School of Information Technologies

Department of Computer Systems

Olatayo Joshua Olukotun 172626IASM

Application of Augmented Reality to Industrial Control Systems using Head-Mounted Display

Master's Thesis

Aleksei Tepljakov
Ph.D.
Researcher

TALLINN 2020

Declaration of Originality

Declaration: I hereby declare that this thesis, Application of Augmented Reality to Industrial Control System using Head-Mounted Display, my original investigation and achievement, submitted for the Master's degree at Tallinn University of Technology, has not been submitted for any degree or examination.

Deklareerin, et käesolev diplomitöö, Peakomplekti põhise täiendatud reaalsuse rakendused tööstuslikke automaatjuhtimissüsteemide jaoks, mis on minu iseseisva töö tulemus, on esitatud Tallinna Tehnikaülikooli magistrikraadi taotlemiseks ja selle alusel ei ole varem taotletud akadeemilist kraadi.

Olatayo Joshua Olukotun

Date: May 14, 2020

Signature: ....

Abstract

Supervisory Control and Data Acquisition (SCADA) is the conventional method for locally or remotely monitoring and controlling industrial systems in the shop-floor and the Human Machine Interface (HMI) helps to visualize data but the entire process may become slow or error-prone as automation data is isolated from the industrial systems.

Augmented Reality (AR) has been applied to many industrial use cases to enhance its advantages. Augmented Reality will play a major role in the industries of the future, enabling virtual contents to be deployed on real industrial systems and also enables real time interaction between user, real objects and virtual objects. Unfortunately, alot of these use cases are presently only deployed in the laboratory and not in the real world because overlaying and registering interactive virtual contents on the real world requires sophisticated hardware and advanced software.

In this thesis, an alternative method for the control and supervision of an industrial control systems and monitor real-time data. AR is applied to the industrial control of a multitank tank system. The Unity3D game engine environment is used to build the desired see through environment, designing the virtual object, controls and data representation on the laboratory model of the real-life industrial multi-tank system. Data from Unity3D is sent to MATLAB/Simulink for the control of the multitank system.

The primary outcome of this project is a control and monitor based AR application deployed on the Meta 2 head-mounted display and experimental testing of the developed model in AR.

The thesis is in English Lnaguage and contains 66 pages of text, 6 chapters, 32 figures, and 6 tables.

Acknowledgement

All glory, honor, and adoration be to almighty God the creator of heaven and earth, the giver of life, wisdom and understanding for his mercy and grace upon me throughout my course of study and this project in particular, without whom it would be impossible to complete.

My sincere gratitude goes to my thesis supervisor Aleksei Tepljakov (Ph.D.) for his priceless advice, instructions, suggestions, guidance and contributions towards successful completion of this thesis. I must equally express my gratitude to all my lecturers for their academics support during my stay in the university.

My sincere appreciation goes to my parents, Mr & Mrs Biodun Olukotun for their love, care, passion and affectionate. I will ever live to appreciate your parental care on me. Cheers to my sisters (Yemisi, Helen and Blessing). To you all, I say thank you for always being there for me. Also, big thanks to all friends who gave supports in one way or another, may the God bless you in all areas.

Conclusively, I give credit and due acknowledgement to all those whose works were cited in this thesis.

Nomenclature

AR	Augmented Reality
VR	Virtual Reality
MR	Mixed Reality
PID	Proportional Integral Differential
UDP	User Datagram Protocol
TCP	Transmission Control Protocol
DTLS	Datagram Transport Layer Security
IPSec	Internet Protocol Security
NTP	Network Time Protocol
DNS	Domain Name System
MATLAB	Matrix Laboratory
D4AR	4-Dimensional Augmented Reality
DT	Digital Twin
BIM	Building Information Modelling
SCADA	Supervisory Control and Data Acquisition
MES	Manufacturing Execution System
RTU	Remote Terminal Unit
DCS	Distributed Control Systems
PLC	Programmable Logic Controllers
HPU	Holographic Processing Unit
HMD	Head Mounted Display
CPU	Central Processing Unit
GPU	Graphical Processing Unit
SAR	Spatial Augmented Reality
ECG	Electrocardiography
SDK	Software Development Kit
GPS	Global Positioning System
SLAM	Simultaneous Localization and Mapping
SP	Set Point
PV	Process Value
HDMI	High-Definition Multimedia Interface
USB	Universal Serial Bus

Contents

List of Figures

List of Tables

1. Introduction

This section gives an introduction to the entire work done in this thesis. It contains the background of the thesis, the problem statement, the motivation driving the thesis, the aim and objectives of the thesis, research methodology adopted in the thesis and lastly, the outline of the thesis.

1.1. Background

Imagine a technology with which you could see more than others see, hear more than others hear, and perhaps even touch, smell and taste things that others can not. What if we had technology to perceive completely computational elements and objects within our real world experience, entire creatures and structures even that help us in our daily activities, while interacting almost unconsciously through mere gestures and speech [2]. AR technologies offer an enhanced perception to help us see, hear, and feel our environments in new and enriched ways that will benefit us in fields such as education, maintenance, design, reconnaissance, to name but a few [2].

In recent years, the growth of Augmented Reality applications can be attributed to solutions that allow people (who use a product or service) to visualize products and imagine what it might be like owning the product or experiencing the service before they actually buy it [3]. This is because AR provides seamless interaction between real and virtual environments, the ability to enhance reality, the presence of spatial cues for face-to-face and remote collaboration, support of a tangible interface metaphor, the ability to transition smoothly between reality and virtuality [4].

In 2016, Microsoft Hololens Developer Kit and Meta 2 Developer Kit were launched, also, AR and VR investment reached $1.1 billion. The following year, ARKit was launched by Apple, and Google released ARCore for Android, both powerful tools for developers to create AR apps. It is predicted that there will be 1 billion augmented reality users and the total value of the AR market is expected to reach $100 billion by 2020 [3]. This is because AR apps, headsets, and smart glasses hold the promise to add value to virtually every industry and as AR technology becomes more sophisticated and the cost saving and business applications expand, the demand and investment in AR will increase [3].

1.2. Problem Statement

SCADA is the conventional method for locally or remotely monitoring and controlling industrial systems on the shop floor and the HMI helps to visualize data but the entire process can become rigorious for the control engineer or error-prone as automation data is isolated from the industrial systems [5]. The thesis provides an efficient and modern way of monitoring and visualizing factory telemetry using the Meta 2 Head Mounted Display (HMD). The ability to immersively interact in real-time with the real system while visualizing process data, troubleshooting, and sending control signals is an advantage over the conventional SCADA. Parameters of the real industrial control system including the tuning of controllers can be changed remotely by manipulating the controls in the AR application deployed in the Meta 2 Headset.

Most work done applying HMD to industrial automation adopt the Microsoft Hololens as a preferred HMD because of its strong spatial tracking capabilities, but the Meta 2 HMD was actually created to replace the 2D desktop displays, and so provides a 90-degree FoV while the Microsoft Hololens retain a 30-degree FoV. This makes the Meta 2 HMD better than the Hololens HMD for this thesis as it will give us a clear line of sight to the industrial system and its environment while using a unique neuroscience-driven interface to access and interact with digital contents easily and naturally providing us with the best immersive AR experience.

The Multitank laboratory model is a laboratory model of a real-life industrial Multitank system used for teaching and researching automatic control in the laboratory [6]. Real-time experiments can be performed on the system design to validate one's linear and nonlinear control methods. The objective of the control is to reach and stabilize the level in the tanks by an adjustment of the pump operation or/and valves settings. This control problem can be solved by several level control strategies ranging from PID to adaptive and fuzzy logic controls, but the PID controller is used in this thesis [7].

1.3. Motivation

The work done in this thesis was motivated by the advancement in augmented reality technologies, especially when integrated with other system. AR shines when applied to the monitoring and control of industrial systems, it provides a modern way of controlling these systems and subsequently help to reduce the downtime of industrial systems.

- Presently, HMI (Human-Machine Interface) technology is used by almost all industrial organizations for process monitoring and data tracking of industrial equipments. Now, with AR, operators can work more efficiently by immersely interacting with the real system while visualizing process data and troubleshooting.
- Training and service materials are time-intensive to produce and maintain. Workers only have access to inaccurate and difficult to understand service information. The AR provided in the Meta 2 headset provides workers with a more effective way to experience technical product information for training, maintenance, and service purposes.

1.4. Aim and Objectives

The aim of the thesis is to design and develop a Meta 2 headset application with it's unique virtual object for complementing the control of a real-life laboratory model of an industrial multi-tank system.

To achieve the design and development of this project, the following were done:

- Review past works that are related to the project.
- Choose suitable components that meet the scope of the project.
- Design the system diagram to integrate various part of the project.
- Create the virtual model
- Develop the required scene
- Develop a closed loop control system to send control signals to the Multitank system
- Establish communication between the computer hosting Unity3D and the computer hosting the Simulink

- Integrate the model, software package and the AR device
- Testing and debugging.

1.5. Research Methodology

For the purpose of this thesis, the methodology of the research is conducted in two parts—reviewing literature and development. A significant part of the thesis was dedicated to literature review on various resources such as articles, books, conference papers, journals and thesis that are related to this project, this was necessary to attain a good understanding for establishing the scope and limitation of the thesis. The development life cycle incudes continuous design, programming, simulation and testing.

1.6. Outline the of Thesis

This thesis consists of five chapters and a brief description of each of the chapters is given thus;

Chapter One is the introductory chapter which introduces the problem being studied and indicates its importance and validity. Other information presented in this chapter includes problem definition, main aim and objectives, methodology employed to solve the problem, and the thesis outline. Chapter Two consists of the literature review which includes review of past works and Chapter Three consists theoretical background of various components and technologies used in the thesis.

Chapter Four is the design and implementation chapter. The chapter is dedicated to system design, simulations and calculation relating to the project. The latter part of the chapter describes the development process; software development, software and hardware integration. Chapter Five is the integration, tests, results and discussion chapter. This chapter describes steps taken to test the developed project. Data are collected and analyzed and the result is compared to the project aim and the result of other studies.

Chapter Six is the conclusion and recommendation chapter. The conclusion summarizes the highlights of the previous chapters and it briefly mentions the findings of the project. It also clearly states the contributions of the work. On the other hand, recommendations section states the constraints of the study and recommendations for other possible future work.

2. Literature Review

Over the recent years, a large amount of work has been done in the field of AR, particularly on the industrial applications of AR using either mobile devices or head-mounted display, some of the past works are reviewed in this chapter, to determine the past and the current state of the technology.

Saleh Kaji et al. (2018) published a paper on an advanced overview of utilizing Augmented Reality (AR) in Smart Cities. Firstly, they recognized that they are six major areas of smart cities but have decided to prioritize three of these areas (Environment, Living and Mobility) as they have more niche for the application of Augmented Reality technology. They have further divided these three areas of smart cities into ten components (Environment- museum, design, construction, air pollution. Living-suggestion, equity, maintenance. Mobility- outdoor, indoor, outdoor-indoor). In their review, they try to collect all articles that are innovative, have industrial perspectives, related to the prioritized areas, and published between 2010 and 2017. They used the conjunction of "Augmented Reality" and "Smart cities" as a keyword to search for these articles in well-known science research databases. They analyzed the articles and try to quantatively determine which area has receive more attention from researchers. The result was that Enviroment section has received attracted more researchers with 61 articles focussing on this area. The Living section has a lower attention from researchers with 32 articles and the area that gained the least attraction from reasearchers is Mobility issues with just 16.3% of the articles reviewed focusing on this area. They went further to categorize the articles into the various components without consideration of the categorization in the main category. They found out that musuem and maintenance components account for 40% of the attention from researchers due to their economic benefits. They concluded that alot of the researchers still try to show proof of concept rather than implemeting the application in a real urban environment and that little is still known regarding some of the reasearch areas [8].

Nayan Kadam, Bharath Chandra Reddy, and Vadde (2017) researched on the Visualization of Industrial Robot with Augmented Reality in their Master's thesis. They worked on demonstrating AR toolkit on the android and windows platform, to design an AR app that can be used to control a virtual object which in their case is a KUKA robot. The KUKA robot was designed using the 3D creation software called blender, and the 3D model is deployed in the Unity game engine. The Unity game engine was integrated with the Vuforia engine to achieve the aim of their work. They reported that the motivation behind their work is to provide humans with seamless ways to control and operate industrial robots without long training period, that even though industrial robots have become more and more autonomous they still need human's help to perform tasks in the real-world. Part of their research work was to analyze various game engines and image recognition technologies. They found the Unity game engine and the Vuforia engine to be the best out there for numerous reason, but their choice was mainly influenced by the fact that Unity is a free software and supports deployment to numerous platforms such as iOS, Android, Windows e.t.c. They explained extensively how to install and integrate the Vuforia Image recognition capability with Unity game engine. Their final product was an android based application from which the virtual model of the six axes KUKA robot can be visualized and cotrolled. They hope to later map the vrtual robot with a real-life industrial robot [9].

