Interactive Interior Design with Hololens

Final report
22.01.2017

Supervisor : Loretta Yi-King Choi

Jack Cheung Ka Kit
2013559878
Acknowledgements

I would like to thank my advisor Dr. Choi for her time, guidance, and insightful feedback, which allowed this project to progress smoothly.

Abstract

Mixed Reality is a new technology that allows users to perform physical interactions with the digital world in real time. It combines features of Virtual Reality and Augmented Reality. The project aims to explore and showcase the potential of this technology by creating a multiplayer application. The main focus of our application is interior design which user can resize, move and rotate the 3D furniture objects. This report presents our technology used, methodology, milestone and future development. The team has already done a research on the feasibility of the devices and started implementing the basic functions of the application.
Table of Content

Section 1: Project Overview

1.1 Background
1.2 Existing application and inspiration
1.3 Objective & Scope & Deliverable

Section 2: Technology used

2.1 Microsoft Hololens
2.2 Unity3D Editor 5.5
2.3 Hololens SDK
2.4 HoloToolkit-Unity

Section 3: Application Design

3.1 Block Building system - Hololens & Tango Tablet
3.2 3D drawing of Art gallery - Hololens only
3.3 Interactive Table tennis - Hololens & Tango Tablet
3.4 Interior Design - Hololens & Hololens

Section 4: Project Methodology

4.1 Configure the hardware and Installed the tool
4.2 Feasibility Research and Study
4.3 Individual development
4.4 Collaborate development
4.5 Multiplayer
4.6 Fine tuning of parameter

Section 5: Progress

5.1 Feasibility Research on Microsoft Hololens
5.2 History of application design
5.3 Individual work
   5.3.1 Old development
   5.3.2 Rotation and Scaling System
5.3 Work by Jerry
5.4 Collaborative Work

Section 6: Project Schedule

Section 7: Conclusion

Section 8: Reference
Section 1 : Project Overview

1.1 Background

In Virtual Reality, A head-mounted device covering the user’s sight and hearing is used to transmit 3D computer graphics and audio. Users can therefore immerse themselves in the digital world as if they were actually in the environment. For example, users can enjoy the view of the Grand Canyon while sitting on sofa.

In Augmented Reality, The digital data is overlaid on the physical world capturing by camera to create the illusion that both aspects are in the same reality. This usually appears on smartphones or cars. For example, map instructions or information can be projected on the dashboard of a vehicle to guide the driver. Popular games such as Ingress or Pokemon GO also uses Augmented Reality.

The idea of Mixed Reality have been discussed for years, but was not realized until 2016. Google [2] and Microsoft [3] are the only companies which released their devices for Mixed Reality development in early 2016. This is the reason that we are going to use Microsoft Hololens and Google Tango Tablet Development Kit in the project. And there are no other MR device products on the market beside the 2 mentioned. MR is rather similar to Augmented Reality. Both digital and physical data are mixed together to create an illusion. However, digital and physical objects would have physical interactions in real time to even enhance the illusion of having both aspects in a same reality. As an example, if you throw a digital ball to a physical wall, the digital ball should be able collide with the wall and bounce back.
1.2 Existing application and inspiration

By now, the Hololens released is developer version for exploration and thus there are no commercial Hololens application on the market. However, there is a demonstration developed by Microsoft and its partner Case Western Reserve University called “Holo Anatomy” gave us some inspiration.

Presenting human body detailed internal structure with high resolution graphics in front of people, the application and Hololens allow us to observe such structure while walking, looking up and down, as if we observe a real human model in reality, not limited to using the mouse and keyboard to watch the 3D model on screen.
It showed a possibility to transform science and medical education in the future. Not limited to education, Hololens may bring evolution to all industries involving designs as well. Instead of 3D games, we think there is greater potential of practical usage, such as interior design, architecture, building maintenance, and machine designs. Besides entertainment, we think Hololens can contribute more to human society. It inspired us to make a final decision on what the application we should develop.
1.3 Objective & Scope & Deliverable

The project is not aiming at providing an APP or WEB solution, it is for research purpose. Therefore the objectives are exploring Mixed Reality Applications as the cutting edge technology and the potential of mixed reality and lastly the interactivity of this technology.

The first goal of project is to develop an interactive multiplayer mixed reality game which can be installed on the 2 different devices (Google Tango and Microsoft HoloLens) and communicate with each other.

I was responsible for Hololens in the 1st semester.

My team member Jerry was responsible for Tango Tablet in the 1st semester.

On January 2017, we decided to switch from a game to a practical application - interior design, which is our final goal.

