Real-World Object Capture in a Mixed Reality Environment

Beej Persson - 40183743

Submitted in partial fulfilment of
the requirements of Edinburgh Napier University
for the Degree of
BSc (Hons) Games Development

School of Computing

April 9, 2018

Authorship Declaration

I, Beej Persson, confirm that this dissertation and the work presented in it are my own achievement.

Where I have consulted the published work of others this is always clearly attributed;

Where I have quoted from the work of others the source is always given. With the exception of such quotations this dissertation is entirely my own work;

I have acknowledged all main sources of help;

If my research follows on from previous work or is part of a larger collaborative research project I have made clear exactly what was done by others and what I have contributed myself;

I have read and understand the penalties associated with Academic Misconduct.

I also confirm that I have obtained informed consent from all people I have involved in the work in this dissertation following the School's ethical guidelines.

Signed:

Date:

Matriculation no:

Data Protection Declaration

Under the 1998 Data Protection Act, The University cannot disclose your grade to an unauthorised person. However, other students benefit from studying dissertations that have their grades attached.

Please sign your name below one of the options below to state your preference.

The University may make this dissertation, with indicative grade, available to others.

The University may make this dissertation available to others, but the grade may not be disclosed.

The University may not make this dissertation available to others.

Abstract

Contents

List of Tables

List of Figures

Acknowledgements

I would like to thank my supervisor, Kevin Chalmers, for his help guiding me through the project.

Special thanks to Willow for being a good dog.

1 Introduction

1.1 Motivations

Mixed reality is an emerging technology that can enhance a user's ability to interact with the world. An application that allows for detection and manipulation of real-world objects is one that can provide new ways to educate students, build virtual structures or even tell immersive stories. The project could benefit technical developers looking for innovative game mechanics or mixed reality enthusiasts seeking an immersive challenge.

The Microsoft HoloLens is a unique and intriguing piece of hardware. It allows for seamless transition between being in the real-world whilst viewing and interacting with the virtual. The possibilities provided by such an innovative device are compelling and developing for it is one of the primary motivators for carrying out the project.

The idea of capturing physical objects in the real-world and then manipulating them in interesting ways in real-time is relatively unexplored. The concept of such an application is enough to warrant further exploration of these technologies and it is this concept from which the project's goals derive.

1.2 Aims and Objectives

The aim of this project is to evaluate the effectiveness of real-world object capture and real-time object manipulation in a mixed reality environment using Vuforia on the Microsoft HoloLens. The goal was to develop a method to recognise and track a simple 3D object in space using Vuforia's Software Development Kit (SDK). If a simple object was successfully detected and tracked then the scope of the project could extend to attempts at manipulation of the object and the capture of multiple and/or more complex objects. The manipulation attempts would include over-laying post-processing effects by applying shaders to the captured object to, for example, make it appear in greyscale or blurred. Further testing could be done to explore to what

extent these manipulations can be performed in real-time given the limiting hardware of the HoloLens.

1.2.1   Overview Of Milestones

The objectives of this project can be summarised in these milestones:

- simple physical object detection on the HoloLens

- simple object manipulation through the use of post-processing shaders

- complex, or multiple, physical objects detected simultaneously

- real-time object manipulation

And, if possible given the scope of the project:

- multiple real-time manipulations of multiple detected objects

- determining the limits of the HoloLens hardware for real-world object manipulation

1.3   Scope and Limitations

1.3.1   Scope

The scope of the project was to work with the Vuforia SDK and build a Universal Windows Platform (UWP) application in the Unity engine that runs on the Microsoft HoloLens. The application, if successfully implemented, would allow a user to detect, track and manipulate objects via a simple user-interface.

1.3.1.1   Main Deliverables

The main deliverables of the project are listed below:

- A report on the background technologies used in the project

- Object capture and manipulation implementation

- Full report of the project detailing the methodology followed, the implementation specifications and an evaluation of the work done

- Poster summarising the key features of the project

1.3.2 Constraints

Vuforia's support of the Microsoft HoloLens is relatively unexplored. Certain features Vuforia offers may not all work with the HoloLens' specific hardware. Due to the nature of the relative niche that Vuforia's SDK seeks to fill, when troubleshooting and reading through Vuforia's forums the desired answers are sparse. This is true for working with Vuforia using more common hardware, but is especially so when working primarily with the HoloLens.

One of Vuforia's targets require models to be 3D printed. A 3D printer was not able to be acquired for the project and as a result during development model target detection was not explored.

Deploying applications to the HoloLens could be done either over WiFi or via a USB connection. Each time the application was deployed it took a significant amount of time for the application to execute. This made trying to fix minor bugs frustrating as, for example, after waiting a while for it to deploy the application may simply not run due a small typographical error. Microsoft provide a HoloLens Emulator that could have enabled faster deployment and debugging times but it required a version of Windows 10 that this project did not have access to during development.

1.4 Structure of this Dissertation

This dissertation is split into three primary sections.

The first section seeks to provide a background to the technologies utilised by the project. The second section details the methodology followed and documents the resulting implementation. The final section is an evaluation of the application and a conclusion to the dissertation and project as a whole.

Section one (Chapter 2) addresses the background information associated with the project. It looks at the three primary concepts and technologies utilised: mixed reality, the Microsoft HoloLens and the Vuforia.

Section two (Chapters 3 and 4) details the approach taken to the project and tools that were used. It includes a look at the specifics of the work undertaken, any problems that were encountered and how that affected the development of the deliverables.

Section three (Chapters 5 and 6) provides an evaluation of the implementation, its relation to the initial goals and a discussion of whether the results were successful. It also provides a conclusion that summarises the findings of the project and highlights future work this project stands to benefit from. This section also includes a reflection on me, the author, and my feelings about the project.

2 Background

This chapter is a look at the context of the project. It aims to provide an overview of the key technologies that appear throughout and a frame of reference for the project's direction thereafter. There are three main background components to discuss. These are mixed reality, the Microsoft HoloLens and Vuforia.

