Year 2022 Master Graduation Thesis

Through Case Study of Smart Lock to Explore Natural Interaction in MR based Smart Home Controller

Supervisor
Professor Nakajima Tatsuo

A Thesis Submitted to the Department of Computer Science and Communications Engineering,
the Graduate School of Fundamental Science and Engineering of Waseda University
in Partial Fulfillment of the Requirements for the Degree of Master of Engineering

Submission Date: January 23rd, 2023

A Thesis Presented to
Department of Computer Science and Engineering
WASEDA University

Research guidance: Research on Distributed System
5120FG30-8 Yiming Sun

Acknowledgement

First of all, I would like to express my deepest appreciation to my supervisor, Professor Nakajima Tatsuo, for his invaluable guidance, patience, inspiration throughout this project, and support my idea. His expertise and insights have been instrumental in shaping my research and helping me to bring this thesis to fruition.

In addition, I would like to thank Yiyi san and all other lab members in Nakajima Lab who have provided support, inspiration, I learned a lot of valuable knowledge related to research through their advice and works.

I would also like to acknowledge the support and encouragement of my family, who have always believed in me and given me the motivation to move forward.

Finally, I would like to express my appreciation to Waseda University, for providing the resources and facilities necessary for the completion of this thesis.

Thank you all for your support and for being a part of this journey.

Abstract

A smart home is a residence that utilizes internet-connected devices to provide enhanced convenience, comfort, energy efficiency, and security. Smart homes have been around for a long time; however, barriers to user adoption of smart homes still exist. There are several barriers, but we will focus on distrust and technological anxiety. To reduce these barriers, we found an interaction paradigm called natural interaction. It refers to how humans interact with technology in a way that mimics natural human behavior and communication. It provides more user-friendly interactions with technology that users are already familiar with, which could reduce the user's cognitive load.

In this paper, we think about how users interact with smart homes in the era of mixed reality popularity. We aim to reduce the barriers mentioned before and enhance the user adoption of smart homes by applying natural interaction to the smart home control application. We developed a smart lock controller on the HoloLens platform. By utilizing advanced features provided by mixed reality head-mounted display, we recreate the interaction of people opening locks in everyday life. We conducted a user study to compare this interaction with the push-button interaction in a mixed reality environment and explored the advantages and limitations of our system based on collected data and user feedback.

Table of content

Contents

Chapter 1. Introduction

In these years, the smart home has become a popular concept. The goal of the smart home is to provide a better quality of user's life in terms of energy efficiency, security, convenience, and entertainment [3]. However, smart homes are not well accepted in the market. Various reasons, such as distrust and technology anxiety, cause it. Natural interaction is an interaction framework that integrates human language and behavior into tech applications. It provides an intuitive, entertaining, and non-instructive interaction experience [12]. By studying the related work, we believe the natural interaction could reduce the user resistance to accepting new technology, which could alleviate the problem of low acceptance of smart homes and lower the barrier to smart home user adoption. Nowadays, the most common solution for smart home control is using mobile applications, such as Apple Home. In the 2D user interface, users interact with home appliances through touch icons and interact with the menu by using their fingers, it is a proven way to interact and has been applied widely in smartphones, personal computers, and other platforms with a 2D user interface. We consider the possible future scenario. The mixed reality (MR) era is as popular as cell phones today. Many interactions will change. Since the user interface is moved from 2D to 3D, and users can use hand tracking to interact with virtual objects directly, the interaction becomes tangible and joyful. We conducted a case study that applied nature interaction to the smart lock to explore the natural interaction experience in MR smart home control system. We reproduced the process of opening the lock in the physical world to the MR application and performed a user study to collect user feedback. Here is the structure of this paper. This paper first explains the background of the study, and after that, we conclude some related work, then state the research goal and introduce the research approach. Then we present our system design and implementation. Finally, we show the user study result, analyze the data, and make conclusions.

Chapter 2. Background

2.1 Mixed Reality

Mixed Reality (MR) is a term that was first introduced in 1994. It blends the physical and digital world, and the best well known of best well known of this is Augmented Reality (AR) 0. Unlike virtual reality (VR), which creates a virtual world that provides a complete immersion experience, the MR merges the virtual and physical worlds and blurs the boundary between them. Nowadays, the MR has gone far beyond original definitions to include Environmental understanding, Human understanding, spatial sound, location and positioning in physical and virtual spaces, and collaboration on 3D assets in mixed reality spaces [2]. The typical device that delivers the modern MR experience is HoloLens. It’s a holographic head-mounted display (HMD) device introduced by Microsoft. HoloLens is equipped with a see-through screen that allows users to see real-world and holographic digital content. Furthermore, users can interact with digital content by tapping their fingers. These features bring more probabilities for interaction design.

2.2 Smart Home

Smart homes are growing in popularity with the Internet of Things (IoT). The goal of the smart home is to provide a better quality of user’s life in terms of energy efficiency, security, convenience, and entertainment. Smart homes are comprised of networks and sensors and can be controlled by smartphones through Apps. Major international IT companies such as Google, Amazon, and Apple have entered the smart home field [13]. By releasing frameworks such as Google Home, Apple HomeKit, and Alexa, smart home devices are centralized and managed by platforms. Usually, smart homes can be visualized by mobile Apps such as Apple Home. Although the concept of a smart home has been brought to the market for several years, it hasn’t been entirely accepted by customers. It involves several reasons, and we will explain the detail in the literature review part.

2.3 WIMP Interaction

Nowadays, the mainstream terminal devices of users are personal computers (PC) and smartphones, and the graphical user interface (GUI) offers an easy way to allow the user to interact with these electronic devices. GUI uses graphic icons and performs through direct manipulation of the visual elements. GUI is usually windows, icons, menus, and pointers (WIMP) based, and pointers always stand for the mouse. However, since touch-based devices such as smartphones became popular, a different approach to GUI design has been applied. To adapt touch input and a smaller screen, post-WIMP is introduced [11]. It simplifies the interface compared to WIMP on the desktop platform, making screen estate more efficient. It could display rich content on a limited-sized screen and provide fast and precise selection operation by utilizing the feature of touch operation. In general, post-WIMP has better focus, is easier to use, and delivers superior user experiences [11].

2.4 Natural Interaction

Natural interaction provides an intuitive interaction. The natural interaction system design focuses on recognizing innate and instinctive human expressions in relation to some object [12]. Applying the natural interaction could reduce the user's cognitive load. In other words, it reproduces the interactions that users are already familiar with and applies them to the virtual object, so the user doesn't have to leave his comfort zone when using the product. Natural interaction can include using speech, gestures, and facial expressions to control and interact with technology, rather than traditional methods such as buttons and touchscreens. In general, natural interaction allows for more user-friendly interactions with technology.