![This is the capture photo of our completed work](image)

User can select different furnitures from menu, and drag the furniture to real world, put it anywhere he looking at, and resize and rotate the furniture at horizontal dimension. Another user can see what the other one selected and put into the real world and he can manipulate with that object too. Under further development, refinement and model collecting, the application can
possibly become a commercial product in the future. We consider such project as a test point. If the project succeeded, we may discover more idea from demands of Architecture or Building Surveying and develop corresponding applications. When CS department possesses the described cutting-edge practical and commercial applications, we can possibly promote the name of “HKU CS” all over the world.

The idea of the project has changed several times during semester. Details of will be described in Section 3 : Application Design.

To achieve the final goal, there are some sub-goals:

To distribute workloads to group members.

In the project, the workload of team members are distributed as follows:

I am responsible for the rotation, scaling and moving system of each 3D object by tap dragging, as well as the Deletion function on each object after being dragged out of the preview menu. Jerry is responsible for the menu system and room scanned model generation.

To finish the individual module according to distributed work.

To merge the modules and find out any incompatible problems, and test the results.

To add multiplayer function, such that user can see any 3D furnitures existing in the world and manipulate with them.

To refine the application.
Section 2 : Technology used

2.1 Microsoft Hololens

It is a head-mounted display unit connected to an adjustable, cushioned inner headband, which can tilt HoloLens up and down, as well as forward and backward. There are some sensors in the front and related hardware, including the cameras and processors. The visor is tinted; enclosed in the visor piece is a pair of transparent combiner lenses, in which the projected images are displayed in the lower half. Before using, the HoloLens must be calibrated to the interpupillary distance (IPD) or accustomed vision of the user.

HoloLens uses the position and orientation of user's head, not eyes, to determine their gaze vector. User can think of this vector as a laser pointer straight ahead from directly between the user's eyes. As the user looks around the room, any applications can intersect this ray, both with its own holograms and with the spatial mapping mesh to determine what virtual or real-world object user may be looking at.
The core interactions on HoloLens are press, release, and bloom. The primary way in which user will select or activate holograms in user’s world is the air-tap gesture, which consists of a press followed by a release.
To air-tap, user makes a fist in front of himself, with the back of his hand facing him. User’s elbow should be bent at his side in a comfortable position. Next, user raises the index finger to the sky and then tap, by flexing his index finger down (the press) and then back up (the release).

Bloom is a gesture that is used to go back to the Start Menu from within a holographic application. It is similar to pressing the Windows key on a keyboard. To do the bloom gesture, user holds out his hand, palm up, with his fingertips together. Then open his hand.
2.2 Unity3D Editor 5.5

It is a free cross-platform 3D engine and development environment developed by Unity Technologies and used to develop video games for PC, consoles, mobile devices and websites. It is in fact the only officially supported application IDE for the 2 mixed reality devices.

Unreal Engine and Unity are currently the only two open source game development software supporting mixed reality. However, Unity has a larger user base, better documentations and cheaper packages. This means searching for resources and problem solving on Unity would be a lot easier comparing to Unreal Engine. Therefore the team chooses Unity as the development engine.

2.3 Hololens SDK

There is no separate SDK for HoloLens. Development tools are also free, and consist of

1. Visual Studio 2015 (integrated development environment developed by Microsoft) Update 3 with the Windows 10 SDK.

2. HoloLens Emulator, which allows users to run apps on Windows Holographic in a virtual machine without a HoloLens device. This is an optional special tool provided by Microsoft, other than ordinary API and SDKs, is the “HoloLens emulator”. We only need the emulator, when we want to carry out the development process without the Hololens device. Although CS department bought 2 sets Hololens, they may be needed by other students anytime for their needs, and therefore we cannot always keep the devices ourselves. Even we have the device, we do not need to carry the devices wherever and whenever we want to develop and test, if we installed the emulator. The emulator is good for a convenient development.
2.4 HoloToolkit-Unity

This is the repository that will contain all Unity specific components.

https://github.com/Microsoft/HoloToolkit-Unity

The HoloToolkit is a collection of scripts and components intended to accelerate development of holographic applications targeting Windows Holographic. The toolkit have been being improved by time, since it is not completed nor matured. When Unity was updated from Version 4 to 5, substantial changes was made, particularly on Input module. The old projects or test cases no longer function on the updated toolkit and Unity 5. It is now required Unity Editor Version 5.5.2f1

The followings are the main modules of the package.

