CSE 566 Virtual Reality, Spring 2020, Assignment 3: Augmented Reality

Tianao Wang 112819772

April 6, 2020

1 Google Drive Link

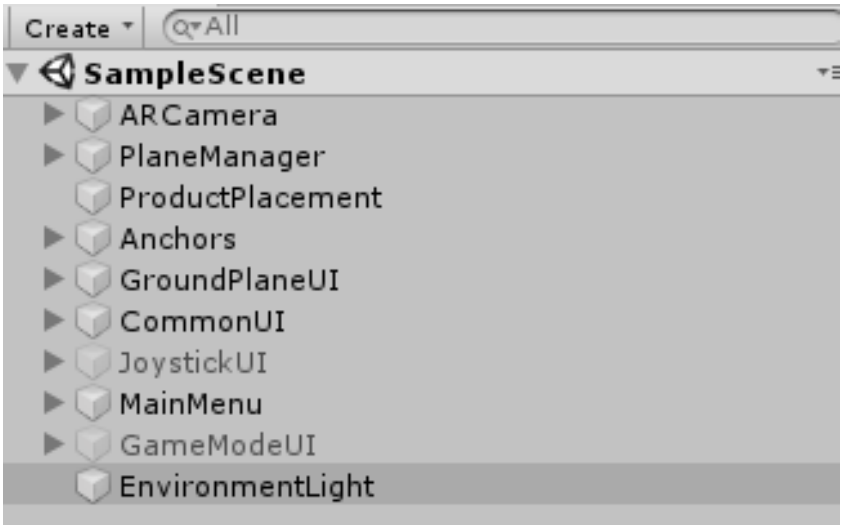
Google Drive Link

2 Background

(1) Unity version:
2018.4.14f1

(2) Hardware used:
iphone11 & Macbook pro

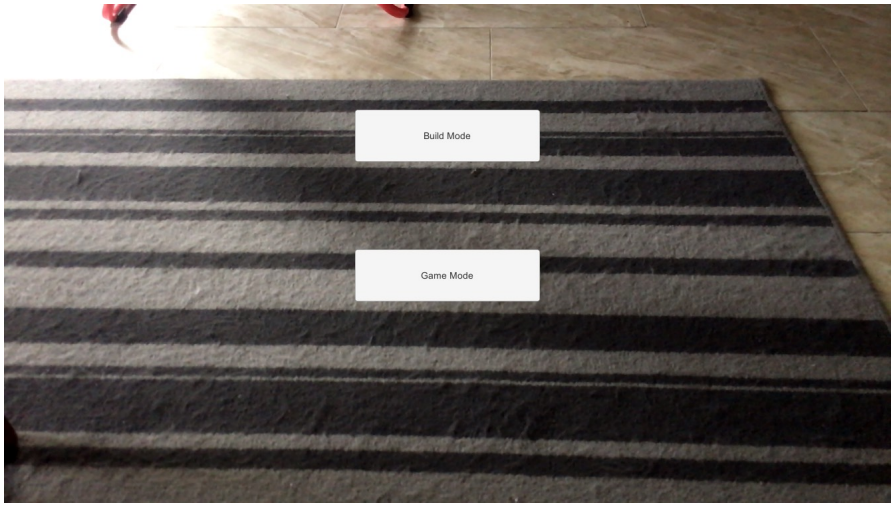
(3) Directory hierarchy:
ARCamera: Vuforia camera
PlaneManager: manager for plane
Anchors: Object used in the project
GroundPlaneUI: Build mode UI
JoyStickUI: UI for JoyStick
MainMenu: UI for main menu
GameModeUI: UI for game mode
Environment Light: Light for day and night mode



3 Implement

3.1 UI

When the user open the application, the main menu will be shown to let user to choose build mode or game mode.



After the user choose the build mode, a build mode UI will appear to let user to choose what object to place.