Chapter 3. Related Work

Smart Home entered the market several years. However, barriers to user adoption still exist. Li et al. analyze the existing adoption barriers and point out the multiple obstacles, including distrust, resistance, and technology anxiety. They conclude that the smart device itself lacks users' trust. The device's reliability, performance, and controllability impacted the adoption intention. The older and less well-educated participants are the most distrustful of IoT and represent strong resistance. Also, the complicated service may have negative presses and feelings on the user's mind [3]. Regarding technology anxiety, Pal et al. state that some users, especially older adults, prefer to use the technology they are familiar with rather than accept new technology due to their reduced cognitive and physical capabilities [4].

Natural interaction has been applied widely in various fields and has proved that it can improve the user experience and make interaction joyful. Kyriakou et al. applied natural interaction to the museum AR system. Users can see virtual artifacts replicas in their system through the AR application. The AR application is deployed on a low-cost HMD like Google cardboard. Additionally, they used leap motion to track users' hands in real time. The experiment results show that the application was accepted positively among all age groups, and most users feel enjoyment [5]. Another application scenario is a video game. Leibe et al. conducted a study that applied natural interaction to the video game Elder Scrolls V. They use Kinect as a pose recognizer to perform tasks such as "Raise shield" and "Cast spell." By measuring the "fun value" of natural and native interaction through a questionnaire, they found that natural interaction provides a more enjoyable and intuitive user experience [6].

Smart home control is a popular topic and has been explored for many years. Such as, Isyanto et al. built voice-based smart home control for disabled people. The work uses Google Assistant to process voice commands and control smart devices [9]. Laehyun et al. built a remote controller for controlling universal home devices. Additionally, it provides several types of haptic feedback for different operations [10].

Although many studies related to smart home control, few existing works explored the smart home in an AR environment. Mahroo et al. created an AR smart home framework called HoloHome. The goal of this framework is to allow AR applications on the HoloLens platform to be able to interact with real objects in the physical world. They use Vuforia image processing to locate the home appliance and overlay the virtual version of the home appliance over the real one. Each time the user looks at the specific smart device, the virtual replica will be activated, and necessary information will appear [7]. The architecture of our project is similar to this study. However, rather than Vuforia image processing to detect the object, we choose to use spatial mapping to locate the real object. Another study conducted by Inomata et al. uses gestures as input commands to control smart devices. Users can use corresponding gestures to operate home appliances. If different home appliances have similar operations, the same gestures are applied. For example, the lights' brightness and the volume level are all adjusted by waving a hand to the left or right [8]. This design takes advantage of MR HMD and offers convenient operation. However, memorizing various gestures can increase the learning threshold, which we are trying to improve.

Chapter 4. Research Goal and Approach

4.1 Research Goal

This research aims to provide a way for users to interact with the smart home in the 3D user interface. Unlike how we use a 2D user interface, such as a button, to control the smart home, we want to reproduce the user's interaction habits from what they usually do in reality. We take advantage of MR HMD by using hand tracking and spatial mapping features, and we developed interaction for a 3D user interface. We want to explore the natural interaction for MR smart home controller, to validate the advantage and limitations of natural interaction for MR smart home controller compared to WIMP interaction. We also want to see if natural interaction could reduce the learning barrier of new tech, improving the user adoption of smart homes.

4.2 Research Approach

In order to validate our approach, we conduct qualitative research. NASA Task Load Index (NASA TLX) is a subjective workload assessment tool that allows users to evaluate the human-machine interface system from a multi-dimensional perspective. It includes mental demand, physical demand, temporal demand, performance, effort, and frustration. In order to achieve our research goal, which is to validate our interaction design, we chose this method to measure the user experience. We also applied other forms of survey. The purpose of the survey is to supplement the questions related to the study's objective, including participants' personal information, a ten-point rating scale for measuring their general preference for experimental and control groups, and a few optional questions to ask participants' opinions about two design groups.

Chapter 5. System Design

5.1 System Overview

The system's feature is to detect the target and place a virtual lock interface in front of the object. The system is divided into two parts, software, and hardware. The software is deployed onto HoloLens, and the hardware is a Bluetooth-controllable lock that Arduino builds. The HoloLens App performs object detection, Bluetooth communication, and object manipulation tasks.

Hand-tracking interaction is the only interaction approach we applied to the project. In order to achieve the goal of natural interaction, we must reproduce the user behavior of opening the lock in our project. Users usually use their hands to grab the key in the physical world. Thus, hand track-based manipulation is essential for our project. In the project, we optimized the hand-tracking approach to ensure the user could freely control the virtual key's position and rotation with less effort.

Communication between HoloLens and Arduino is also an essential part of this project. To open the lock, the user needs to insert the key as they did in the real world. The 3D lock model contains a box collider. After the collider detects the overlap, HoloLens will send the signal to the Arduino, and Arduino will listen to the movement through the Bluetooth module. The received signal will be the trigger to control the status of the physical lock.

5.2 Software Environment

5.2.1 Development Tools

We develop the program on Unity. Unity is a cross-platform game engine that supports all popular platforms. HoloLens 2 is built on UWP, which Unity also supports. We chose the Unity 2020.3.42

LTS version since it's the recommended stable version for Microsoft MRTK2 and Azure Object Anchor (We will explain both terms in the next section).

5.2.2 SDK

Microsoft Azure Object Anchors (MAOA) 0.25.0 version is applied. MAOA is a mixed reality service that can align 3D virtual objects with physical objects. It also performs object detection tasks by using the geometric approach. To achieve the task, the developer must upload a one-by-one sized 3D model with the same shape as the target object. This service will compare the 3D model with the geometry mesh to detect the object based on the percentage of coverage, which could bring a rich and immersive MR experience.

Mixed Reality Toolkit 2 (MRTK2) is an open-sourced toolkit we used in the project for object manipulation. MRTK2 is a powerful toolkit that offers a set of components and features for XR development. It supports XR devices, including HoloLens and VR devices like Oculus. Unity provides developers with various examples to help them familiarize themselves with XR development. MRTK also offers pre-configured prefabs and input configurations to simplify the workload of XR App development dramatically.

Arduino Unity Plugin is a plugin for Bluetooth communication between Unity App and Arduino. Its methods are easy to understand and support both BLE and Bluetooth Classic. Also, it supports multiple platforms, such as iOS, Android, UWP, and Windows.

5.3 Hardware Setups

HoloLens 2 is a holographic MR device that uses see-through displays that do not isolate the user from the physical environment. It contains spatial mapping features which could represent real-world surfaces. It allows us to use real-world positions to place an anchor in the world. It also provides a hand-tracking feature that will enable us to implement natural interaction.

