

COMS W4172 3D UI and AR (2020 Spring)

HW 3

Shusen Xu - UNI: SX2261

2020.4.16

Miscellaneous

1. Software Version

Unity Version: 2019.3.0f6

Vuforia version: Vuforia Engine 9.0.12

2. Scripts and functionality introduction:

(a) Enemy Base

- i. EnemyBase.cs: Controlling the spawning of enemies in an appropriate frequency.
- ii. EnemyBaseController.cs: Controlling the life cycle of enemy bases by calculating health value which is affected by projectiles from tower base. When its health value turns to zero, then it is destroyed. And it destroys all projects contacting with it.
- iii. FireSystemEnemyBase.cs: Controlling the launcher firing in a fixed frequency and aiming at the TowerBase.

Note: The controller of EnemyBase repawning after being destroyed is implemented at TowerSystemController.cs.

(b) Projectiles

ProjectileController.cs: Controlling the life cycle of the projectiles

(c) Tower Base

- i. TowerBaseController.cs: Controlling the life cycle of tower base by calculating its health value which is affected by projectiles from enemy bases and enemies. When bases When its health value reaches zero, then it is destroyed. And it destroys all projects and enemies contacting with it.
- ii. TowerFireController.cs: Controlling the firing of the turret in a fixed frequency and aiming.
- iii. TowerSystemController.cs: Containing the main logic of this game, such as respawning EnemyBases after they were destroyed, the wall block creation and switching selection and manipulation modes.

(d) Wall

WallController.cs: Controlling the life cycle of walls by calculating health value which is affected by projectiles and enemies. When its health value turns to zero, then it is destroyed. And it destroys all projects and enemies contacting with it and produces physics response to projectiles.

- (e) Selection and Manipulation
 - i. VirtualHandController.cs: Controlling the selection and manipulation operations(rotate, translate, scaling) by simple virtual hand.
 - ii. RayCastingController.cs: Controlling the selection by Ray Casting and manipulation operations.
- (f) AutoFocus and Extended Tracking
 CameraControlling.cs, CameraSettings.cs and TrackableSettings.cs: controlling autofocus and extended tracking.

User Interface Description

corresponding to 10 rules

1. Image targets

As Figure 1 shows, there are three image targets, one is for game presentation, and another two are wands which separately are the virtual hand and the ray casting pointer.



Figure 1: Left: Main Image Target, Middle: Image Target for the wand of virtual hand, Right: Image Target for the wand of ray casting pointer

2. Model prefabs As the Figure 2 and Figure 3 show, we can have an overview of adopted models:

3. Button trigger:

Considering the rule called Visibility of system status, when a button was pressed, the color of will turn green. As the Figure 4 shows, the description of several buttons as a trigger is as follows:

- (a) "SwitchMode" button:
Switching between virtual hand mode and ray casting pointing mode.
- (b) "CreateWall" button:
Trigger the creation of walls for supplementing protective blocks to the Tower Base.
- (c) "ClutchOrNot" button:
Confirmed selection operation or deselect the currently manipulated object. For example, if meets with an object which is not selected, clicking this button will confirm the selection operation(i.e attaching the object to the wand). If an object is selected and the user want to deselect it, the user could click on this button once again.
- (d) "Scale", "Rotate" and "Translate" buttons:
When an object is in the selected status, clicking on these three above buttons for triggering different manipulation operations.

