At startup I call `TimeOfDaySeed()` to vary random sequences. Planet positions on their circular orbits are randomized by a random angle $\phi \in [0, 2\pi)$. The point-cloud jitter uses `Ranf(-1, 1)` along the local normal.

Interaction

- Mouse left-drag: free 3D rotation (X/Y).
- Keys +/-: uniform scaling (kept above a min scale).
- Right-click menu: axes, depth cue, projection, reset, quit (unchanged from base).

Colors & transparency

I use more than five distinct base colors: warm yellow sun; and eight planets with clearly separated hues (Mercury warm gray-brown; Venus golden; Earth saturated blue; Mars orange-red; Jupiter tan; Saturn pale yellow; Uranus teal-cyan; Neptune deep blue). Alpha blending (SRC_ALPHA, ONE_MINUS_SRC_ALPHA) is enabled with depth writes disabled while drawing translucent bodies to avoid hard edges.

Vertex count (meets requirement)

The sun uses $160 \times 120 \approx 19,200$ points. Each planet uses $96 \times 64 \approx 6,144$ points ($\times 8 \approx 49,152$). Total $> 68,000$ vertices $\gg 100$.

3D thickness

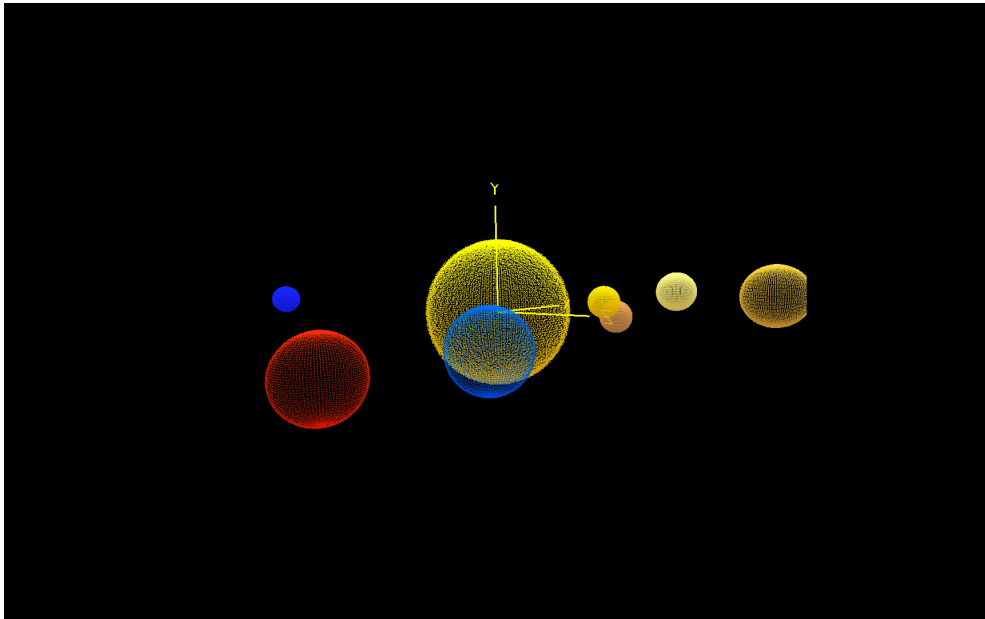
All bodies are true 3D spheres; the camera is pulled back with perspective, and fog is tuned for a deeper space feel.

Build & run

On macOS:

```
make -j  
./sample
```

Screenshot



Video (Kaltura/Media Server)

Link: https://media.oregonstate.edu/media/t/1_wh9m3810

Please ensure the permission is *unlisted*; I verified by opening it from a second account.

Notes

To match the requested style, I switched the sphere renderer to GL_POINTS with outward jitter and a pole-to-belt gradient. I also remapped scaling to \pm as requested.