R.C Tarca et al. (2015) presented a paper on "A Telerobot Control System Using Augmented Reality". The paper showed how the application of Augmented Reality to the control of a Telerobot improved the performance of the telerobot system without adding or changing its components or features. The telerobot system comprises of a Mitshubishi Movemaster RV-2AJ robot with 5 axes, a server, local and remote clients. The system obtains the scene image with the objects placed in it workspace using a web camera and calibration was made to enhance the precision and usability of the robot, they implemented a specific protocol over IP to communicate between the server and the client. The application of AR was used to map the 2-dimensional coordinates of the acquired image to 3-dimensional in the real-world after the edge extraction of the acquired image. The method involves superimposing edge-extracted image with the real scene in which the robot acts. Their work nonetheless only took advantage of the visualization and calibration benefits of augmented reality, they propose to later work on using haptic gloves and head-mounted display to command and control the process [10].

Mani Golparvar-Fard (2010) researched on 4-Dimensional Augmented Reality- Models for Automation and Interactive Visualization of Construction Progress Monitoring in his Doctoral thesis. He applied Augmented Reality to improve the performance in construction activities. He was motivated with the objective of detecting ongoing or potential problems with performance in construction activities inorder to either prevent or eradicate these problems. He reported that to improve performances in construction activities, the project manager has to be able to identify in real-time the difference between “as-built” and “as-planned” performances and for this to be possible, the project manager should be able to collect and process data easily and able to present it in format that is accessible to all project paticipants. The research addressed this challenge by presenting the necessary theoretical and mathematical knowledge for generating an augmented reality imagery that combines unorderd construction sites photographs with Building Information Models (BIM). The 4D Augmented Reality environment generated in this work represents progress monitoring metrics as a joint representation of as-built and as-planned information for automated progress tracking and visualization of construction performance deviations. During the research, the 4-Dimensional Augmented Reality models were generated for seven ongoing projects ranging up to $326 million. D4-AR models allow project manager, supervisors, owner and even architect to remotely visualize the as-built scene by browsing through the collection of photographs, visualize performance deviations gnerated by the comparison of the as-build data and the as-planned data [11].

Simone Garagnani and A.M Manferdini (2011) presented a paper on Virtual and Augmented Reality Applications for Cultural Heritage. It is important to note that the research was done before the advent of technology like Vuforia and Wikitude, which could have been used to achieve many things that were performed manually in this work. They researched on ways to apply VR and AR to share information about cultural heritage such as archaeological artefacts which institutions such as Museum are made to promote and protect especially as digital technologies have advanced over the years in collecting real-time data and reproduce good copies of them. For their case study, they picked the Situla Arnoaldi bronze, a structurally complex archaeological artefact dated to the full 500 B.C which is preserved in the Archaelogical Museum of Bologna. The laser scanner is used to obtain quality colour and radiometric data and apply them to generating the 3D geometry and used 3D Resharper to produce quality results regardng the specific allignment between 3D model and the project Image. They chose the Universal 3D file format to save and share the resulting digital content beacuse it is cross-platform and can be easily shared online. The final product of their research was the visualization of Situla Arnoaldi in virtual reality environment. They propose to apply Augmented Reality to the same application, where users could easily get information while interacting with them and their components [12].

Vladimir Kuts et al. (2019) made a industrial robotics targeted research on Digital Twin Based Synchronised Control and Simulation of the Industrial Robotic Cell using Virtual Reality. The aim of their research is to enable universal control algorithms through Virtual Reality experience which can be easily modified for a wide range of industrial equipments. They built a Digital Twin (DT) of industrial robot - Motoman GP8 in VR. They used the Unity3D, 3DS Max and Maya for developing the desired virtual reality environment and the HTC Vive head-mounted display was used to visualize the designed environment. The final product of the research is the VR software deployed to the HTC Vive HMD enabling synchronisation model of the real industrial robot and the virtual robot. The work can be integrated with any industrial robot, and any platform that can display augmented reality software due to the modular nature of the development. They concluded after much experimental and practical testing that the DT concept is a practical viable solution that can drive the monitoring and control of industrial system in the future [13].

T.P. Caudell and D.W. Mizell (1992) published Augmented Reality: An Application of Heads-Up Display Technology to Manual Manufacturing Processes. In this article, Caudell and Mizell pinned the difficulty in automating the manufacture and assembly of modern aircraft to two main reasons: (1) the small lot size of parts and (2) robots still need human's help to perform tasks in the real-world as they cannot boast the skills required in the real world. Seeing that fabrication and assembly in the aerospace industry are still dominated by complicated, human involved processes, Caudell and Mizell introduced a "see-thru" virtual reality goggle to augment the factory worker's visual field of view with useful and dynamically changing information to improve the efficiency and performance of manufacturing activities. The technology that provides this access was tagged as HUDset by Caudell and Mizell, which means Heads-Up Display. The mentioned main objectives of their research are to improve the precision by which a virtual content can be overlayed onto the real-world space, and extend the radius from which the digital content can sense the user head position and orientation [14].

In Conclusion, as technology continues to advance over the years the application of AR to the industrial and manufacturing processes have increased. Many rsearchers adopt VR to remotely control industrial systems and robot by designing a DT of the industrial equipment, but this method requires that there is a mapping between the DT and the real industrial robot or equipment. This mapping can in some case be difficult to implement and may cause problems in the future.

3. Augmented Reality, Industrial Control System and User Datagram Protocol

The solutions that drive innovation in manufacturing are increasingly data-centric. In this age of digital transformation, value is established by creating and capturing data, deriving insights from data, or executing complex automated tasks based on data. Augmented reality (AR) has emerged as a powerful new tool to bridge the gap between the digital and real worlds for assemblers, operators, and technicians. By adopting AR solutions for employee training and guidance applications, manufacturing organizations empower their factory workers with information [15]. This helps to:

- Optimize asset and personnel performance
- Reduce costs
- Ensure product quality
- Increase on-time delivery

3.1. Different Types of Immersive Technology

3.1.1. Virtual Reality

Virtual reality (VR) is a powerful and interactive technology that changes our life unlike any other. Virtual reality, which can also be termed as immersive multimedia, is the art of simulating a physical presence for the audience in places both real and imaginary. It usually involves two senses namely sight and sound. The key property that distinguished VR from all previous media types is “presence”. Presence is the psychological sense of being there, of actually being immersed in and surrounded by in the environment [16].

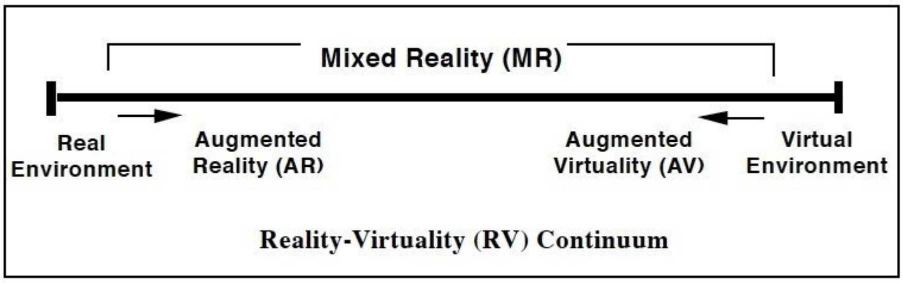


Figure 1. Immersive Technology Spectrum

3.1.2. Augmented Reality

Augmented Reality (AR) is a real-time visualisation technique involving direct or indirect view of a physical real-world environment that has been enhanced by adding computer-generated information to it. AR requires a device to function as a mean of visualisation, the device directly affects the immersion level in any augmented reality application. Milgram's Reality-Virtuality Continuum is defined by Paul Milgram and Fumio Kishino as a continuum that spans between the real environment and the virtual environment comprise Augmented Reality (AR) and Augmented Virtuality (AV) in between, where AR is closer to the real world and AV is closer to a pure virtual environment, as seen in Figure 1 [17]. Augmented Reality aims at simplifying the user's life by bringing virtual information not only to his immediate surroundings, but also to any indirect view of the real-world environment, such as live-video stream. AR enhances the user's perception of and interaction with the real world [18].

3.1.3. Mixed Reality

Mixed reality (MR) is the merging of real and virtual worlds to produce new environments and visualizations, where physical and digital objects co-exist and interact in real time. Mixed reality does not exclusively take place in either the physical or virtual world, but is a hybrid of reality and virtual reality, encompassing both augmented reality and augmented virtuality via immersive technology [17]. Mixed reality is often termed as the combination of virtual reality and augmented reality which is misleading. Mixed reality is simple an upgraded version of augmented reality where interactions are prioritized.

3.2. Types of Augmented Reality Applications

3.2.1. Marker-based Augmented Reality

Marker-based augmented reality (also called Image Recognition AR) uses a camera and some type of visual marker, such as a QR/2D code, to produce a result only when the marker is sensed by a reader. Marker based applications use a camera on the device to distinguish a marker from any other real-world object. Distinct, but simple patterns (such as a QR code) are used as the markers, because they can be easily recognized and do not require a lot of processing power to read. The position and orientation are also calculated, in which some type of content and/or information is then overlaid the marker [19]. Figure 2 shows a user using a QR code to distinguish the target from other area in the real world.

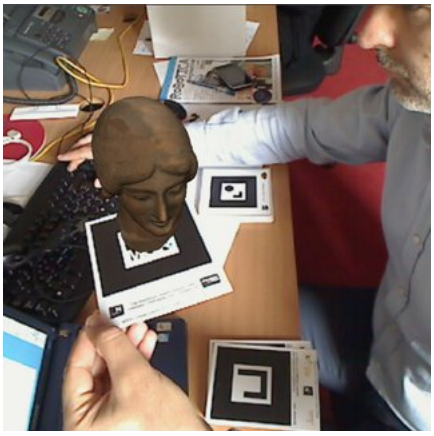


Figure 2. Marker-based AR used to overlay a model on a marker

3.2.2. Marker-less Augmented Reality

As one of the most widely implemented applications of augmented reality, marker less (also called location-based, position-based, or GPS) augmented reality, uses a GPS, digital compass, velocity meter, or accelerometer which is embedded in the device to provide data based on your location. A strong force behind marker less augmented reality technology is the wide availability of smartphones and location detection features they provide. It is most commonly used for mapping directions, finding nearby businesses, and other location-centric mobile applications [19]. Figure 3 represent an example of marker-less AR, outling the road side by using location based AR to overlay digital content on the side of the road.

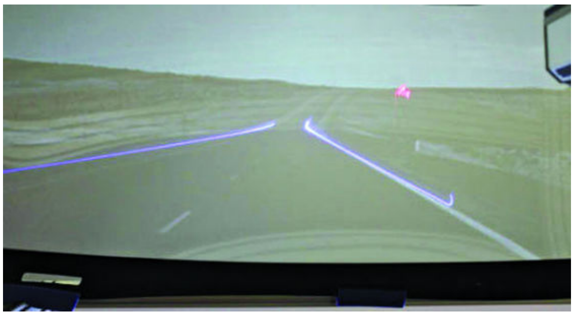


Figure 3. Outlining the road using location based AR can help save drivers from accidents

3.2.3. Projection based Augmented Reality

Projection based augmented reality works by projecting artificial light onto real world surfaces. Projection based augmented reality applications allow for human interaction by sending light onto a real-world surface and then sensing the human interaction (i.e. touch) of that projected light. Detecting the user's interaction is done by differentiating between an expected (or known) projection and the altered projection (caused by the user's interaction). Another interesting application of projection based augmented reality utilizes laser plasma technology to project a three-dimensional (3D) interactive hologram into mid-air [19]. Figure 4 shows the projection of light as a phone dialer on a user palm.

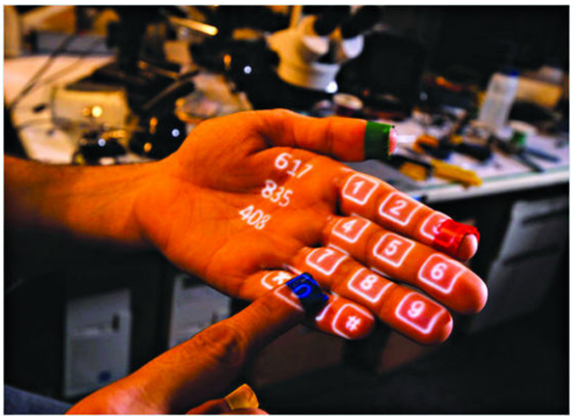
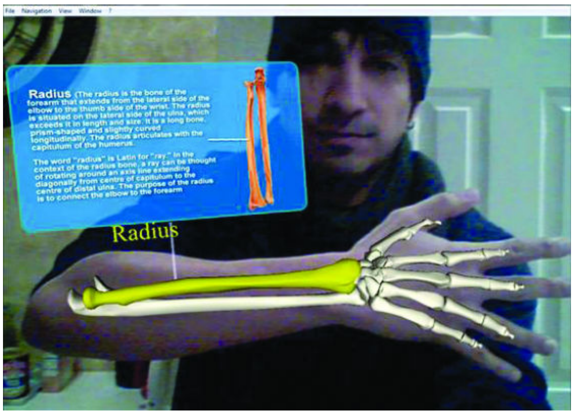


Figure 4. Using projection-based AR to create a dialer in your hand

3.2.4. Superimposition-based Augmented Reality

Superimposition based augmented reality either partially or fully replaces the original view of an object with a newly augmented view of that same object. In superimposition based augmented reality, object recognition plays a vital role because the application cannot replace the original view with an augmented one if it cannot determine what the object is. A strong consumer-facing example of superimposition based augmented reality could be found in the Ikea augmented reality furniture catalogue. By downloading an app and scanning selected pages in their printed or digital catalogue, users can place virtual idea furniture in their own home with the help of augmented reality [19]. Figure 5 is an example of superimposition-based AR where a part of a user hand is replaced with digital content.