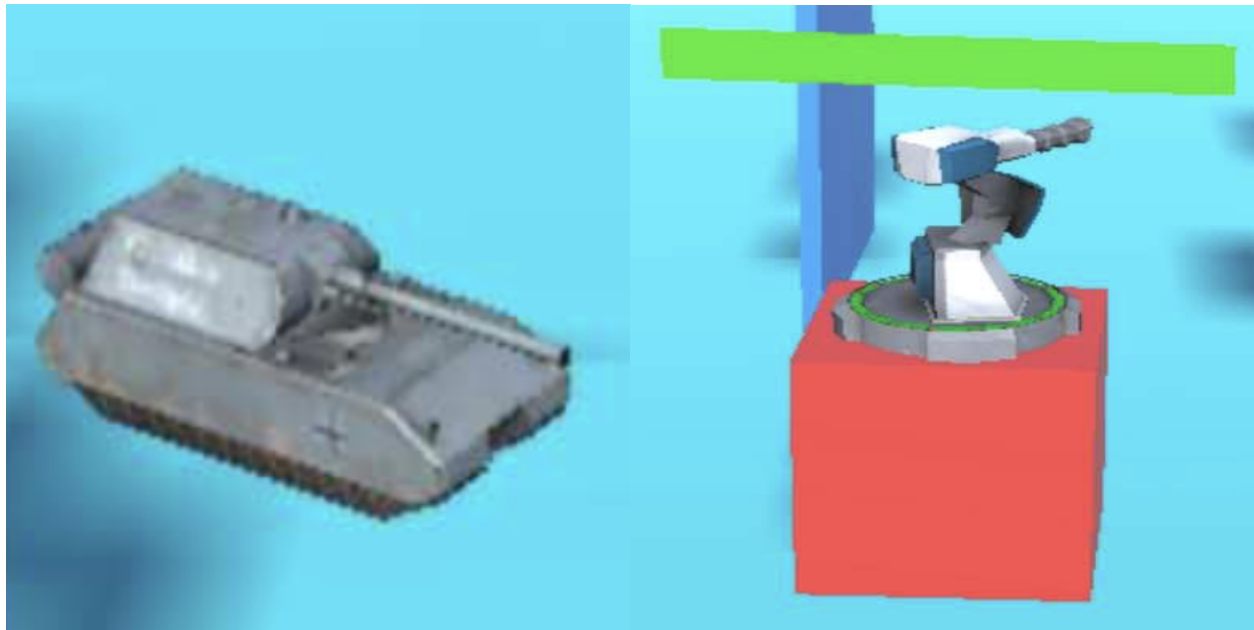


Figure 2: Left:Enemy Prefab, Right: Enemy Base Prefab



Figure 3: Left:Wall Prefabs, Right: Tower Base Prefabs

(e) "Undo" button:

Concerning flexibility and efficiency of use and controlling, the "Undo" button was set. If users carelessly manipulating the selected objects and get a unwanted result, users can bring the selected object back to its original transform and material without complex operations. Also, in my application, when finding a selected object, the wand(virtual hands or ray casting stick) will change colors to yellow which reminds user currently contacting with selected objects. At this time, if user do not want select this object, the "undo" button can make the want back to its original status, recovering its original materials and color.