**Input**

The module was mainly used throughout my development, since my work mainly related to user hands actions and 3D objects responses. The module allows developers to handle various types of input and send them to any game object being currently gazed at, or a state that not gazing at particular objects. It also includes a cursor similar to the HoloLens shell cursor that fully leverages the Unity's animation system. Each input source (hands, gestures, others) implements a **InputSource** interface. The interface defines various events that the input sources can trigger. The input sources register themselves with the **InputManager**, whose role it is to forward input to the appropriate game objects. Input sources can be dynamically enabled / disabled as necessary, and new input sources can be created to support different input devices. One or many input interfaces could be implemented, such as:

**IFocusable** : for focus enter and exit. The focus can be triggered by the user's gaze or any other gaze source.

**IHoldHandle** : for the Windows hold gesture.

**InputHandler** : for source up and down. The source can be a hand that tapped, a clicker that was pressed, etc.

**InputClickHandler** : for source clicked. The source can be a hand that tapped, a clicker that was
pressed, etc.

**IManipulationHandler**: for the Windows manipulation gesture.

**INavigationnHandler**: for the Windows navigation gesture.

To make a special note that there is a *HoloLensCamera.prefab* under the module of Input module. The prefab was created with the substantial update of Holotoolkit, simulates user head movement when wearing Hololens on head, user input such as tap and user body movement such as walking around an environment. That means developer can test input and Hololens camera movement even in Unity editor by pressing Run button. This greatly enhance the convenience of debugging or testing during application development, since I can reduce the number of times of exporting applications to Hololens Emulator.

**Sharing**

Sharing library allows applications to span multiple devices, and enables holographic collaboration. The tool was originally developed for OnSight, a collaboration between SOTA (a Microsoft studio) and NASA to enhance their existing Mars rover planning tool with HoloLens, HoloToolkit. Sharing enables users to use multiple devices for a task by allowing the apps running on each device communicate and stay in sync seamlessly in real time. The module was not used.

**Spatial Mapping**

The module contains scripts that leverage SpatialMapping related features. The most useful component among which is *SpatialMapping.prefab*. This is the base prefab which allows developers to visualize and access spatial mapping data on the HoloLens. It can also save or load room models that were captured from the Windows Device Portal.

**Spatial Understanding**
The module contains scripts and prefabs that leverage Spatial Understanding related features. However, it is not completed, and to be fixed. There are insufficient explanations or discussion of the usage of the module. The module was not used.

**Spatial Sound**

The module relates to the audio features. It was not explored.

**Utilities**

The module was not explored.
Section 3 : Application Design

In the final application, all users would be able to see virtual objects created on both devices but they can not perform the features provided on the other type of device. Among the 3 main objective, interactivity is the most difficult and important component.

Note that the design of the project has been modifying with the progress of study of the devices, documentations of the devices and the development. Since we can discover better idea by learning various kinds of limitation and utilize the concept of Mixed Reality under the limitations. And since the change of design is part of the process of our project, we will describe the designs in this section and describe reasons and the changes in Section 5 : Current Progress.

3.1 Block Building system - Hololens & Tango Tablet

The building system allowed users to build different virtual objects on top of the real world. Users could choose to add simple shapes such as cubes or spheres or some realistic 3D models like buildings or bridges. It was similar to playing lego or minecraft but it involves a lot of mixed reality features. After selecting an object from user menu, the user could tap on a flat surface through the device (Tango) or perform an air-tap (Hololens) to indicate where he wants to place the object. Once the object was added on top of the surface, the scale of the object can be modified by the user. Other than using pre installed models, users could also upload their own 3D models to the application. Users could achieve this by using the various scanners on the tablet to scan a real object. The user had to circle around the real object holding the device to conduct the scan. Once all the surfaces of the object were covered, a 3D model could be generated by the device. Users with Hololens & Tango Tablet can build and manipulate the blocks simultaneously.
3.2 3D drawing of Art gallery - Hololens only

What was influenced in this idea is Hololens. Works of Tango of Jerry remains on building system.

Users could do the air-tap gesture, as if holding the left mouse button to draw anything in MS windows paint or photoshop, and drag the fingers in real world to draw 3D objects into the world. Those 3D objects could be cubes, spheres or cylinders. Based on these simple base 3D models, we can still achieve variation by applying different colours and shapes. We would also allow 2 different types of objects to be draw, either behaving as Rigidbody that bind to real world physics or non-Rigidbody that floating in the mid-air. We would also allow users to switch to shooting mode upon a selection on an UI panel. Shooting mode allowed users to shoot a 3D object to the drawn objects set by an air-tap, as if a mouse click. Tango users could see what users with Hololens drawn.