Other than HoloLens 2, we use Arduino UNO Rev3 to control the lock. The Arduino UNO is a microcontroller board. It uses Arduino IDE as the development environment, so we can write C++ script and burn it into the board. The details of the lock system will be described in the System implementation part.

Object Detection

Object detection is the core part of this project because we want to place virtual objects in the fixed position corresponding to the target object. The App will search for the thing based on the geometry mesh if we set the target in the search area. The mesh is generated through the spatial camera that HoloLens equipped. In our case, the geometric search method is better than computer vision because computer vision cannot get an object's 3-dimensional coordinate in the real world. After all, it cannot recognize the depth of the thing. Compared to computer vision, the geometric search approach can sense the object's depth to place the virtual lock in the given position in the 3D space. To access the geometric search approach, we used Azure object anchor SDK. This SDK can return search results that obtain the object position in Unity coordinates. Unity can use position data directly, so the script will spawn a virtual lock to the object's location after we get the object's position. Before the detection starts, it is required to define a search area. The service only detects the things that are in the specified search area. MAOA has four options to define the search area: bounding box, field of view, location, and sphere. We initially used the bounding box method.

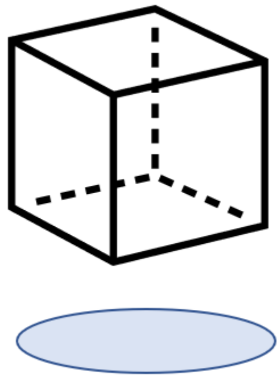


Figure 1 Bounding box search area

In this way, the user must drag a box to cover the target before processing the object detection. We found that the box will sometimes teleport to other places due to HoloLens incorrectly tracking the hand position. Also, it makes the whole workload complicated because users must manually set the search area. The field of view (FOV) approach could simplify the entire workload because, in this way, the search area will be the area of the user's view, and the user does not need to do additional work to specify the search area. However, we found a significant performance drop after we change the approach for setting up the search area. The most obvious cost is that after applying the FOV method, the object detection accuracy is much lower than the bounding box method. It takes longer to find the target, and the service returns sometimes do not match the object's position in the real world. So when we place the virtual object in the real world based on the MAOA result, there will be a significant deviation from our expected position. However, the cost is acceptable because this study aims to validate the user experience of the interaction. The accuracy of object detection is not the primary concern in this study. However, the accuracy issue can also affect the user experience since this object detection approach currently needs to be mature. We expect to solve this problem in the future.

Hand Tracking

Object manipulation is an essential feature of the project. We used MRTK2 manually. Initially, users can use their hands to manipulate the virtual key. Initially, we let users interact with virtual keys using the object manipulator. An object manipulator is an MRTK component that makes objects movable, scalable, and rotatable using both or a single hand.

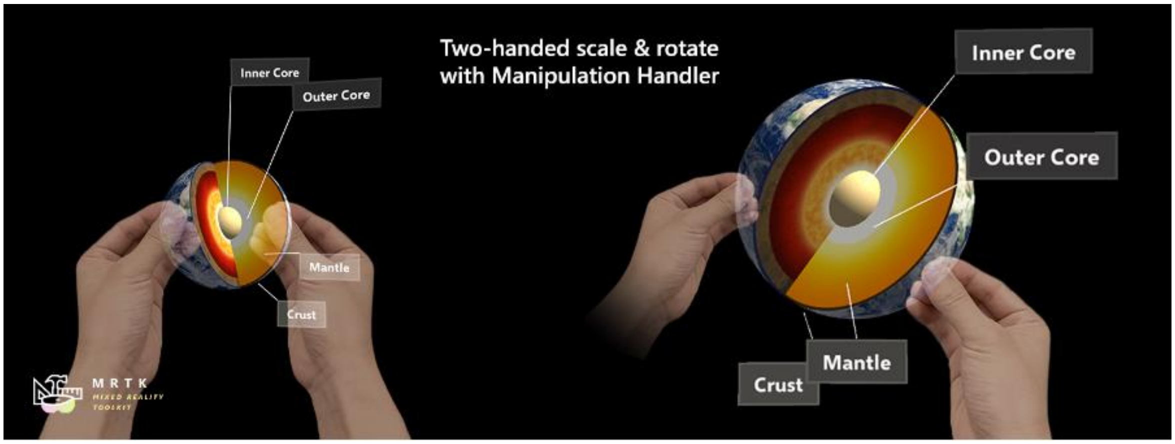


Figure 2 Object manipulator [14]

However, we found that in some instances, HoloLens may lose track of hands. Also, it's hard for the user to rotate the object at a slight angle. We hope to reproduce the manipulation experience in the physical world. Users are supposed to turn the virtual thing in any direction they like, so we must seek another approach. We found that the MRTK SDK provides a method to allow developers to access the data of each joint on both hands. In other words, we can get the user's hand joint position in real-time through MRTK.

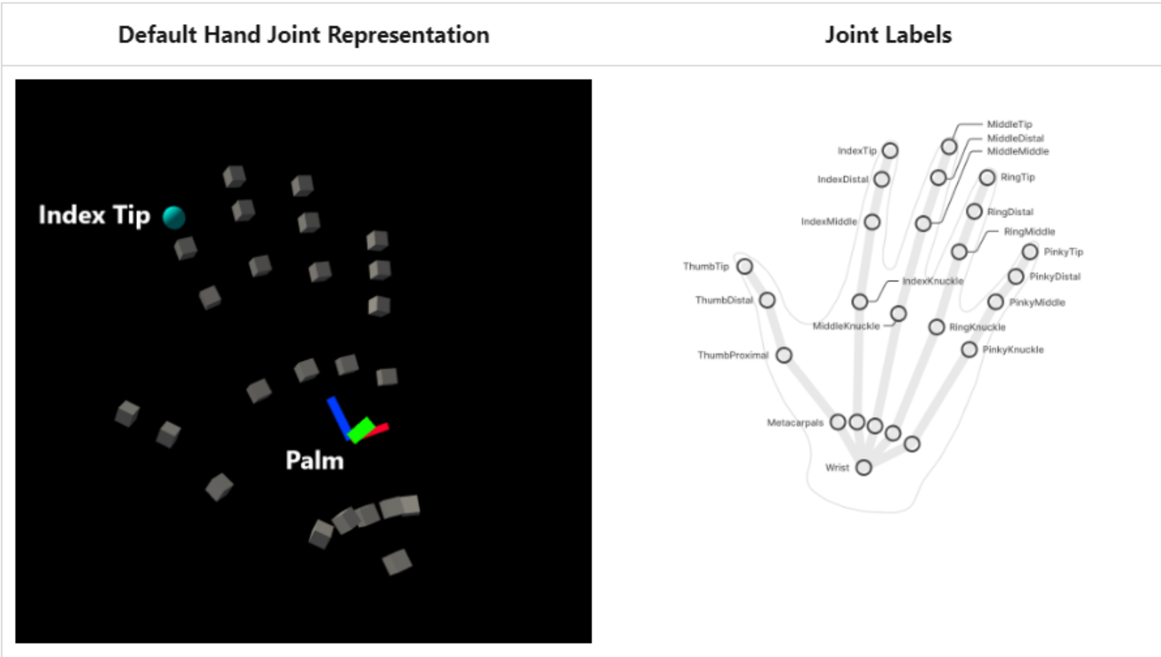


Figure 3 Joint index[15]

By utilizing this method, we tracked the position and rotation of the user's index tip and assigned these values to the virtual object. In this way, the virtual thing will follow the movement of the user's index tip. Although the user does not conduct the grabbing operation, the whole procedure becomes much smoother than the object manipulator.