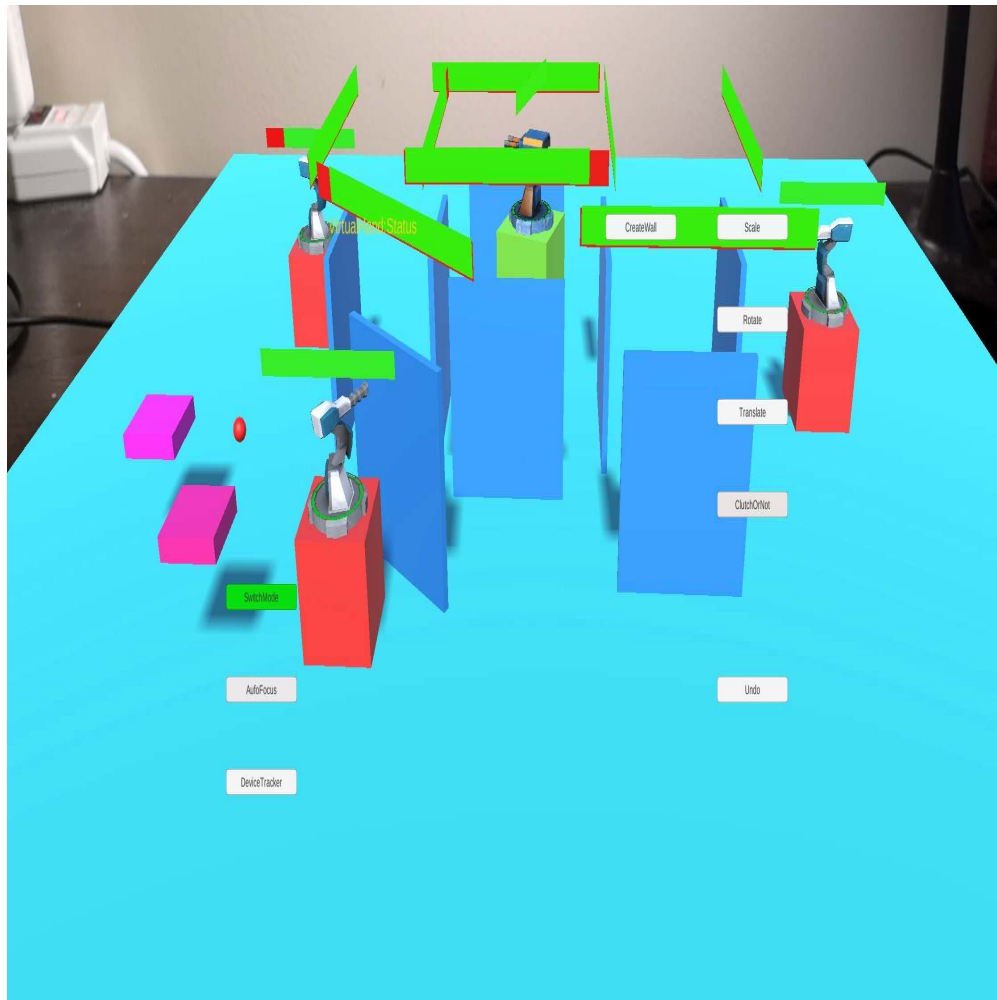


Figure 4: Main User Interface

- (f) "AutoFocus" button: Trigger the camera to auto focus the recognized image targets
- (g) "Device Tracker" button: Trigger Extended tracking.

4. Status Info text:

There is a text UI providing the game status and reminding the next operation. This UI meets with the requirement of design rule called Recognition rather than recall, guiding user. Also, it helps user clearly recognize the current status and avoid making errors. For example, the text can point out the current mode (virtual hand mode or ray casting mode), remind the current status such as manipulations (Rotate, Translate, Scale), which object is being holding by virtual hand or pointed at by ray casting.

Virtual Hands Mode

Description: In my application, I adopt the simple virtual hand which I simplified to a pink 3D cube which intersects with objects. After selection, position/orientation of the image target (as shown in Figure 1) are mapped to virtual hand (3D cube) and the changing position and orientation of this virtual hand could manipulate the held object. The detailed procedures are as follows:

1. Selection:

Putting the image target (the middle one as shown in the Figure 1), when the image target is recog-