3.3 Interactive Table tennis - Hololens & Tango Tablet

The Tango device would be replaced by Oculus Rift, a VR headset. Hololens remained in the idea. Both users would use real rackets and real table, while the ball was a 3D object, which was going to be shared across Hololens and computer. Note the Oculus is only responsible for conveying the graphics from games running on computer to user, it is not a computer as Hololens. The game in this idea was a cross-platform application between Hololens and computer. Moreover, a camera would be paired with Oculus such that to let user see the real world, since the game was supposed to run accompanied by reality. The moving rackets should be mapped by Hololens into 3D word to interact with the 3D ball, such that users wearing Hololens and the Oculus headset could play table tennis. The game was very dynamic and interactive.
3.4 Interior Design - Hololens & Hololens

This is the final design with only Hololens involved. Jerry will be switching from Tango to Hololens. Series of furniture 3D models will be used in the design. Users can drag them from user menu into a room environment, such as sticking to wall or lying on a table, for room design purpose. The reality surface will be scanned into computer, such that 3D objects can only be placed on an available surface, for example, a 3D computer object cannot be placed on a real small chair if the area of the seat was too small. Additionally, users can drag to rotate and resize the 3D object by user menu button.

Other than the main interior design feature, there is also a small interactive feature where a robot would exist in mixed reality to guide and interact with the user.
Section 4 : Project Methodology

4.1 Configure the hardware and Installed the tool

All the development tools required for Hololens can be downloaded from:

We chose Microsoft Hololens SDK for our development, since it is the only SDKs available at the time being. We can only trigger the devices to run the software by using the SDK. Furthermore, it is an official release to support Unity3D Editor, the most compatible SDK for our development.

To build simple single player demos for each device. The test demos should utilize the functions provided by the device, to test any practical executional limitation. Then, we can focus on the functions that are only needed for our project and develop demos to test and learn other limitations.

4.2 Feasibility Research and Study

Before development, we should learn

1. The limitation and capability of the device, such as virtual object positioning, depth camera distances, such that we can know how far the device can detect the real world environment, how far we can place the virtual objects in the 3D world from our device and how great the visibility of the virtual objects.

2. The structure and components of the SDKs and function of each components.

3. How the sample demo within the SDKs function, and what do they look like.

Next, we should

1. decide and design the scope and scale of our project, and improve the idea of which.

2. know how to position the virtual objects to let user have the best experience

3. modify the correct source code files and define our own methods without breaking the Hololens system by knowing how the components cooperate to realise the corresponding functions,

We need to study the details of tutorials and documentation provided by Microsoft from:
4.3 Individual development
According to the original plan, after learning about the devices, the team would start develop a single player application for each device individually. Two members will work on Microsoft HoloLens and the remaining two will work on Tango Tablet Development Kit.

Since we have changed to the another objective. We will develop and test in Unity3D separately as distributed workload. I will as much as I can break my rotation-move-scaling system into module or scripts. This is easier to test, debug and maintain, and even merge with team member's work. Then the merging could begin.

4.4 Collaborate development
At this stage, I will merge my system into Jerry’s menu system. We could learn any incompatibility when we are dragging furnitures out of menu; when we are viewing furnitures on preview menu; and when we are re-expand menu to drag another furniture out of menu.

4.5 Multiplayer
After completing single player applications for individual devices, the team will start transferring the applications to a multiplayer platform. First, a multiplayer player server would be built using the Photon Networking SDK. The team chooses to set up a fully authoritative backend server rather than using the cloud servers provided by Exit Games. This is to prevent any potential problems in the connection to occur since cloud servers require the use of the internet but a local server can be host in a Local Area Network(LAN). Then, the team will merge the applications from Tango Tablet Development Kit and HoloLens into one so two different types of devices can run the same application and communicate with each other.

4.6 Fine tuning of parameter
After multiplayer prototype being accomplished, we will start to refine the parameters, such as rotation, resizing and moving sensitivity, and user interface.
Section 5 : Progress

In this chapter, our team will conclude milestones we achieved in previous phases.

5.1 Feasibility Research on Microsoft Hololens

The table is the results of feasibility research of Hololens, related to Hololens spatial mapping and hologram range. By learning the numbers, we can make an appropriate game design about how far from the Hololens the objects can be placed, moved, and created.

<table>
<thead>
<tr>
<th>Range Name</th>
<th>Minimum Distance</th>
<th>Maximum Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Depth Camera</td>
<td>0.85 m (2.8 ft)</td>
<td>3.1 m (10 ft)</td>
</tr>
<tr>
<td>Hologram Placement</td>
<td>0.1 m (4 inches)</td>
<td>infinity</td>
</tr>
<tr>
<td>Optimal Zone</td>
<td>1.25 m (4 ft)</td>
<td>5 m (16 ft)</td>
</tr>
<tr>
<td>Comfort Zone</td>
<td>1.0 m (3 ft)</td>
<td>infinity</td>
</tr>
</tbody>
</table>

The table[4] is going to be explained in the following. Note that *Holograms* is the terminology refers to virtual objects in Hololens constructed virtual world.