2.1 Mixed Reality

2.1.1 Virtuality Continuum

Mixed reality (MR) is a variation of virtual environments, or virtual reality. A virtual environment is one in which a user is immersed inside a synthetic environment. Whilst in this environment the user cannot see the real world around them, what they see is entirely virtual. In a mixed reality environment, however, the user is able to see aspects of both the real world and super-imposed virtual objects. MR "supplements reality, rather than completely replacing it" [1].
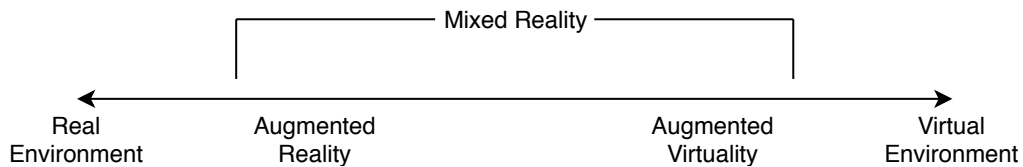
Figure 1: The Virtuality Continuum

MR is often considered as encompassing both augmented reality (AR) and augmented virtuality (AV), both of which reside along the "virtuality continuum" (see Figure 1), a term coined by Milgram and Kishino in 1994 [2], with the real world at one end and the virtual at the other. AR refers to an otherwise real world environment in which there can be seen virtual augmentations, often viewed through a clear screen. AV, conversely, attempts to merge real world scenery or objects into an otherwise virtual world.

2.1.2 Applications and Devices

The nature of combining real and virtual objects can be utilised in a variety of ways and can enhance a user's ability to perform certain tasks and interact with the world. Whilst initially prevalent in the arts and entertainment industries, MR has a number of practical uses and is being taken advantage of by businesses today, particularly in the fields of military training, manufacturing and education [3,4]. Many devices utilise these virtual environments, such as the Oculus Rift and the HTC Vive, headsets that need to be connected to a computer or console, both released in 2016, but the Microsoft HoloLens is one of the few standalone devices available that offers mixed reality capabilities in a head-mounted display.

2.2 Microsoft HoloLens

The Microsoft HoloLens is a mixed reality holographic head-mounted display unit with an adjustable cushioned headband released in 2016 [5]. It is "the first fully untethered holographic computer running Windows 10", enabling a mixed reality experience with no wires, no external cameras and no connection to a PC required [6].

2.2.1 Hardware, Sensors and Features

Table 1: Microsoft HoloLens Hardware Specifications

CPU	Intel Atom x5-Z8100 @ 1.04GHz
GPU/HPU	HoloLens Processing Unit
RAM	2.0GB
Storage	64GB
OS	Windows 10 32-bit

The optics are two separate displays, which are viewed through glass lenses, on to which images are projected and layered to produced what the user can see in their space. The HoloLens' system is essentially mobile hardware (see Table 2) featuring an Intel Cherry Trail Atom chip [7]. Additionally,

however, it runs a few processors that make it unique. There is an inertial measurement unit (IMU), which includes an accelerometer, gyroscope, and magnetometer, and there is an aptly named Holographic Processing Unit (HPU), that handles where the user is looking, gesture tracking, spacial mapping and more. [6].

It has four "environment understanding cameras", two on each side, which provide the basis for tracking the user's head; a depth camera, which helps with hand tracking and performs surface reconstruction (for placing holograms on physical objects); a video camera and an ambient light sensor. These sensors work with the optics module and the IMU and is packaged for the Intel Atom chip by the HPU to produce fast response times to movement so that the user's position is updated and displayed in less than 10ms to ensure the holograms feel part of the world. The HoloLens also features four microphones for receiving user commands, mounted stereo speakers providing spatial sound and both WiFi and bluetooth [8].

2.2.2 Practical Applications

As of 2018 there has only been a few practical applications of the Microsoft HoloLens and a number of them have only been proof of concepts so far. Notably there has been a number of applications developed by NASA in partnership with Microsoft. In 2015 they teamed up for the OnSight and Sidekick projects, utilising the HoloLens on the International Space Station for communication and real-time guidance with ground operators, and later again in 2017 to help find the best places on Mars to build bases [9,10].

There are also a couple of examples of use in medicine. In 2017 a team of surgeons used the HoloLens to visualise MRI and radiography information whilst operating on a patient with a malignant tumour patient [11,12]. Also in 2017 an ultrasound training simulation was designed to teach healthcare workers proficiency in anatomy through interaction with 3D holograms of internal human structures [13,14].

Microsoft are also working on using the HoloLens as a means of communica-

tion as part of a research project called Holoportation [15]. Users will have high quality 3D models of themselves rendered in each other's local space. The goal is to allow them to see, hear and interact with one another as if they were in the same room [16].

2.3 Vuforia

Vuforia is an augmented reality software development kit designed for Android, iOS and Universal Windows Platform (UWP) applications with support for phones, tablets and eyewear [17, 18]. The Vuforia engine is natively integrated with the latest versions of Unity and provides the technology to recognise and track images and simple 3D objects in real time. Vuforia provides a developer portal with access to the Vuforia Target Manager that allows the creating and storing of multiple targets to be used by Vuforia applications [19].

2.3.1 Target Types

Vuforia supports a variety of 2D and 3D targets that it stores in a databases that it can reference when needing to detect and track objects. These features are listed below:

Table 2: Vuforia SDK Features

Images	Objects	Environments
Image Targets	Object Recognition	Extended Tracking
Multi-Targets	VuMark	Ground Plane
Cylinder Targets	Model Targets	

2.3.2 Related Projects

There have been a few examples of applications that make use of Vuforia's tracking capabilities to build interactive experiences. These applications tend to be built for mobile devices. In 2014 a mobile advertising application was

designed to take advantage of Vuforia's markerless image detection to detect image targets that possessed information relevant to the user [20].

There are far fewer applications that use Vuforia on the HoloLens. In 2017 a simple proof of concept assembly instruction application was attempted in Unity that featured user interfaces, dynamic 3D assembly instructions and spatially registered content placement [3].

2.3.3 Summary of Findings

Attempting to capture and manipulate real-world objects on the Microsoft HoloLens is mostly unexplored territory. The HoloLens provides spatial recognition hardware and sensors that allow for head and hand tracking but no built-in object recognition. Vuforia provides the tools to recognise real world objects through the use of a variety of targets stored in the online target databases. The SDK is integrated into Unity and supports development of UWP applications, as well as specifically stating support of the Microsoft HoloLens.

3 Methodology

This chapter details the approach taken and the tools that were required during the development of the deliverables. It will detail the work that was undertaken and the problems that arose as a result. The problems encountered ultimately defined the outcome of the deliverables and the way in which they affected the project are detailed in this chapter.

