

CSE 566 Virtual Reality, Spring 2020, Assignment 2: User Interaction

Tianao Wang 112819772

March 3, 2020

1 Google Drive Link

[Google Drive Link](#)

2 Background

(1) Unity version:
2018.4.14f1

(2) Hardware used:
Oculus quest

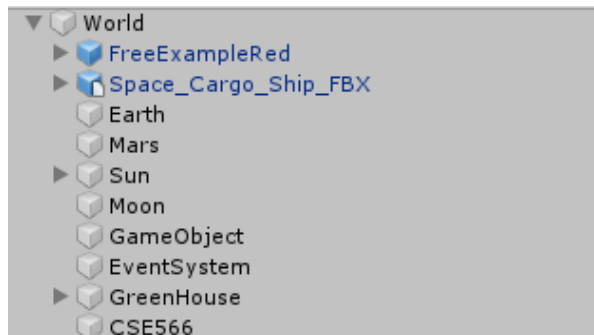
(3) Directory hierarchy:

FreeExampleRed:The spaceship. There are many items in it. Player is a player controller of Oculus. Visualizer represents the end point of the ray which casted by the controllers. UICanvas is a 3D UI which let player choose the interaction with cargo. TeleportationTube is the teleportation tube which can make player teleport to the green house. WateringCan2 is the tool which need to be carried to the greenhouse. The cardBox is a box to contain the tools. The minimap is a mini map for users to notice where they are.

The Space cargo ship is a cargo ship which can release the cargoes for player.

The Greenhouse is the green house which let user to do some tasks.

The Camera_Map is a camera which takes a image of greenhouse to let user see the map when teleporting.



3 Implement

3.1 Flying the spaceship

To implement flying ability, we need to implement rotation and movement. For rotation part, we cannot use euler angles because this will cause gimbal lock. So for this assignment, I use quaternion to rotation. I use the rotation of controller to change the rotation of spaceship. The core code is shown below.

```
//interpolate the spaceship's rotation to match the controller's rotation  
transform.rotation = Quaternion.Slerp(transform.rotation, controllerRotation, 10*speed * Time.deltaTime);
```

Then I just need to implement the movement of spaceship. I use a target in front of the spaceship which represent the place where spaceship will move to in next moment. Using two buttons of controller, I can accelerate and deaccelerate my spaceship.

```
//move cam  
TargetLocation.Translate(Vector3.forward * speed * Time.deltaTime);  
//calculate spaceship's target location  
target.position = TargetLocation.position + (controllerRotation * Vector3.forward);  
//interpolate the spaceship's rotation to match the controller's rotation  
transform.rotation = Quaternion.Slerp(transform.rotation, controllerRotation, 10*speed * Time.deltaTime);  
//move the spaceship to the target position  
transform.position = Vector3.Lerp(transform.position, target.position, speed * Time.deltaTime);
```

3.2 Minimap

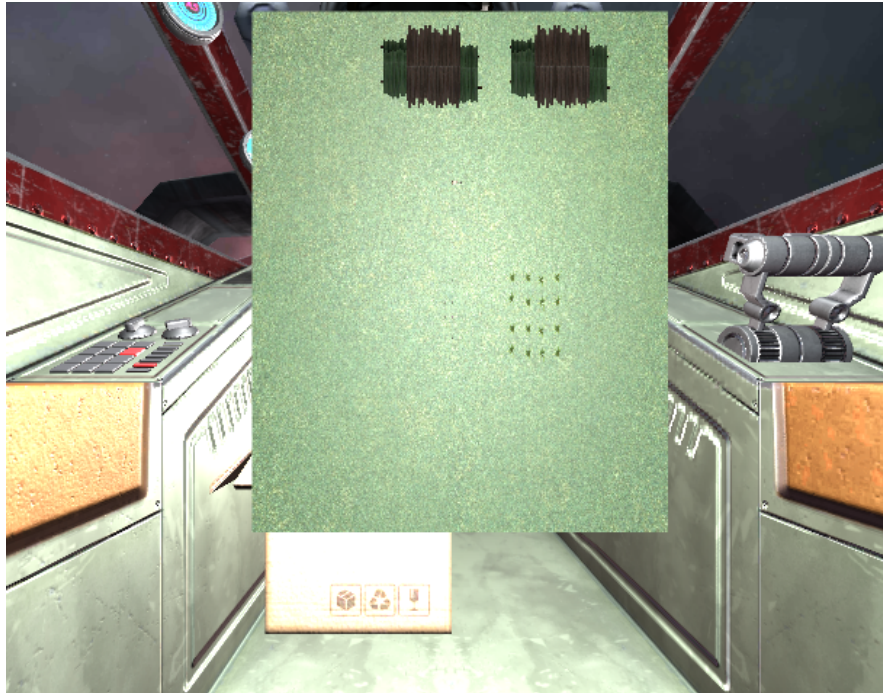
It's easy to implement a 2D minimap using camera and canvas. However, in this assignment, we need to implement a 3D minimap. To make sure the synchronize of the minimap and the whole world, I directly copy a same world and change the scale of this world. Let the mini world small enough and set it into the spaceship. In this way, the real world and this miniworld share the same script and reference which makes sure the movements of plants are the same. The only thing I need to change is the speed of my spaceship by the ratio of real world and mini world.



