NTNU

Kunnskap for en bedre verden

Augmented reality for Offshore Crane Operations

Demy Patrick Gielesen

Spring 2019

Master thesis in Simulation and Visualization

Department of ICT and Engineering, Faculty of Information Technology and Electrical Engineering

Norwegian University of Science and Technology

Supervisor 1: Robin T. Bye

Supervisor 2: Arne Styve

Keywords: Augmented Reality, Offshore, Cranes, HoloLens, Unity3D, Situational Awareness

Acknowledgement

I would like to thank my supervisors, Robin T.Bye and Arne Styve for providing insight, feedback and helping with the thesis setup.

I would also like to thank my family and friends both for their support and ideas given throughout the thesis.

Lastly, I would like to thank OSC for giving me access to a HoloLens and Crane Simulator and for providing a job offer to work as a Software Developer.

Abstract

In my thesis, I explore the potential usage of Augmented Reality for crane operators of offshore cranes. By suggestion of Offshore Simulator Center (OSC), the idea was to use the Microsoft HoloLens to develop prototype software and use their crane dome-simulator for testing purposes.

Three problems commonly found in crane-operations were addressed and Augmented Reality solutions were proposed for each. The problems addressed were: Blind zones, location of information vital for the job, and perspective problems inside the crane's cabin. All solutions were validated through the use of a digital survey.

The result is that Augmented Reality has the potential to be used in crane operations. 4/5 operators agreed that Augmented Reality is usable for today's jobs and came with further suggestions on how to expand the solutions.

Abbreviations

m Meter

cm Centi-Meter

AR Augmented Reality

HUD Heads Up Display

HMD Head mounted Display

ms Milliseconds

MP Mega Pixels

CPU Central Processing Unit

RAM Random Access Memory

3D 3 Dimensional

2D 2 Dimensional

App Application, usually in context of phone applications

FOV Field of View

IDE Integrated development environment

API Application Programming Interface

LIFO Last In First Out

FIFO First In First Out

MRU Motion Reference Unit

RPRS Relative Positioning Reference Systems

List of Figures

List of Tables

Listings

Contents

Chapter 1

Introduction

Augmented reality, or AR, is a technology that enables us to see and interact with virtual 3D objects by superimposing them on our field of view. By "augmenting" our reality we can add visuals, smells and sound that only we are able to perceive[40]. Building on the science-fiction idea of "holograms", which are 3D objects made of sound and light[25], Augmented Reality enables us to see these holograms with specialized display technology. The idea of Augmented Reality was first tested in 1968 by the use of wearable cathode ray tube screens [47] and was gradually researched further over the years. It wasn't before 1990 that this idea really gained traction and it was given the name "Augmented reality" by scientists of the Boeing Company[18]. AR technology has gained a bigger focus in recent years because of the improvement of display and computing technology. The release of the standalone AR headsets, the Microsoft HoloLens in March 2016 [22] and Magic leap One in August 2018[48], created a minor AR boom by being released for commercial and professional use. Prior commercial solutions were to use either a tablet or a Heads up Display (HUD) to accomplish a similar thing. In contrast, these new AR headsets enable us to portray AR elements directly in our perspective without the need for external handheld devices or fixed screens. This gives us the freedom to move around and interact with our surroundings along with the ability to use gestures to interact with the AR objects. Considering professional use cases we can portray virtual guidelines, system statuses, notify the user of imminent dangers and much more. This thesis looks further into the usage of AR and how it can be applied for offshore crane scenarios.

1.1 Motivation

The initial idea of using AR for offshore cranes was provided by Offshore Simulator Centre (OSC) [4]. Their idea was to use their simulator software to predict positions of the load hanging in the crane and then display it with a Microsoft HoloLens. The user of the HoloLens headset would then be able to see what the predicted outcome is from his/her input from within the crane's cabin. As part of this project would this thesis focus on how to implement connecting the HoloLens to their simulator software and as additional topic find other potential things that can be displayed with the HoloLens. For this, OSC provided a Microsoft HoloLens headset and access to a crane dome-simulator for testing.

1.2 Problem Formulation

Can Augmented Reality function as a visual aid for crane operators to increase situational awareness?

Problems to be addressed

- Find specific cases where visual problems occur within crane operations
- Suggest and develop prototype Augmented Reality software for said cases
- Analyze the usability of the solutions made

1.3 Literature Survey

A preliminary study was conducted on Augmented reality and its usage in offshore applications prior to the writing of the thesis. This study functioned as a basis for the knowledge on Augmented Reality. The case of offshore cranes was a later revision suggested by OSC. This resulted in a continuous literature survey throughout the writing of the thesis.