3.1 Approach

There were three principal stages to the development process. The first stage was the project set-up. The second stage details the initial attempts at implementations of object capture using Vuforia's object targets. The final stage was an adaptation to the previous lack of success by changing the implementation to use another target type and later changing the environment in which the detection was desired.

3.1.1 Stage One: Set-Up

The first stage was the set-up. This stage included acquiring and getting to grips with the tools and technologies that were to be used. The Microsoft HoloLens was acquired, the required software downloaded and sample projects installed and explored. This stage was important to gain confidence in HoloLens usage and development and allow a more fluid transition to the second stage.

3.1.2 Stage Two: Object Recognition

In the second stage the initial attempt to detect real-world objects using Vuforia on the HoloLens was carried out. Vuforia's Object Recognition was chosen as the best targets to use to attempt this given the research done earlier. The additional software required (Vuforia's Object Scanner APK) was installed and an object was scanned in and made a target in Vuforia Target Manager. An application was built using Unity and multiple attempts were made to detect the object, but all were ultimately unsuccessful. This led to the final stage.

3.1.3   Stage Three: Multi-Targets

In the final stage of development an adaptation was made and Vuforia's
Multi-Targets were explored instead. Images were taken of the multiple sides
of a cuboid object and a target was created using Vuforia Target Manager.
Another application was built in Unity and more attempts were made to
detect the object, but again there was no success. This led to the decision to
try to change the environment in which the attempted detection was being
made. The project was moved from a bright area with wide windows that
provided directional light to a dimmer area with a diffuse overhead light in
the hopes that this would enable detection. Both applications were tested
and adjustments made but object detection on the HoloLens still didn't oc-
cur.

3.2   Technologies

The tools that the project required included hardware and software. This
included the Microsoft HoloLens, Unity, Visual Studio, Vuforia and its asso-
ciated tools. The technologies required are detailed below:

3.2.1   Microsoft HoloLens

The project was to be developed for the Microsoft HoloLens. The HoloLens
was chosen due to its immersive experience when worn, its mixed reality
capabilities and its Windows operating system that aligned with Vuforia's
support of Universal Windows Platform applications. Microsoft also provide
HoloLens Emulator software, however this required an a particular version of
Windows 10 that the project did not have access to. The University provided
the HoloLens for the duration of the project.

3.2.2   Unity

Unity is a multi-platform engine with a comprehensive user interface in which
immersive applications can be built and features a wide variety of developer
tools for 3D virtual environments [21]. As of Unity 2017.2, Vuforia Engine
is delivered as a built-in tool with the majority of the required functionality

coming as standard. Unity was used to build the applications that were ran on HoloLens and were primarily composed of Unity's scenes, game objects and cameras.

3.2.2.1   Scenes

Scenes provide the environments that the application will run in. It contains the camera for the user view-point, lighting and any game objects required [22]. Each scene in the applications built during this project were used to house the targets required by Vuforia and the objects to be rendered on those targets as an indication of target detection.

3.2.2.2   Game Objects

Game objects are Unity named entities that represent every object used in the scene, including the camera, lighting and the Vuforia targets [23]. In the applications built these were primarily the camera, any Vuforia targets that were required in the scene, lighting and canvases used for user interfaces.

3.2.2.3   Camera

Unity's camera is a required game object that provides the user their field of view and lets them see the environment [24]. The camera used in this project is a special one provided by the Vuforia SDK that supports augmented reality by removing the background skybox and adjusting the clipping planes to better suit digital eyewear.

3.2.3   Visual Studio

Visual Studio is an integrated development environment made by Microsoft. It provides a user interface in which software can be designed, debugged, tested and developed [25]. For this project Visual Studio will be used to deploy the Unity applications to the Microsoft HoloLens. Applications can be deployed from the development machine to the device over WiFi or via a USB connection. Visual Studio also allows debugging to occur whilst the

application is running on the HoloLens.

3.2.4   Vuforia

The Vuforia SDK provides the tools to detect and track real world objects through the use of targets. It has a developer portal that allows for easy access to the SDK and associated tools. The tools utilised for the project were Vuforia's Target Manager, Object Scanner and a variety of their pre-defined targets detailed below:

3.2.4.1   Target Manager

The Vuforia Target Manager is a tool that allows the creation and management of target databases via the Vuforia developer portal. The databases can be assigned license keys by developers and enables easy management of the targets needed for separate applications. When creating applications users can choose which databases the SDK will use when attempted to detect targets. The tool can also be used to create new targets by uploading images and 3D models to the database and manipulating them to be in-line with the required format for specific targets [19].

3.2.4.2   Image Targets

Figure 2: An Example Image Target, retrieved from [1].

Image targets are flat images, such as printed media, that the Vuforia SDK can detect and track. They do not require special black and white spaces or codes to be recognised but instead the SDK tracks features that are found in the image itself by comparing those features against its target resource database. Once initially detected the SDK will continue to track the target whilst at least partially in the device camera's field of view and can perform an action or link virtual content to the target [26], such as the 3D rendered teapot that can be seen in Figure 2.

3.2.4.3   Multi-Targets

Multi-Targets are created by combining multiple image targets and arranging them into a simple 3D geometric shape, such as a cube. All the faces of a multi-target can be tracked simultaneously as they possess a pre-defined position relative to the target's origin. As a result the entire object can be tracked when any one of its images has been detected [27]. These targets are created by defining a relationship between pre-existing image targets using the Vuforia Target Manager.

3.2.4.4   Object Recognition and the Vuforia Object Scanner

Object recognition allows the detecting and tracking of more intricate 3D objects by creating object targets made using the Vuforia Object Scanner. The object scanner is provided as an Android Package (APK), officially it is only supported on the latest Samsung Galaxy devices but can be configured to work on other Android devices [28]. It is an application that can be used to scan a physical 3D object, produces an Object Data (*.OD) file and allows a visualisation of the object target and its coverage. The *.OD file produced by the application can then be uploaded to produce Vuforia's target manager and an object target to be produced [29]. The SDK will then be able to recognise the real-world object that was scanned by comparing it with the digital representation of the features and geometry that is provided by the object target.

---

[1]`https://vuforialibrarycontent.vuforia.com/Images/devGuide_ImageTargets.jpg`

3.2.4.5 VuMark

Figure 3: Example VuMarks, retrieved from [2].