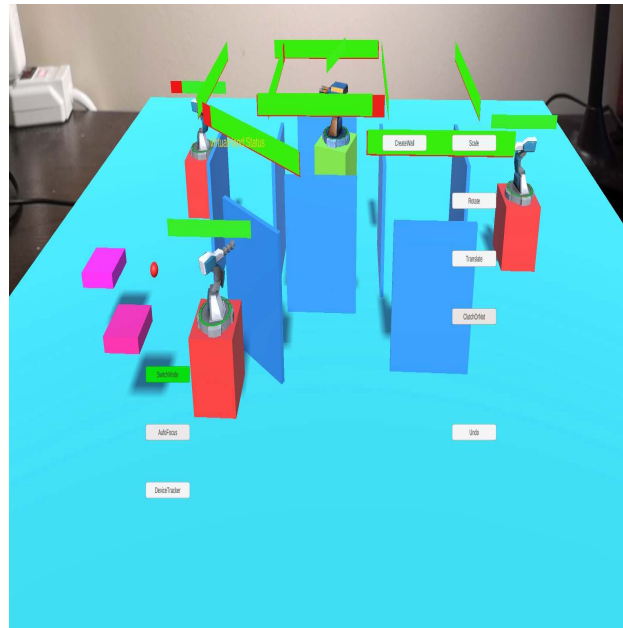


Figure 5: Switching to Virtual Hand Mode by "SwitchMode" button

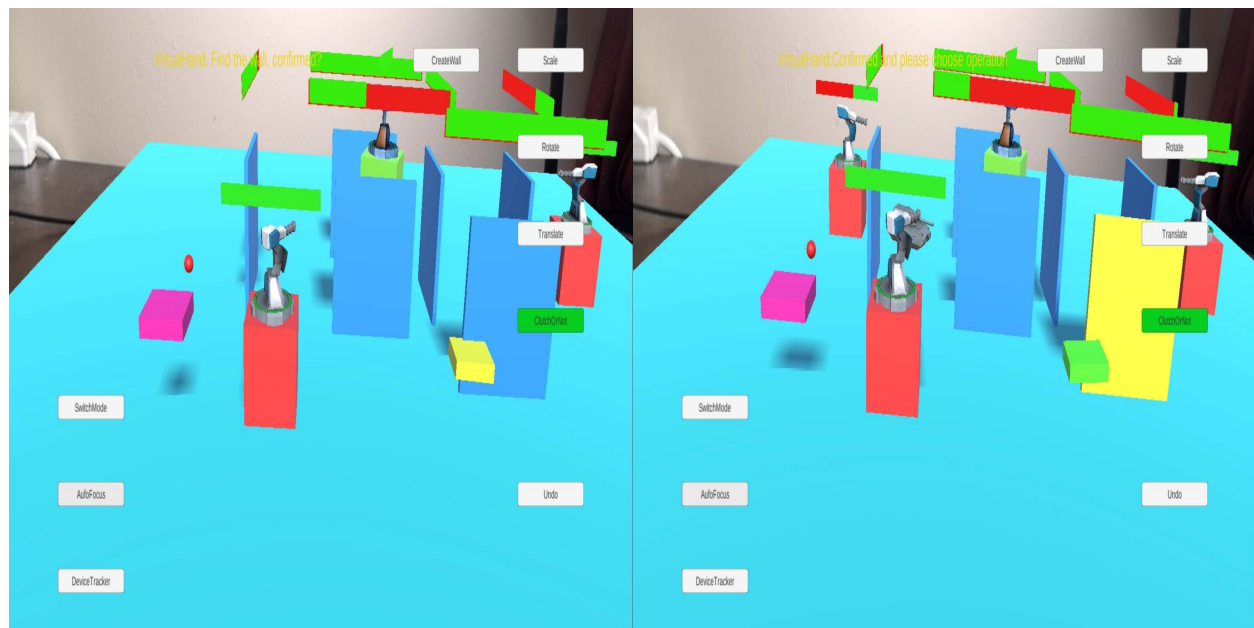


Figure 6: Left: Finding the selectable wall, Right: Clutching the selectable wall

nized, there is a 3D pink cube appearing as a virtual hand. Changing the position and manipulation of the image target(wand) will directly transform the virtual hand.

For selection, users need to move the virtual hand(3D pink cude) towards the object you want to select, in order to directly touch the object. When the virtual hand and object meet, the the material of the virtual hand will turn yellow(as shown in the left pictures in Figure 6 and Figure 7.), reminding users that the virtual hand has find the selected object. In the same time, the Status Info text will remind the user and guide to the confirmation of selection.