1.4 Objectives

This thesis will explore the usability of the current state of AR and how AR can be used for offshore crane operations. The focus will be to find AR solutions for real-life problems in crane lifting scenarios. These solutions will be considered as standalone solutions, independent of if they are viable with today's AR technology. Current technologies will be proposed for the given solutions and shortcomings will be discussed. The research questions for this report are:

1.4.1 Research Questions

- What problems in offshore crane operations can be addressed with Augmented Reality?
- How can Augmented Reality be implemented for offshore crane operations?
- What are the advantages and disadvantages of using augmented reality in crane operating scenarios?

1.5 Approach

Considering the three research questions there are several steps needed to our approach. The first question requires preliminary research in what Augmented Reality actually can be useful for. This will need research in typical crane operating scenarios where the objective is to find potential problems that can be solved with AR. These problems are then to be addressed with proposed AR solutions. This will be an ongoing process where solutions will draw inspiration from both the research and from the potential implementation itself. This bleeds into the next research question where we implement said solutions. The goal is to find out if these solutions are both feasible and viable with today's technology and a study has to be conducted to find the shortcomings. An iterative coding process will be used to find how the solutions can be implemented. If any limitations are found these should be addressed as part of the discussion. Lastly, a study has to be done on the general usefulness of AR and if AR itself is viable in crane operations. Considering that AR is purely a visual tool, great care has to be taken into how we want to measure the usefulness. For this purpose, a survey was chosen to get subjective data on the use of AR.

1.6 Limitations

The following limitations have been applied to this project:

- This thesis is restricted to offshore knuckle-boom pedestal cranes that have a crane cabin.
- The implementation will only be tested in a crane simulator.
- Microsoft HoloLens will be used as main augmented reality tool, but solutions will be theorized to work with the concept of augmented reality in general.
- Because of difficulties in getting crane operators on site, test subjects are only shown a digital video of the implementation.
- Because of the nature of crane operators work, crane operators are generally busy and/or reluctant to participate in surveys. This made for a small data set.

1.7 Assumptions

- We assume that current Augmented Reality technology is a perfect holographic technology.
- We assume that the simulator used gives us accurate representations of real-world crane-scenarios.
- We assume that at a minimum, 5 years experience is seen as having expertise status. (Based on the survey participants with the least experience)

Chapter 2

Theoretical basis

2.1 Coding constants

2.1.1 Most common C# data types sorted by size

Variable	Size	Type	Max Digits
Bool	1 or 4 bytes	Boolean	1 (zero and one)
Byte	1 byte	Unsigned integer	3
Char	2 bytes	Unicode character	1 char
Short	2 bytes	Signed integer	5
Float	4 bytes	Floating point	7
Int	4 bytes	Signed integer	10
Double	8 bytes	Floating point	15
Long	8 bytes	Signed integer	19
Decimal	16 bytes	Precise Decimals	28-29

2.1.2 Quaternion

Quaternions are mathematical notations that represent orientation and rotation in 3D space. Consisting of 3 complex values "x,y,z" and one real value "w", quaternions provide an easy to compute way to calculate rotations in 3D space along with avoiding the gimbal lock problem. The sum of the squares of the x,y,z components always correspond to 1 because they represent

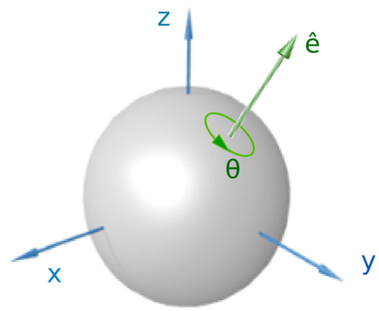


Figure 2.1: Visualisation of the quaternion principal where rotation θ, is applied around unit vector ê, image from [10]

a unit vector. One can think of it like that we rotate the 3D object w units around the vector created by x,y,z (see figure 2.1). Rotating a vector "p" with quaternions is given by the formula:

$$p' = q * p * q^{-1} \tag{2.1}$$

Where q and p are defined as:

$$q = \cos(\frac{w}{2}) + \sin(\frac{w}{2}) * (p_x * i + p_y * j + p_z * k) \tag{2.2}$$

$$q^{-1} = \cos(\frac{w}{2}) - \sin(\frac{w}{2}) * (p_x * i + p_y * j + p_z * k) \tag{2.3}$$

2.1.3 Vector3

A distinction has to be made between a point and a vector. A point is defined as a location in euclidean space while a vector is displacement in space. From this notion does a vector have both a direction and a magnitude. Both are defined with by x,y,z coordinates and can sometimes be used interchangeably if the the point and the vector share a common origin point. In Unity3D both vectors and points are stored as Vector3 Variables.

2.1.4 Stack buffer (LIFO)

A Last In First Out (LIFO) buffer, or stack buffer, is a type of buffer where we store incoming data onto a "stack" and take the last submitted data when we read from it again. The process of writing data with a stack buffer is called Push() and to read is called Pop() (see figure 2.2). These type of buffers are useful when we want to ensure that we always read the newest data first. An example where we can use a stack buffer is when we want to reverse the order of a set of numbers. Filling the buffer with our list of numbers and then reading from it again will result in a backwards order.