Radius (The radius is the bone of the forearm that extends from the lateral side of the elbow to the thumb side of the wrist. The radius is situated on the lateral side of the ulna, which exceeds it in length and size. It is a long bone, prism-shaped and slightly curved longitudinally. The radius articulates with the capitulum of the humerus.

The word "radius" is Latin for "ray." In the context of the radius bone, a ray can be thought of rotating around an axis line extending diagonally from centre of capitulum to the centre of distal ulna. The purpose of the radius is to connect the elbow to the forearm

Figure 5. Superimposition-based AR used for educational application

3.3. Augmented Reality Hardware

This section focuses on the hardware aspect of the technology needed to support an AR application. AR hardware devices requires some special core capabilities and it is important to note that the capabilities of these hardwares are changing rapidly. One can see that there is a constant progression towards faster, smaller and more importantly less expensive hardwares that can be deployed to develop AR applications [20].

3.3.1. Displays

There are various kind of displays, but in this section we would discuss the Head-Mounted Display (HMD), the Handheld display and the spatial display. Table 1 compares the various kinds of displays used in AR.

In AR, head-mounted displays are devices attached to the user head that take the form of an helmet or a glass which allows the user to see the real world and project the virtual object or environment into the real world. It is important that the user are able to see the real for HMD used in AR, this is the main difference between VR HMDs and AR HMDs. There are also various didtinctions between HMDs used in AR.

- The medium through which the device allows the user to see the real world. Some uses optical means and others utilize video technology.
- Whether they display on both eyes or a single eye. If they display on both eyes, whether they display the distinct visual on both eyes to achieve stereoscopic 3D viewing.

Handheld dispays refer to small computing devices with a display that the user can hold and use in thier hands. The handheld is currently the most common type of display used for AR applications. The reason is due to the widespread adoption of smartphones which already possess a display, processor and a sensor. They have the added advantage of portability, as they can easily fit into the user's pocket. Handheld displays use either panels such as in smartphone, or a mobile projector to display.

Spatial Augmented Reality (SAR) uses video projectors and optical elements, holograms, and other technologies to provide graphic information directly to the physical objects, without requiring the user to wear or hold the display [21].

Table 1 shows comparison of various types of displays. In the table we compared three types of displays: HMD, Handheld and Spatial. We reviewed what type of technology use the various displays and the advantages and disadvantages of the various displays.

Table 1. Comparison of different types of displays [1]

Type	HMD	HMD	Handheld	Spatial	Spatial	Spatial
Technology	Video transparent	Optical transparent	Video transparent	Video transparent	Optical transparent Direct supplemen	Direct supplement
Advantages	Visual control, synchronization of the virtual and real environments, waiting video processing from cameras	Half silver mirror technology, the natural perception of the real environment	Portable, powerful, combination of many sensors	Price, can be adapted to the standard equipment	The most natural perception of real objects	Display directly on the surface of a physical object
Disadvantages	It is necessary to fix the device on the head, unnatural perception of real world	The time delay, jitter of virtual elements	A small display, weight (for tablets)	It does not support mobile systems	It does not support mobile systems	Do not depend on the user (everyone can see the same thing)

3.3.2. Sensors

Sensors take information from the environment and provide that information to AR application. Sensors can provide information like the environment temperature, pressure, light. However, the primary purpose of sensors in AR application is to provide location information to enable the application to determine the location and orientation of different things in the real world.

Camera is a sensor used for optical tracking, which is the most tracking method used for AR applications. The camera gathers light through a lens and provides a signal that represents an image of what the camera "sees." Tracking information is derived from the image by analizing it. There are lot of cameras that are used in AR system, ranging from smartphone cameras, tablet cameras, webcams, and industrial cameras that can be installed into an AR system depending on the kind of application. Other tracking devices are GPS, accelerometer, compass.

Optical tracking are widely used because many devices already have cameras preinstalled into them, and very good cameras that provide digital signal are readily available for inexpensive amount. One major disavantage of using optical tracking is that there need to be enough light in the real world for the camera to provide a good image for the computer vison software used on the computer or on the HMD. Another issue is that there must be a clear line of sight between the camera and the real world and the object being tracked [20].

3.3.3. Computers

Computers used in AR application should have a fast processor and enough video memory. The processors receive signals from the sensors, perform the instruction required and send the signal that provide display of the system. The processing system in AR applications consists primarily of one or more general-purpose microprocessors as the central processing unit (CPU) and perhaps one or more special-purpose graphics processing units (GPUs). GPUs are hardware specially optimized for executing threedimensional graphics computations [20].

Some of the most common architectures for the processing of AR applications are shown in Table 2.

Table 2. The capabilities of several architectures available for AR systems

	Handheld	Handheld with Server	Desktop/Laptop	Desktop/Laptop with Server	Web Based	Cloud Based	Combination
Portable	Yes	Yes	No	No	No	No	Maybe
Portable with network	Yes	Yes	No	No	Yes	Yes	Maybe
Sufficient computation alone	Maybe	Yes	Yes	Yes	Yes	Yes	Maybe
Requires network	No	Yes	No	Yes	Yes	Yes	Maybe
Cross-platform solution	No	No	No	No	Maybe	Maybe	Maybe
Latency important	Maybe	Yes	No	Yes	Yes	Yes	Maybe

3.4. Augmented Reality Visualization Devices

Nowadays there are many devices used to visualize AR application ranging from a simple smartphone to state of the art technology such as the Microsoft Hololens 2. In this section, some of the most popular AR Visualisation devices are briefly described;

3.4.1. Smartphone

Smartphone is one of the easiest devices to implement your augmented reality solution without worrying about scalability. The augmented reality application accesses your smart phone camera to run image process and deep learning algorithms, giving the smartphone ability to superimpose computer generated information in the physical world. Figure 6 shows a smarphone been used to monitor the temperature of different parts of an industrial machine.

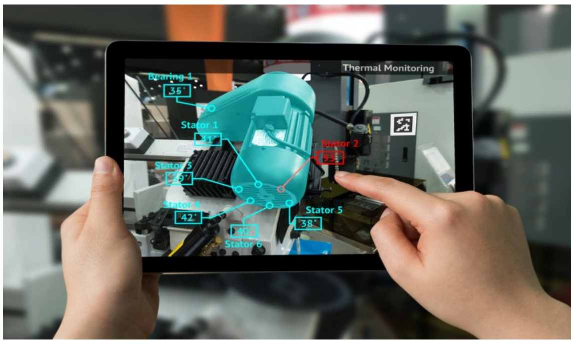


Figure 6. A smartphone used to visualize AR application

3.4.2. Microsoft HoloLens

Microsoft HoloLens this is one of the best ever discovery done in augmented reality field, the device superimposes the virtual world in real world so that one can interact with virtual object in real world instinctually. As a standalone headworn computer capable of projecting images that appear to be in the real world, the HoloLens is the first of its kind [22]. HoloLens has sensor to capture information about what action a person is doing and environment around that person. It uses a high-definition lenses and advanced optical projection system to generate multi-dimensional full-colour images with very low latency so that holograms can be seen in real world. Custom holographic processing unit HPU is custom silicon that processes a large amount of data/information per second from the sensors. Microsoft HoloLens understands gestures, gaze, voice and map the world around you, all in real time. It also has camera so that it can detect object in real world and It then uses that information to project 3D images on top of and even inside them and also has inbuilt speaker [23]. The Microsoft hololens is shown in Figure 7.

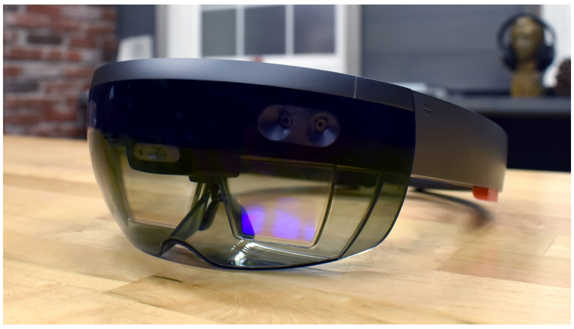


Figure 7. The Microsoft Hololens 1

3.4.3. Magic Leap One

The Magic Leap One is a futuristic pair of augmented reality (AR) glasses, designed to enhance your world with digital objects and images while still letting you interact with everything real that's going on around you. Magic leap one is a bridge between the Microsoft Hololens and the Meta 2, a small cable connects the head-mounted display to a small computer positioned on the hip of the user. This computer handles the data and graphics processing. The headset is much smaller than the hololens and meta 2, as a result can not accomodate people who wear corrective lenses [22]. The Magic Leap One is shown in Figure 8.

3.4.4. Google Glasses

The google glasses is a minocular optical see-through head-mounted display, designed to display smartphone-like information on a screen whilst allowing users to remain hands-free. It has demonstrated usefulness in various fields including surgery, electrocardiography (ECG) monitoring and as a teaching and training tool [24]. The glass can be interacted with using gestures, touchpad and voice recognition. The

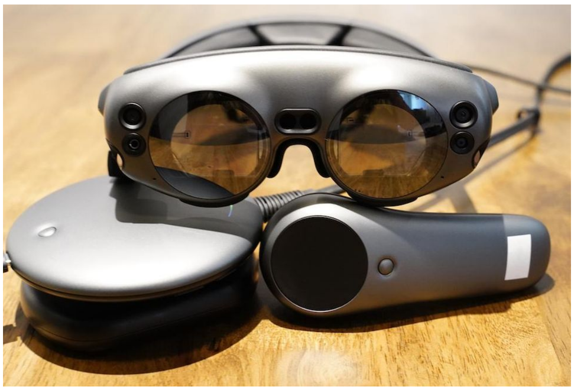


Figure 8. The Magic Leap One

wearer can swipe through applications installed on its android operating system by intracting eith the touchpad placed on the side of the glass . The entreprise edition of the Google glass is shown in Figure 9.

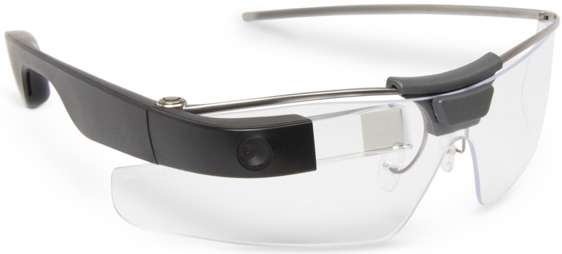


Figure 9. The Google Glass Enterprise Edition

3.4.5. Nreal Light

The Nreal Light is a lightweight mixed reality glass, it weighs around 88g with USB-C compatibility. It comes with a 52-degree field of view (FOV) and 6D of tracking, plane detection, image tracking, supports simultaneous localization and mapping. The nreal light comes with a 23g 3Dof controller and houses 170g android computing unit [25]. The Nreal Light mixed reality glass is shown in Figure 10.

3.4.6. Meta 2

Meta 2 uses a unique neuroscience-driven interface to access, manipulate and share digital information easily and naturally [26]. One difference between the Meta 2 and other head-mounted displays like the Microsoft Hololens and the Magic Leap One is

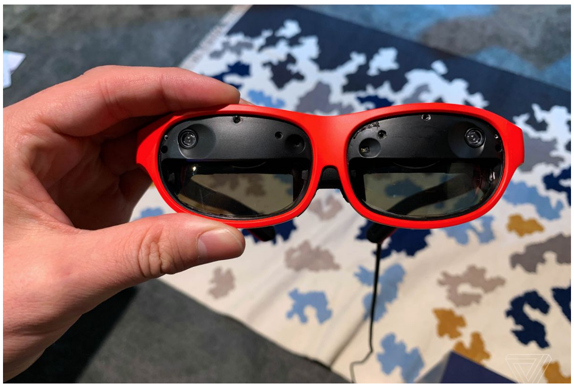


Figure 10. The Nreal AR Glass 3.0

that the Meta 2 requires a connection to a computer, so it is referred to as a tethered, head-mounted device. It was built by its creator as a replacement for the traditional 2D PC screen, it plugs into the PC HDMI port , allowing sensor input and image processing to be handled by the connected PC's video graphics and processors [22]. Figure 11 shows the Meta 2 head-mounted display.

Recommended PC Requirements [27]

- Graphics: NVIDIA GTX 960 / AMD R9 280
- CPU: Intel Core i7 (desktop CPU)
- USB Ports: USB 3.0
- OS: Windows 8.1 64-bit or newer
- Memory: 8GB RAM
- Storage: 10GB
- Video: HDMI 1.4b
- Sound Card: Intel HD-compatible sound card
- Game Engine: 64-bit Unity 5.3x

Figure 11. The Meta 2 Headset by Meta

3.5. Industrial Control System

An industrial control system (ICS) is an information system used to control industrial processes such as manufacturing, product handling, production, and distribution [28]. ICSs include supervisory control and data acquisition (SCADA) systems, distributed control systems (DCS), and programmable logic controllers (PLC) [29]. ICS are typically found in the electric, water, oil and gas, chemical, pharmaceutical, pulp and paper, food and beverage, and discrete manufacturing (automotive, aerospace, and durable goods) industries as well as in air and rail transportation control systems [28]. Presently, Industrial control systems (from sensors to SCADA, MES) are connected to Internet for various purposes such as data exchange for business-oriented plant operations and remote maintenance [30]. Figure 12 shows how data are excahnged between the contol room and the plant.