1st Row. Depth spatial mapping cameras is from 0.85 meters to 3.1 meters. This means a plane put in front of the Hololens less than 0.85 meters would not be remembered by the devices, but only blocking the camera sight. And in an open area that the closest surface to the Hololens farther than 3.1 meters would neither be remembered. The virtual world where holograms to be put into is constructed by depth camera capture [4]. This feature will not influence our project goal, since open area spatial mapping is not required for our the project.

2nd Row. When users use finger pinch gesture to place any holograms, the closest distance between which and the Hololens is 0.1 meters. Holograms can also be put to infinite distance from the Hololens. The infinity means holograms will be shrunk into very small size, therefore users may feel that holograms are being moved to infinitely away from them [4]. Note that though...
theoretically holograms can be put 0.1 meters from Hololens, users can not perceive it. Because holograms will appear in users’ eyes only 0.85 meters away from Hololens. No matter users move too close to or too far away from the holograms, they will disappear. This is due to the concern of eye comfort, that Microsoft want to prevent the focus of each eye from being too great. Thus, the closest distance of the virtual object to the display lens should be 0.85m [5]. This feature reminded us when users create holograms in our game, the objects should not be generated less than 0.85m from the Hololens.

3rd Row. Optimal Zone is referring to optimal hologram placement distance range. It indicates holograms placing between 1.25 meters and 5 meters from the Hololens look better to players [4]. Because the objects users see in the 2 display lenses overlap relatively better in this range of distance. If objects were placed beyond this range, situation that objects being seen in one eye while not in another will occur [5]. This feature reminds us that holograms should not be generated less than 1.25m from the Hololens.

4th Row. Comfort Zone means our eyes and hands will be comfortable when holograms placed within such distance range from the device. Microsoft suggest that application designers should structure content to encourage users to interact with holograms at least 1 meter away from Hololens. The “interact” means users to perform the tap gesture to select holograms. The 1 meter minimum distance here is to prevent users’ arms from passing through holograms, which may ruin user experience. This may relate to object starting to disappear in less than 0.85m distance. Since we have concluded that holograms should not be generated less than 1.25m from the Hololens, the comfort zone will not influence our project.
Besides the above table, there are other limitations for Hololens.

Hololens is a standalone, mobile computer, the available storage and memory are insufficient for large scale application. The hard limit of the memory allocation to each single Hololens app and games are 900MB.

Graphical processing power seems not strong enough, jitter happens when high poly graphics are used. Fortunately sophisticated computer graphic and complex game system are not our objective this time. The project will remain simple.

Battery life is short. Less than 3 hours, the battery falls from 100 to less than 10 during the testing of applications. It caused some inconveniences to development, especially the device is kept in locker of CS laboratory and shared by different students. When the device is not being used, it will be put in the locker, not possible to be charged. When it is being used, it is frequently being put on and down from head. It is difficult to charge in the case either.
5.2 History of application design

Note that the change of design is concurrent with the study of the devices, documentations and tutorial, and with the development.

The first idea was 3.1 Block Building system. However, after discussion, the gameplay might not sufficiently utilize the concept of MR and might be slightly dull. Users may feel bored soon, therefore the design was extended to our second idea 3.2 Art gallery.

The idea consisted of “3D Drawing” on Hololens and “Photo Gallery” on Tango. Note that the “Photo Gallery” was assigned to another student in another group. The Tango task on our group remained as building system. In the idea of 3D Drawing, We expected a different gameplay experience than only dropping objects to build, and user could make full use of their creativity by drawing their own imagination.

In early December, we thought that Art gallery was not fun enough to attract users, since the pace is slow and the interactivity between users was still low. Then we came up with the third idea, 3.3 Interactive Sport game: Table tennis. The description of the idea is on the above section. This idea was fully dynamic and thus full interactivity between users was required and less boring. However, there were several difficulties.

1. we could not pair the the camera with Oculus VR headset to transform Oculus into transparent glass as Hololens.
2. Hololens at this stage could not detect fast moving small real world objects into 3D world. The spatial mapping power was limited.
3. 3D objects 0.85 meters or closer from the Hololens would not display on the lens, which means users could not see the 3D objects even if they exist in the 3D world. This is the hardware limit of Hololens, stated in the above “Feasibility Research” section.

Thus the idea was changed again.