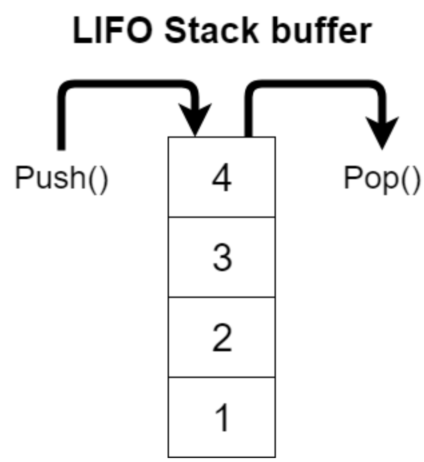


Figure 2.2: LIFO buffer concept, we push the numbers in order first only to take them out in reverse order with pop.

2.1.5 Queue buffer (FIFO)

A First In First Out (FIFO) buffer, or queue buffer, is a type of buffer where we "queue" up data for reading. By writing data to this type of buffer the data will fill up until we read from it again by retrieving the oldest data. Writing data to a queue buffer is called Enqueue() and reading is called Dequeue() (see figure 2.3). An example where we can use a queue buffer type is when we want to execute some commands in order, but some commands take longer time to execute then it takes to receive a new command. Hence we can fill a queue buffer with commands that have to wait for the first command to execute.

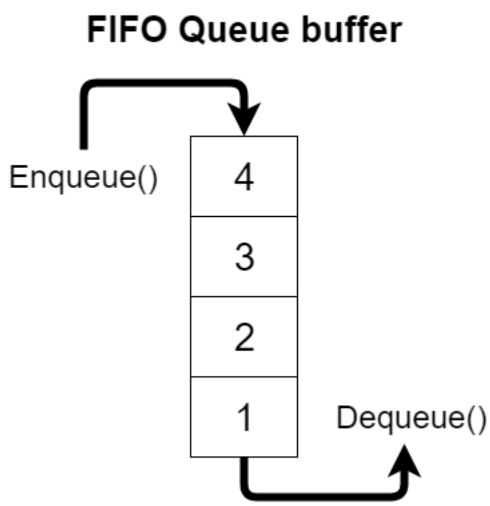


Figure 2.3: FIFO buffer concept, we enqueue and dequeue the numbers in order.

2.2 Math

In this thesis we use vector and byte math. For storing any variable as a string we use the following equation to find the max byte size.:

$$(MaxDigits + DecimalPoint + Sign) * BytesPerChar = StringByteSize \tag{2.4}$$

2.2.1 Vector math

Creating a vector from two points P and Q in 2.4 in 3D space gives us the formula:

$$\vec{PQ} = (Q_x - P_x, Q_y - P_y, Q_z - P_z) \tag{2.5}$$

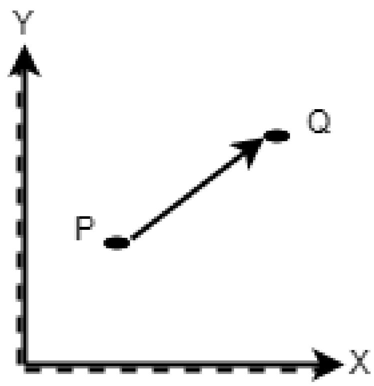


Figure 2.4: Vector from point P to point Q

Scaling a vector V by scalar a:

$$a * \vec{V} = \{a * x, a * y, a * z\} \tag{2.6}$$

Downscaling a vector V by scalar a:

$$\frac{\vec{V}}{a} = \{\frac{x}{a}, \frac{y}{a}, \frac{z}{a}\} \tag{2.7}$$

2.3 Augmented reality

Figure 2.5: The popular Augmented reality phone game, Pokemon Go[16]

Augmented Reality (AR) is a concept where our environment is "Augmented" by virtual objects. The term "Augmented Reality" was first coined by Thomas P. Caudell and David Mizell in 1990 [19][18], but the technology was only made viable through recent advances in computing technology. Augmented Reality is part of the Reality-Virtuality continuum (figure 1.) as part of Mixed Reality. As opposed to Augmented Virtuality, where we add real objects to a virtual space, and Virtual Reality, where we are fully immersed in a virtual space, Augmented Reality solely adds objects to our real environment [51].