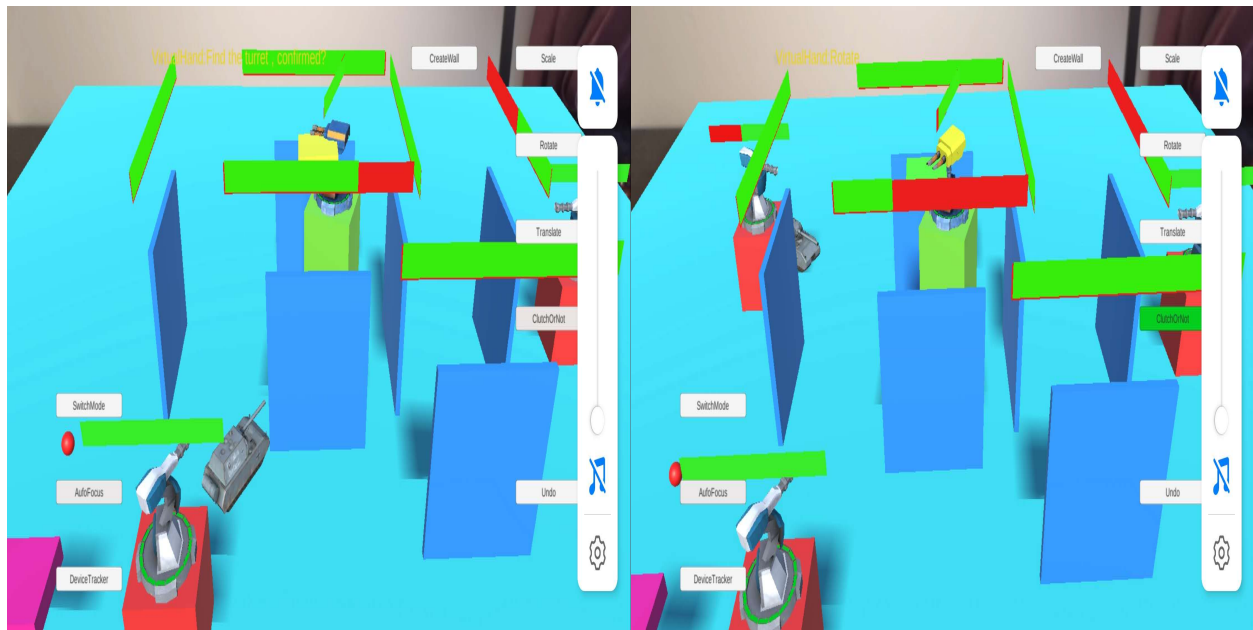


Figure 7: Left: Find the selectable turret, Right: Clutch the turret

As shown in the right pictures in Figure 6 and Figure 7, after clicking on "ClutchOrNot" button, the object will be attached to the virtual hand. What's more, the material color the virtual hand will turn green, suggesting there is a objected selected by this hand. In the same time, the clutched object will turn yellow, suggesting this object is currently being manipulated. The Staus Info text will also remind the next operation, stating that "Confirmed and please choose manipulation operation"

The deselection requires users to click "ClutchOrNot" button again, declutching the object with the vital hand. And their both material turns to their original material.

Note that: after being selected, the object cannot be affected by physics and hurts from projectiles and enemies.

Also, for wall objects, it requires users to choose the way they want to manipulate them (rotate, translate, scaling). However, for turret model, after selection, rotation will be set as the default manipulation.

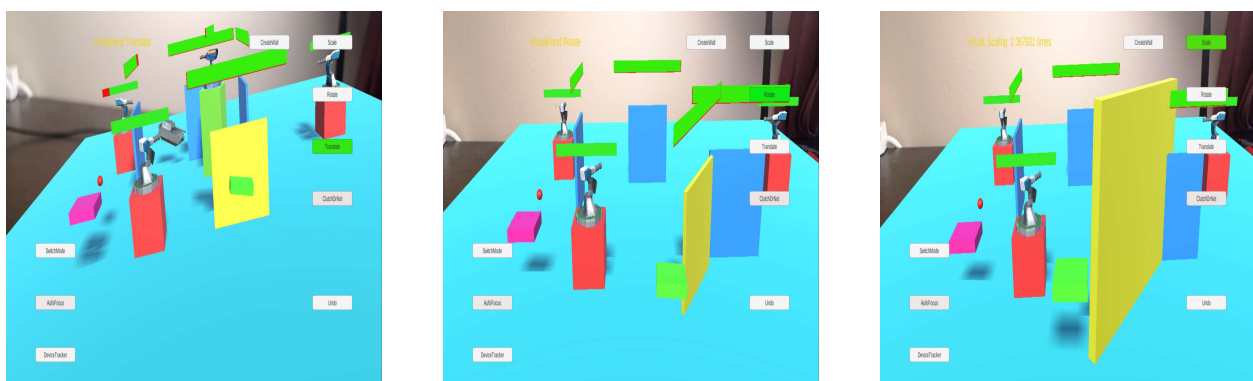


Figure 8: Left: Translate the wall, Middle: Rotate the wall Right: Scaling the selectable wall