The final design is 3.4 Interior Design. The path changed from interactivity and dynamism to practicality. We have been focusing on making a game and providing fun, however, after discussing with supervisor, the better idea to respond to “Innovation of HKU Department of CS” is a practical application. Although the design appear to be boring since users only drag and then stand to observe the 3D objects existing in reality. The design is more useful and constructive to society and business in the future, and it is also extendable. It could be applied further to Architecture upon further development. Thus the idea will have greater potential.
5.3 Individual work

5.3.1 Old development

The first demo was tap to instantiate objects in the environment with gravity to free fall upon instantiation. The objects were held by reality surfaces, such as sofa and table. One important note here was that the objects looked not as 3D and seemed dull. In the final product, we have to use more sophisticated model and beautiful texture to convey the 3D graphical effect. One more problem was found that when the total number of 3D objects such as spheres or cylinders is over 300, the frame rate of Hololens drop drastically, serious lagging occurs. This is the hardware limit.

Some small features such as gazing at an object changing its colour or tapping to move objects were also tested on the demo. Then I thought if physics effect and dynamics could be added into the game, more fun could be provided to user, and at the same time, 3D drawing was being developed by another student, I changed my direction on trying to do a shooting demo.

Problems arose during development. Objects after instantiation still dropped as free falling, not performed a projectile as expected. Some answers from official forum gave hints. The codes were modified and placed in another new script file; movement of objects were triggered by accessing the velocity parameters directly rather than applying Unity embedded physics methods.

After weeks of trying by the air-tap gesture, projectile motion of an 3D object was performed and bouncing can be applied to the object. What users will see is that they can tap to shoot a...
projectile cube to hit other existing 3D objects in the scene, the objects being hit will bounce as well. If the shooting cube hit spatial mapping surface, which means the reality surface remembered by the computer, the cube will also bounce. Thus users will have experience that the 3D cube has interaction with the real wall or table or something else.

The link to the video demo is
https://drive.google.com/file/d/0B49TxTEydDtRNFJXWmp5VkOZEk/view?usp=sharing

Unfortunately, the development progress was not satisfied in first semester. Since the tutorials and documentation on Hololens API are not sufficient. The development requires a lot of time from me to study the codes. The toolkit itself is neither completed yet. Unlike traditional SDK tools, which were usually completed when released to public and no breaking changes need to be made in a short periods; however, this one is open to public before ready and not entirely developed by Microsoft staff. Microsoft intends to collect feedbacks and opinions from interested developers all over the world, and then improve the toolkit. Therefore the toolkit is being modified and improved constantly. I do not think the content is sufficient for each module so far. Some scripts and other components were removed and some with bugs are left in the toolkit. Therefore the progress was slow. In December of 2016, a breaking change on Input module was made with a update of the toolkit, some mechanisms become substantially different. I have to study the new Input module, my works made in the first semester was no longer functionable. There is shortage in explanation on the changes and how to smoothly shift old projects to new ones.

However, there are more advantages in the new toolkit. If developers began from the new one, the progress might be better. In earlier builds of the toolkit, there were three main scripts that managed the basis of interactivity within a scene: Gaze Manager, the Gesture Manager, and several other components that would support voice input. To set up a basic gaze and gesture interaction, a script implement an “OnSelect()” function needs to be added to object, which would receive messages sent from the gaze and gesture managers. It was a challenge for me to understand the scripts thoroughly on how to implement more interaction by enabling different modes with a mode manager, due to being unskillful in C#.

The new version in fact changed the input module to be more extensible and easy to work with. HoloLens apps built in Unity now use a global input manager, and as introduced in section 2, different interfaces corresponding to different inputs are provided. If we want to implement any input action on objects, we now can attach scripts extending those interfaces to each object, without handling of sending message between scripts and event listeners.
Another advantage is Hololens camera simulation. Now we can use the console output to debug and test any error. The old toolkit did not provide such function, the application would not respond until writing a camera simulation system myself. I have to spend time export the application to Emulator through Visual Studio. Furthermore, I could not use console to debug if any problems emerged.
5.3.2 Rotation and Scaling System

The planned system is to create a transparent box based on the centre of furniture, technically the 3D model of the object, surrounding the furniture. And then 8 scale handlers will be stucked at the 8 vertices of the box. Users can tap on each of the handlers to resize the furniture. In the same way, 4 rotation handlers will be stucked at the middle of 4 vertical edges of the box. Users can tap on each of the handlers to rotate the furniture horizontally, which means rotation around the axis that going through the centre and the top and the bottom. The system will set to be child component of each furniture in Unity. The system will appear on any furniture being gazed and tapped by user, and disappear if being tapped again.

The first step is to locate the centre of an 3D model in Unity.