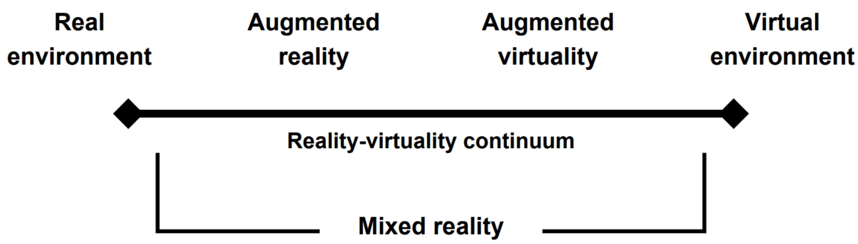


Figure 2.6: The Reality-Virtuality continuum[51]

Virtual augmentation can be accomplished by a wide range of technologies such as Head-Mounted displays (HMD), Heads-Up displays (HUD) or other display technologies. As an display technology, AR adds "Holograms" (see section 2.4) to our surroundings by either projecting or displaying a virtual object. The current most widespread solution of AR is to use either a phone or a tablet. The use of a phone/tablet as AR device has been largely popularized by the gaming industry with games such as Pokemon Go [16](see figure 2.5). In this case the phone/tablet functions as a video see-through device. The user would film their environment and the phone/tablet would display the recording along with additional virtual objects. AR apps would search for reference points in the recording and anchor the virtual object to those points. All modern phones/tablets include a built in gyroscopic sensor that can stabilize the virtual object on the screen further. Another common implementation of augmented reality is HUDs. HUDs consist of a piece of semi see-through mirror called a "Beamsplitter" that functions as a reflector. The virtual objects we want to see are projected onto the beamsplitter by the use of mirrors, projectors or directly from a display. This idea builds on the Pepper's Ghost Illusion as explained in section 2.4. HUDs are commonly used in aviation and traditionally used as fighter pilot situation displays [18][46]. They are also found in cars to display directions and speed of the car(see figure 2.7). A new and upcoming Augmented reality technology is HUDs incorporated as Head-mounted displays (HMD). With HMDs the beamsplitter is wearable as a pair of glasses. HMD's make it possible to superimpose AR objects directly on our surroundings, giv-

Figure 2.7: Hudway car HUD [32].

Figure 2.8: A screenshot from the movie "Ironman 2", showcasing a blue hologram of the iron-man suit.

ing a "true" immersive AR experience. But because it has to be wearable, every component has to be tiny. E.g the Microsoft HoloLens uses tiny pico-projectors and specialized waveguides to project the light onto the beamsplitters [30](see section 3.2.2 for more info). HMD's also require advanced tracking capabilities to be able to project stable AR objects, or else the user might experience HMD sickness [49].

2.4 Holograms

In science fiction movies such as in Star Wars and Iron Man (see figure 2.8) we have these magical floating 3-dimensional objects made of light with which we can interact [52]. These objects are called holograms. Holograms are as Microsoft puts it: "3D objects made of light or sound" [25]. More technically this means that a hologram is a recording of the light that an object scat-

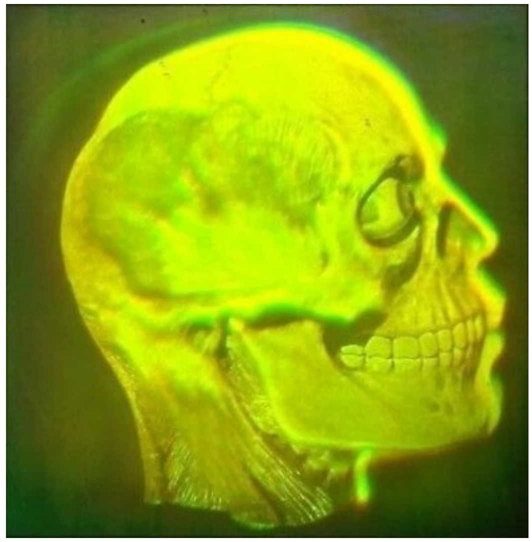


Figure 2.9: A rainbow hologram of an anatomy head.[2]

ters or emits. By encoding this recording on a medium we get a 3D optical representation of the object. The word "hologram" can either be referring to the encoded material or the optical representation itself. Typically when we refer to a hologram we refer to the 2D holographic image such as a rainbow hologram [31](see figure 2.9). These holograms are made by bouncing a laser of an object onto a medium to capture an interference pattern that scatters light similarly as the object. Because of the expensive setup, high precision and reference object needed to create these images, this technology is used in many security applications such as bank notes or credit cards. In the case of Augmented Reality, we instead capture the form of a virtual object. As explained in section 2.3 AR headsets project the virtual object onto glasses to create the 3D hologram. This is, in reality, a pseudo-hologram as AR doesn't give an actual representation of the light scattered by the object, only the perspective. It only looks like a 3D object because we can generate what each eye perceives to create a similar perspective. In reality the object we see consists of two non-holographic 2D images of the object. This is similar to the Pepper's Ghost illusion where we project a copy of the object with mirrors onto a glass pane(see figure 2.10). As with AR headsets, the Pepper's Ghost illusion is not a real holographic solution. Pepper's ghost illusion was first popularized in 1862 to create a ghostly effect on theater stages [45] but is now also used in museums and for HUDs. For ease of understanding, we assume that AR generates real holograms and will call the virtual objects displayed with AR for holograms in this thesis.