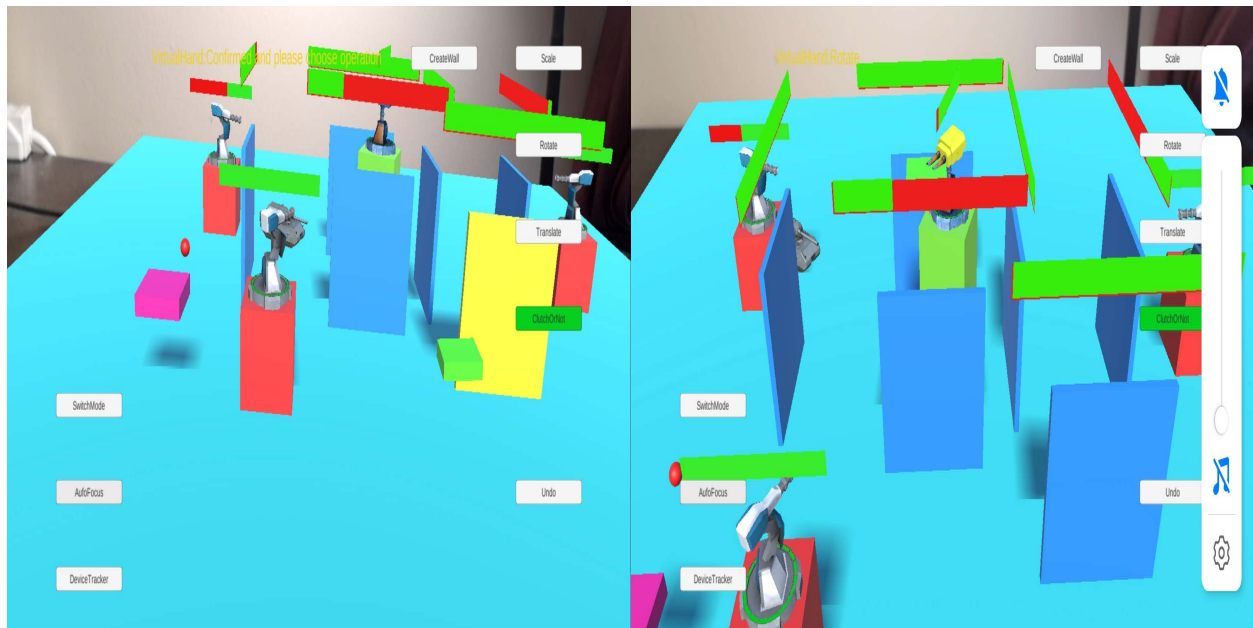


Figure 9: Rotate the turret

2. Translate :

As shown in the left picture in Figure 8. After selection, then clicking "Translate" button, the position of virtual hand are mapping to the selected object. Thus moving the virtual hand brings about the translate of the selected object.

3. Rotate:

As shown in the middle picture in Figure 8. After selection, then clicking "Rotate" button, the game adopts "Transform.LookAt()" function keeps the selected objects always faces(forward vector direction) towards the virtual hand. By this way, we can easily changing the orientation of the selected object by moving slightly the virtual hand.

4. Scale:

As shown in the right picture in Figure 8. After clicking "Scale" button, user can scale up by moving the virtual hand farther, scale down by moving the virtual hand closer to the center of the selected object in whichever direction they want.(maybe left or right is the easiest direction.) Also, the Status Info text will represent the current size of the selected object.

Note, only if there is a currently selected object, user can flexibly switch among these three above operations. And "Undo" button can bring the selected object back to its original transform status if something unwanted error happens.

5. The evaluation and Rationale

(a) Advantages(Rationale selecting this technology):

Isomorphic: Virtual hand operation perform similarly as the hand manipulation in the natural scenes. User can easily understand and become familiar with this operation with low learning cost.

(b) Disadvantages(Limitation):

User can only select objects within arm's reach, so they may need to travel to select objects farther away, which is quite inconvenient.

Ray Casting Mode

Description: In this mode, I adopts the ray casting as the selection method. After selection, position/orientation of the image target for ray casting mode are mapped to a 3D cube and a small red sphere which used for determining the ray casting direction. Also, the changing position and orientation of this wand could manipulate the attached object. The detailed procedures are as follows:

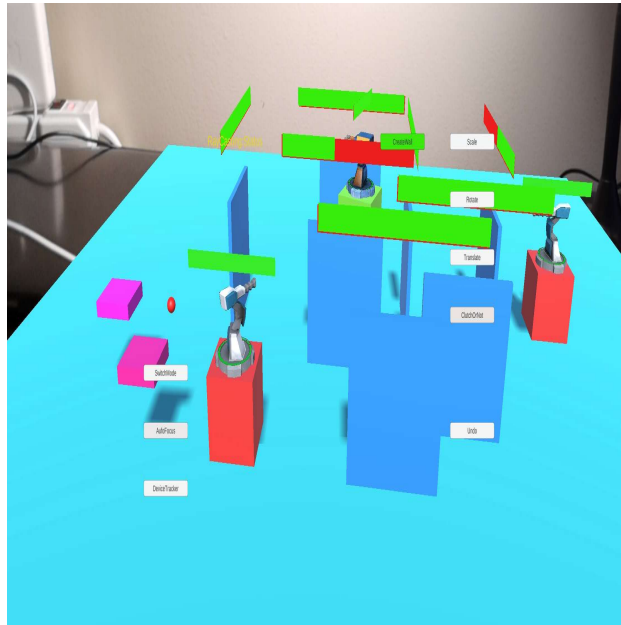


Figure 10: Switching to Ray Casting Mode(default) by "SwitchMode" button

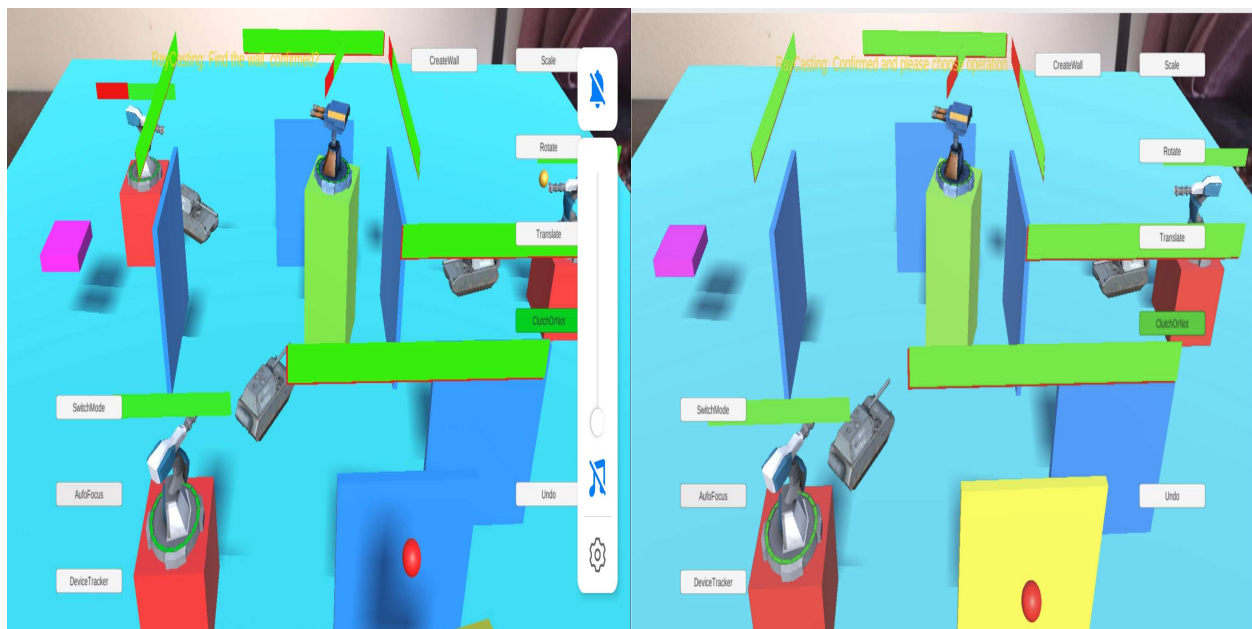


Figure 11: Left:Find the selectable wall, Right: Clutch the wall

Selection:

Unlike the virtual hand mode, ray casting selection does not require users to move, hold or contact with selectable object. There are a ray casting continuously radiate from the wand(a 3D pink cube) in the direction pointing from the wand(a 3D pink cube) to the a red sphere marker(used for indicate the direction only). When the ray hits with the object, the status info text will remind user which object has been founded and guide the next step to confirm this selection by clicking "ClutchOrNot" button. After clicking on that button, the material color of selected object will turn yellow and the material color of this wand will turn green as shown in Figure 11.

When there is a object being selected, clicking "ClutchOrNot" button will deselect the selected object. And both material color of the wand(cube) and selected objects turn to their original material.

Note that: after confirmed selection for "wall", the status info text will remind user that "confirmed and choose manipulation operation"(rotate, translate, scaling). However, for "turret", user do not have to choose manipulation options since the default manipulation has been set to rotation in pitch or yaw.