VuMarks are customised markers that encode data and support unique identification and tracking. They are similar to image targets in that they are 2D targets that can be detected and tracked, but provide a lot more than image targets cannot. As the same VuMark design can be used to encode a range of unique IDs or data, they enable the SDK to distinguish between identical looking products using their unique instance ID. VuMarks are designed to enable companies to create AR targets that subtly integrate with their brand [30], as shown in the example VuMarks in Figure 3.

3.2.4.6 Model Targets

Model targets enable physical objects to be detected and tracked using a 3D model of the object. The real-world objects are recognised by the Vuforia SDK using a "specially prepared target database that is generated by processing a digital 3D representation of the object" using the Model Target Generator tool. Supported objects must be rigid and have matte surfaces. For this to work, the 3D representation of the object should be accurate to a high degree of precision to the physical object. Vuforia's model target test

---

[2]https://vuforialibrarycontent.vuforia.com/Images/Vuforia_6_Images/
DesignGuide/ExampleVuMarks.png

application features a model that is to be 3D printed to be recognised by the SDK [31].

3.3 Project Set-Up

The first stage of work that was undertaken for the project was to get to acquire the hardware, install the software and begin to get to grips with Microsoft HoloLens development.

3.3.1 Microsoft HoloLens

The Microsoft HoloLens was provided by the University for the duration of the project. Initially it required setting-up the device for development, which included creating a Microsoft account, enabling developer mode on the device and ensuring the correct Visual Studio settings were used on the deployment device. It was during this stage that the features and capabilities of the Microsoft HoloLens were explored as part of an attempt to become more familiar with the hardware.

Microsoft provides a comprehensive set of online documents to guide users new to mixed reality development [32]. It was these guides that were key to the determination of the tools that would be used during the development process of the project. Some of the required tools were downloaded, installed and set up. Initially a basic UWP application was built in Visual Studio and deployed to the HoloLens. This was done to gain a better understanding of how the development machine interacts with the HoloLens and the deployment process.

3.3.2 Vuforia SDK

At this stage the focus of the work was on understanding how the Vuforia SDK worked and how to build applications that took advantage of the functionality it supports. During the early stages of development Microsoft's guides recommended that the Vuforia SDK be installed separately and integrated into Unity 5.6 manually. This changed, however, during the early

development stages and with the release of Unity 2017.1 the Vuforia Engine was delivered with it.

3.3.2.1   Vuforia's Samples

Vuforia provide a comprehensive asset package that can be imported into Unity[3]. This contains the majority of the default target types, such as image targets, object recognition and multi-targets. It also contains a lot of miscellaneous media, such as splash and loading screen images, meshes and materials, as well as an AR camera object.

Vuforia also provide sample applications for digital eyewear that provide some basic scenes containing example targets[4]. Much of the steps taken to get these samples running was determined by combining information detailed in the guides provided by Microsoft, Unity and Vuforia. However troubleshooting Vuforia's forums was required as often the guides lacked necessary information.

The process involved acquiring and applying a Vuforia *Eyewear App License Key* to the Unity project, defining the desired Target Database, fixing the provided camera so that it did not have a skybox and changing the clipping panes to the recommended values for the HoloLens. There were also numerous settings that needed to be changed or enabled in Unity. These included, but are not limited to, enabling virtual reality support in the *XRSettings* in *PlayerSettings*, setting the device type to *Digital Eyewear*, loading the HoloLens Device Config, setting the platform build configurations and enabling extended tracking on each target game object in the advanced section of their behaviour script. After everything had been correctly configured, and after a not-insubstantial amount of minor changes, the sample could on occasion be successfully deployed to the HoloLens. Even until the very end of development there were certain errors that could appear at different stages that had no discernible pattern or reasoning to their appearance.

---

[3]https://assetstore.unity.com/packages/templates/packs/vuforia-core-samples-99026

[4]https://developer.vuforia.com/downloads/samples

Troubleshooting only revealed workarounds, such as rebuilding the Visual Studio solution, restarting Unity and, for one particularly inexplicable bug, starting again with a new Unity project and reinstalling the asset packages. This was to be the norm for all subsequent stages of development.

3.4   Implementation

3.4.1   Object Recognition

At this stage no testing to ensure that Vuforia could detect the simpler targets was undertaken. The downloadable image targets that could be printed off were not used. Instead an attempt at capturing objects using Vuforia's object recognition was undertaken.

3.4.1.1   Choice Over Other 3D Targets

Vuforia's HoloLens sample contained multiple sample scenes, each showcasing a different target type. One of these was a scene named *ObjectReco* that included the object target as a Unity game object. The decision to begin to attempt object capture using Vuforia's object recognition feature rather than its two other 3D targets, VuMark and Model Targets, was twofold. Firstly, using model targets were disregarded earlier due to the constraint of lacking a 3D printer that is required for its functionality. There was also no example scene of model targets provided in the HoloLens sample at the time due to them being relatively new. Secondly, VuMarks were disregarded as they are not true object capture, rather they recognise the 2D VuMark that is applied to a physical object and gather the physical object's information as a result.

3.4.1.2   Creating an Object Target

Vuforia's HoloLens sample contained a object recognition game object that was a representation of a Mars habitat. Vuforia provided media that could be printed off and folded such that it would become physical paper representation the object target. Since Vuforia provided the tools to create new object

targets using the Vuforia Object Scanner and the first goal of the project was to capture real-world objects, the Mars habitat wasn't used.

3.4.1.3   Vuforia Object Scanner

Vuforia Object Scanner is provided as an Android Package (APK) and was installed onto an Android device using Android Debug Bridge (adb)[5]. Vuforia also provide an Object Scanning Target[6] that was to be printed off and used by the object scanner to determine the position of the Object Target relative to it's local origin.
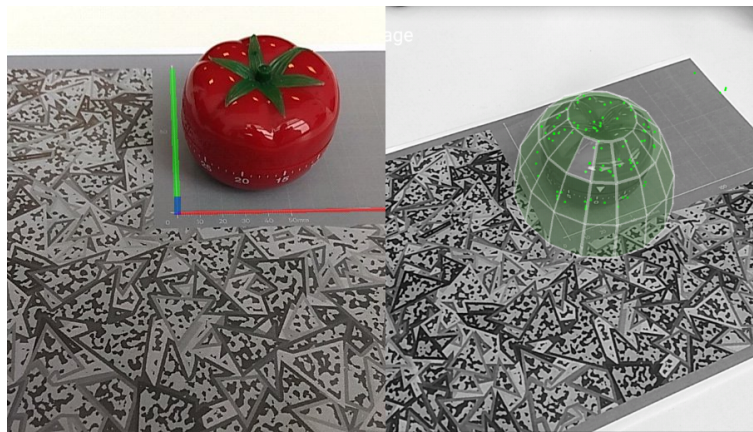