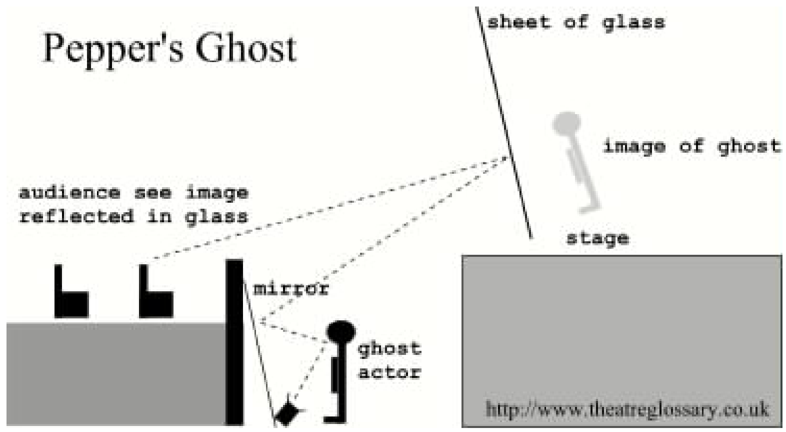


Figure 2.10: Pepper's ghost illusion being used on a stage[45].

2.5 Offshore Cranes

Offshore cranes are a type of pedestal crane that can be either mounted on a ship, barge or on an oil platform. The advantage of mounting the crane on a pedestal is that it enables us to theoretically rotate the crane 360 degrees. The downside of this, on the other hand, is that these cranes only have a certain reach. This type of crane is typically used to transport materials and personnel to platforms, marine vessels, and other structures. There are two types of pedestal cranes used on offshore vessels, normal pedestal cranes and a stiff-boom variant for heavy lifting. The general structure of these two types remains largely the same besides that on a stiff-boom variant the boom is typically lifted through hydraulics as opposed to cables. Pedestal cranes consist of 4 parts; the pedestal, a crane cabin, a boom that can consist of one or two parts, and the hook and hoist-rope structure. Offshore pedestal cranes also often have an extra auxiliary hook for better control of the load.

A point of interest for this thesis are crane cabins and the crane's controls. This is the main interaction hub for the operator and is what we want to improve. Visibility in the cabin is usually restricted by the physical shape of the cabin, resulting in a field of view (FOV) of only 60 degrees horizontally and 50 degrees vertically [34]. Controls are usually grouped together and centered around where one would place your hands on the armrest of the operator's chair(see figure 2.12). The main controls are two joysticks for Boom, Knuckleboom (If the crane is a stiff-boom variant, otherwise its Boom-extension), Slew and Hoist motions. The joysticks are accompanied with buttons for starting/stopping the crane, along with alarm buttons and emergency stop. In our

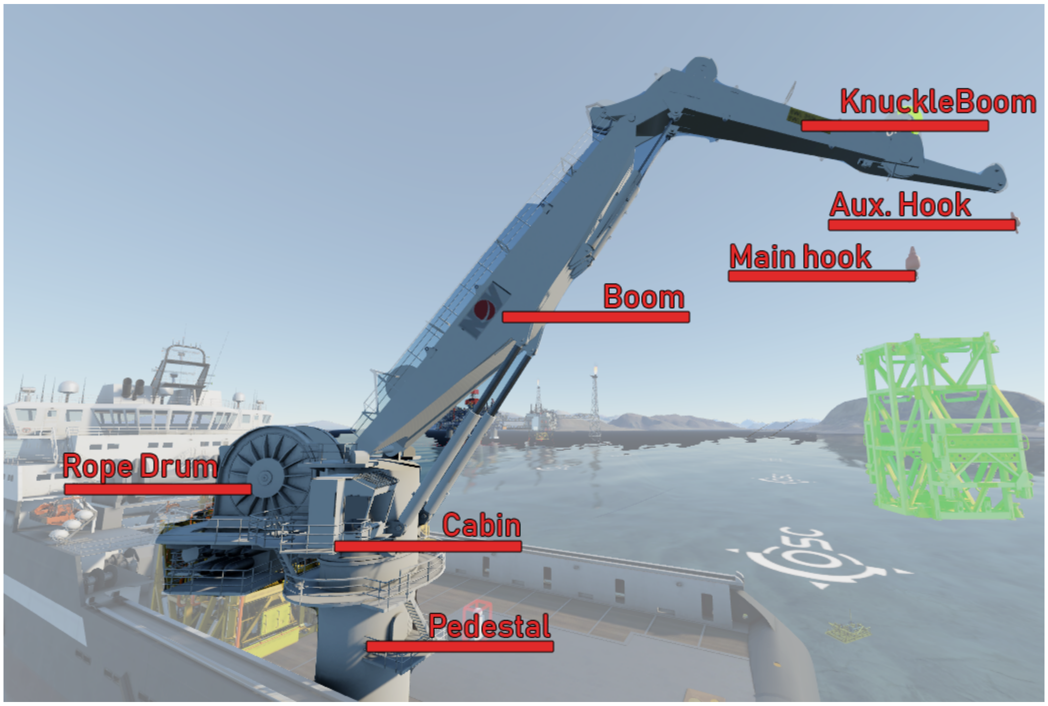


Figure 2.11: An offshore pedestal crane as seen in OSC's simulator software.

case, there is a CCTV camera showing a top-down view of the load situated in the top right corner of our main window pane and a tablet with Crane-level, load, Safe Working-load level (SWL) and current active alarms situated on the left armrest of the chair.