3.5.1. History

Process control of large industrial plants has advanced through various stages. At first, control was panel local to the process plant. This result in the need for personnel to monitor the distributed panels, and there was a lack of an overall view of the process. Later, an ever-manned control room was used to monitor all incoming plant measurements, and exclusively transmitted all automatic and manual control outputs back to the plant in the form of pneumatic or electrical signals. This result in the decreased deploration of manpower and a centralized view of the entire process [31]. While there is the advantage of a central control center, this arrangement was not so

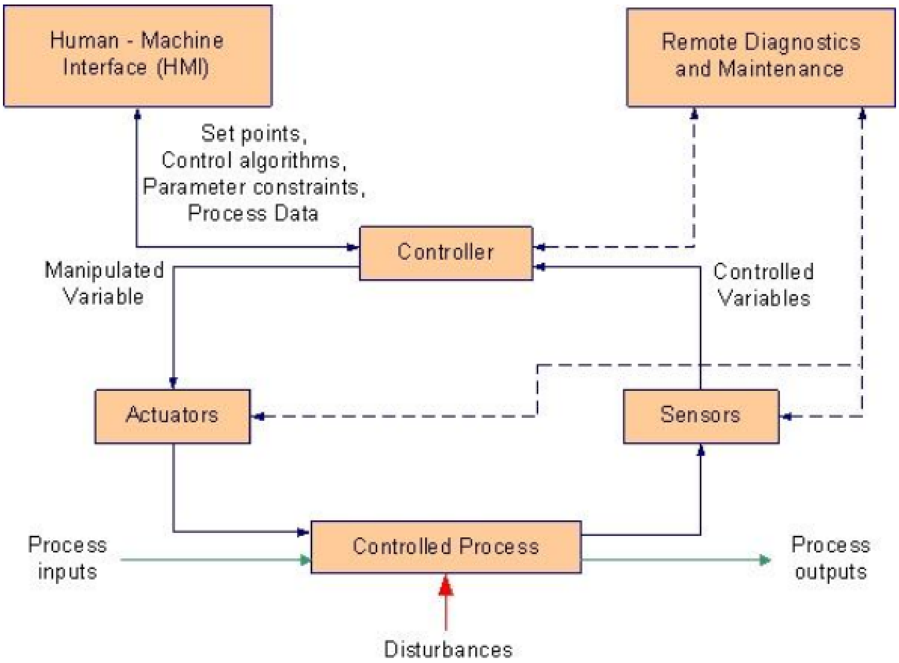

Human - Machine Interface (HMI)
Remote Diagnostics and Maintenance
Set points, Control algorithms, Parameter constraints, Process Data
Controller
Manipulated Variable
Controlled Variables
Actuators
Sensors
Process inputs
Controlled Process
Process outputs
Disturbances

Figure 12. Industrial control system operation

flexible as each control loop as its own controller hardware. So any changes in the system framework requires reconfiguration of signals by re-piping and re-wiring. Also, for the operator to monitor the full process there need to be ceaseless movement in the large control room. These discrete controllers were later replaced with a computer software hosted on a network of input/output racks with their own processor. This was made possible with advancement in electronics processors, high speed electronic signalling network and electronic graphic display technology [31].

DCS was born out of the need to avoid the rigidity of the former set-up in large continuous industrial processes, which was created from establishing a central location for all controllers hardwares. DCS was adaptable to changes made, it aid the interconnection and the re-configuration of plant controls such as cascaded loops and interlocks, and connecting with other production devices. Also, control racks are now located on the field, in this way, there is less cabling runs. The racks of relay amd timer used for event driven control were usually very difficult to troubleshoot and reconfigure, this problem drove the development of PLC especially in indiustries that were mainly involved in combinational and sequential logic control [31].

The focus of a SCADA is data acquisition and the presentation of a centralised Human Machine Interface (HMI), although they also allow high level commands to be sent through to control hardware [31]. SCADA systems use open-loop control with sites that are widely separated geographically. A SCADA system uses remote terminal units (RTUs) to send supervisory data back to a control center [32]. Figure 13 shows the indutrial control levels with DCS (including PLCs or RTUs) operate on level 1. Level 2 contains the SCADA software and computing platform.

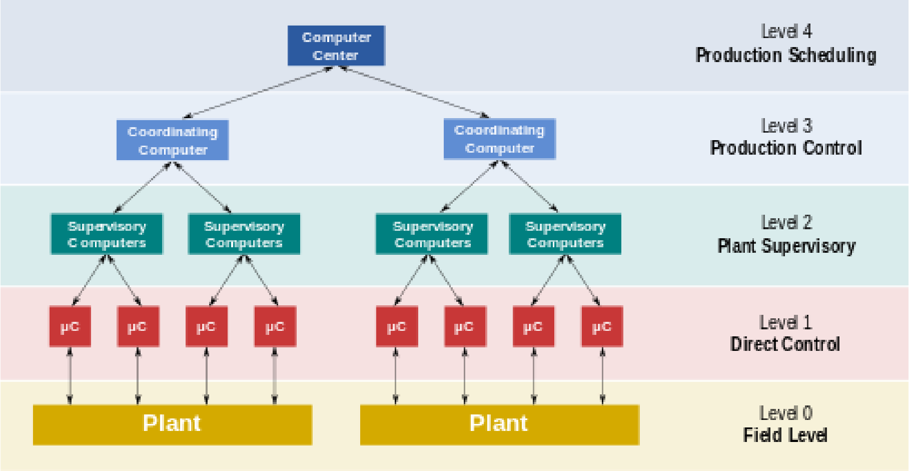


Figure 13. Functional manufacturing control levels

3.5.2. PID Controller

PID controllers are found in a wide range of applications for industrial process control. Approximately 95% of the closed loop operations of industrial automation sector use PID controllers. PID stands for Proportional-Integral-Derivative. These three controllers are combined in such a way that it produces a control signal. Proportional or P- controller gives output which is proportional to current error $e(t)$. It compares desired or set point (SP) with actual value or feedback process value (PV). The resulting error is multiplied with proportional constant to get the output. If the error value is zero, then this controller output is zero [33].

Due to limitation of p-controller where there always exists an offset between the process variable and set point, I-controller is needed, which provides necessary action to eliminate the steady state error. It integrates the error over a period of time until error value reaches to zero. It holds the value to final control device at which error becomes zero. Integral control decreases its output when negative error takes place. It limits the speed of response and affects stability of the system. Speed of the response is increased by decreasing integral gain K_i. I-controller doesn't have the capability to predict the future behavior of error. So it reacts normally once the set point is changed. D-controller overcomes this problem by anticipating future behavior of the error. Its output depends on rate of change of error with respect to time, multiplied by derivative constant. It gives the kick start for the output thereby increasing system response [33].

Figure 14 shows that the output of a PID Controller, which is equal to the control input to the plant, is calculated in the time domain from the feedback error as follows:

$$u(t) = K_p e(t) + K_i \int e(t)dt + K\frac{de}{dt} \tag{1}$$

The transfer function of a PID controller is found by taking the Laplace transform of Equation (1).

$$K_p + \frac{K_i}{s} + K_d s = \frac{K_d s^2 + K_P s + K_i}{s} \tag{2}$$

where K_P= proportional gain, K_i= integral gain, and K_d= derivative gain.

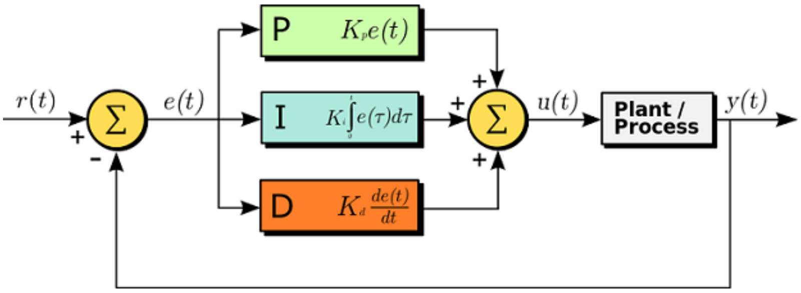


Figure 14. Working of PID Controller

The general effects of each controller parameter (K_P,K_i ,K_d) on a closed-loop system are summarized in Table 3. Note, these guidelines hold in many cases, but not all. If you truly want to know the effect of tuning the individual gains, you will have to do more analysis, or will have to perform testing on the actual system[34].

Table 3. Summary of the effects of each controller parameter on a closed-loop system

CL Response	**Rise Time**	**Overshoot**	**Settling Time**	**Steady state error**
K_p	Decrease	Increase	Small change	Decrease
K_i	Decrease	Increase	Increase	Decrease
K_d	Small Change	Decrease	Decrease	No change

3.6. User Datagram Protocol (UDP)

UDP is a primary member of the internet protocol suite, referred as UDP/IP suite [35]. It is a communication protocol that is mainly used to form a low-latency and loss-tolerating connection between hosts on an Internet Protocol (IP) network. It is unreliable and connectionless protocol [36]. So, there is no need to establish connection prior to data transfer. As a result, UDP is beneficial in time-sensitive communications, including Voice over Internet Protocol (VoIP), domain name system (DNS) lookup, and video or audio playback. UDP is an alternative to Transmission Control Protocol (TCP) [37] which has the ability to break large data sets into individual packets, check for and resend lost packets, and reassemble packets in the correct sequence [36]. UDP has features that make it beneficial for use with applications that can tolerate lost data. For example:

- It permits packets to be lost and found in different order than they were transmitted, making it suitable for real-time applications where latency might be a concern.
- It can be used for transaction-based protocols, such as Domain Name System (DNS) or Network Time Protocol (NTP).
- It can be used where a large number of clients are connected and where real-time error correction isn't necessary, such as gaming, voice or video conferencing, and streaming media.

3.6.1. UDP Header Composition

Figure 15 shows that the UDP header has a fixed size of 8 bytes. The UDP header has four fields, each of which is 2 bytes. They are the following:

- **source port number**, which is the number of the sender;
- **destination port number**, the port the datagram is addressed to;
- **length**, the length in bytes of the UDP header and any encapsulated data; and
- **checksum**, which is used in error checking – its use is required in IPv6 and optional in IPv4.

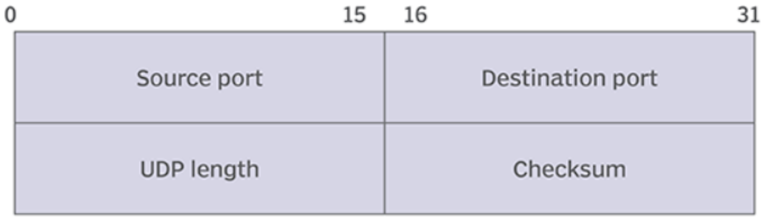


Figure 15. UDP header format

3.6.2. UDP Packet Security

UDP are used in many application protocols to transport application packets, and most of these applications thrive on secure local networks. However, as cyber-security threats mount-up everyday, it will be dangerous to have an insecure protocol in a network managing the crucial services of configuration management and addressing[38].

Datagram Transport Layer Security protocol (DTLS) is the primary option used for UDP security. DTLS is a communications protocol that provides security for datagram-based applications by allowing them to communicate in a way that is designed to prevent eavesdropping, tampering, or message forgery [39]. DTLS can be implemented easily using OpenSSL, a library that contains Transport Layer Security implementations, which also includes DTLS implementations. A UDP option suited for any application should be found easily in this library[38].

Another option is to use a lower-layer security, so security used for Internet Protocol can support any Transport Layer protocol. Internet Protocol Security (IPSec) does not work with port numbers as it is below the Transport Layer, so the fact that UDP cannot create a session does not matter when using IPSec [38].

4. Working System and Implementation

This chapter deals explicitly with the design methodology of the AR-based system proposed in this project. In this chapter, the system implementation, design tools, implementation tools are discussed. Before the design process was carried out, the following factors were considered: the design cost, its suitability for purpose, reliability, size, simplicity and availability of the components.The design process was then carried out in a manner such that the whole system was broken down into functional units. The project consists of two functional part (Unity3D side and the MATLAB side). But as regard the work performed, the project consists of three aspects;

1. The Hardware programming
2. The AR Software Design and Development
3. The Integration of the whole system

Details of the procedure working principles of the various stages are briefly described; The hardware part consists of the design of a working simulink in MATLAB that satifies the control criteria, it also involves the proper setting up of the router to be able to host our network, deployment of the designed AR application to the Meta 2 head-mounted display. While the software part is involves the design of the vitual tank, configuring Unity3D for AR developement, the design of the desired scene in Unity3D, the programming implementation of the Unity scene using Visual Studio and then integrating the scene with MATLAB. Finally, the developed AR application was deployed into the head-mounted display.

