CSE 366 Spring 2024

Assignment 1: Basic VR Monday, February 26, 2024, 11:59 p.m. (EST)

Now that you have completed Assignment 0 and have gained some experience with working in Unity let's use those skills and what you have learned in class to design an interactive virtual environment. In this assignment, you will design a VR world and implement basic movement!

For a true VR experience, we strongly recommend that you meet the following minimum hardware requirements:

- A Google Cardboard (or similar)
- A mobile device with either Android version 8.0 or higher or iOS version 13.0 or higher (note that for later assignments you might need a higher version of the OS).

Overview

For this assignment, you will design a 3D virtual scene similar to the Roth Pond area. Therefore, your virtual scene should have an oval-shaped **pond**, a **path** around the pond, some **trees**, a few cuboid-shaped **buildings**, and a **hot air balloon**. Of course, we want to make the virtual scene fun, so we would also like you to include a small **cart** that will go on the path around the pond. The details about each object will be explained later in this assignment. Though we would love you to be creative with this first assignment, however, you are allowed to download models from the Unity Asset Store or any other source, provided you have permission to download them, and you <u>cite each source properly</u> in your report.

The scene will start with the user standing outside a building facing the pond and the cart should be on the path right in front of the user. To include basic interaction, the user should be able to *teleport* (transport instantly) inside the cart. Don't worry, we will learn together how to make that happen! Once the user teleports inside the cart, the cart should start moving on its route around the path and the user should be able to rotate their head and see the virtual scene from this new viewpoint, while the cart is moving. Additionally, the user should also be able to experience how the virtual scene looks during the *day* and at *night*.

Note: Please read the entire Assignment document first to gain a full understanding of the components.

Virtual Roth Pond Scene

Your assignment's virtual scene should have:

1. One oval-shaped pond

Oval is the simplest shape to represent the pond. However, you may be as creative as you want with the shape of the pond. The pond should have a water-like texture: a shade of blue with reflective properties (see Unity manual for <u>Textures</u> and <u>Materials</u>). Though you may design your own texture and material if you wish to learn more in-depth graphics in Unity, however, there are many freely available textures on the internet.

Extra credit: Download / implement a Unity shader that creates waves in the water.

2. A path around the pond

The path should be along the circumference of the pond and noticeably wide. Similar to the pond, you should add a texture of your preference: gravel, sand, or plain solid color.

3. A cart placed on the path

The cart should be hollow. You may download a cart model from the internet or design it using basic 3D polygons: one cuboid and 4 disks or spheres for the wheels. The cart should be on the path around the pond. The cart should be stationary except after the user teleports inside it and then it moves. The cart should also have headlights that turn on at night. The headlights should be two spot lights.



Fig. Example of a 3D cart

Tip: Recall that an ellipse can be mathematically defined using: $x = centerX + (major_{axis} * \cos T)$ $y = centerY + (minor_{axis} * \sin T)$

So, you will have to write a script that transforms the position of the cart every frame (recall: Update() function that is called every frame).

4. <u>Three</u> buildings around the pond area

Similarly, building models can be downloaded or you may design it using basic 3D hollow cuboids.



Fig. Example of a 3D building

Extra credit: Add a light source inside at least one building. The outer face of the building should have 2 large transparent windows or many small ones to allow light to shine out during the *night* mode.

5. One hot air balloon

The hot air balloon can be a downloaded model or designed using 3D polygons and it should have a constant and continuous trajectory (example shown in the figure below) – that is, once the hot air balloon completes its trajectory, it should repeat the motion from its original starting point.

The trajectory arrows shown in the figure are perpendicular to the ground – so the arrow in orange shows the balloon gaining elevation while also moving to the right. The starting point for the hot air balloon should be above the ground, in the sky. You can choose any speed for the hot air balloon to float in its trajectory. The figure shows the simplest path that is expected from this assignment, feel free to design any path that may be more complex than the one shown here.



Fig. Example of a 3D hot air balloon Fig. Diagram of the hot air balloon's trajectory

Tip: Consider using Lerp

6. Vegetation

There should be at least 5 trees in your virtual scene. You may add other kinds of vegetation as well, for example grass patches, flowers, bushes, to make your tour scenic. The trees can be a downloaded model or designed using a 3D cylinder and cone or sphere. Distribute the trees evenly around your virtual as you will need this in Assignment 2.



