10/24/24 – 10/31/24

Group number: **40**

Project title: 3D Gaussian Splatting With Dynamically Raytraced Lighting

Client: Jackson Vanderheyden & Brian Xicon

Advisor: Simanta Mitra

Team Members/Role:
Ethan Gasner - Documentation Manager.
Kyle Kohl - Communication Manager.
Jackson Vanderheyden - Graphics Scope Manager.
Brian Xicon - Machine Learning Scope Manager.
Luke Broglio - Schedule Manager.

- **Weekly Summary:** The Graphics Team finalized the initial hybrid Gaussian rendering pipeline in Unity. Small adjustments might be made further in the development life cycle, but the core foundation is in place. In addition completed a demo which renders gaussians from our .ply files in 2D and worked on expanding the Gaussian file and parser to handle relightable Gaussians. The Machine Learning team discussed implementation for our Gaussian point optimizer with a very trivial prototype created. Additionally, data sets were added to our team git repository.

- **Past week accomplishments**

Ethan Gasner: This past week I continued to learn about techniques and formulation for our optimizer. I was also able to help Brian with the structure for the prototype he started.

Kyle Kohl: To continue my education of Raytracing, I wrote a bare bones java raytracer. I also watched videos about compute shaders.

Jackson Vanderheyden: In Unity, I finalized the initial architecture of the hybrid Gaussian rendering pipeline. The command buffer will be inserted before image effects. It consists of the following steps: data buffer preprocessing, primary ray generation, loop through ray intersection, ray shading, and ray reflection compute shaders for however many specified bounces, and a shader blitz command to write to the camera's texture. I finished all the buffer preprocessing in the pipeline, and the primary ray generation compute shader.

Luke Broglio: I finished debugging the demo of rendering Gaussians in 2D. I then moved into working to expand the gaussian .ply file and parser by adding in normal vectors, spherical harmonics, and PBR properties.

Brian Xicon: This week I curated all of our main datasets we will be using and put them on our repository. I also began development on a rudimentary version of our Gaussian Point Optimizer in which I hope to add more features to it as we continue with this project. I also tested our current hardware to make sure it would be able to train and render our project within a reasonable time.

o **Individual contributions**

NAME	Individual Contributions	Hours this week	HOURS cumulative
Ethan Gasner	Practiced/ developed Machine Learning techniques and assisted in the first steps of optimizer creation.	6	42
Kyle Kohl	Learned more about compute shading. Worked on researching running Pytorch models inside of a Unity project.	6	42
Brian Xicon	Curated datasets, began development on Gaussian Point Optimizer, and tested current hardware's capability of testing and running our project.	6	42
Jackson Vanderheyden	Finalized the hybrid Gaussian rendering pipeline in Unity. Finished all path buffer data preprocessing stage and the primary ray generation stage.	6	42
Luke Broglio	I finished debugging the demo of rendering Gaussians in 2D. I then moved into working to expand the gaussian .ply file and parser by adding in normal vectors, spherical harmonics, and PBR properties.	6	42

o **Plans for the upcoming week**

Ethan Gasner: This upcoming week I intend to help brian continue with the optimizer and add more functionality while also testing what is currently there. My goal is to properly create gitlab issues and merge requests as well.

Kyle Kohl: For this next sprint (2 weeks), I plan on continuing to research how to run Pytorch models inside of Unity. Ideally I would like to have this code working by the end of the sprint. I also plan to continue to learn about compute shading. I plan on learning more about Ray Tracing. I had hoped to get further along on this front but encountered some errors.

Jackson Vanderheyden: For this upcoming week, I plan to continue coding the remaining compute shaders needed to do the bare minimum of a ray tracer with triangle intersections (potentially 2+ weeks). Gaussian support will be added once the pipeline can render traditional triangle meshes.

Luke Broglio: I want to work on having my changes adding example gaussian files and the parser merged into the main branch. Then I plan to start assisting in implementing ray triangle intersection.

Brian Xicon: Next week, I plan on continuing my implementation of the Gaussian Point Optimizer which will consist of resolving some errors I was getting with my current implementation and then find ways to render my rudimentary .ply file without a fully made renderer.

- o **<u>Summary of weekly advisor meeting</u>**

During this week's advisor meeting with Professor Mitra, we presented the work we have done so far and discussed our plans for the upcoming two weeks. This included the prototype optimizer and the developments made on the graphics prototype as well.