4.1. System Design

Figure 16 shows the system block diagram used. The system diagram utilizes two mode of communication which are wireless and wired. The Meta 2 head-mounted display connects via a USB and HDMI cables to a lab PC while the multitank system connects to another lab PC via a data acquisition box. Both PC are connected to the same router, making it possible to send and receive UDP packets from each other. The main task of the laboratory computers is to establish a bidirectional data communication between MATLAB and Unity3D, which in turn provide signal to the Meta 2 HMD and the Multitank System. The tools and software used are mentioned below:

1. MATLAB/Simulink 2012
2. Unity3D Editor version 2018.4 withe Vuforia SDK
3. Account in vuforia development portal
4. Visual Studio 2019
5. Router
6. 2 PCs
7. Meta 2 and Meta SDK
8. Multitank System
9. Cables and Connectors

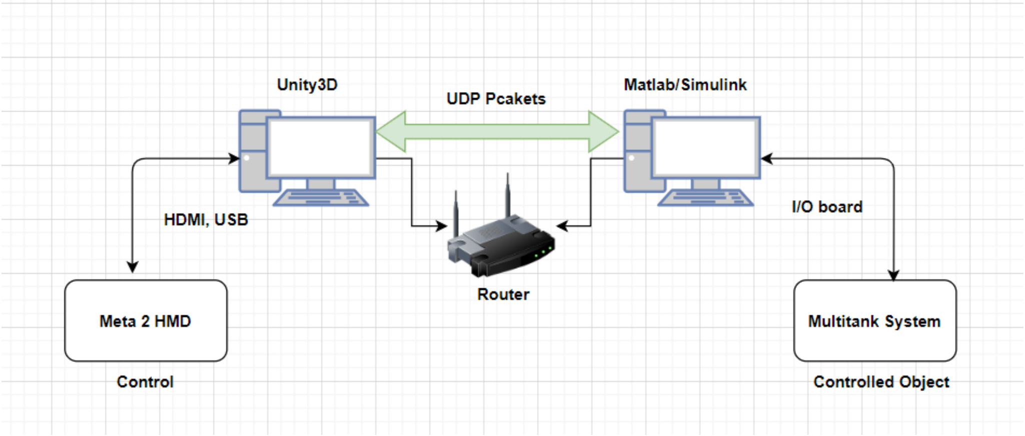


Figure 16. System block diagram

4.2. Multitank System

The industrial device model used in this work is a multitank system. To ensure that non-liner and linear control methods satifies all defined control requirements before deploying the control system into real-life industrial system where a design mistake can result in production losses, one must first implement this control method on industrial system models available in the laboratory.

The Multitank system is a model developed by INTECO. The system comprises of four separate tanks, the base tank is used as a reservoir. The uppermost tank has a constant rectangular sectional area while the second and third tank are conical and spherical respectively, so having variable cross section and this results in non-linearities in the system. As the water outflows the tank due to gravity, the tank valves act as flow resistors. The area ratio of the valves is controlled and can be used to vary the outflow characteristic. Each tank is equipped with a level sensor based on hydraulic pressure measurement[7].

4.2.1. Hardware

The Multitank System is shown in Figure 17, it has been designed to operate with an external, PC-based digital controller. The control computer communicates with the level sensors, valves and pump by a dedicated I/O board and the power interface. The I/O board is controlled by the real-time software which operates in MATLAB/Simulink RTW/RTWT rapid prototyping environment[7]. The components of the multitank system are listed below:

- 3 tanks made of acrylic glass
- 2 controlled valves
- 3 manual valves.
- pump: variable flow, driven by 12 V DC motor.
- 3 level sensors, piezoresistive.
- power interface.
- RT-DAC I/O internal PCI or external USB board

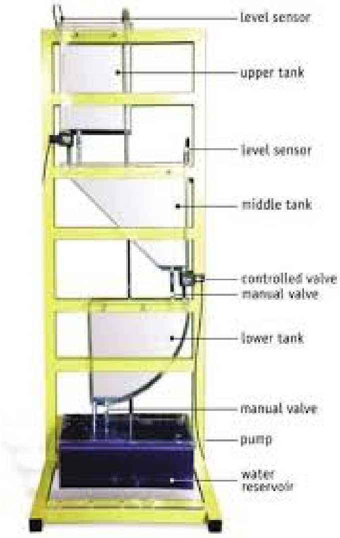


Figure 17. The Multitank system

4.3. Matlab Simulink Diagram

The design of controllers for level control in the first tank is carried out by considering linear fractional-order approximations of the nonlinear model of the process while discrete PID controller is designed for the control of the second and third tanks. In addition, an extended Kalman filter is designed to reduce measurement noise propagation into the control law thereby enhancing the performance of the pump[40]. The input packet receives the data sent by Unity3D in real time and inputs the data into the various control system of each tank. The output packet send data back to Unity3D also in real-time.

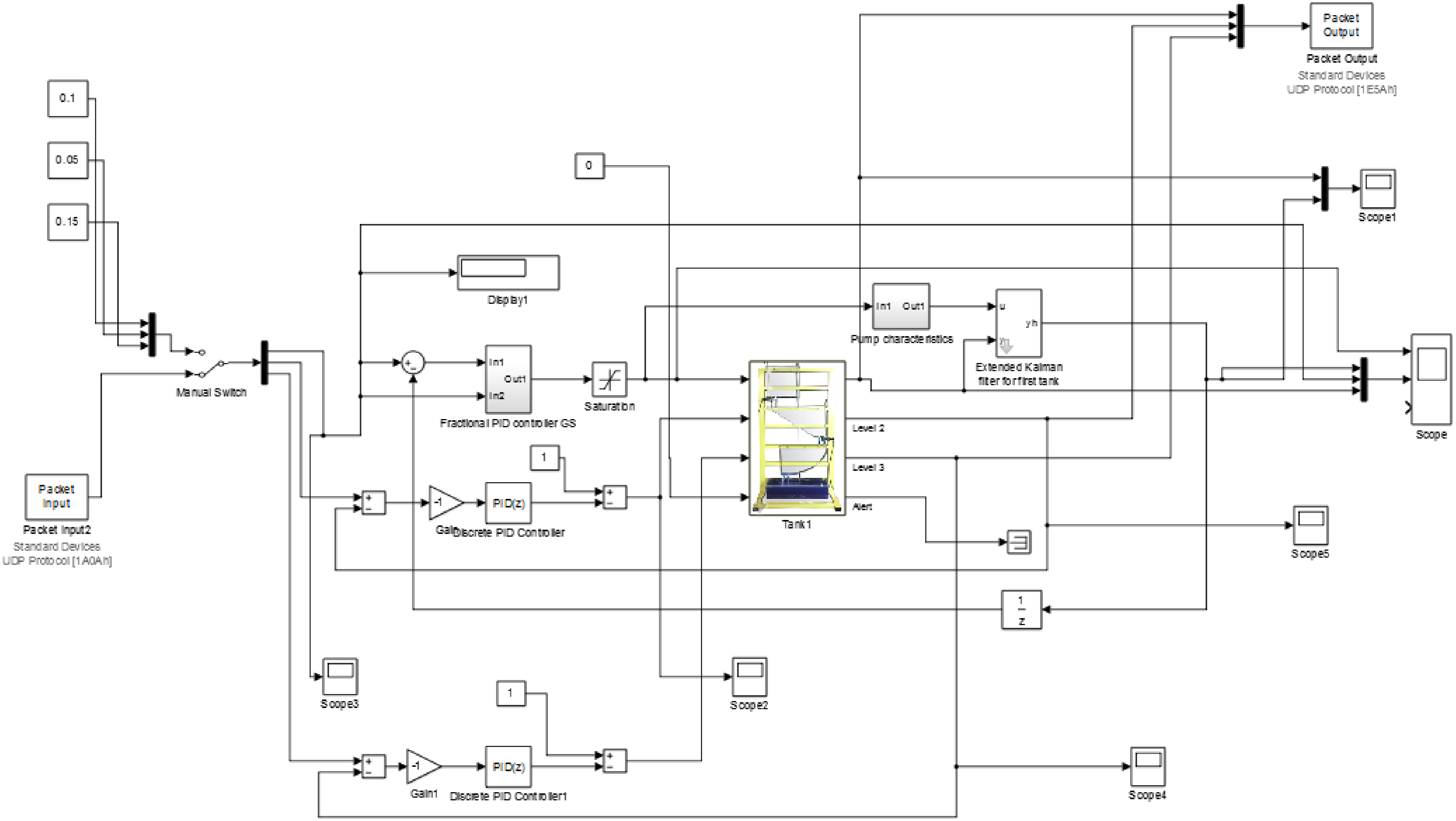


Figure 18. Simulink diagram used for the real-time control

4.4. Matlab Simulink Network Settings

A UDP board is installed in the input and the output packet. The input packet recieves an array of double data type. In total, the input packet recieves a packet of size 24 bytes as shown in Figure 17, 8 bytes each representing payload for each tank. The first 8 bytes represent the payload for the first tank, the next 8 bytes represent the payload for the second tank while the last 8 bytes represent the payload for the third tank. The setpoint for each tank is available in the payload. The output packet send a 24 bytes packet to the designnated port number as shown in Figure 20.

After correct Board setup. Sample time represents the number of times in 1 second we want the block to run. Sample time is set to “dt” in all blocks, this will enable us set a system sample time instead of setting individiual sample time for each block. The system sample time is set as 0.01. Missed ticks is the number of timer ticks that your model can lag behind the real-time kernel, the value 10 is be set by defaut. In Input/Output section we declare input packet size in bytes and data type. Each byte range should be multiplied by 8 double data type. Input packet field order is by default - Little Endian[6].

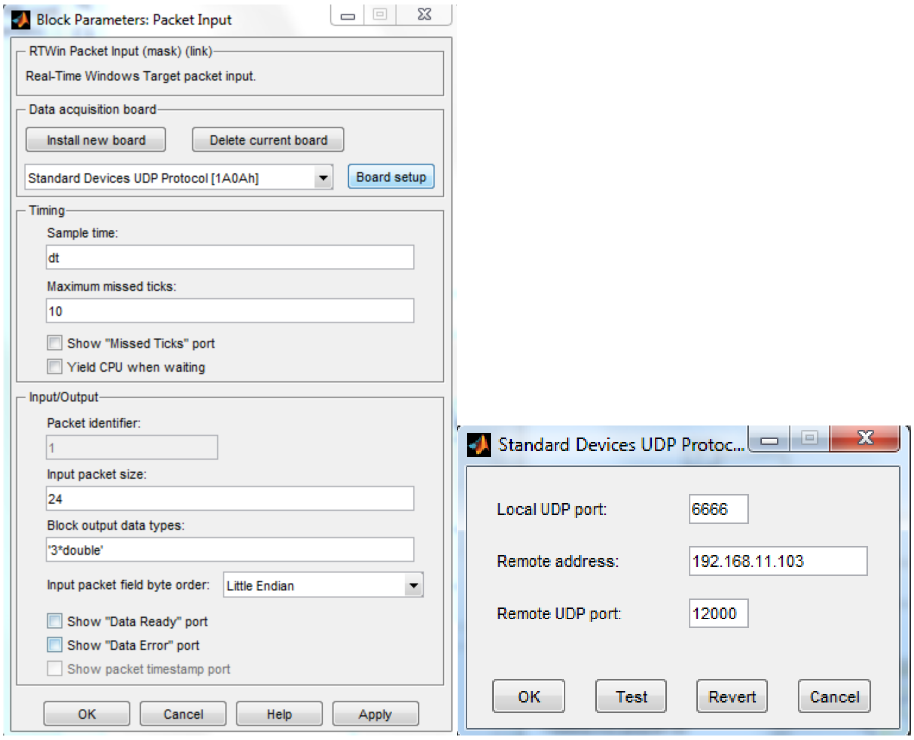
4.5. Augmented Reality Software Development

In this section, I would be discussing the steps in creating the AR application for this thesis. The tools used for the development are mentioned below:

4.5.1. Unity

Virtial reality and Augmented Reality applications are built inside a game engine. There are many game engines out there, but the type of game engine we use depend on the type of application we are trying to develop. Since we are trying to develop an AR application, we should pick a game a game engine that has both VR and AR capability. The other factors that influence the choice of game engine were

- The capability for the game engine to both send and receive data in real time.



Block Parameters: Packet Input

RTWin Packet Input (mask) (link)

Real-Time Windows Target packet input.

Data acquisition board

Install new board | Delete current board

Standard Devices UDP Protocol [1A0Ah] | Board setup

Timing

Sample time: dt

Maximum missed ticks: 10

☐ Show "Missed Ticks" port

☐ Yield CPU when waiting

Input/Output

Packet identifier: 1

Input packet size: 24

Block output data types: '3*double'

Input packet field byte order: Little Endian

☐ Show "Data Ready" port

☐ Show "Data Error" port

☐ Show packet timestamp port

OK | Cancel | Help | Apply

Standard Devices UDP Protoc...