3.3 Teleportation Tube

The teleportation tube supports two direction. This the spaceship, when player enters the teleportation tube, it will show a 2d map of greenhouse. The 2d map is generated by a camera above the

greenhouse and is shown on a canvas which is always in front of the player. After the player confirm to teleport, the position of player will be changed to the place in the greenhouse which is on plant CSE566. In the green house, we just need to directly teleport the player back to the spaceship.



3.4 Interacting with the cargo ship

To implement this requirement, I need to let the controller cast rays and make the cargoes interact with the rays.

I follow this [tutorial](#) to let my controller cast the rays. Then I import a cargo ship model and use four cubes to represent the cubes. In the scene, if the player move ahead for a little distance, a 3D UI will be shown to ask whether the player want to accept the cargoes.

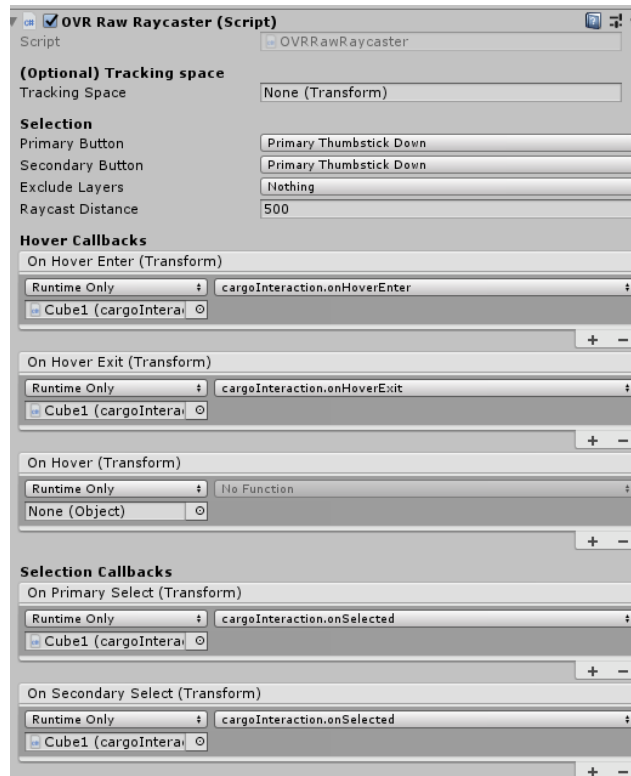


If the player select yes, the cargo will fly out of the cargo ship. It should give four cubes a force in random direction. For this assignment, I move four cubes to four specific places to avoid they run too far away from the spaceship which will make them quite hard to be recollected.