Chapter 6. System Implementation

The whole system is divided into two parts: software and hardware. The software part is the MR program that creates the virtual space that places the virtual objects in the physical world and allows the user to manipulate the virtual objects. The hardware part is a lock system that was built oriented with Arduino, which two parts communicate through classical Bluetooth. This image shows the structure of the HoloLens program.

We have a static class called Data, which stores two Boolean variables that present the system's current status. The first variable is isCollide. This variable represents if the key is inserted into the lock or not. Another variable is isSent, which defines whether the program sends the message to the Arduino. These two variables will help the program determine whether it should send the signal to the Bluetooth device. The following picture presents the logical diagram for triggering send signal condition. We only want the signal to be sent if the key is inserted into the lock, which is the case when two colliders overlap, and the signal has yet to be sent. If the signal has already been sent, we will wait until HoloLens 2 receives the signal from Arduino to set isSent to false.

6.1 Software Part

Step 1: Setting Up MAOA

MAOA uses the spatial mapping feature on HoloLens 2. It calculates the percentages of coverage between the 3D asset of the target and the 3D mesh generated by spatial mapping. That is to say. We need to make a one-by-one scaled 3D model for the object that is being detected. After making the model, we need to upload the 3D asset to the Azure service, which will convert the 3D asset to the service-compatible format.

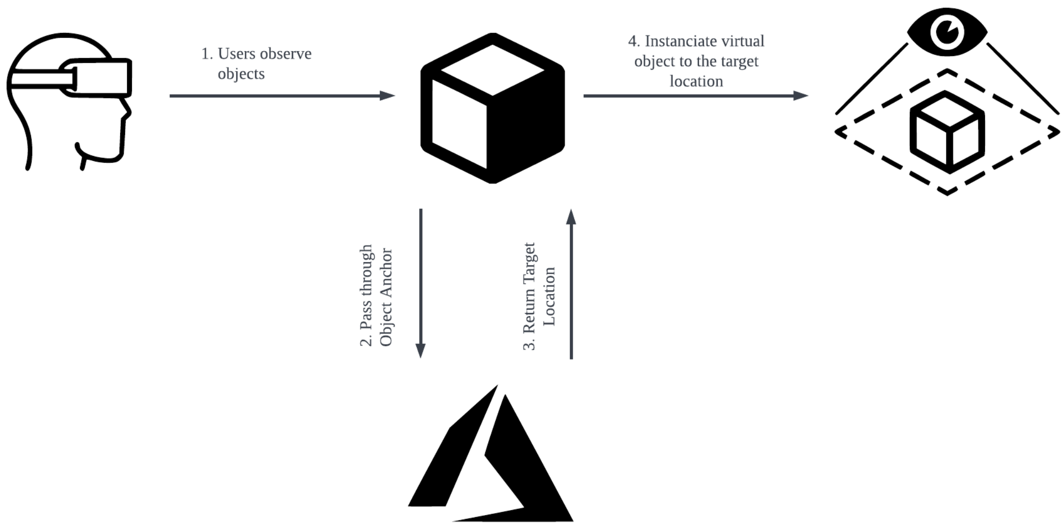


Figure 4 MAOA workflow

We can visualize this converted file through a Unity scene that Microsoft provides. The screenshot shows the converted result. It generates a mesh file that includes the same data as the mesh generated by spatial mapping.

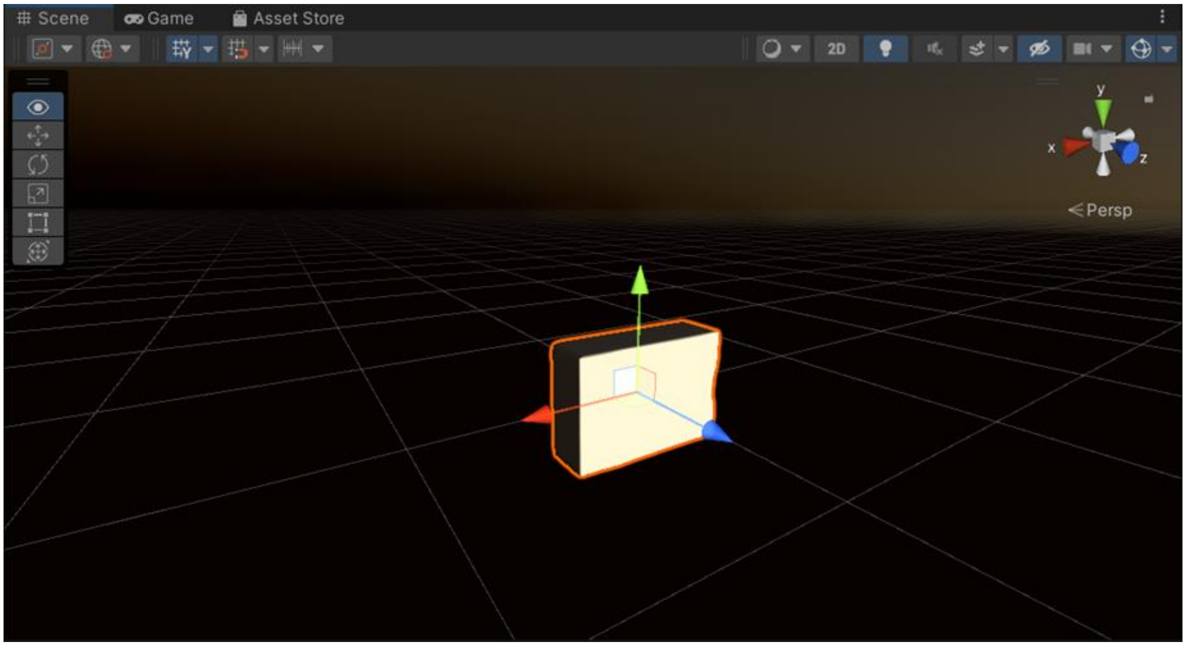


Figure 5 Uploaded model visualization

After we converted the 3D asset, we needed to configure the service through the C# script. We have a script called Object Search that configures the service and runs it with every tick. The script initializes the service by creating a new service. After that, it loads the converted 3D asset file from the local directory. The object detection task will run asynchronously and is called in every tick. If the object is detected, it will be added to the list of tracking results. The result includes various information about the target, and the information we need is its location. We can access the result from anywhere by instantiating the class IObjectAnchorService. Finally, we configure the option of managing the search area. We chose to use FOV and modify the value of MinSurfaceCoverage to 0.7. MinSurfaceCoverage is used as a parameter to determine the similarity of the object to the 3D assets. The smaller the value of MinSurfaceCoverage, the shorter the time it takes to detect the thing, but at the same time, the lower the accuracy.