The second step is to create a bounding box based on above and find 8 vertices of the bound, only then the 8 scale handlers can be made and stucked to the vertices. The handlers are in fact simple spheres. The rotation handlers were not considered here since if I can succeed in the second step, they can be easily created by the same principle.

There is a class of Unity library to handle this issue.

https://docs.unity3d.com/ScriptReference/Bounds.html

By using this class, I could locate the centre of irregular objects of different models. But the next challenge is to create a simple transparent cube slightly bigger than the objects and surrounding the objects without distortion, no matter the objects were moved or rotated or resized. Therefore I started with standard cubes, spheres, and cylinders of Unity, but later I had to test on different models.

Problems arose upon different mesh renderers in Unity and models even with the same renderer; Problems also arose when handling position of the bounding box. Since there are 2 scales of position in Unity, Local and World, Mesh renderer and Mesh filter also influence the setting of position.

The bounding box became too big if the object was enlarged, the growth of size of bounding box and the 8 spheres were not proportional.

Mesh renderer is set to be in World space, while Mesh filter is set to be in Local space. At the beginning, the bounding box and the 8 spheres and the object appeared in different locations, sometimes 2 of which stayed together.

The bounding box size relative to the objects distorted, even with the same Mesh renderer though different 3D models I downloaded.
The 8 spheres were swallowed by the bounding box, but not at the vertices of the box.

The followings are the early development captures.
Finally, I used bounds property of Mesh renderer added to furniture for locating the position of the bounding box and all the handlers, and bounds property of Mesh of Mesh filter added to furniture for proper scaling of the bounding box.
When the box and the handlers can be created stably, development of scaling and rotation functions started. Followed the design suggestion of Holo Application of Microsoft. I intended to use the `IManipulationHandler` interface to handle rotation and scaling. It seemed simple at first since only a few codes were required to achieve the 2 functions. But in fact this interface could not achieve the the purpose that I expected. My desirable design is that user firstly tap on the handler and then drag to rotate or resize a furniture, and the degree of rotation or scaling is proportional to the degree of the movement of user’s hand. If the user hold the gesture, the rotation or scaling should also remain still. But the `IManipulationHandler` starts and keeps rotating or scaling as long as user hold the tap gesture on the handler and only if user keeps gazing at the handler. Similarly it stops rotating or scaling as long as user gaze away or remove their hand gesture.

I had to give up this approach. And then I made use of another script called `HandDraggable` from the toolkit input module. The script demonstrates how to use hand dragging to move objects. I studied the file and made modification on it to create 3 files corresponding to `rotation` and `scaling`. By comparing the coordinates between previous frame and current frame of a factor related to hand movement, I can realise my desirable design mentioned in the last paragraph.
To note that rotation is carried out by comparing the coordinates of a Vector3 variable storing info of hand movement direction between previous frame and current frame. The variable is measured always relative to the camera, which is the Hololens camera, in another word, user’ head.
To note that I used to apply the same way as I handling rotation, there are distortion however. Now the scaling is carried out by comparing the distance of hand in terms of all the 3 axises from the position of the furniture between previous frame and current frame. Thanks to Prof. Loretta giving a suggestion.
However, in later stage, the bounding box still distorted on different 3D models. And it seemed that the bounding box is no longer needed to indicate my system being activated, as long as the handlers are correct. Thus I decided not to waste time on fixing the box, and disable it.
I think there should be a mechanism that user can remove furniture from world.

I added a delete function that a red delete box is added to the handler set.

Only when the furniture is dragged out of the preview menu, put on the world, the delete box will appear. Then user can tap on it to remove the furniture.

Even in preview menu, users can also tap on the furniture to call out the handler set, but this time, the red delete box WILL NOT appear. It would be weird to delete the furniture in previewing.
5.3 Work by Jerry

This is the menu system and the scanned room model generation function by Jerry. Details is left to be discussed in Jerry’ report.
Please note that we designed that even in preview menu, users can also tap on the furniture to call out the handler set. As I mentioned above, the red delete box WILL NOT appear if furniture was in previewing.
5.4 Collaborative Work

At the end, since we could not have 2 Hololens for development, the multiplayer feature development was cancelled. However, we still have to add a feature that when user move the furniture, the furniture should not penetrate the floor or walls. To note that the floor and walls will be scanned by Hololens and Unity API automatically put a mesh collider on the spatial mapping. By adding response scripts when our 3D furnitures interacting with the collider, we can create an illusion that the 3D furniture can interact with the realistic environment.