Then I implement the OnHoverEnter and IsSelected funtion to make the cargoes move to spaceship. I add a OVR Raw Raycaster and choose the function which called by different events. In the scene, when I hover on the cubes, they will change color to show that they are now selected. After selecting, I can use joystick to attract these cubes to the window. Once a cube enters the windows, the 3D UI will show that a cargo is received.

```
void Update()
{
    if (flag & flag2)
    {
        outText.text = "";
        Object.transform.position = Vector3.MoveTowards(Object.transform.position, ReceivePlace.transform.position, speed * Time.deltaTime);
    }

    if (Object.transform.position == ReceivePlace.transform.position)
    {
        flag = false;
        flag2 = false;
        outText.text = "<b>cargo loaded successfully</b>";
        Debug.Log(Object.name + "InBox");
        BoxPlace = GameObject.Find(Object.name + "InBox");
        BoxPlace.transform.localScale = new Vector3(0.25f, 0.25f, 0.25f);
        Object.SetActive(false);
        //Object.transform.position = BoxPlace.transform.position;
    }
}
```



3.5 Getting ready for the mission

For this requirement, I move a water can which is originally in the spaceship into the box and bring it to the greenhouse.

3.6 Planet CSE566 Greenhouse

In this assignment, I need to do two tasks on CSE566 greenhouse. One is watering and the other is harvesting. The player can use the A button on controller to see the 3D UI task list. After the user finish the tasks, the toggle will be set true automatically.



In the greenhouse, the player need to finish these tasks. For watering task, player can hover the ray to the water can and it will follow the player. Then player can use hand trigger for watering and the color of ground will change after watering. At the same time, the toggle of watering task will be set true.

```
public void water(Transform transform)
{
    if (transform.gameObject.tag == "wateringcan")
    {
        if(Object.transform.localRotation == Quaternion.Euler(0, 0, 0))
        {
            Object.transform.localRotation = Quaternion.Euler(0, 0, 45);
            wateringground.SetActive(true);
            task1Tog.isOn = true;
        }
        else
        {
            Object.transform.localRotation = Quaternion.Euler(0, 0, 0);
        }
    }
}
```

The second task is harvesting which is quite easier. When the ray enter the fruit which means player is harvesting the fruit, the fruit will move to the player's hand and the task toggle will be set true.

```
void Update()
{
    if (flag)
    {
        Object.transform.position = Vector3.MoveTowards(Object.transform.position, ReceivePlace.transform.position, 1 * Time.deltaTime);
        if (Vector3.Distance(Object.transform.position, ReceivePlace.transform.position) < 0.5)
        {
            Object.SetActive(false);
            task1Tog.isOn = true;
        }
    }
}

0 个引用
public void onHoverEnter(Transform transform)
{
    if (transform.gameObject.tag == "turnip")
    {
        Object = GameObject.Find(transform.gameObject.name);
        Debug.Log("on");
        flag = true;
    }
}
```

4 Reference

- (1) Starfield Skybox,PULSAR BYTES, <https://assetstore.unity.com/packages/2d/textures-materials/sky/starfield-skybox-92717>
- (2) SolarSystemPackage,Simon István <https://assetstore.unity.com/packages/essentials/legacy-image-effects-83913>
- (3) HiRezSpaceshipsCreatorFree,Ebal Studios,<https://assetstore.unity.com/packages/3d/vehicles/space/hi-rez-spaceships-creator-free-sample-153363>
- (4) Oculus Integration,Oculus,<https://assetstore.unity.com/packages/tools/integration/oculus-integration-82022>

- (5) Garden Realistic Tools,GBAndrewGB,<https://assetstore.unity.com/packages/3d/garden-realistic-tools-68960>
- (6) Bhutan Farmhouse,Bretwalda Games,<https://assetstore.unity.com/packages/3d/environments/urban/bhutan-farmhouse-19360>
- (7) Ground Materials FD Free,Fausto D,<https://assetstore.unity.com/packages/2d/textures-materials/floors/ground-materials-fd-free-140364>
- (8) Cartoon Farm Crops,False Wisp Studios,<https://assetstore.unity.com/packages/3d/vegetation/plants/cartoon-farm-crops-79777>
- (9) Cardboard Boxes Pack,PolyWorkshop,<https://assetstore.unity.com/packages/3d/props/cardboard-boxes-pack-30695>
- (10) Space Cargo Ship,Silas Ferigato,<https://assetstore.unity.com/packages/3d/vehicles/space/space-cargo-ship-19537>