Figure 4: Pomodoro Timer Object Scanning

3.4.1.4   Scanning

As seen in Figure 4, the first object chosen to be scanned in was a Pomodoro timer. The origin is the lower left hand corner of the Object Scanning Target's grid region and corresponds to the local (0,0,0) coordinates of the Object Target instance's bounding box. The Pomodoro timer was placed on the grid ensuring none of the outer edges of the object crossed the grid's axes. The object scanner APK uses the Android device's camera to scan the object and starts when the record button is pressed. The device was moved slowly

---

[5]https://developer.android.com/studio/command-line/adb.html

[6]https://vuforialibrarycontent.vuforia.com/Images/Fall2014/ ObjectScanningTarget.PNG

around the object whilst recording, keeping it within view. The APK over-
lays a simple indication of the object's geometry that is made up of multiple
faces that turn green when the corresponding sections have successfully been
scanned in. Once the majority of the relevant surface areas were green the
recording was stopped and the scanning session was named and the resulting
Object Data (*.OD) was saved. The APK provides the capability to test the
results of the scan by rendering a green cuboid augmentation at the target's
origin when the object is detected. The target was tested in this way against
a variety of backgrounds and was successfully recognised as shown in Figure
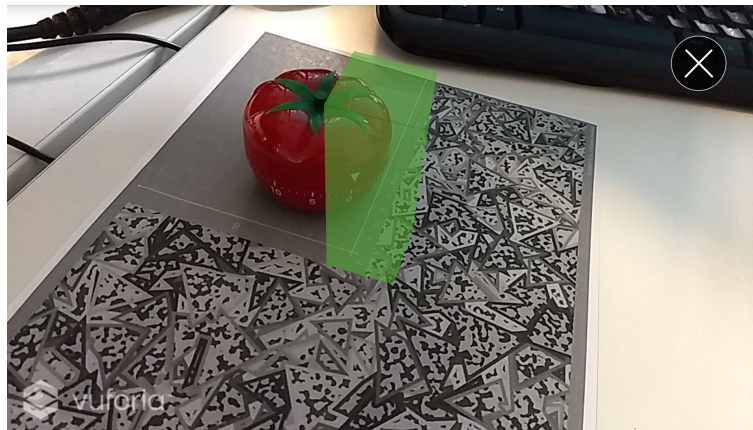5.

Figure 5: Object Detection in the APK Testing Feature

The object data was then sent back to the development machine as an *.OD
file. This file was then uploaded to a new target database created for the
project using Vuforia Target Manager.

3.4.1.5   Application

The Unity project was updated to direct Vuforia to the newly created target
database and a new scene was built that contained the ARCamera and the
new Object Target. The 3D Mars habitat render was left as the object that
would be overlaid and would be used to determine if the Pomodoro timer
was successfully detected. The build settings were configured for UWP, the
target device set to the Microsoft HoloLens and the debugging option was

enabled. The application was exported from Unity into Visual Studio and then built, run and deployed to the HoloLens from there. Despite multiple attempts at reconfiguring the project it was clear the Pomodoro timer could not be detected.

3.4.1.6   Troubleshooting

After all the usual guides provided by Microsoft, Unity and Vuforia and their listed solutions to common problems had been exhausted, online forums were turned to in the hope of finding the answers. Quickly it became clear the consensus of other developers was that using Vuforia's Object Recognition tool was very inconsistent when deployed to the Microsoft HoloLens, even the one provided in the sample[7,8]. A Vuforia Support reply on one post suggested that the user wait for the then unreleased Model Target tool[9]. As so many minor things could affect the success of this object detection method (such as the surrounding environment during the object scanning phase, the material of the object itself and the lighting in the desired detection environment) and there was doubt it could be made to work on the HoloLens at all, it was at this point decided that another type of Vuforia target should be attempted.

3.4.2   Multi-Target

Some further research was done into whether anyone had successfully detected real-world objects using Vuforia on the HoloLens and a two-part blog post by Mike Taulty, a Microsoft UK developer, was found [33]. This page (and the previous part) detailed his attempts at simple object recognition using Vuforia's Multi-Targets.

3.4.2.1   Choice of Target

Given that the goal was still to capture real-world objects, simply detecting image targets would not be enough. Once again the reasons for not trying

---

[7]https://developer.vuforia.com/forum/hololens/object-recognition-sample-doesnt-work-hololens
[8]https://developer.vuforia.com/forum/hololens/improve-object-tracking
[9]https://developer.vuforia.com/forum/hololens/object-recognition-unsatisfactory-hololens

Vuforia's other 3D targets, VuMark and Model Targets, were the same. Vu-Marks were disregarded as not true object capture, and Model Targets as the project did not have access to the 3D printer that they required to be implemented. Multi-Targets can be recognised even if a single child image target is detected and this raised the concern of whether they didn't represent true object capture either, but it was still the closest to the project's goals. As a result it was determined that attempting to capture real-world cuboid objects through the creation of a Multi-Target was to be the next stage of the development.

3.4.2.2   Creating a Multi-Target

FRONT
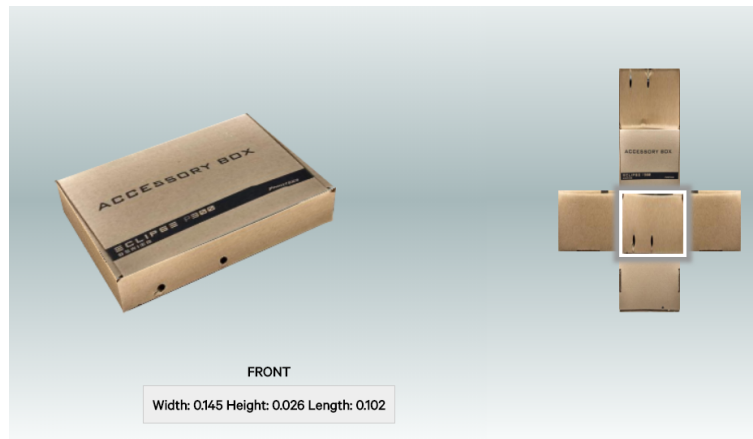Width: 0.145 Height: 0.026 Length: 0.102