2.6 Tracking possibilities

The notion of tracking is based on the idea to find a specific position of an object relative to reference points in 3D space. A typical example is the Global Positioning System, or GPS, that uses known locations of satellites as reference points to geolocate vehicles. Many technologies enable us to track the position of objects for a wide range of scenarios. In this project, we are looking at tracking technologies that are specified for use in offshore scenarios and more precise solutions to accurately detect positions within a room. Three types of tracking possibilities were considered. For offshore applications, we have several options available that provide tracking in the form of encapsulated devices that are mounted on the object we want to detect. Two commonly used tracking devices are MRU's and Radar systems. A Motion Reference Unit (MRU) is a self-contained unit that provides positional and rotational data through a wired con-

Figure 2.12: An operator chair as seen from above. Note the controls on each armrest and an alarm system tablet mounted to left. Picture was taken at OSC.

nection. The main purpose of these units is to supply tracking data for systems such as Heave compensation systems, ROV orientation tracking, antenna stabilization, and Dynamic Position (DP) systems [36]. Tracking with these type of units is done by logging the displacement from an initial measured position that functions as a reference point. Radar systems, on the other hand, work with external reference points, also known as radar beacons. These systems create what is known as a Relative Positioning Reference System (RPRS) networks where a transponder mounted on the object we want to track responds to radio waves from the various beacons. The timing of the response is then used to triangulate the position of the object through interferometric methods [37]. The general use case for radar systems is to locate ships and smaller boats in a range of approx 500m and is generally less precise and responsive as MRU's.

As for our accurate tracking possibility, we look at another commonly used tracking method called image recognition. With image recognition, we use algorithms to detect certain reference points that can be seen in images or video footage. Typically are QR codes used as reference points because of how easy they are to distinguish from the surroundings. Considering that images are 2D there are several steps needed to create an accurate 3D representation of the position. The simplest solution is to use multiple cameras from various angles as reference points. Measuring the distance from the camera is detected by looking at the size of the reference image. More advanced image recognition algorithms such as Vuforia can use a single camera to accurately describe a position relative to the room that is recorded.

Chapter 3

Materials and methods

3.1 Simulator

The simulators provided by OSC are multi-functional offshore scenario simulator used for training purposes. The advantage of using simulators for training purposes is that any type of scenario can be loaded, making it easier to train for dangerous and otherwise rare situations. OSC's solution to simulating offshore environments is to install entire ship-bridges or crane-cabins into large fiberglass domes(see figure 3.1). The virtual world is projected on the inside walls of the dome, creating a encapsulating virtual environment. Surrounding the user with a projection of the virtual world in such a way creates high immersion through replacing the real world with a virtual one.

Figure 3.1: Dome crane-simulator by OSC [3].

The simulators can load any scenario given by a scene file. Scene files consist of a snapshot of the simulators state at a given time. The core of the simulator is a stand-alone server. This server will handle all the physics of the scenario and store the positional data. Any visualization of the scenario is done with external visualization clients that connect to the core. For visualization purposes do all clients need to have a local repository of the 3D models that are present in the scenario. The core and the visualization clients communicate through RakNet networking protocol.

3.1.1 How the simulator will be used

The simulator would be used for various purposes throughout the thesis. First and foremost does the simulator provide a testing environment for any prototypes build. By being able to load the specific scenes are we able to highlight bugs and proof test the application. This also lets us try out the problematic scenarios that are found through research. Another factor is that by using a simulator for both the visuals for the physical simulator dome and as a source for displaying any information we can measure any potential delays that are induced by the software. By having access to a simulator we can also ask people to come to try out the application and give a verbal opinion of the software while the software is in production. Being able to run the same scenario can also be useful for testing with test subjects, making sure that the test is similar for every test subject.