Upon discussion with Jerry, we started with a way that when furniture intersect with the spatial mesh collider, a boolean state indicating furniture entered spatial collider or not is triggered, and then the dragging is immediately disabled, and put the furniture back to the position of the previous frame, and then immediately re-enable the moving function so that user can drag to move it again. However, we found a problem that if there are a lot of realistic furnitures or people in a room, they will also be scanned and recognised as environment. Eventually, user can hardly drag to move the furniture smoothly, since the 3D furniture will be too easy to intersect with the spatial mesh collider.

Eventually Jerry discovered a solution that rather than stopping user from dragging the furniture when it intersects with spatial colliders, it remember the previous furniture dragging position under the state of not entered spatial colliders. Such that when the furniture intersects with spatial colliders, it keeps returning to that remembered position mentioned above, while dragging is still functioning. When users drag the furniture away from spatial colliders, the furniture move freely as before. But instead of directly changing the position of furniture, a Unity Physics method Vector3.MoveTowards() was used, This solution is using less codes and simpler to manage and provide better user experience.
## Section 6: Project Schedule

<table>
<thead>
<tr>
<th>Scheduled Time of Completion</th>
<th>Task</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>October</td>
<td>Researched on Tango Tablet Development Kit feasibility</td>
<td>Completed</td>
</tr>
<tr>
<td></td>
<td>Researched on Hololens feasibility</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Determined the project goal</td>
<td></td>
</tr>
<tr>
<td>November - December</td>
<td>Tango Tablet Development Kit single player demo</td>
<td>Completed</td>
</tr>
<tr>
<td></td>
<td>Hololens single player demo</td>
<td>Completed</td>
</tr>
<tr>
<td>December - January Mid</td>
<td>Adapting Photon Unity3D Networking SDK</td>
<td>Terminated</td>
</tr>
<tr>
<td></td>
<td>Tango Tablet Development Kit multiplayer development</td>
<td>Was developing and Terminated</td>
</tr>
<tr>
<td></td>
<td>Hololens multiplayer development</td>
<td>Was developing and halted</td>
</tr>
<tr>
<td>January Mid - February</td>
<td>Hololens UI</td>
<td>Completed</td>
</tr>
<tr>
<td></td>
<td>resize, rotate and move objects</td>
<td></td>
</tr>
<tr>
<td>February - April</td>
<td>Hololens UI</td>
<td>Completed</td>
</tr>
<tr>
<td></td>
<td>resize, rotate and move objects</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AI Non player character</td>
<td>Cancelled</td>
</tr>
<tr>
<td></td>
<td>Multiplayer development</td>
<td></td>
</tr>
<tr>
<td>April</td>
<td>Project refinement, Optimization</td>
<td>Completed</td>
</tr>
</tbody>
</table>
Section 7 : Conclusion

The goal of the project is to develop a practical application of interior design by allowing users putting 3D furnitures into the world, and user can rotate, resize, move and remove furnitures from the world. Thus users can assess whether the design of the furniture is satisfied. The application is still simple and far from commercial product. But the project purpose is researching, and we consider which as a starting point. In the future :

The multiplayer feature could be added to allow multiple people discuss the design work together.

Distance/area/volume measurement in reality scale on 3D furniture (such as distance between foot and the head of a furniture, or just any 2 points user selected, and area of any plane of the furniture, and the volume) can be added to fulfill the practical design needs, the result also need to adapt to the user resizing.

3D furniture will automatically attach to suitable reality planes such as walls and floors when users drag to move it to near reality planes. For example, a sofa should be and only able to attach to floor when the destination is a smooth flat surface and no other obstacles. Similarly, a wall lamp or a poster should be and only able to attach to smooth flat wall surface.

Small AI objects walking through the room and 3D furnitures can be added to emphasize the illusion of Mixed Reality and thus provide better user experience.

We started with feasibility research. Because we need to know our project limitation and decide our scope. Although there are changes on the project design and difficulties, the process is as an iterative model of software engineering in commercial development out of school. Thus it is part of the valuable learning experience. During development process, I met problems, I learnt how to solve them by trying different approaches, breaking the each problem into smaller ones. Sometimes, if I solve the particular small problems, the other small problems were automatically solved or avoided. Most of time, I searched solution on web and studied the documentation of C# and Holotoolkit. There are too many possible solutions on web, and how to efficiently locate a correct solution I need is also a skill that I need to learn. The project offered me a precious opportunity to be a student pioneer to explore the cutting-edge technology. It opened my mind on how the future technology can transform human life. It also offered me a precious opportunity to apply the knowledge I learnt over the academic years, such experience is important to consolidate my knowledge. As a technology student, learning and applying is a life commitment, only then professionalism can be achieved.
Section 8 : Reference