EE/CprE/SE 4920 SPRINT REPORT 1

2/7/25 - 2/14/25

Group number: 40

Project title: Hybrid Relightable 3D Gaussian Rendering

Client: Jackson Vanderheyden & Brian Xicon

Advisor: Simanta Mitra

Sprint Summary:

	Sprint Tasks	Accomplished Tasks	Tasks for Next Sprint
Ethan	-ML: Finish SfM Python Script to automatically conduct SfM -Update Team Website with current documents	ML: Finished core functionality for SfM Colmap automation script, creating sparse and dense .ply files to be used in gaussian creation and optimization.	ML: Work with Kyle to get python scripts working within the Unity Environment. Machine Learning: Modify SfM script to accept user selected paths
Kyle	 Create a team account for registering Unity Package to the store Test importing PyTorch modes to ONNX files for importing to Unity 	-GP & ML: Successfully created a team account for registering Unity Assets the the Unity Public Store.	-GP: Work with Ethan to get Python scripts working within the Unity Environment. This is part of importing Pytorch models to Unity.
Jackson	-GP: Diffuse rendering -GP: Model texturing render support -GP: Physically based path reflection rendering	-GP: Diffuse rendering -GP: Started Model texturing render support	-GP: Model texturing render support -GP: Physically based path reflection rendering

Luke	 Write tests to confirm BVH generation is fully working. Fix any issues exposed by BVH generation Merge BVHs into main branch 	 Write tests to confirm BVH generation is fully working. Fix any issues exposed by BVH generation 	 Merge BVHs into main branch Write Ray-gaussian intersection code
Brian	-ML: Optimize camera angles for training ML models off our premade images. -ML: Create ML model and optimize it to work with different values of Gaussians like color, texture, etc.	-ML: Finished retrieving the camera angles of our point cloud to reference with the premade images, just want to optimize them to make sure the camera intrinsic and extrinsic values match with the original images perfectly.	-ML: Finish optimizing the camera angles. -ML: Create the ML model used to turn our normal point cloud into a Gaussian point cloud.

Ongoing Tasks:

Graphics Programming (GP) Team:

□ Model texturing render support

□ Physically based lighting calculations

□ Merge BVHs into the main branch

□ Write ray-gaussian intersection code

Machine Learning (ML) Team:

Create a ML model to convert a point cloud into a Gaussian point cloud.