3.2 HoloLens

The HoloLens is a standalone Augmented Reality headset that was released by Microsoft in March 2016[22] at a launch price of $3000 for the development kit. The headset itself consists of an adjustable headband with a large visor attached to the front. The overall weight of the headset is 579gram. The top of the visor includes an array of sensors while the lower part has two small windows that function as our AR view. The sensor array consists of an IR depth camera, 4 IR emitters, 4 environment understanding (grayscale) cameras, a 2.4MP RGB camera and an Inertial Measurement Unit (IMU). Along with that, it has 4 microphones, an ambient light sensor, built-in speakers, volume and brightness buttons, Wifi 802.11ac, Bluetooth 4.1 LE and a battery

Figure 3.2: The Microsoft HoloLens.

that lasts up to 5hours depending on use. For computing, it has an x86 1Ghz Intel Cherry Trail CPU, a Holographic Processing Unit (HPU), 2GB RAM and 64GB flash memory[53].

3.2.1 How the HoloLens tracks

All of the sensors are used to find both the orientation and position of the headset relative to the room you are in. When entering a new room, the HoloLens will first try to "map" the surroundings with the IR depth camera and make a 3D representation of the space. This spatial map will be linked to the current Wifi connection the HoloLens has so that it always knows which room it is in by looking up the Wifi name. The IR depth camera will continue updating the map and also provide tracking of hand-gestures made in front of the headset. While this was happening the IMU would try to find the orientation of the headset. After a map has been made, the 4 environment understanding cameras and the RGB camera are enabled to help the IMU find the orientation of the headset by comparing image data to the map. Because of the orientation of the 4 environment understanding cameras, they will continue tracking when a gesture is performed and the IR depth camera is blocked. This greatly increases the IMU's stability in finding the orientation of the headset even when gestures are performed. Because of the heavy reliance on both the grayscale and RGB cameras, the HoloLens will work best in well-lit areas[15]. This also means that dark-colored objects can be hard to track and are seen by the headset as gaps in the spatial map.

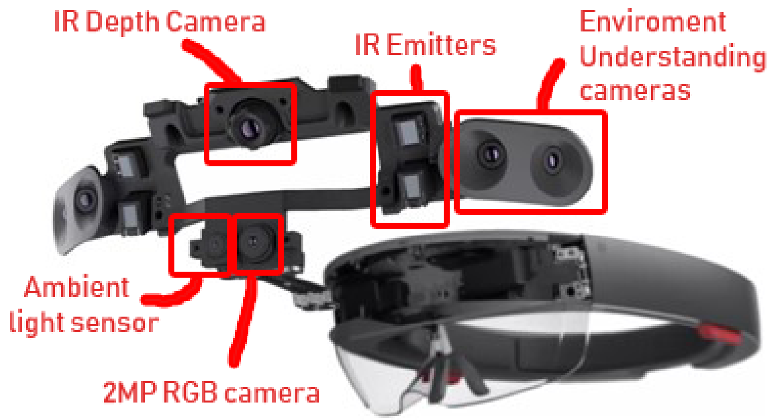


Figure 3.3: An exploded view of the front of the HoloLens, showcasing the sensor array [53] [55]

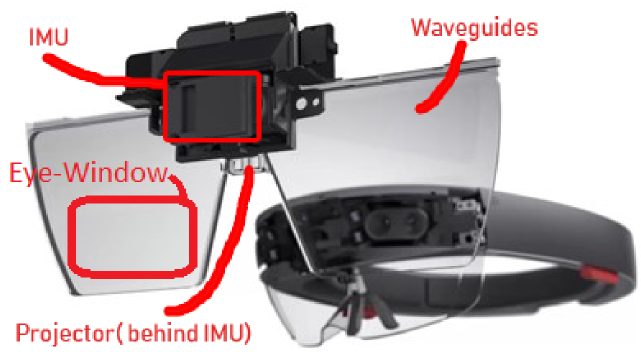


Figure 3.4: Hololens's optics system.

3.2.2 Optics

The viewing portion of the visor from the HoloLens consists of two See-Through Holographic Lenses, or waveguides[20]. A tiny pico projector is situated on the bridge of your nose that projects directly into both waveguides. The waveguides guide the light to two small windows, one for each eye, where the AR objects are projected on[30](see figure 3.4). Each window has a field of view (FOV) of 30* degrees horizontally and 17.5* degrees vertically. These windows consist of 3 layers of beamsplitters that only reflect either Red, Green or Blue colors. The projector will iterate through each color at 240Hz "layering" the colors to get a fully colored image. The actual screen frame rate is 60Hz. Because the display is a sequential display it is possible to get color separation when the user moves his/her head rapidly. This results in the AR objects "trailing" receiving a rainbow effect.

Figure 3.5: An AR object trailing[35]

3.3 Unity3D

Unity3D is a game development platform used to build 3D and 2D games. Unity itself is a specialized game engine developed by Unity-Technologies which can run on multiple device platforms. At the time of writing, Unity supports up to 25 different platforms[50]. In 2018 Unity worked as a basis for 50% of all mobile games and 60% of all VR/AR applications[13]. Along with the many devices, it supports it also natively supports tools such as VR, AR and image detection. The editor comes with an easy to use interface that lets you drag and drop scripts and models into your scene. The editor also includes its own asset store where you can easily obtain scripts, models, and shaders for your project. For this project Unity version 2018.3.3.f1 was used.