Step 2: MRTK 2 and Hand Tracking

MRTK 2 provides an input system and MR UI components. We used this toolkit for the interaction part of our project. As we mentioned in the system design part, we manually used the script to track the user's finger joint. We have a script called HandTracker, which contains all the functions used to optimize hand tracking. There are several ways to access the hand controller. The way we used is to access data through a static class called HandJointUtils. We call the function TryGetJointPose from this class. To do that, we need to specify the specific joint label, which should be the index tip; We also need to set the hand to be tracked, and the script supports tracking two hands simultaneously, which is also the option we applied. Finally, we need to pass a variable with the type MixedRealityPose. We call it pose. This variable is used for storing the joint pose data so we can access it later.

```
if (HandJointUtils.TryGetJointPose(TrackedHandJoint.IndexTip, Handedness.Any, out pose))
```

Figure 6 Getting hand tracking data from MRTK

We instantiated the 3D model of a key when hands were detected and destroyed it when hands were out of view. The key's position will be updated in every frame based on the position of the index tip. Lerp is an interpolation function that can transfer from A to B by the interpolant. We use this function to translate the key's position from its previous position so it can move smoothly when it follows the hand movement.

Step 3: Bluetooth Communication

We use Bluetooth to transfer the data between HoloLens 2 and Arduino. In the software part, we need to write scripts in Unity to set up Bluetooth communication. We used a Unity plugin from the asset store called Arduino Unity Plugin. This plugin is a user-friendly class that helps users establish the connection between Arduino and Unity. The plugin provides two different ways to establish the connection. It depends on which type of Bluetooth module the user is using. There are two types of Bluetooth modules, one is Bluetooth Classic, and another one is Bluetooth Low Energy (BLE). To set up Bluetooth, initially, we need to get an instance of BluetoothHelper. It's the core of this plugin. We need to specify the Bluetooth device; the default name of our Bluetooth module is "HC-05". For the Bluetooth Classic, we have to pair it with HoloLens 2 first, so the program can find the Bluetooth device through the paired list based on the specified list. BLE works differently from Bluetooth Classic. It doesn't have to be paired with the device externally. After we determine the name of the device, we can call the connect function to connect the device. There are several events that the given methods can invoke. Here are three events we will use in this program.

1. **OnConnected:** It will be invoked when the connect method successfully connects the device.
2. **OnConnectedFailed:** This event will be invoked when the connect method fails to connect the device, we used this event for debugging purpose.

3. **OnDataReceived:** It will be invoked when the listener method receives data from the Bluetooth device.

When the connect function is called, and the device is connected successfully, we call StartListening function in the Onconnected event to listen for the incoming messages. To send a message to the Bluetooth device, we can send the message we want through the calling SendData function. We have a function that monitors the two variables from Data class in every frame, and if two variables match the condition for sending the message, it will call the SendData. After we send the message to Arduino, we need to set isSent to true. When the program receives the message from Arduino, the OnDataReceived event will be invoked, and inside the event, we will update the isSent in the Data class to false.

Step 4: Collider Setup

The final step is to set up the scene. In our demo, we need to instantiate the virtual lock and key, and user needs to insert the key into the lock to open the physical box. In order to detect if the key is inserted, we need to add box collider components to both 3D assets of key and lock.

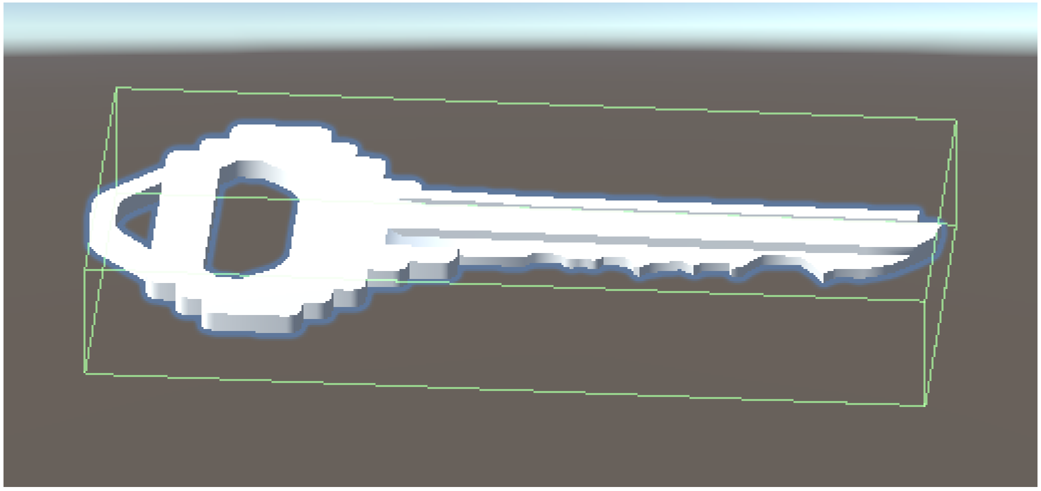


Figure 7 Setting collider

We have a script called CollideTrigger. We attach this script to the prefab of the lock. When two colliders overlap, it will trigger the event OnTriggerEnter. In the event, we will update the isCollide variable to true. When the two colliders no longer overlap, it will invoke OnTriggerExist. We will set isCollide to false in this event. Figure 8 shows the screenshot of the completed Demo.

Figure 8 Screenshot of the completed demo

6.2 Hardware Part

Arduino * 1
HC-05 * 1
Solenoid lock * 1
Relay * 1
Battery box * 2

Table 1 Components for lock system

We used Arduino Uno to build a remote controllable lock. We used the Bluetooth module model HC05 for communicating with HoloLens 2. It's a Bluetooth Classic protocol module, which has poorer energy efficiency compared to the BLE, an updated standard of Bluetooth. To let Arduino control the lock's status, we use a solenoid lock. A solenoid lock is a type of lock that uses electromagnetism. When it is powered on, Arduino will unlock the lock. A relay module is a switch that can open or cut a circuit. We use a relay module to operate with a solenoid lock by controlling the current that passes through the lock. The lock requires direct current with five volts and two amps; we have a battery box connecting to the relay module to provide enough power to the lock. When connecting the lock with the relay, we need to connect the lock to the NO port. The difference between NC and NO is NO port allows the device only to be turned on when the relay is active. NC is quite opposed.