Translate:

As shown in the left picture in the left picture of the Figure 12. After selection, then click "Translate" button. By this way, the position of wand can be mapped to selected object in a way keeping related position with the wand as if the selected object has been attached to a pointing stick.

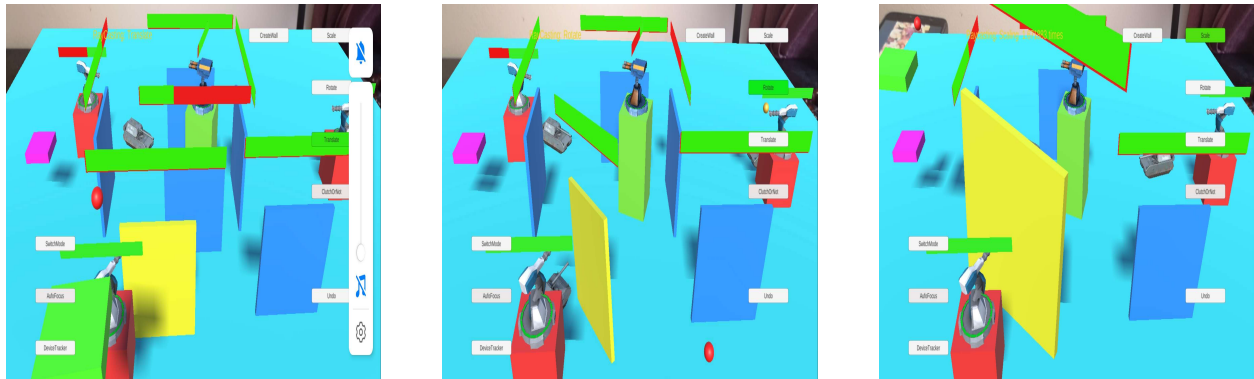


Figure 12: Left: Translate the wall, Middle: Rotate the wall, Right: Scale the wall

Rotate:

As shown in the middle picture in the Middle picture of Figure 12. After selection, then clicking "Rotate" button, similar to virtual hand mode, the game adopts "Transform.LookAt()" function keeps the selected objects always faces(forward vector direction) towards the wand. Thus, changing the position and orientation of the wand will change the rotation of the selected object.

As shown in Figure 13, similar to the method used for wall. However, the rotation of the turret has been limited in pitch and yaw.

Scale:

As shown in the right picture in Figure 12. After clicking "Scale" button, user can scale up by moving the wand further, scale down by moving the wand closer to the center of the selected object in whichever direction they want.(maybe left or right is the easiest direction.) For objects with different distance, the game assigns different scaling coefficients. Also, the Status Info text will represent the current size of the selected object.

The evaluation and Rationale

1. Advantages(Rationale selecting this technology)

First, Unlike virtual hand, it allows selection/manipulation beyond user's reach. Also, for manipulation, this ways requires less physical movement.

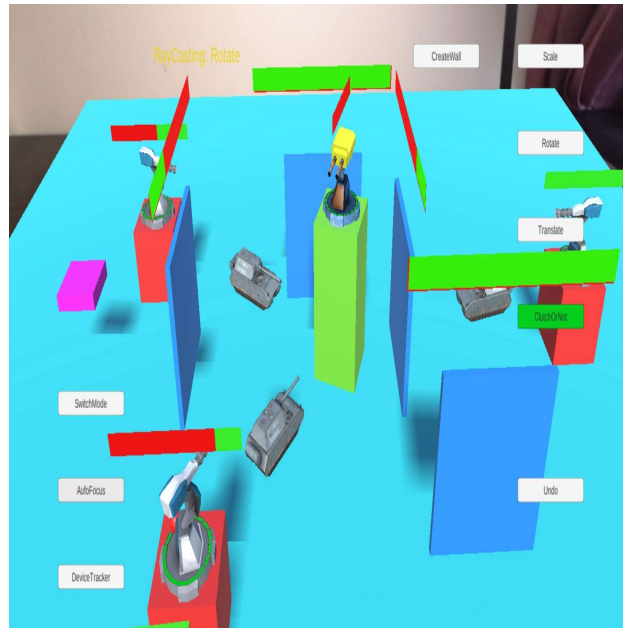


Figure 13: Rotate the turret in pitch and yaw

2. Disadvantages(Limitation):

It is quite poor positioning except radially around user. Which one object pointed by this way may be quite unclear.