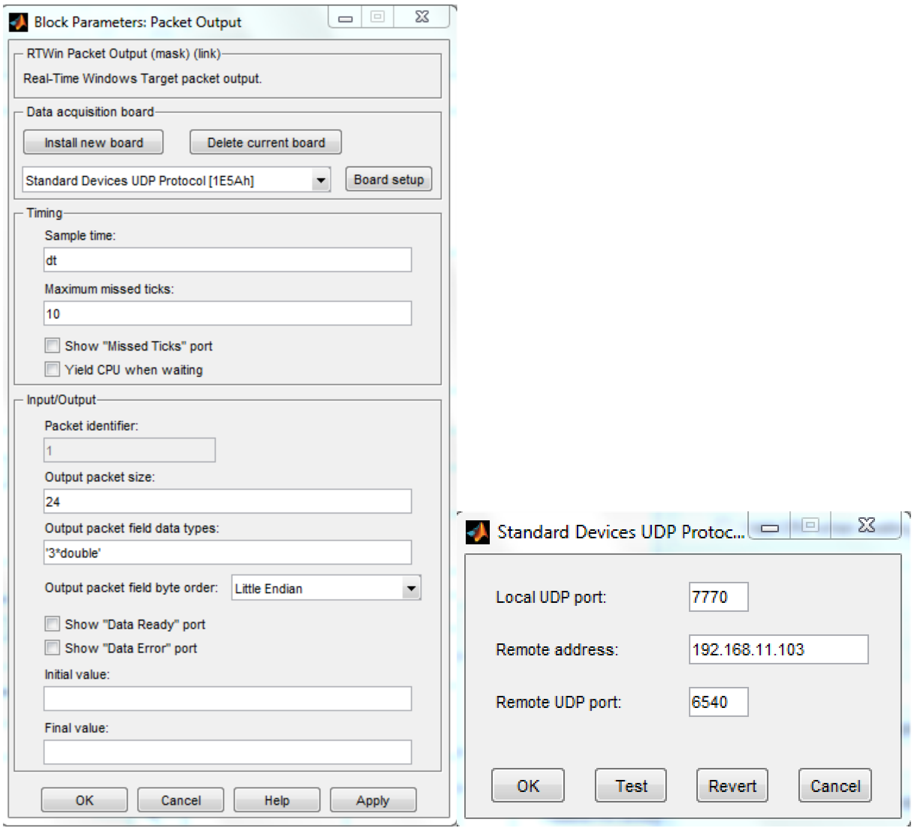
Local UDP port: 6666

Remote address: 192.168.11.103

Remote UDP port: 12000

OK | Test | Revert | Cancel

Figure 19. Packet input block and board setup



Block Parameters: Packet Output

RTWin Packet Output (mask) (link)

Real-Time Windows Target packet output.

Data acquisition board

Install new board | Delete current board

Standard Devices UDP Protocol [1E5Ah] | Board setup

Timing

Sample time: dt

Maximum missed ticks: 10

☐ Show "Missed Ticks" port

☐ Yield CPU when waiting

Input/Output

Packet identifier: 1

Output packet size: 24

Output packet field data types: '3*double'

Output packet field byte order: Little Endian

☐ Show "Data Ready" port

☐ Show "Data Error" port

Initial value:

Final value:

OK | Cancel | Help | Apply

Standard Devices UDP Protoc...

Local UDP port: 7770

Remote address: 192.168.11.103

Remote UDP port: 6540

OK | Test | Revert | Cancel

Figure 20. Packet output block and board setup

- The ability of the game engine to deploy designed scene into numerouss platform like android, Universal Windows Platform (UWP), iOS, hololens and other immersive headsets.
- Ease of usage. Most of the game engine have intimidating UI and it can be overwhelming trying to learn various parts of the engine and so they have long learning curve.
- The programming language used for development in the game engine played a significant role in choosing a game engine..
- Proper documentaions and large community

Unity is a cross-platform game engine developed by Unity Technologies and used to develop video games for PC, consoles, mobile devices and websites. Unity can be downloaded free of charge. Unity is undoubtedly a very popular gaming platform. This impressive engine accounts for 45% of all game development, almost half of the market. If revenues are more than $ 100,000, one will need to buy the PRO version for $ 1,500 or $ 75 per /month. Currently, Unity has the largest community. Unity uses C# and JavaScript. Unity has a powerful store - Unity Asset Store (just for this store alone, one can safely select Unity), which has absolutely everything from textures and 3D models to finished projects. None of the competitors can boast a range as wide as that. Asset Store brings Unity in leaders of market [41]

Mostly, game, VR and AR are tilted to the Unity engine because compared to other progressive 3D engines like Unreal engine the learning curve is shorter. Unity has many features, and its versatility is incredible. First, one can create almost anything using just this one engine: 2D and 3D projects of varying scale. The UI is pleasant to use, very intuitive and requires little time to become proficient in despite its rich functionality. Unity's usability allows high prototyping speed achieved by means of live tweaking and debugging support, which enable the developer to make adjustments while playing. Unity stands out in the fast-growing mobile gaming industry, and many of the most popular 3D mobile games were designed with it. And good news is, Unity is adding more VR technology to its game development[42]. In the Table 4 are shown results of comparison.

Mostly beacuse of my background, having used the Unity3D environment for AR development previously and also my experience developing enterprise application

Table 4. Games engines, table of alternatives and criteria [1]

Criteria/Engines	Cross-platform	Availability of documentation	Programming languages	Additional services +AR and MRS (Score)	License type
Unity	All platforms	Much	C#, JavaScript	Excellent	Free for students
CryEngine3	Windows PC, PlayStation 4, Oculus Rift, Xbox One, Linux PC	Few	C#, C++	Bad	Free
Unreal Engine 4	Windows, OSX, Android, Linux, Oculus Rift, PlayStation, Steam, HTML5, Xbox.	Enough	C++, Blueprints	Good	Free for students

with the C# programming languge in the .Net environment, I decided to develop the AR application needed for this thesis work using the Unity3D game engine.

4.5.2. Virtual Object Creation

AR is augmenting virtual object into the real world. The virtual object which can be a 2D or 3D model of an object and is designed using a 3D graphics design software such as Blender, 3D Max, Maya e.t.c.. In this case, the Blender software was used to design the 3D model of a metre rule attached to the water tank model. Blender is a free software but one of the best 3D design software available. Blender is the free and open source 3D creation suite. It supports the entirety of the 3D pipeline—modeling, rigging, animation, simulation, rendering, compositing and motion tracking, video editing and 2D animation pipeline[43]. In order to more clearly present the results of the review of programs for creating virtual objects, the results are summarized in the Table 5:

After reviewing the options and their comparison, it was decided to use Blender for virtual object creation. As illustrated in the Table 3, all the 3D modelling software offer the same packages and have very similar technologies and services. The water

Table 5. Graphics software, alternatives and criteria [1]

Criteria/Alternatives	Blender	Maya	3D Max
Cross-Platform	Windows, Linux, OSX	Windows, Linux, OSX	Windows, Linux, OSX
Availability of documentation/easy to learn	Much	Much	Much
Possibility to export objects to Unity3D	Yes/Yes	Yes/Yes	Yes/Yes
Additional services (Score)	Excellent	Excellent	Excellent
License type	Free for students	Free for students	Free for students

tank model was designed with Unity3D game object design option as it was easy to scale at point of creation to desired size in Unity3D scene. Using third party applications mentioned above does help some of the development challenges but also cause some other problems affecting the optimization of our AR application mentioned below:

1. It could increase the size of the final AR application by a lot
2. Scaling the imported design in the Unity3D scene to required size can be problematic
3. Application Compilation

AR application requires the integration of virtual object with the real environment and when developing AR application for smart devices such as the Meta 2 HMD in which the field of view is a constraint it would be beneficial for the developer to consider some of the points below

1. The size of the virtual object relative to the real object
2. The position of the virtual object in the world space

3. The material and the quality of the virtual object
4. The ease to match the virtual object with the real object

Figure 21 shows the developed virtual object which includes the water tank, the metre gauge, the red marker and the water. The water tank comprises of different cubes representing the different sides of the tank which all have a transparent white material.

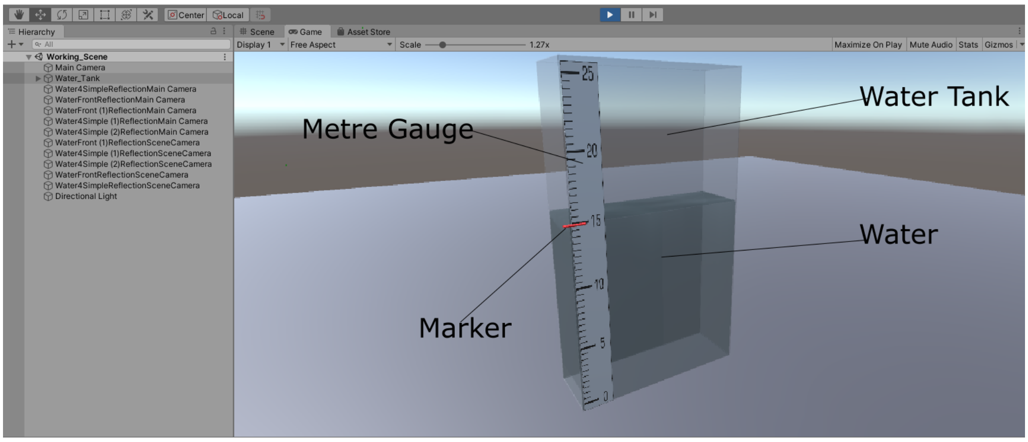
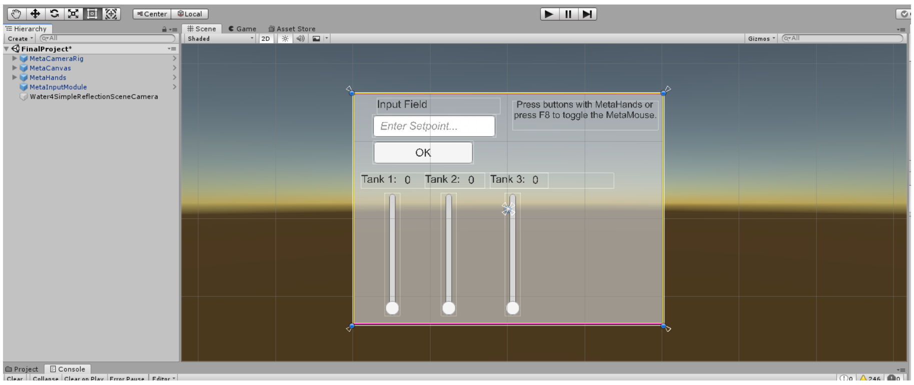

Center
Local
Hierarchy
All
Working_Scene
Main Camera
Water_Tank
Water4SimpleReflectionMain Camera
WaterFrontReflectionMain Camera
WaterFront (1)ReflectionMain Camera
Water4Simple (1)ReflectionMain Camera
Water4Simple (2)ReflectionMain Camera
WaterFront (1)ReflectionSceneCamera
Water4Simple (1)ReflectionSceneCamera
Water4Simple (2)ReflectionSceneCamera
WaterFrontReflectionSceneCamera
Water4SimpleReflectionSceneCamera
Directional Light
Scene
Game
Asset Store
Display 1
Free Aspect
Scale
1.27x
Maximize On Play
Mute Audio
Stats
Gizmos
Water Tank
Metre Gauge
Water
Marker
25
20
15
10
5
0

Figure 21. A 3D view of the Unity scene showing the water tank with metre rule

The Meta 2 Unity SDK is installed into Unity3D by importing the Meta 2 Unity SDK package downloaded from Meta 2 website. The SDK includes SLAM, hands interaction tracking, example code, and documentation. Now, we are able to interact and manipulate 3D holographic content by adding the MetaCameraRig, MetaHands and MetaInputModule to the Unity3D scene. Figure 22 shows the developed virtual object which includes the sliders and input field for the setting the setpoints for the various water tanks.

Figure 23 shows the developed virtual graph. The graph displays how the process value compares with the set point of the various water tank levels.

Figure 24 shows the integration of all developed virtual objects. The arrangement was done with the aim to make it easy for a user to interact with the sliders easily, while also visualizing the data on the graph.



Press buttons with MetaHands or press F8 to toggle the MetaMouse.

Figure 22. The Unity scene showing canvas that contains the sliders to control the three tanks

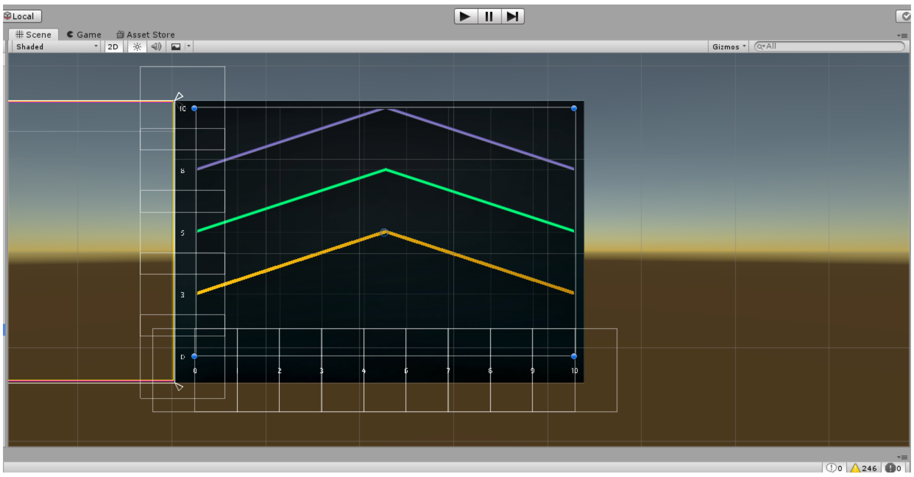
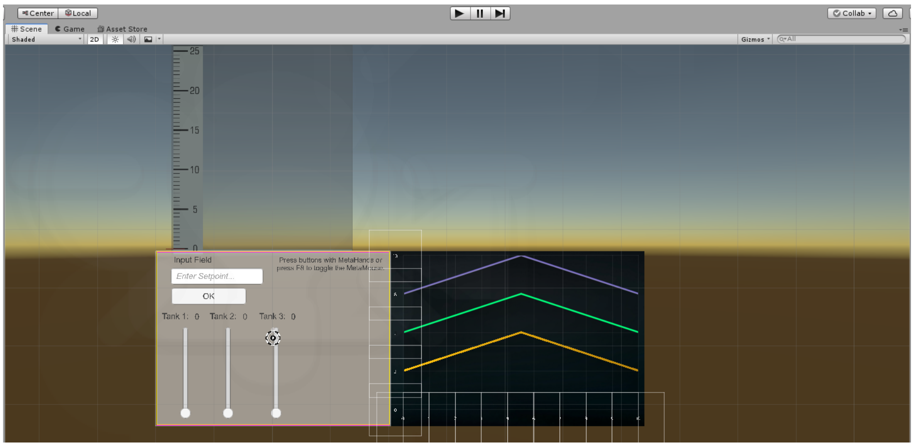


Figure 23. The Unity scene showing the graphical representation



Press buttons with MetaHands or press F8 to toggle the MetaMouse.