In the script, we read the message in each tick. If it receives the message from HoloLens 2, we will set the pin mode of the relay module to high, which increases the voltage of the relay module, which will activate it so that the lock will open. We set a timer for five seconds. After five seconds, the relay will be inactive, and the lock will be closed automatically. Also, at that point, the Bluetooth module will send a signal to the HoloLens 2, and the software part will invoke the event OnDataReceived and do its job.

Figure 9 A Box with lock system

Chapter 7. Experiment and Result

7.1 Experiment Procedure

To validate the natural interaction design, we set a control group that uses the button to open the smart lock in the MR environment, representing the WIMP interaction.

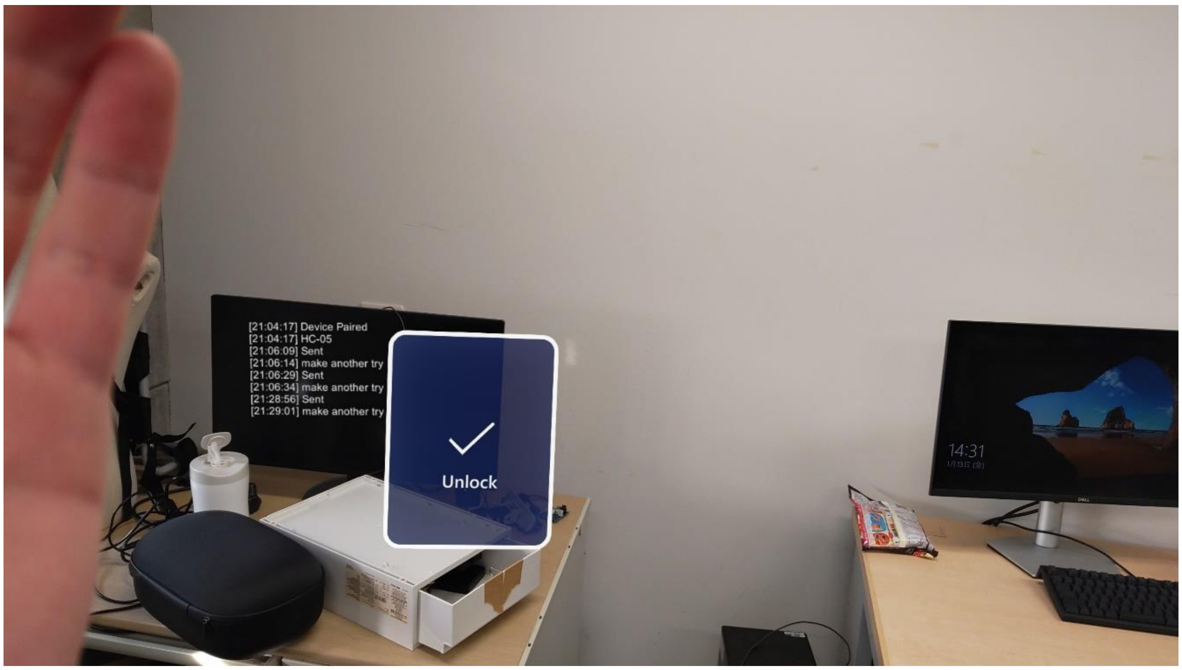


Figure 10 Push-button type controller

We labeled our natural interaction demo as "Case 1" and our control group as "Case 2". We asked participants to try both demos and fill in the NASA TLX for each case. After they had experienced both cases, we asked them to finish another survey for additional feedback. The survey includes the following question.

- Participant's age
- Do you have prior experience with AR
- 0 to 10 rating scale for the preference of Natural interaction, 0 stands for dislike, 10 stands for very like.
- 0 to 10 rating scale for the preference of WIMP interaction, 0 stands for dislike, 10 stands for very like.

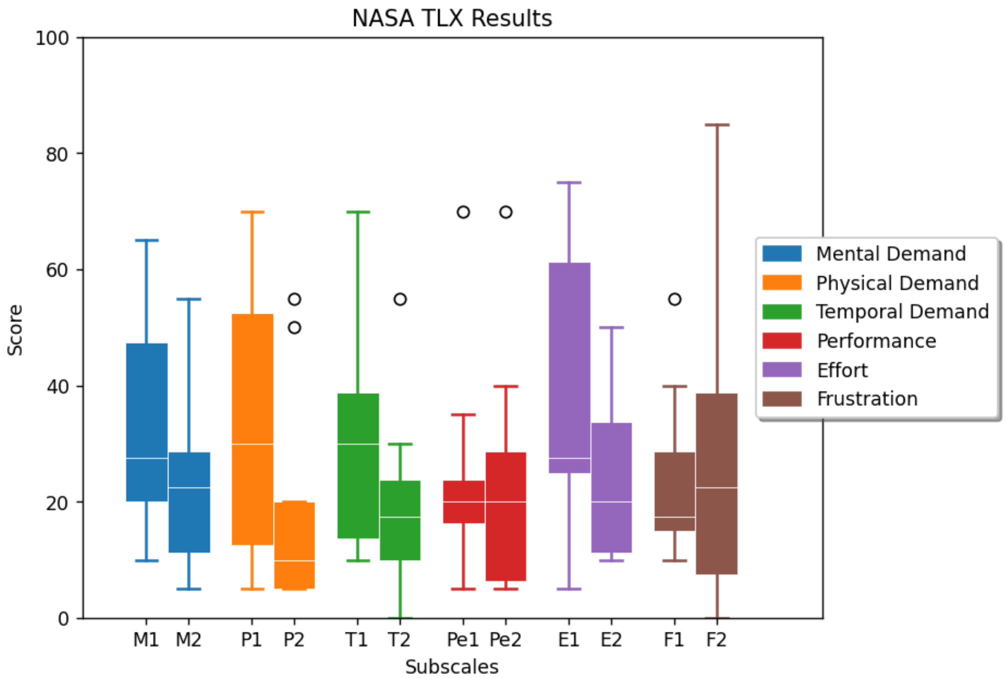
- Based on the demo, what aspects of natural interaction do you prefer compared to WIMP interaction.
- Based on the demo, what do you think are the shortcomings of the natural interaction compared to the WIMP interaction.
- Do you think natural interaction could reduce the learning barrier of new tech, and why.