Figure 6: Accessory Box Multi-Target in the Vuforia Target Manager

A multi-target is a Vuforia target that consists of multiple image targets in a defined geometric arrangement. The position and orientation of each image target is defined relative to the Multi-Target's origin and allows a the entire target to be detected and tracked whenever one of the child image targets is detected. A Multi-Target was created using the Vuforia Target Manager. A simple cuboid box was chosen and images were captured of each of its faces and uploaded to the manager. The blank mutli-target was built by creating a new target, defining it as a cuboid, inputting its dimensions (in scale with the real box) and then adding it to the previously created target database. At this point the created target could be edited to apply each image to the

corresponding face on the virtual cuboid, as shown in Figure 6. Meeting the tool's image upload requirements was convoluted, the image needed to be a specific format, under a certain size, an exact aspect ratio was needed and if it was a pixel off the upload would fail without explanation.

3.4.2.3   Application

Once again the Unity project was updated to reflect the changes represented by the chosen target. The same target database was selected and a new scene was added that contained the ARCamera and the Multi-Target as a game object. This time, a wireframe box that was slightly larger than the Multi-Target object was created, applied to target and configured so that it would render over the real-world object when detected. This was done so that it would be clear once the application had recognised the box. Once again the application was exported to Visual Studio and deployed to the HoloLens, and once again detection was unsuccessful.
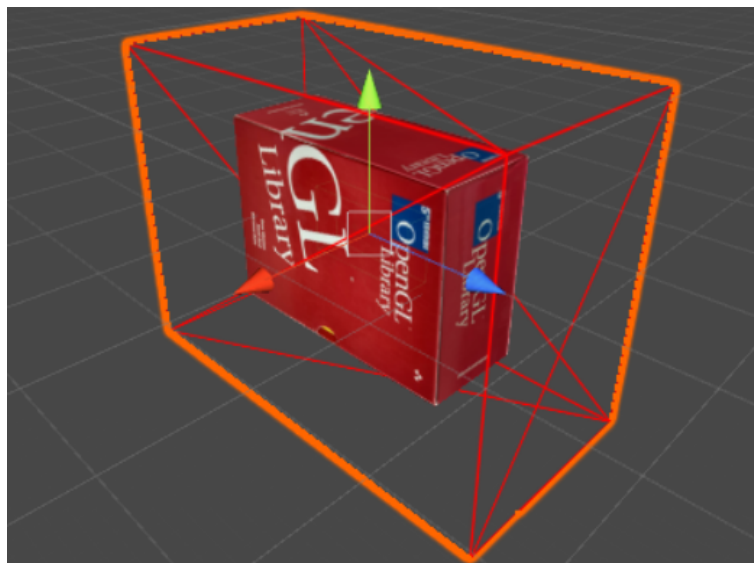

Figure 7: OpenGL Box Multi-Target with Wireframe Cube in Unity Editor

3.4.2.4   Troubleshooting

After checking settings and trying to slightly adjust the environments lighting a decision was made to try a different Multi-Target that was richer in detail

in the hopes that it could be more easily detected. For the creation of the second target care was taken to ensure there was even lighting on each of the surfaces, no glare was visible and no details of any surface were cut off when the pictures were taken. The target was created in much the same way as before, with the multiple images taken applied to a target with the dimensions of the new box in the Vuforia Target Manager. Shown in Figure 7 is the Multi-Target with its surrounding wireframe box as shown in the Unity Editor. Even with the care taken, there was never any indication of successful object detection. After the usual checks, rebuilding and redeploying, it was determined that a potential underlying issue was the environment in which the object detection was being attempted.

For all the stages of development thus far the work had been undertaken in a wide open room with large floor to ceiling windows on one side that provided the environment with ample daylight. However, this light was noticeably coming from one direction with all objects producing shadows. It also produced glare on otherwise fairly matte objects, such as the second box used to produce a Multi-Target. Given the lack of success it was determined that testing would be taken to a smaller room with less directional light that featured a single overhead light in the hopes of providing a more diffuse light setting. In this new environment each of the previous implementations and all the available 3D targets were tested.

Width: 0.086 Height: 0.305 Length: 0.086

Figure 8: Whisky Box Multi-Target in the Target Manager

After multiple attempts at detecting these objects, a third Multi-Target was created using a box that better suited Vuforia's listed requirements for multi-targets. The box chosen was a whisky box as it was rich in detail, made of a matte material, had contrasting regions and none of the sides were the same. The target in its finished form in the Vuforia Target Manager can be seen in Figure 8. Whilst a significant amount of thought and care was put into ensuring an even lighting distribution across each face of the object and a diffuse and well-lit environment the object was still not detected.

3.4.3 Development Conclusions

By this stage the development period that was available was coming to a close. A few final attempts were made at detection by changing a number of variables that could have had an affect on the success of the application. The ARCamera's field of view was varied and tested, a range of capabilities in Unity's *Publisher Settings* were enabled and the wireframe box was edited in the hopes that it simply wasn't visible enough before. None of these attempts were successful and the development stage ended without achieving any of the projects initial goals and only delivering some implementations without much functionality.

4   Evaluation

The aim of this project was to evaluate the effectiveness of real-world ob-
ject capture and real-time object manipulation on the Microsoft HoloLens
using the tools provided by the Vuforia SDK. Due to the lack of success at
producing any functional applications that could recognise and track simple
objects, the project did not progress to the stage of simple object manipu-
lation, let alone real-time manipulations. As a result, there is unfortunately
nothing to evaluate in terms of the effectiveness of real-time application of
simple post-processing effects to manipulate real-world objects. However, the
project does allow for a discussion on the effectiveness of real-world object
capture, especially with regards to the extent by which it can be achieved on
the HoloLens using Vuforia.

4.1   Object Capture

When the project was in its infancy, the aim was to capture simple objects in
a mixed reality environment whilst using the HoloLens. This was envisioned
as detecting and highlighting new objects in real-time. Of course, very early
on during the initial research it was realised that such software didn't exist.
Instead, via the features provided by the Vuforia SDK, detecting objects
required a database of information that can be referenced when scanning
the environment for the natural features that make up the target object.
Many varied attempts were made, with multiple Vuforia tools and features
utilised, but on no occasion was object detection achieved. As a result of
the work done in this project it can be said that real-world object capture
using Vuforia on the HoloLens is rather ineffective. That statement is also
supported by the many others, referenced earlier, who were also facing issues
when attempting the same.