Figure 24. 2D view of the all developed virtual contents for the Unity scene

5. Integration, Results and Testing

This chapter focuses on how the project design was integrated, put to test, and the result expected of the design.

5.1. Establishing Connection between Unity3D and MATLAB/Simulink

For the two stand-alone applications hosted on two different PCs to communicate with each other, we need to establish a remote connection between the two hosts. Since we will be sending and receiving streams of data, the UDP connectionless protocol provides us a viable solution than TCP. To establish a UDP connectionless protocol, both hosts were connected to the same router(same network), the C# scripting available for Unity3D was used to create an instance of a UDP client, set up both sending and receiving ports, set up IP addresses, and assign the number of bytes we want to send. A receiving and sending UDP packets have already been installed on the Simulink side to correspond with what the Unity is ending and receiving (see section 4.4).

Figure 27 shows the function implemented for sending the UDP packets to MATLAB/Simulink

```
public void Send(double[] val)
{
for (int i = 0; i < val.Length; i++)
{
IPEndPoint serverEndpoint = new IPEndPoint(IPAddress.Parse(senderIp), senderPort);
byte[] sendBytes = BitConverter.GetBytes(i);
udpClient.Send(sendBytes, sendBytes.Length, serverEndpoint);
}
}
```

Figure 25. Program code for sending UDP packets from Unity3D to Simulink

Figure 28 shows the function implemented for recieving the UDP packets sent by MATLAB/Simulink

```
private void ListenForMessages(UdpClient client)
{
IPEndPoint remoteIpEndPoint = new IPEndPoint(IPAddress.Any, 0);
while (threadRunning)
{
byte[] receiveBytes = client.Receive(ref remoteIpEndPoint);
        double[] values = new double[receiveBytes.Length / 8];
        for (int i = 0; i < values.Length; i++)
{
    values[i] = BitConverter.ToDouble(receiveBytes, 8 * i);
            lock (incomingQueue)
{
incomingQueue.Enqueue(values[i]);
}
}
     }
}
```

Figure 26. Program code for receiving UDP packets in Unity3D from Simulink

5.2. Establishing Connection between Unity3D and Meta 2 HMD

The Meta 2 Unity SDK is installed into Unity3D by importing the Meta 2 Unity SDK package downloaded from Meta 2 website. The SDK contains the Meta 2 Camera prefab which was dragged into the scene, this helps to see the virtual content available in the Unity3D world space through the Meta 2 HMD camera. The Meta Hand prefab enables us to interact with the virtual content with our hands while the Meta 2 Input Module enables us to give input to the virtual content displayed. Some other settings were configured in Unity3D such as enabling its AR and VR capability, improve the quality setting to ensure we see high-quality digital content, and finally we connect the host and the Meta 2 HMD using both HDMI and USB.

5.3. Results

The pictures below show snapshots taken while a user put on the Meta 2 headset to utilize the AR application developed and deployed into it. Figure 27,28 and 29 show the Meta views while a user is using the developed AR application to send control signals to the Multitank system.

The list of what the user is able to do is listed below;

1. The user is able to see the digital information
2. The user is able to interact and give input to the digital UI using the hand
3. The user is able to use the sliders to set set-points for the tanks in the Multitank system
4. The user is able to see the graphical representation of the set-points and the process values in real-time

5.4. System Testing

After the system implementation, a series of tests were performed on the Multitank system to validate the implementation of the control system designed in MATLAB/Simulink, The PID controllers parameters were set and the goal is to control the liquid level of the three tanks. The setpoints of the three were chosen randomly and the display block was added to the simulink to identify the unreachable water levels in the second and third tanks. Doing this we were able to find out the reachable and reachable levels in the second and third tanks. This occur because they depend on the first tank as this is where the liquid intake comes in from and the liquid outflows the tanks due to gravity, the system will perform better if each tank has its pump so we can control the pumping of water to each tank thereby eliminating the dependencies between the tanks. Table 6 shows the result of the experimental testing, unrecheable levels are identified with the letter “U”.

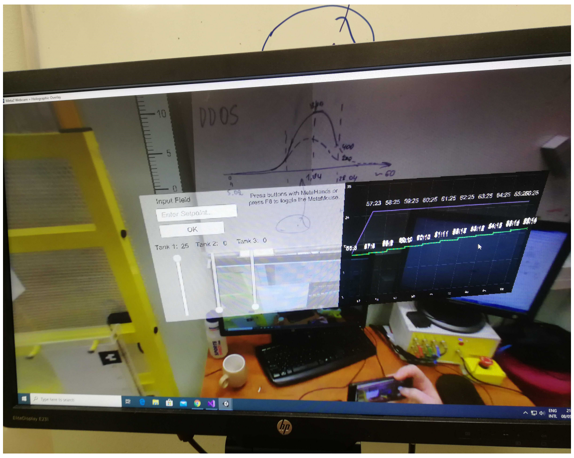

Input Field
Enter Setpoint...
OK
Tank 1: 25
Tank 2: 0
Tank 3: 0
Press buttons with MetaHands or press F8 to toggle the MetaMouse
DDOS

Figure 27. Meta 2 view showing the digital content in the AR application

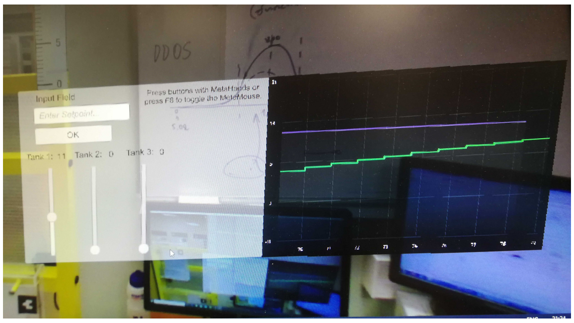

Input Field
Enter Setpoint...
OK
Tank 1: 11
Tank 2: 0
Tank 3: 0
Press buttons with MetaHands or press F8 to toggle the MetaMouse
DDOS

Figure 28. Meta 2 view showing the process value approaching the set-point of value 11cm for th first tank

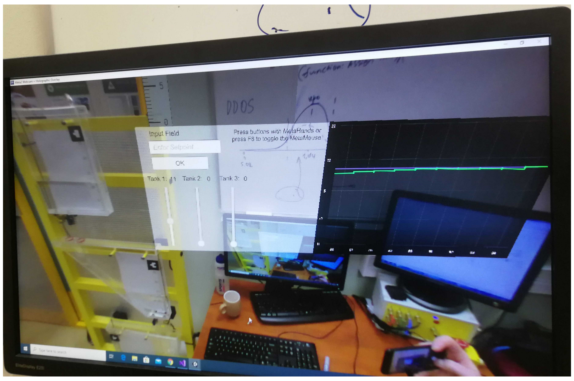

Input Field
Enter Setpoint...
OK
Tank 1: 11
Tank 2: 0
Tank 3: 0
Press buttons with MetaHands or press F8 to toggle the MetaMouse
DDOS

Figure 29. Meta 2 view showing minimal error between PV and SP

Table 6. Experimental testing showing unreachable water levels. U represents levels that are unrecheable

Tank 1 (cm)	Tank 2 (cm)	Tank 3 (cm)
5	5	5
5	10 (U)	5
5	15	5 (U)
5	20 (U)	5 (U)
5	25 (U)	5 (U)
5	10	10
5	10 (U)	15
5	10 (U)	20 (U)
5	10	25 (U)
5	20 (U)	5 (U)
10	10	10
10	10	20 (U)
10	15 (U)	20
15	10	20
15	20 (U)	25
20	10	10
20	10 (U)	20

6. Conclusion and Recommendation

6.1. Conclusion

AR is a field that can play a major role in industry 4.0 where an Engineer can work remotely sending control signals, tuning controllers, and visualizing automation data of industrial systems. Conventionally, these processes are done using the traditional SCADA system where Engineers need to be in a control station while monitoring and sending signals. The work done in this thesis presents a more interactive and cost-efficient alternative as it will help to reduce the plant downtime because problems would be identified early and quickly enough.

The thesis aimed to design and develop a Meta 2 headset application with its unique virtual object for complementing the control of a real-life laboratory model of an industrial multi-tank system and some objectives were identified to achieve the aim. To a very good extent, this aim was achieved. The result showed that we can send control signals to the Multitank system from our device and visualize automation data in real-time. Also, work was done trying to integrate Vuforia image recognition algorithm with the HMD, but this was not possible due to a lack of documentation and support on the device.

6.2. Recommendation

Several improvements can be done, and later integrations with technologies that are not currently possible. But two major improvements are necessary if the AR application is going to ever be deployed to the shop-floor. They are narrated below;

During my thesis work, Meta, the company that produced the Meta 2 augmented reality headset furlough two-thirds of the company's workers due to the company lacking investors and as result did not provide support for the headset. A new company called Meta View has purchased the old Meta's assets, this new company is presided by the former founder of the Vuforia technology before it was sold to PTC. So it seems the future is looking interesting for the high definition HMD. At the moment, the Meta 2 kit either does not support Vuforia Image Recognition technology or just lack of documentation on how to integrate both technologies. So a proposed upgrade that can be performed on the work done in this thesis is to later integrate the functionalities provided by the Meta 2 SDK with Vuforia Image Recognition capability or integrate with some other AR computer vision technology such as Wikitude, enabling us to be able to use state of the art image recognition technology to recognize the real Multitank before overlaying the digital content on it. Another obvious recommendation is to include in the digital content displayed the possibility to either decrease or increase the controller parameters (proportional gain, integral gain, and the derivate gain) to minimize the controller error.

References

[1] M. Tarasevich, “Virtual objects integration and motion recognition for augmented reality applications,” Master’s thesis, Tallinn University of Technology, 2016.

[2] R. V. Krevelen, “Augmented reality: Technologies, applications, and limitations,” Department of Computer Science, Vrije Universiteit Amsterdam, Tech. Rep., 2007.

[3] B. Marr, “9 powerful real-world applications of augmented reality (ar) today,” 2018. [Online]. Available: https://www.forbes.com/sites/bernardmarr/2018/07/30/9-powerful-real-world-applications-of-augmented-reality-ar-today/#16941ca62fe9

[4] A. W. Ismail and M. S. Sunar, “Collaborative augmented reality: Multi-user interaction in urban simulation,” in *Lecture Notes in Computer Science.* Springer Berlin Heidelberg, 2009, pp. 382–391.

[5] S. S. Miro Antonijevic and H. Keserica, “Augmented reality applications for substation management by utilizing standards-compliant SCADA communication,” *Energies*, vol. 11, no. 3, p. 599, mar 2018.

[6] A. Smolov, “Control of multitank system using raspberry pi,” Master’s thesis, Tallinn University of Technology, 2017.

[7] *INTECO Multitank system - User Manual Multitank System-User manual*, Inteco, 2012.

[8] S. Kaji, H. Kolivand, R. Madani, M. Salehinia, and M. Shafaie, “Augmented reality in smart cities: applications and limitations,” *Engineering Technology*, 2018.

[9] V. BharathChandra Reddy, Nayan Kadam, “Visualization of industrial robot with augmented reality,” Master’s thesis, University of Applied Science Ravensburg-Weingarten, 2017.

[10] R. Tarca, F. Popentiu-Vladicescu, T. Vesselenyi, I. Tarca, and I. Barda, “A telerobot control system using augmented reality,” *Augmented Reality*, vol. 4, 01 2009.

[11] M. G. Fard, “4 dimensional augmented reality- models forautomation and interactive visualization ofconstruction progress monitoring,” Ph.D. dissertation, University of Illinois at Urbana-Champaign, 2010.

[12] S. Garagnani and A. M. Manferdini, “Virtual and augmented reality applications for cultural heritage,” *Virtual and augmented reality*, September 2011. [Online]. Available: http://amsacta.unibo.it/3151/

[13] V. Kuts, T. Tahemaa, Y. Bondarenko, and T. Otto, “Digital twin based synchronised control and simulation of the industrial robotic cell using virtual reality.” *Machine Engineering*, 2019.

