CSE 493V
VR SYSTEMS

3D Gaussian Splatting Video Chat

Jason Zhang

University of Washington

PROBLEM

Conventional video chat applications are only capable of providing a basic visual and audio connection, displaying users on a screen and transmitting voice through device speaking. They lack interactivity and immersive features that replicate the natural dynamics of in person conversations

RELATED WORK / MOTIVATION

Some related work are like Google's project Starline, which requires depth sensors, LiDAR, and light field display to capture and reconstruct a person in real-time 3D
Another approach was Apple's vision pro , which is a mixed reality head that that integrates VR and AR capabilities, including FaceTime with avatars.

Both approaches lack immersion, ability to interact, and the ability for users to explore the scene. They also require specified hardware that are expensive and less accessible to general consumers

YOUR APPROACH / SOLUTION

With the emergence of 3D Gaussian Splatting, we are now capable of generating photo realistic 3D scene from just a few images. Which we want to implement in Unity, a VR application that renders 3D footage of these Gaussian Splatting scenes in real-time to simulate video calling in 3D

METHOD / PIPELINE / ALGORITHM / PROCESS

1. Capture Input Images – Use groups of 6 images taken simultaneously (representing a single frame of the scene).
2. Generate Gaussian Splatting Scene with InstaSplat – Process each frame using InstaSplat, training for 1000 iterations to obtain a point cloud representation of the scene.
3. Import into Unity – Read the generated point clouds into Unity using the "Unity Gaussian Splatting" extension.
4. Simulate Video Playback in VR – Alternate between successive point cloud scenes in time order to simulate playing a video within the VR environment.

RESULTS

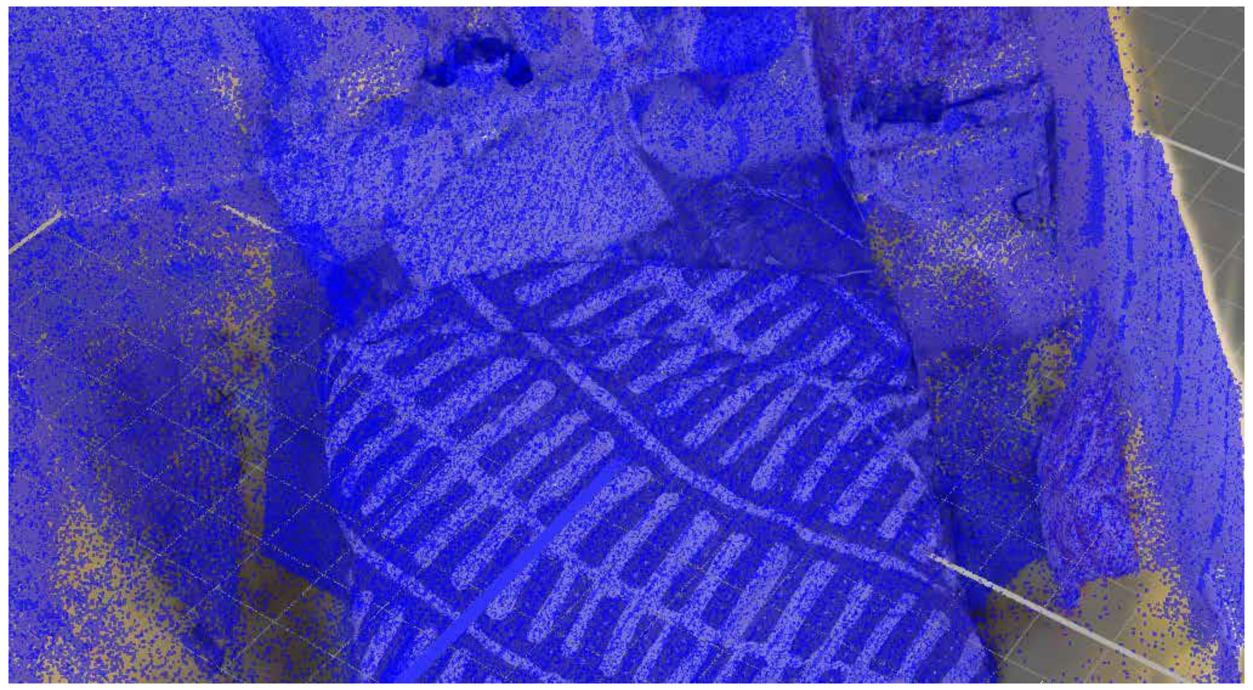

Figure: Point Cloud Generated through 6 images of bedroom

We successfully achieved 60 FPS on PC and 30 FPS on the Meta Quest 2, demonstrating that real-time playback of Gaussian Splatting scenes is feasible within VR environments. While performance on PC was smooth, the computational limitations of standalone VR headsets like the Quest 2 resulted in a lower frame rate, indicating the need for further optimizations.

Future work could focus on improving real-time rendering efficiency, such as GPU acceleration techniques and optimized point cloud compression. Additionally, integrating real-time scene updates using hybrid computer vision techniques (e.g., object tracking with YOLO) could enable dynamic interactions, reducing the need for frequent full-scene reconstructions and making Gaussian Splatting viable for interactive VR experiences and telepresence applications.

REFERENCES

[1] A. Pranckevičius, Gaussian Splatting Playground in Unity, GitHub Repository, 2023. Available: https://github.com/aras-p/UnityGaussianSplatting
[2] J. Kerbl, T. Leimkühler, T. Müller, A. Keller, 3D Gaussian Splatting for Real-Time Radiance Field Rendering, arXiv preprint, 2023. Available: https://arxiv.org/abs/2308.04079