3.2 BuildMode

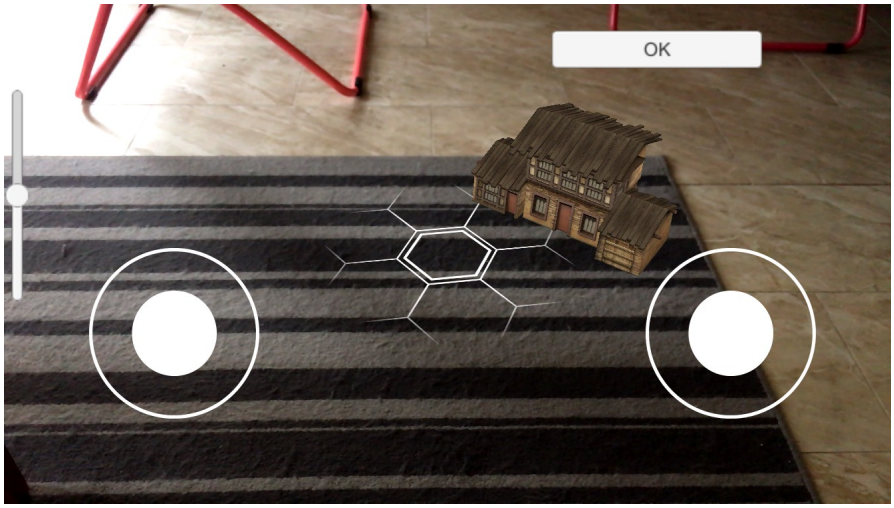
3.2.1 Choose and set object

I refer the Vuforia official example 3-groundplane to implement this requirement.
It can detect the ground and create planes to set up objects. It can set object into mid air or on the ground. Each time the user select to set up a object, it will change the mode and find the current plane to put the object in that plane.

```
if (CurrentPlaneMode == PlaneMode.MIDAIR)
{
    Debug.Log("PlaceObjectInMidAir() called.");

    // With each tap, the Drone is moved to the position of the
    // newly created anchor. Before we set any anchor, we first want
    // to verify that the Status=TRACKED/EXTENDED_TRACKED and StatusInfo=NORMAL.
    if (TrackingStatusIsTrackedAndNormal)
    {
        this.contentPositioningBehaviour.AnchorStage = this.midAirAnchor;
        this.contentPositioningBehaviour.PositionContentAtMidAirAnchor(midAirTransform);
        UtilityHelper.EnableRendererColliderCanvas(this.midAirAugmentation, true);

        this.midAirAugmentation.transform.localPosition = Vector3.zero;
        UtilityHelper.RotateTowardCamera(this.midAirAugmentation);
        JoyStickUI.SetActive(true);
        CurrentObject = this.midAirAugmentation;
        Debug.Log(CurrentObject.name);
        Debug.Log(CurrentObject.transform.position);
        modifyFlag = true;
    }
```

3.2.2 Joystick

Once user choose an object and place it into the space, a joystick UI will be shown for user to change the position and rotation of the obejcts.



The user can use the left joystick to change the x axis and y axis of spaceships, plants and change the x axis and z axis of greenhouse.

```
if (xyjoystick.Horizontal!=0|| xyjoystick.Vertical != 0)
{
  CurrentObject = PlaneManager.CurrentObject;
  if (CurrentObject.tag == "greenhouse")
  {
      rigidbody = CurrentObject.GetComponent<Rigidbody>();
      rigidbody.MovePosition(rigidbody.position + new Vector3(xyjoystick.Horizontal * 1f, 0, xyjoystick.Vertical * 1f) * Time.deltaTime);
  }

  if(CurrentObject.tag == "spaceship"||CurrentObject.tag == "plant")
  {
      rigidbody = CurrentObject.GetComponent<Rigidbody>();
      rigidbody.MovePosition(rigidbody.position + new Vector3(xyjoystick.Horizontal * 1f, xyjoystick.Vertical * 1f, 0) * Time.deltaTime);
  }
```

User can rotate the objects with the right joystick.

```
if(rotatejoystick.Horizontal != 0 || rotatejoystick.Vertical != 0)
{
    CurrentObject = PlaneManager.CurrentObject;
    Vector3 dir = new Vector3(rotatejoystick.Horizontal * 3f, 0, rotatejoystick.Vertical * 3f);
    Quaternion qua = Quaternion.LookRotation(dir.normalized);
    CurrentObject.transform.rotation = Quaternion.Lerp(CurrentObject.transform.rotation, qua, Time.deltaTime * 0.3f);
}
```

User can change the z axis of spaceships and plants with the slider.

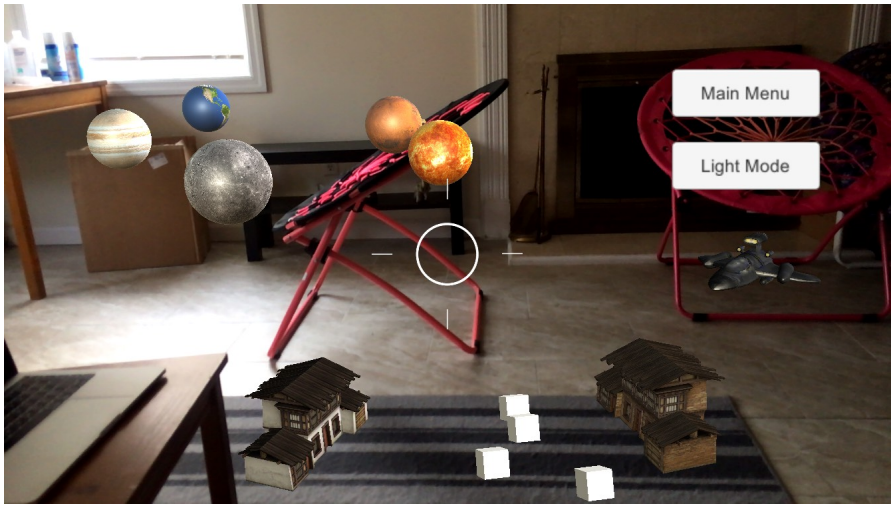
```
if (sliderMove)
{
    CurrentObject = PlaneManager.CurrentObject;
    if (CurrentObject.tag == "greenhouse")
    {
        rigidbody = CurrentObject.GetComponent<Rigidbody>();
        rigidbody.MovePosition(rigidbody.position + new Vector3(0, (ZSlider.value - tempSliderValue) * 1f, 0) * Time.deltaTime);
        tempSliderValue = ZSlider.value;
        sliderMove = false;
    }

    if (CurrentObject.tag == "spaceship" || CurrentObject.tag == "plant")
    {
        rigidbody = CurrentObject.GetComponent<Rigidbody>();
        rigidbody.MovePosition(rigidbody.position + new Vector3(0, 0, (ZSlider.value - tempSliderValue) * 1f) * Time.deltaTime);
        tempSliderValue = ZSlider.value;
        sliderMove = false;
    }
```