[14] T. Caudell and D. Mizell, “Augmented reality: an application of heads-up display technology to manual manufacturing processes,” in *Proceedings of the Twenty-Fifth Hawaii International Conference on System Sciences.* IEEE, 1992.

[15] W. Hastings, “Augmented reality for manufacturing: Bringing digital transformation to skilled workers,” *Digital Transformation*, 2018.

[16] K. Mathivanan, S. T, A. B, and S. R, “A study of virtual reality,” *International Journal of Trend in Research and Development*, vol. 4, pp. 2394–9333, 06 2017.

[17] P. Milgram and F. Kishino, “A taxonomy of mixed reality visual displays,” *IEICE Trans. Information Systems*, vol. vol. E77-D, no. 12, pp. 1321–1329, 12 1994.

[18] J. Carmigniani and B. Furht, *Augmented Reality: An Overview*, 07 2011, pp. 3–46.

[19] A. Solutions, “5 ways how augmented reality will impact the future,” *Augmented reality*, Dec. 2019. [Online]. Available: https://www.ambiguousit.com/blog/how-augmented-reality-will-impact-the-future/

[20] A. B. Craig, “Understanding augmented reality : Concepts and applications,” *ProQuest Ebook Central*, 2013.

[21] J. Carmigniani, B. Furht, M. Anisetti, P. Ceravolo, E. Damiani, and M. Ivkovic, “Augmented reality technologies, systems and applications,” *Multimedia Tools and Applications*, vol. 51, no. 1, pp. 341–377, dec 2010.

[22] N. Reality, “What’s the difference between hololens, meta 2 & magic leap?” *Augmented Reality*, 2017. [Online]. Available: https://next.reality.news/news/whats-difference-between-hololens-meta-2-magic-leap-0181804/

[23] V. Shetty, V. Rai, and M. Patil, “Study of hardware and software used in: Virtual and augmented reality device,” 01 2016.

[24] M. H. Iqbal, A. Aydin, A. Lowdon, H. I. Ahmed, G. H. Muir, M. S. Khan, P. Dasgupta, and K. Ahmed, “The effectiveness of google GLASS as a vital signs monitor in surgery: A simulation study,” *International Journal of Surgery*, vol. 36, pp. 293–297, dec 2016.

[25] Nreal, “Nreal light - ready to wear mixed reality glasses,” *Augmented Reality Glasses*, Jan. 2017. [Online]. Available: https://www.nreal.ai/specs/

[26] Wikipedia, “Meta 2,” *Augmented Reality Devices*, 2017. [Online]. Available: https://xinreality.com/wiki/Meta_2

[27] XinReality, “Meta 2,” *Augmented Reality Devices*, 2017. [Online]. Available: https://xinreality.com/wiki/Meta_2

[28] L. Johnson, “Industrial control systems characteristics,” *Security Component Fundamentals for Assessment*, 2016.

[29] O. Manso, “Overview of industrial control systems,” *26th European Symposium on Computer Aided Process Engineering*, 2016.

[30] Y. Hashimoto, T. Toyoshima, S. Yogo, M. Koike, S. Jing, and I. Koshijima, “Conceptual framework for security hazard management in critical infrastructures,” in *Computer Aided Chemical Engineering.* Elsevier, 2012, pp. 1266–1270.

[31] Wikipedia, “Industrial control system,” 2020. [Online]. Available: https://en.wikipedia.org/wiki/Industrial_control_system

[32] B. Galloway and G. P. Hancke, “Introduction to industrial control networks,” *IEEE Communications Surveys and Tutorials*, vol. 15, no. 2, pp. 860–880, 2013.

[33] Elprocus, “The working principle of a pid controller for beginners,” *Last accessed 2020-05-05. [Online]. Available: https://www.elprocus.com/the-working-of-a-pid-controller/*, 2020.

[34] U. of Michigan, “Introduction: Pid controller design,” 2020. [Online]. Available: http://ctms.engin.umich.edu/CTMS/index.php?example=Introduction&section=ControlPID#27

[35] Wikipedia, “User datagram protocol,” 2020. [Online]. Available: https://en.wikipedia.org/wiki/User_Datagram_Protocol

[36] M. Rouse, L. Rosencrance, G. Lawton, and C. Moozakis, "Udp (user datagram protocol)," 2019. [Online]. Available: https://searchnetworking.techtarget.com/definition/UDP-User-Datagram-Protocol

[37] GeeksforGeeks, "User datagram protocol (udp)," 2018. [Online]. Available: https://www.geeksforgeeks.org/user-datagram-protocol-udp

[38] S. Cooper, "A guide to udp (user datagram protocol)," 2019. [Online]. Available: https://www.comparitech.com/net-admin/guide-udp-user-datagram-protocol/Security_for_UDP

[39] Wikipedia, "Datagram transport layer security," 2020. [Online]. Available: https://en.wikipedia.org/wiki/Datagram_Transport_Layer_Security

[40] A. Tepljakov, E. Petlenkov, and J. Belikov, "Gain and order scheduled fractional-order PID control of fluid level in a multi-tank system," in *14 International Conference on Fractional Differentiation and Its Applications 2014*. IEEE, jun 2014.

[41] Unity, "Last accessed 2020-05-05. [online]. available: http://www.unity3d.com." *Game Engine Software*, 2020.

[42] Y. Kronberg, "How to choose a game engine," 2017. [Online]. Available: https://www.linkedin.com/pulse/how-choose-game-engine-yann-kronberg/

[43] Blender, "Blender," *Open Source 3D creation*, 2020. [Online]. Available: https://www.blender.org/

Appendix 1 - How To Install the Vuforia Unity Extension

1. Create a Vuforia Developer Portal account using `https://developer.vuforia.com/vui/auth/register`
2. Download Vuforia SDK for Unity here `https://developer.vuforia.com/downloads/sdk`
3. Import Vuforia SDK into Unity. Make sure the SDK is compatible with the version of Unity

Appendix 2 - How to apply Vuforia SDK to get a good Image Target for Unity3D

Vuforia is an augmented reality software development kit (SDK) for mobile devices and HMD that enables the creation of augmented reality applications.It uses computer vision technology to recognize and track planar images and 3D objects in real time. The image recognition capability available in Vuforia can be used to positon the designed virtual object to desired position in the real word. The steps to be taken to recognize the multitank model are;

1. A good picture of the multitank model should be captured using a smartphone.
2. Uploaded the picture into the Vuforia developer database
3. Run the image through the Vuforia Image recognition algorithm
4. If rating is good (usually a rating of 4 or 5 is good enough), deploy image into the Unity3D environment, if not take a new picture.
5. Add a game object to the surface of the image target
6. Add the vuforia AR camera to the scene.
7. Insert a valid liscence into the Vuforia AR camera configuration
8. Ensure that AR capability is enabled in Unity3D
9. deploy the package and run the simple AR application on a smartphone

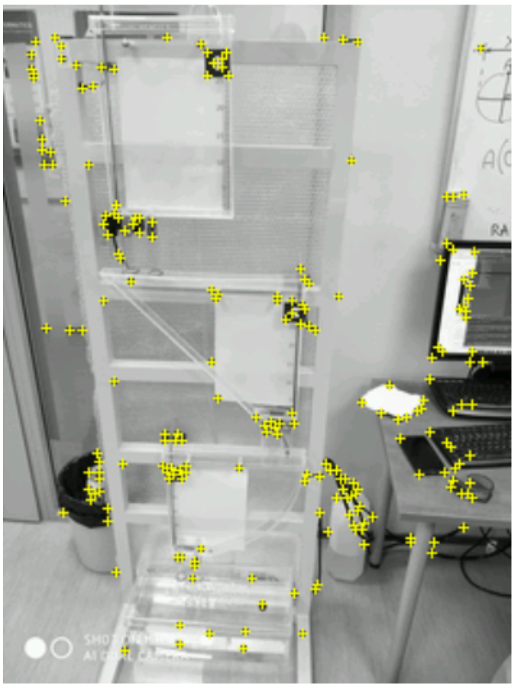


Figure 30. The Vuforia Engine Recognition of Smartphone camera picture

Following these procedures, we are able to test how well the target is recognized from the AR camera and how stable virtual object looks relative to the target. These processes were done repeatedly to ensure a good Image target. We had the problem of inconsistent target recognition, the AR camera could not recognize at all time the target after we deploy the simple AR application into a smartphone either due to change in the room lighting condition or changes around the target in the real world. To tackle the p oblem, we decided to capture the target (the multitank system) using the AR camera by following just step 6 to step 9 mentioned above. Now, it is possible to take the picture of the target (the multitank system) using the AR camera itself, then we proceed from step 2 to step 9. This method was tested and it helped solve the problems of inconsistent target recognition and unstable virtual object. Figure 31 shows the picture taken by the AR camera been recognized inside Vuforia Image recognition engine and Figure 32 shows the Image Target uploaded into Unity3D game engine.

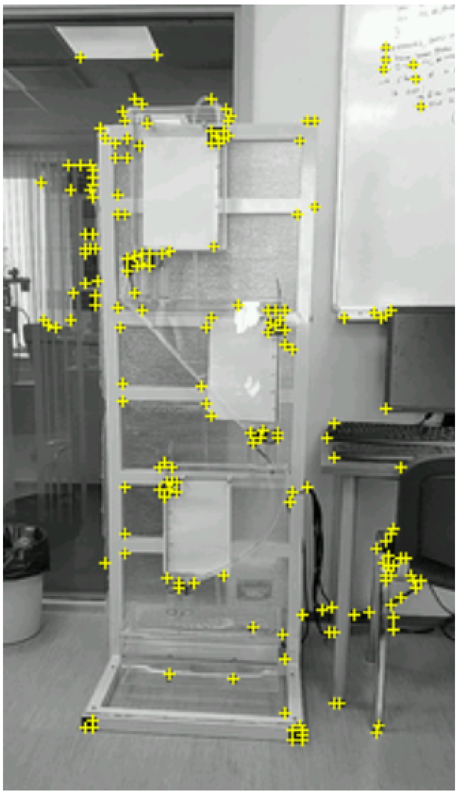
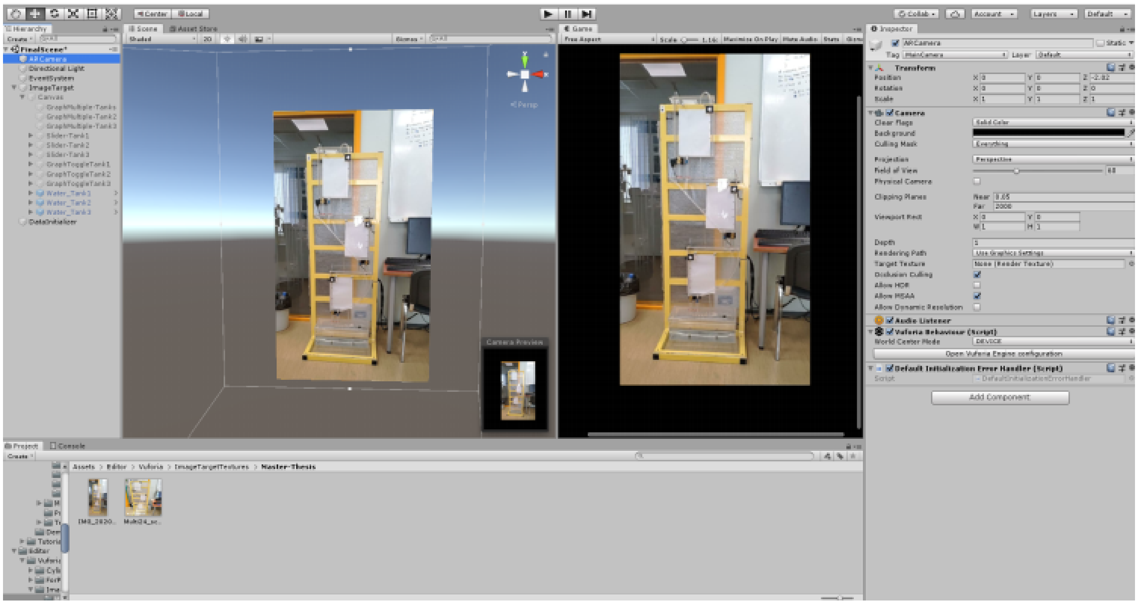


Figure 31. The Vuforia Engine Recognition of AR Camera picture



ARCamera
Tag MainCamera Layer Default

Transform
Position X 0 Y 0 Z -2.82
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

Camera
Clear Flags Solid Color
Background
Culling Mask Everything
Projection Perspective
Field of View 60
Physical Camera □
Clipping Planes Near 0.05 Far 2000
Viewport Rect X 0 Y 0 W 1 H 1
Depth 1
Rendering Path Use Graphics Settings
Target Texture None (Render Texture)
Occlusion Culling ☑
Allow HDR □
Allow MSAA ☑
Allow Dynamic Resolution □

Audio Listener

Vuforia Behaviour (Script)
World Center Mode DEVICE
Open Vuforia Engine configuration

Default Initialization Error Handler (Script)
Script DefaultInitializationErrorHandler

Add Component

Figure 32. Image Target inside Unity3D

Appendix 3 - Source Code

- The application program source code is available at: `https://github.com/tysjosh/MultitankControlUnity`
- Make sure you download Unity 2018.4.18 and import the Meta 2 SDK package available in the above github folder into Unity.
- Use Matlab 2012 to run the simulink package