



Generating Virtual Objects

Boe Zienko, Matthew B. Thacker

Advisor: Dr. Jeremy Becnel

SFASU Department of Computer Science

Abstract

Our goal is to develop an application that generates virtual objects from real world input in the form a video and allows these objects to be shared and manipulated in real time.

To process the video to generate a virtual object, we are using Neuralangelo, which is an artificial intelligence model developed by Nvidia.

We are connecting two or more Oculi via a Photon Unity Networking (PUN) server, so one device can act as an instructor and scan objects to import into the virtual world as well as manipulate those objects. The other Oculi will act as spectators and only be able to watch.

Introduction

Virtual reality technology has many applications in the field of education. One such application is sharing objects between instructors and students. Currently, this would be difficult for instructors to implement, as manual creation of virtual objects with tools like ProBuilder is challenging, error prone, and time-consuming. The research presented here examines how to implement Neuralangelo to automate the process of generating virtual objects which can then be imported into a shared virtual environment, updated by a PUN server .

Methods and Materials

Materials

- Unity 2022.3.10f1
- PUN2 (FREE)
- Docker
- Neuralangelo
- COLMAP
- Meta XR All-in-One SDK

Once set up, Neuralangelo will take an input of 2D videos or a group of images and return a 3D-rendered object or scene.

To set up a shared scene that multiple users can simultaneously exist in, a PUN server was set up. Then PUN2 (FREE) was downloaded from the Unity Asset Store and imported into the Unity project. A loading scene was then created with the purpose of being the first scene a user is loaded into. Within this scene is a C# script that connects the user to the PUN server. A lobby scene was added to serve as an intermediate scene with a C# script which allows users to create or join a room on the server. Once the room is created or joined, the user will be taken to the SharedSpace scene, where users can interact with one another and objects.

```

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using Photon.Pun;
5 using UnityEngine.SceneManagement;
6
7 public class ConnectToServer : MonoBehaviourIPunCallbacks
8 {
9     private void Start()
10    {
11        PhotonNetwork.ConnectUsingSettings();
12    }
13
14    public override void OnConnectedToMaster()
15    {
16        // Join lobby once connected to photon unity network
17        PhotonNetwork.JoinLobby();
18    }
19
20    public override void OnJoinedLobby()
21    {
22        // load lobby scene
23        SceneManager.LoadScene("Lobby");
24    }
25 }
26

```

Figure 1. ConnectToServer Script

```

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using UnityEngine.UI;
5 using Photon.Pun;
6 using TMPro;
7
8 public class CreateAndJoinRooms : MonoBehaviourIPunCallbacks
9 {
10    public TMP_InputField roomNameInput;
11
12    public void CreateRoom()
13    {
14        PhotonNetwork.CreateRoom(roomNameInput.text);
15    }
16
17    public void JoinRoom()
18    {
19        PhotonNetwork.JoinRoom(roomNameInput.text);
20    }
21
22    public override void OnJoinedRoom()
23    {
24        PhotonNetwork.LoadLevel("SharedSpace");
25    }
26 }
27

```

Figure 3. CreateAndJoinRooms Script

Expected Results

We plan to use the Neuralangelo AI to create a virtual reality application that scans an object and returns a 3D-rendered object that can be manipulated by the user. In order to scan an object, we will have the user record a video that takes a 360° view of the object. The video will be passed to Neuralangelo and a 3D render will be sent back to the user and then to the server. This 3D render will be able to be manipulated by the users on the server in real time.



Figure 2. Screenshot of the last 10,000 iterations of Neuralangelo running.

Conclusions

We look forward to developing this application and continuing our research into the subject. Our next steps include but are not limited to the following:

- Get XR Interaction toolkit up and running in SharedSpace scene.
- Take videos from the VR headset to send to Neuralangelo.
- Send polygon file back to VR headset from Neuralangelo.

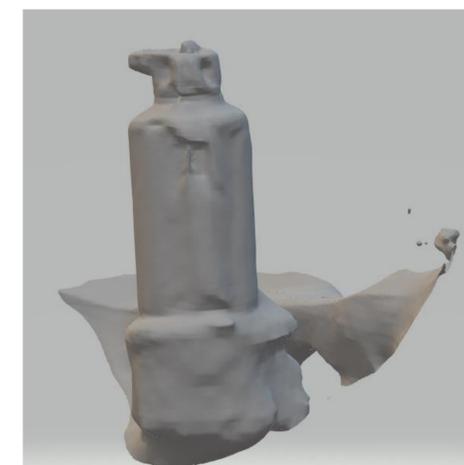


Figure 4. Water Bottle Rendered by Neuralangelo

Contact

Boe Zienko, Matthew B. Thacker
Department of Computer Science
P.O. Box 13063, SFA Station
Nacogdoches, Texas 75962
zienkoba@jacks.sfasu.edu
thackerm@jacks.sfasu.edu
936.468.2508

References

- Li, Z. et al. (2023). *Neuralangelo*. Research at NVIDIA. <https://research.nvidia.com/labs/dir/neuralangelo/>
- <https://doc-api.photonengine.com/en/pun/current/annotated.html>
- <https://developer.oculus.com/documentation/unity/unity-package-manager/>