This survey aims to get participants' feedback and perform additional analysis in conjunction with the research goal. After we collected data from participants, we used a boxplot to visualize the distribution of the NASA TLX score and rating scale, so we could compare the results of two cases and make some conclusions.

We found ten participants from Waseda University; the age range was 22 to 33 years old. Forty percent of participants haven't used AR devices before.

7.2 NASA TLX Result

The picture below shows the boxplot for the NASA TLX result. The Y axis represents the overall score for each subscale. By comparing mental demand for case 1 and case 2, we can see the median of case 2 is lower than case 1. Also, the upper quartile of case 2 is much lower than case 1, meaning most participants think the mental demand for case 2 is lower than case 1. Similarly, most participants believe the physical and temporal demands of case 2 are lower than in case 1. For the performance box, the median of case 1 and case 2 are very close. The Q3 of case 1 is lower than case 2, but the Q1 of case 1 is higher than case 2. Although some participants feel more satisfied when they finish case 2, certain groups feel much less comfortable when they finish case 2. But overall, the two cases provide similar satisfaction for participants. We also found participants make more effort when they perform task 1. For the frustration box, the median of case 1 is lower than case 2, and Q3 of case 1 is much lower than case 2. Although the Q1 of case 2 is less than case 1, most participants are less frustrated when they experience case 1.



.

Figure 11 Result for NASA TLX score

Figure 11 shows the box plot for the NASA TLX Result. X axis presents each subscale. Each label under the x axis presents the abbreviation of each subscale's name, for instance, M1 presents the mental demand for case1, and P2 means physical demand for case 2, vice versa. The bubbles that above box presents the outliers, we dropped these values because they are significantly different from all other values. Keeping these values could affect the result. By reviewing the plot, we can see that the overall score of NASA TLX shows the mental, physical, and temporal demand of the natural interaction demo is higher than the WIMP interaction demo. Participants feel satisfied by completing both demos, and the natural interaction takes more effort compared to WIMP. Finally, participants feel less frustration when they use natural interaction. The overall WIMP interaction experience is better than the natural interaction demo.

7.3 Rating Scale

When we look at the result from user's rating scale, the median of natural interaction is significantly higher than WIMP, and both Q1 and Q3 of natural interaction are all higher than WIMP. Thus, it looks like overall participants prefer to use natural interaction.

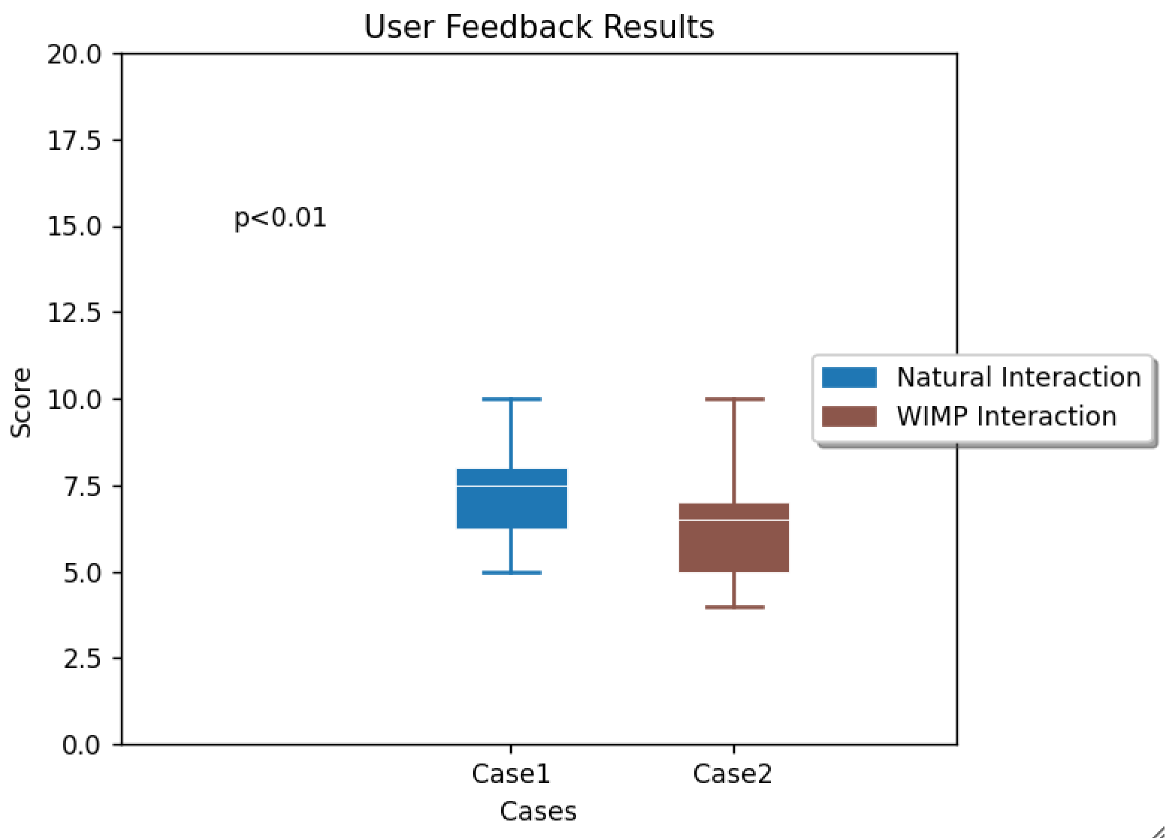


Figure 12 User rating score for two cases

When we look at the result from the user's rating scale, the median of natural interaction is significantly higher than WIMP, and both Q1 and Q3 of natural interaction are higher than WIMP, and p value is smaller than 0.01. Thus, overall, participants prefer to use natural interaction. These results are interesting because, based on the NASA TLX, it seems WIMP interaction has a better overall score than natural interaction. We think one of the reasons is that a single task has some limitations, it makes the whole interface for case 2 too clean, so some shortcomings of case 2 were not exposed. We got a suggestion from one of our participants that we could add smart devices into our system so that they can have a more detailed experience of the whole system. Another reason is people are already familiar with WIMP interaction, and all the participants are college students. They are not strangers to modern technology. Also, because more than half of the participants had never used an AR headset before, some were stressed out when they first used HoloLens. The natural interaction demo was the first case they tried, which can also affect their score.

7.4 User Feedbacks

We received some feedback from participants based on the two questions we asked. For the positive side of natural interaction, participants think it's easy to understand. Because natural interaction is close to the operation in the real world, it has better feedback than the WIMP interaction. One participant mentioned that "it makes me feel the experience is smoother and more seamless" another participant said, "You can experience the practicality of the process of unlocking the door and strengthen your perceptions and know your mission objectives."