Fig. Simple 3D tree

Day and Night

Your scene should start during the day. For day-time, you will need a directional light source placed at a high point to act as the Sun. The day/night cycle should be scripted by time, for example the day/night mode can be 100/100 seconds long.

For the night mode, you should have a moon that illumines the scene. All the lights in your scen(e.g.: headlights of the cart) should be turned on during the night mode. The intensity of the

moonlight should be enough that the virtual objetcs in your scene are visible. You may design your virtual scene as beautifully as you want for the night mode, so that a user may enjoy the night-time view of the cart ride.

Extra credit: Design the Sun movement such that the Sun "rises" and "sets".

Camera and Teleportation

The user should be able to teleport in two different modes: the default scene mode and inside-the-cart mode.

<u>Scene mode (default)</u>: The camera (i.e., the user's eye) should be initially located and oriented such that the pond, the cart, at least one building, at least 2 trees, and the hot air balloon, is within its frustum (visible). The camera orientation may change as the user navigates through the scene. Thus, this camera mode should cater for head rotation. Input from your mobile device's gyroscope will be used to adjust the parameters for the camera orientation.

<u>Inside-the-cart mode:</u> Once teleported inside the cart, the camera should be attached to a view that looks from inside the cart. Only the user's head rotation should be tracked. When the user is teleported inside the cart, the cart should automatically start its motion. Once the user teleports out of the cart, the cart should be placed at its default location on the path and should be stationary.

Teleportation

The main interactions in this assignment will be how to select and how to teleport. For this assignment and given our hardware limitation, the simplest implementation is by *gaze* (staring at an object).

To teleport inside the cart, the user should *gaze* at the cart, say for at least 5 seconds. This can be implemented using Unity's Ray casting functionality (for reference, see <u>Gaze in Unity</u>). To help the user interact, a loading object (icon or simple 3D object like a cube) should appear when staring at an object to show the user that she/he is ready to teleport. If the user looks away, the loader should disappear and the timer reset.

In order to teleport out of the cart and back to the default scene mode, the user should gaze at their feet for at least 5 seconds (virtual food for thought: how do you figure if the user is looking at their feet? Hmmm.... Hint: think about it in terms of your phone's gyroscope). Similar to teleporting inside the cart, a loading icon should appear when staring down, to show the user that the game is ready to teleport. If the user looks away, the loader should disappear, and the timer reset.

Some guidelines

Before starting this assignment, if you are not familiar with Unity and have not tried out Assignment 0, we strongly suggest that you go through Assignment 0 and complete it first. Additionally, check out the Unity Manual and see the Unity reference page on Input for a comprehensive overview of its functionality.

Before starting the assignment, think carefully about how you structure your scene's object hierarchy. Begin by designing each object separately and test that your camera works fine. Then, put all the modes together in your virtual world and test out your teleportation and navigation.

<u>Setup</u>

This assignment should run in stereo view, which you can view using Google Cardboard or any head-mounted phone holder with stereo lenses.

To enable stereo rendering for Google Cardboard, please follow the following documentation: <u>Quickstart for Google Cardboard for Unity</u>.

I also came across this decent YouTube tutorial video.

Submission

It is strongly recommended that you test your application before submitting your work. Please save your Unity project in a folder named "CSE366 Assignments" in Google Drive and enable the *share of this folder* rather than sharing the single project file. In Brightspace, please submit the link to your Google Drive folder.

For this assignment, include the following for full credit: your Unity project folder a report, and video.

Your Unity project folder should contain your Unity saved scene and all your Scripts and Assets that will be required to rebuild your project. Please do not include your project executable in the folder. You will also need to upload a recording of your application in <u>stereo-view</u>. To do that, turn on screen capture on your phone and run the application. Interact with your application and make sure you show <u>all</u> aspects of the assignment: head rotation, gaze, teleportation, day/night mode, all required VR objects for this scene, and extra credit functionalities (if you have implemented them).

For your report, you should include *at least* the following:

- A title: "CSE 366 Virtual Reality, Spring 2024, Assignment 1: Basic VR"
- Your name and Stony Brook ID
- Unity version
- Hardware used
- Directory hierarchy
- Any extra functionalities/ features that you implemented for this assignment
- Bulleted list of all the extra credit tasks achieved
- Any references to the downloaded 3D models or scripts.

Extra Credit

Extra credit of up to additional 5% of the assignment value can be granted if you complete tasks highlighted in the green boxes. Each extra credit task is worth equal points. There is no partial credit for the extra credit points. You will only receive the extra credit for the task if you fully complete the task described.