4.1.1   Applicability of the Tools

The tools Vuforia provide were much more easily implemented on an Android
device. This was shown when the Vuforia Object Scanner used produced an
object target that the phone was able to detect very easily. When trying

to detect the same object target using the HoloLens there was no such success.

5   Conclusion

5.1   Has the Project met its Aims?

5.2   Personal Reflection

5.2.1   Learning Outcomes

5.2.1.1   L.O.1: Manage a substantial individual project, including planning, documentation and control.

5.2.1.2   L.O.2: Construct a focussed problem statement and conduct a suitable investigation, including literature or technology review, into the context of that problem.

5.2.1.3   L.O.3: Demonstrate professional competence by applying appropriate theory and practice to the analysis, design, implementation and evaluation of a non-trivial set of deliverables.

5.2.1.4   L.O.4: Show a capacity for self-appraisal by analysing the strengths and weaknesses of the project process and outcomes with reference to the initial objectives and to the work of others.

5.2.1.5   L.O.5: Provide evidence of the foregoing in the form of a report.

5.3   Future Work

References

[1] R. T. Azuma, "A Survey of Augmented Reality," *Presence: Teleoperators and Virtual Environments*, vol. 6, no. 4, pp. 355–385, 1997.

[2] P. Milgram and F. Kishino, "A Taxonomy of Mixed Reality Visual Displays," *IEICE TRANSACTIONS on Information and Systems*, vol. E77-D, no. 12, pp. 1321–1329, 1994.

[3] G. Evans, J. Miller, M. I. Pena, A. MacAllister, and E. Winer, "Evaluating the Microsoft HoloLens through an augmented reality assembly application," *Proc.SPIE*, vol. 10197, pp. 101 970V–1 – 101 970V–16, 2017. [Online]. Available: https://doi.org/10.1117/12. 2262626

[4] C. E. Hughes, C. B. Stapleton, D. E. Hughes, and E. M. Smith, "Mixed Reality in Education, Entertainment, and Training," *IEEE Computer Graphics and Applications*, vol. 25, no. 6, pp. 24–30, 2005.

[5] M. Corporation, "The World's First Holographic Head-Mounted Display — Microsoft HoloLens." [Online]. Available: https://www. microsoft.com/en-gb/hololens/hardware

[6] T. Holmdahl, "BUILD 2015: A closer look at the Microsoft HoloLens hardware," April 2015. [Online]. Available: https://blogs.windows.com/devices/2015/04/30/build-2015-a-closer-look-at-the-microsoft-hololens-hardware

[7] D. Rubino, "Microsoft HoloLens - Here are the full processor, storage and RAM specs," May 2016. [Online]. Available: https://www. windowscentral.com/microsoft-hololens-processor-storage-and-ram

[8] S. Colaner, "What's Inside Microsoft's HoloLens and How It Works," August 2016. [Online]. Available: http://www.tomshardware.com/ news/microsoft-hololens-components-hpu-28nm,32546.html

[9] Jet Propulsion Laboratory, NASA, "NASA, Microsoft Collaborate to Bring Science Fiction to Science Fact," June 2015. [Online]. Available: https://www.jpl.nasa.gov/news/news.php?feature=4642

[10] Microsoft Reporter, "NASA: Microsoft HoloLens is helping us find the best sites for bases on Mars," July 2017. [Online]. Available: https://news.microsoft.com/en-gb/2017/07/10/nasa-microsoft-hololens-is-helping-us-find-the-best-sites-for-bases-on-mars

[11] A. Bernardo, "Virtual Reality and Simulation in Neurosurgical Training," *World Neurosurgery*, vol. 106, pp. 1015–1029, 2017.

[12] 3ders.org, "Microsoft HoloLens helps Spanish doctors cut surgery time in half," October 2017. [Online]. Available: http://www.3ders.org/articles/20171023-microsoft-hololens-mixed-reality-headset-helps-spanish-doctors-cut-surgery-time-in-half.html

[13] J. Lynn, "CAE Healthcare announces first mixed reality ultrasound simulation solution with Microsoft HoloLens," January 2017. [Online]. Available: https://www.emrandehrnews.com/2017/01/27/cae-healthcare-announces-first-mixed-reality-ultrasound-simulation-solution-with-microsoft-hololens

[14] F. Mahmood, E. Mahmood, R. G. Dorfman, J. Mitchell, F.-U. Mahmood, S. B. Jones, and R. Matyal, "Augmented Reality and Ultrasound Education: Initial Experience," *Journal of Cardiothoracic and Vascular Anesthesia*, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1053077017309916

[15] B. Cutler, S. Fowers, and W. Chang, "Holoportation - Microsoft Research." [Online]. Available: https://www.microsoft.com/en-us/research/project/holoportation-3/

[16] S. Orts-Escolano, C. Rhemann, S. Fanello, W. Chang, A. Kowdle, Y. Degtyarev, D. Kim, P. L. Davidson, S. Khamis, M. Dou *et al.*, "Holoportation: Virtual 3d teleportation in real-time," in *Proceedings of the 29th Annual Symposium on User Interface Software and Technology.* ACM, 2016, pp. 741–754.

[17] Vuforia, "Getting Started." [Online]. Available: https://library.vuforia.com

[18] ——, "Vuforia for Digital Eyewear." [Online]. Available: https://library.vuforia.com/articles/Training/Vuforia-for-Digital-Eyewear.html

[19] ——, "Vufora Target Manager." [Online]. Available: https://library.vuforia.com/articles/Training/Getting-Started-with-the-Vuforia-Target-Manager.html

[20] Y.-g. Kim and W.-j. Kim, "Implementation of Augmented Reality System for Smartphone Advertisements," *International Journal of Multimedia and Ubiquitous E ngineering*, vol. 9, no. 2, pp. 385–392, 2014.

