Spatial augmented reality: practical assignment

Project setup

1. **Assets ➔ Import package ➔ Effects ➔ Projectors**
2. Create the following folders for your assets: *Materials, Meshes, Textures, Scripts*
3. Import the provided meshes in the *Mesh* folder
4. Import the provided textures in the *Textures* folder

Scene creation

1. Create two empty objects named *VirtualScene* and *RealScene* (*Note: in the paper, the RealScene is called “Scene representing the real world”).
2. For the *VirtualScene*
   a. Place *VirtualScene* at position (0, 0, 0), without any rotation
   b. Add a *Plane* that represents the ground at position (0, 0, 0) with dimension 3x3x3
   c. Create and apply to the ground a new material named *RedReflective*
   d. Move the camera *Main Camera* in the hierarchy of *VirtualScene* at position (0, 2, -7)
   e. Modify the attribute *Viewport Rect* of *Main Camera*: X=0 ; Y=0 ; W=0.5 ; H=1
3. For the *RealScene*
   a. Place *RealScene* at position (35, 0, 0), without any rotation
   b. Add a *Plane* that represents the ground at position (0, 0, 0) with dimension 3x3x3
   c. Create and apply to the ground a new material named *GreenReflective*
   d. Duplicate the camera *Main Camera* of the *VirtualScene*, then move it to position (0, 2, -7) in the hierarchy of *RealScene*
   e. Modify the attribute *Viewport Rect* of *Main Camera*: X=0.5 ; Y=0 ; W=0.5 ; H=1

Exercise 1: Screen simulation

1. Add the textured mesh in the hierarchy of *VirtualScene* at position (0, 0, 0) with dimension 2x2x2, and call it *VirtualObject*
   a. Drag & Drop the mesh into the hierarchy of *VirtualScene*
   b. Drag & Drop textures onto the created object
2. Add a Quad named RealScreen into the hierarchy of RealScene at position (0, 2.5, 0) with dimension 5x5x5
3. Add a new camera named VirtualCamera to the hierarchy of VirtualObject. This camera will be used to acquire images that we want to display on RealScreen. Place it where you want, as long as it “sees” the object VirtualObject
4. Create a RenderTexture named virtual_render_camera, then attach it to the attribute TargetTexture of VirtualCamera
5. Drag & Drop virtual_render_camera onto RealScreen in order to associate them

**Question:** What happens if you modify the dimension of RealScreen to 8*5*5? How can you resolve the problem?

6. Add a new script C# named SetAspectRatio, then attach it to VirtualCamera
   a. Follow the instructions
7. To activate the script, press the button Play on top of the window

**Exercise 2: Projector simulation**

1. Deactivate RealScreen
2. Reduce the light intensity of the Directional Light to 0.5 (in order to improve the legibility of the scene)
3. Add the diffuse mesh in the hierarchy of RealScene at position (0, 0, 0) with dimension 2x2x2, then call it RealObject. Modify its material if necessary.
4. Add a BlobLightProjector named RealProjector in the hierarchy of RealObject.
   a. You can find the BlobLightProjector in the following:
      Assets ⇒ Standard Assets ⇒ Effects ⇒ Projectors ⇒ Prefabs
5. Display the image of VirtualCamera with RealProjector
   a. Attach the texture virtual_render_camera to the attribute Cookie of the property LightProjector

**Question 1:** Does the projection superimpose correctly on RealObject. If not, how can you solve the problem?
   ⇒ You need to set the transformation between VirtualCamera and VirtualObject to the same as the one between RealProjector and RealObject

6. Copy paste the Transform component of RealProjector into the Transform component of VirtualCamera

**Question 2:** Does the projection superimpose correctly on RealObject. If not, how can you solve the problem?
   ⇒ You need to set the intrinsic parameters of VirtualCamera to the same as the ones of RealProjector

7. Modify the aspect ratio and the field of view of VirtualCamera to correspond to the one of RealProjector