On the negative side, participants think the natural interaction takes too much time to complete the task. As NASA TLX measured, it takes more physical demand than WIMP interaction. One participant shared that "implementing natural interaction might need much more physical effort than the WIMP interaction. The scale of the virtual object needs to be proper. Also, the texture of virtual objects needs to be detailed."

We conclude that participants think the natural interaction is intuitive and has better feedback than WIMP. However, it takes more effort and time, which is less convenient if the user is already familiar with the task.

We also asked if participants thought natural interaction could reduce the learning feature of new tech, and seventy percent of participants agreed. The participants who agree with this opinion commonly think that using real-world interaction will take less effort to learn complicated tasks, and users can apply their real-life habits to the task flow.

We found the overall result presented by NASA TLX deviated slightly from the user's feedback in the additional survey. Results from partial subscales match what we learned from the user's feedback. Such as physical demand and temporal demand. Since the user needs to pick up the key to open the lock, this requires additional operations. It does take more time and physical energy compared to pressing the button. However, there are some differences between the mental demand scores for this interaction and the user feedback in the survey. Mental demand is the subscale for measuring if the task is simple and easy to understand. Based on the user feedback, we can see that natural interaction is easy to learn and straightforward. This may be caused by the fact that participants could not show their preferences directly through the NASA TLX. However, there can be other reasons. Since this assessment is exceptionally subjective, it is normal to see slight deviations in the data.

Chapter 8. Conclusion and Future Work

This paper presents a design that applies natural interaction to the MR smart home control system. We evaluate the user experience by comparing it with WIMP interaction in 3D space and explore the possibility of natural interaction to lower the learning barrier. We conducted a case study of opening the smart lock. Through the hand tracking and spatial mapping feature of HoloLens 2, we reproduced the physical lock opening experience in digital space. We also made a demo for WIMP-based interaction as the control group for comparison with our design.

The overall feedback from the survey shows that participants prefer natural interaction; they think natural interaction is intuitive, and users do not have to redevelop their usage habits. However, it takes more time and effort, so it may not fit the scenario when the task is simple. Also, the experiment answered our question that natural interaction could reduce the learning barrier of new technology. Thus, if MR became popular, we could apply this interaction to smart homes and other fields to facilitate the user adoption barrier. Users could accept new technology faster than before.

This study does have some limitations. We only developed a single case which makes the whole task too simple, so the shortcoming of WIMP interaction in the 3D user interface cannot be fully exposed, such as the shortage of feedback and the accuracy of selection. Also, because the task is simple, the time to complete the task is too short, and the user's experience of the interaction may need to be deeper, which may also affect the result.

In the future, we should improve our experiment and add more cases to make more complete test scenarios. Also, we need to improve our experiment procedure, such as adding a session to help participants get familiar with the essential operation of HoloLens. It will reduce the external influencing factors and improve the accuracy of the test data. Furthermore, we can apply natural interaction to other fields to explore the application scenarios of natural interaction beyond the smart home.

Reference

[1] Milgram, P., & Kishino, F. (1994/12/25). A Taxonomy of Mixed Reality Visual Displays. IEICE TRANSACTIONS on Information and Systems E77-D, 8.

[2] qianw. What is mixed reality? - Mixed Reality. Retrieved December 22, 2022, from https://learn.microsoft.com/en-us/windows/mixed-reality/discover/mixed-reality

[3] Li, W., Yigitcanlar, T., Erol, I., & Liu, A. (2021). Motivations, barriers and risks of smart home adoption: From systematic literature review to conceptual framework. Energy Research & Social Science, 80, 102211. https://doi.org/10.1016/j.erss.2021.102211

[4] Pal, D., Funilkul, S., Charoenkitkarn, N., & Kanthamanon, P. (2018). Internet-of-Things and Smart Homes for Elderly Healthcare: An End User Perspective. IEEE Access, 6, 10483-10496. https://doi.org/10.1109/ACCESS.2018.2808472

[5] Kyriakou, P., & Hermon, S. (2019). Can I touch this? Using Natural Interaction in a Museum Augmented Reality System. Digital Applications in Archaeology and Cultural Heritage, 12, e00088. https://doi.org/https://doi.org/10.1016/j.daach.2018.e00088

[6] Nogueira, P. A., Teófilo, L. F., & Silva, P. B. (2015). Multi-modal natural interaction in game design: a comparative analysis of player experience in a large scale role-playing game. *Journal on Multimodal User Interfaces*, *9*(2), 105-119. https://doi.org/10.1007/s12193-014-0172-1

[7] Mahroo, A., Greci, L., & Sacco, M. (2019). HoloHome: An Augmented Reality Framework to Manage the Smart Home. Augmented Reality, Virtual Reality, and Computer Graphics, Cham.

[8] Inomata, S., Komiya, K., Iwase, K., & Nakajima, T. (2020). AR Smart Home: a Smart Appliance Controller Using Augmented Reality Technology and a Gesture Recognizer.

[9] Design and Implementation of IoT-Based Smart Home Voice Commands for disabled people using Google Assistant. 2020 International Conference on Smart Technology and Applications (ICoSTA).

[10] Laehyun, K., Wanjoo, P., Hyunchul, C., & Sehyung, P. (2010, 9-13 Jan. 2010). An universal remote controller with haptic interface for home devices. 2010 Digest of Technical Papers International Conference on Consumer Electronics (ICCE),

[11] Day, A. (2012). Welcome to the Post-WIMP Era. Welcome to the Post-WIMP Era. Retrieved December 25, 2022, fromhttps://blogs.gnome.org/aday/2012/01/12/welcome-to-the-post-wimp-era/

[12] Baraldi, S., Bimbo, A. D., Landucci, L., & Torpei, N. (2009). Natural Interaction. In L. Liu & M. T. ÖZsu (Eds.), Encyclopedia of Database Systems (pp. 1880-1885). Springer US. https://doi.org/10.1007/978-0-387-39940-9_243

[13] Amazon, Samsung, Google, Apple: Big 4 Driving Smart Home Device Sales | Tech Times. (n.d.). Retrieved January 15, 2023, from https://www.techtimes.com/articles/192188/20170113/amazon-samsung-google-apple-big-4-driving-smart-home-device-sales.htm

[14] thalbern. Object manipulator - MRTK 2. Retrieved January 19, 2023, from https://learn.microsoft.com/en-us/windows/mixed-reality/mrtk-unity/mrtk2/features/ux-building-blocks/object-manipulator

[15] keveleigh. Hand tracking - MRTK 2. Retrieved January 19, 2023, from https://learn.microsoft.com/en-us/windows/mixed-reality/mrtk-unity/mrtk2/features/input/hand-tracking