[21] Unity Technologies, "Unity." [Online]. Available: https://unity3d.com/unity

[22] ——, "Unity - manual: Scenes." [Online]. Available: https: //docs.unity3d.com/Manual/CreatingScenes.html

[23] ——, "Unity - manual: Gameobjects." [Online]. Available: https: //docs.unity3d.com/Manual/GameObjects.html

[24] ——, "Unity - manual: Camera." [Online]. Available: https: //docs.unity3d.com/Manual/Cameras.html

[25] M. Corporation, "Visual studio ide." [Online]. Available: https: //www.visualstudio.com/vs/

[26] Vuforia, "Image Targets." [Online]. Available: https://library.vuforia. com/articles/Training/Image-Target-Guide.html

[27] ——, "Multi-Targets." [Online]. Available: https://library.vuforia. com/articles/Training/Multi-Target-Guide.html

[28] ——, "Vuforia Supported Versions." [Online]. Available: https:// library.vuforia.com/articles/Solution/Vuforia-Supported-Versions.html

[29] ——, "Object Recgonition." [Online]. Available: https://library. vuforia.com/articles/Training/Object-Recognition.html

[30] ——, "Vumark." [Online]. Available: https://library.vuforia.com/ articles/Training/VuMark.html

[31] Vuforia, "Model targets." [Online]. Available: https://library.vuforia. com/articles/Solution/model-targets-user-guide.html

[32] Microsoft Corperation, "Development for Mixed Reality." [Online]. Available: https://docs.microsoft.com/en-us/windows/mixed-reality/ development

[33] M. Taulty, "HoloLens, Unity and Recognition with Vuforia (Part 2)," December 2016. [Online]. Available: https://mtaulty.com/2016/12/29/ hololens-unity-and-recognition-with-vuforia-part-2/

Appendices

A   Initial Project Overview

SOC10101 Honours Project (40 Credits)

Title of Project:
Real world object-capture in a mixed-reality environment.

Overview of Project Content and Milestones:
This project will look to evaluate the effectiveness of capturing real world objects using Vuforia on the Microsoft HoloLens. Initially, a method to capture a simple object will be developed, before attempts to capture more complex (or multiple) objects and real-time object manipulation will be implemented. The project aims to explore to what extent this can be done.
The milestones will be:

- simple object capture

- simple object manipulation

- complex object capture

- real-time object manipulation.

The Main Deliverable(s):
Object capture and manipulation software.
Dissertation.
Poster.

The Target Audience for the Deliverable(s):
Technical mixed-reality game developers looking for new game mechanics.
Enthusiasts interested in new game technologies.

The Work to be Undertaken:

Research similar attempts at mixed-reality object manipulation.

Design a method to capture a cube (or similar object) using Vuforia.

Build and apply simple shaders to the object.

Expand on above methods to capture more complex objects.

Explore and evaluate the extent to which the objects can be manipulated in real-time.

Document and report findings.

Additional Information / Knowledge Required:

Creating and managing Unity projects.

How to use Vuforia.

Improve understanding of shader usage.

Information Sources that Provide a Context for the Project:

General Development Page for the HoloLens:

`https://developer.microsoft.com/en-us/windows/mixed-reality/development`

Some related downloadable tools:

`https://developer.microsoft.com/en-us/windows/mixed-reality/install_`
`the_tools`

Vuforia Specific:

`https://developer.microsoft.com/en-us/windows/mixed-reality/vuforia_`
`development_overview`

`https://developer.vuforia.com`

The Importance of the Project:

Mixed-reality is an emerging games technology with great potential for immersive story-telling and innovative game design. This project looks to explore an aspect of that and if successful could be beneficial to those interested in designing such games.

The Key Challenge(s) to be Overcome:

Lack of similar projects documented.

Fairly niche/specialist/obscure software, mostly new territory.

B   Report on the Interim Review Meeting

SOC10101 Honours Project (40 Credits)
Week 9 Report

Student Name: Beej Persson
Supervisor: Kevin Chalmers
Second Marker: Gregory Leplatre
Date of Meeting: 20/11/2017
Can the student provide evidence of attending supervision meetings by means
of project diary sheets or other equivalent mechanism? **no***
If not, please comment on any reasons presented
No evidence provided but no indication from supervisor that there was a
problem with weekly meeting attendance.

Please comment on the progress made so far

The work done focused on identifying ways of working with a Hololens, i.e.,
the output side of the project. This was useful, but the main challenge of
the project is object recognition and augmentation, which would have been
useful to deal with earlier. This would also have allowed you to engage with
relevant literature on the subject.

Is the progress satisfactory? **unsure**
Can the student articulate their aims and objectives? **Partly**
If yes then please comment on them, otherwise write down your suggestions.

The overall goal (altering the appearance of real-world objects using AR)
is clear. What specific alterations, to which objects and in which context
remains to be determined. Familiarity with relevant literature will help you

specify your project more accurately. Many things are possible, some more complex than others. For this type of project, a good approach would be to consider what operation would have the highest visual impact while having a manageable development cost. There is definitely potential in your project.

Does the student have a plan of work? **Yes**
If yes then please comment on that plan otherwise write down your suggestions.

Unfortunately, the plan is limited to work already done.

Does the student know how they are going to evaluate their work? **partly**
If yes then please comment otherwise write down your suggestions.

The proposed approach is sensible but it is difficult to be more specific at this stage as the functionalities of the system aren't clearly defined.

Any other recommendations as to the future direction of the project

With Vuforia, you should be able to capture objects in a scene (in advance) and then track these objects. The acquisition/tracking effectiveness probably depends on the complexity of the object, which you should establish. Once you know the limits of what you will be able to track, you can focus on real-time alterations. Your incremental development approach is suitable, but it needs to be better controlled. Once you know what you are aiming to do, you can define the incremental steps that will lead you there. The core of the research in this area is about achieving realism, which involves lighting and material acquisition. Potentially impactful results can also be achieved by taking slighting different directions. Giving an object a stylised look is one of them. Anecdotally, magic tricks are high impact and can be technically cheap. A magic trick you could implement could be: give a physical object

to a user (that you will have previously scanned with Vuforia). Ask them to put it on a table (that you have also previously scanned), and make the object disappear (assuming the user is wearing a Hololens).

Signatures: Supervisor   Kevin Chalmers
                 Second Marker   Gregory Laplatre

                 Student

Please give the student a photocopy of this form immediately after the review meeting; the original should be lodged in the School Office with Leanne Clyde

C   Diary Sheets (or other project management evidence)

Insert diary sheets here together with any project management plan you have

D   Appendix 4 and following

insert content here and for each of the other appendices, the title may be just on a page by itself, the pages of the appendices are not numbered, unless an included document such as a user manual or design document is itself pager numbered.