After setting up the object, User can press the OK button to finish.

3.3 GameMode

3.3.1 rotate around and move

In the game mode, the plants will rotate around the sun using the code I write in the homework1 and the spaceship will move between the greenhouses.



3.3.2 day and night mode

I add some points light at the spaceships and greenhouses and direction light in the environment. User can use the light mode button to change day and night mode. When day mode is on, the environment light will turn on and point lights will turn off. Vice Versa.

3.3.3 drop and drag cargo

For dropping cargoes, I set up a random value and in each random seconds the spaceship will drop a cargo to a random place around the greenhouse.

```
void Start()
{
    int random = UnityEngine.Random.Range(2, 10);
    isDraged = false;
    InvokeRepeating("dropCargo", 0f, random*1f);
}
```

```
0 个引用
void dropCargo()
{
    if (PlaneManager.gameModeFlag)
    {
        newposition = position.transform.position + new Vector3(UnityEngine.Random.Range(-0.001f, 0.001f),
            0, UnityEngine.Random.Range(-0.001f, 0.001f));
        isDraged = true;
    }
}
```

```
if (isDraged)
{
    cargo.transform.position = Vector3.MoveTowards(cargo.transform.position, newposition, 1f * Time.deltaTime);
}
else
{
    cargo.transform.position = startPosition.transform.position;
}
```

For drag object, I refer the official document of Vuforia to drag object.
It cast a ray from the screen and hit the object to select the object. Once the object be selected, user can use finger to move the object. It will calculate the move of finger can transform the movement to the object to implement the move of cargo.

```
if (touch.phase == TouchPhase.Began)
{
    Debug.Log("test1 began");
    //Struct used to get info back from a raycast
    RaycastHit hit = new RaycastHit();
    if (Physics.Raycast(ray, out hit, 1000))
    { //True when Ray intersects colider.
        pickedObject = hit.transform;
        Debug.Log("test1 hit " + hit.transform);
        lastPlanePoint = planePoint;
    }
    else
    {
        Debug.Log("test1 set null1 " + pickedObject);
        pickedObject = null;
    }
```

```
else if (touch.phase == TouchPhase.Moved)
{
    Debug.Log("test1 moved");
    if (pickedObject != null)
    {
        Debug.Log("test1 planePoint " + planePoint);
        pickedObject.position += (planePoint - lastPlanePoint)*0.5f;
        lastPlanePoint = planePoint;
    }
    //Set pickedObject to null after touch ends.
}
else if (touch.phase == TouchPhase.Ended)
{
    Debug.Log("test1 ended");
    Debug.Log("test1 set null2 " + pickedObject);
    pickedObject = null;
}
```

If the cargo touch the greenhouse, it will be received by the greenhouse and will be reset as a new cargo in the spaceship.

```
0 个引用
private void OnTriggerEnter(Collider other)
{
    if(other.tag == "greenhouse")
    {
        cargo.transform.position = startPosition.transform.position;
        //newposition = position.transform.position;
        newposition = position.transform.position + new Vector3(UnityEngine.Random.Range(-0.001f, 0.001f),
            0, UnityEngine.Random.Range(-0.001f, 0.001f));
        isDraged = false;
        Debug.Log("cargo received!");
    }
}
```

4 Asset Reference

(1) SolarSystemPackage,Simon István,Link

(2) Joystick Pack,Fenerax Studios5,Link

(3) Galactic Heroes Cartoon Spaceship, Funky Finger Productions Inc.,Link

(4) Bhutan Farmhouse,Bretwalda Games,Link

5 Code Reference

(1) Drag.cs,Vuforia official document,Link

(2) PlaneManager.cs, Vuforia official toturial scenes 3-groundplane,Link