3.3.1 Unity Scripting

Scripts are written in C# and are attached to "GameObjects" within the scene. Visual Studio is the primary IDE used to write these scripts and is included with the install of Unity. Most of Unity's scripts derive from a base class called MonoBehaviour which gives us access to basic controls of Unity's physics. A typical MonoBehaviour script has two main functions, Start() and Update(), that lets us run physics based on the engine's update rate. The Start() function readies all variables when the engine starts and Update() is called for every game tick. There are many other default functions that get called by the engine but these are just the basic functions that allow us to run our physics.

3.4 Vuforia

Vuforia Engine is a computer vision library that can track known tracker-images in real time and anchor AR objects to these trackers. Vuforia can track up to five image targets simultaneously and detect up to 1000 different images with a single app[8]. Using any type of device camera, the Vuforia Engine runs in the background of your app to search for image matches from a predefined image database. Image databases can be generated by making an account on Vuforias webpage and uploading the images you want to track. These images should be a minimum of 12cm wide and a general rule from Vuforia is to make the target at least 1/10th the size of the total distance between camera and target. Meaning, if we want to detect a target 3m away from the camera, the target has to be around 30cm wide for optimal detection[8]. Image targets aren't limited to pure black and white images such as QR codes or data matrix codes[5]. The images can be anything that has a known size in the real world and has some discerning features. For optimal tracking, it is recommended to use images with good contrast which have non-organic and non-repeating shapes to optimize the number of discerning features the engine can detect[7]. Another consideration is the lighting conditions. Vuforia is mainly built for well-lit scenarios such as indoor environments.

Another feature of the Vuforia Engine is the Extended tracker feature. Extended tracking enables to keep the AR objects position compared to the environment even when the target goes out of view. This is especially useful for larger 3D objects where the user has to be able to look around without having the target within the camera view. The built-in Device Tracker for each device is used to continue the tracking of the AR object when the target goes out of view. For the HoloLens this is the Spatial mapper feature. Extended tracking takes advantage of the HoloLens's spatial mapping by converting the tracker's position from Vuforia tracking space to HoloLens spatial mapping coordinate space and providing it to the HoloLens. This allows the HoloLens to take over and project the AR objects in our view space without needing to care about targets and instead use the HoloLens's tracking capabilities for better stability[6]. A downside of Extended tracking is that it expects that the targets are stationary relative to its environment[5].

Vuforia introduced support for HoloLens with version 6.1 and enabled their Extended Tracking feature by version 7.2. As of Unity 2017.2 Vuforia is also integrated directly into Unity.

3.5 Blender

Blender is a free and open source 3D creation tool by the Blender foundation [26]. Blender is used for creating 3D models, model rigging, visual effects and rendering of images and videos. Blender has cross-plattform support for Linux, MacOS and Windows.

Key features as described in their documentation [27]:

- Blender is a fully integrated 3D content creation suite, offering a broad range of essential tools, including Modeling, Rendering, Animation & Rigging, Video Editing, VFX, Compositing, Texturing, and many types of Simulations.
- It is cross-platform, with an OpenGL GUI that is uniform on all major platforms (and customizable with Python scripts).
- It has a high-quality 3D architecture, enabling fast and efficient creation workflow.
- It boasts active community support, see blender.org/community for an extensive list of sites.
- It has a small executable, which is optionally portable.

For this project, Blender 2.79b was used.

3.6 Google services

Two services by Google were used in this project. The first is to create the survey for questioning crane operators. This service, called Google Forms, is one of 4 administration applications included with Google's Drive Office suit. Using this application we can easily generate a digital survey or questionnaire consisting of a wide range of question types:

- Short answer
- Paragraph

- Multiple choice
- Checkboxes
- Dropdown
- File upload
- Linear scale
- Multiple choice grid
- Checkbox grid
- Date
- Time

The second service that is used is YouTube. YouTube is a digital streaming service where you can upload and share videos. This is used to upload an explanatory video of the solutions created in this project. Uploading the video to YouTube also made it possible to implement the video directly in the digital survey created with Google Forms.

3.7 Project Organisation

Prior to the thesis, preliminary research was conducted in the field of AR usage on ships in general. This was later changed and narrowed down to the specific case of offshore cranes. Because of this change, further research was required specifically in crane scenarios. This results in the need for continuous research throughout the project. A meeting schedule consisting of a single meeting every second Tuesday was made with the supervisors to ensure constant feedback. Several digital systems were also used for easier access and writing. The thesis itself was to be written with LaTex on Overleaf.com while the school provided an Atlassian Confluence service for documentation. A 2-week retrospective document was made before every meeting which was easily shared with Confluence. Meeting notes were also published in the same fashion after the meetings were completed. Confluence also provided common storage space for research documents that had been found, along with general notes taken throughout the thesis.

3.8 Programming Approach

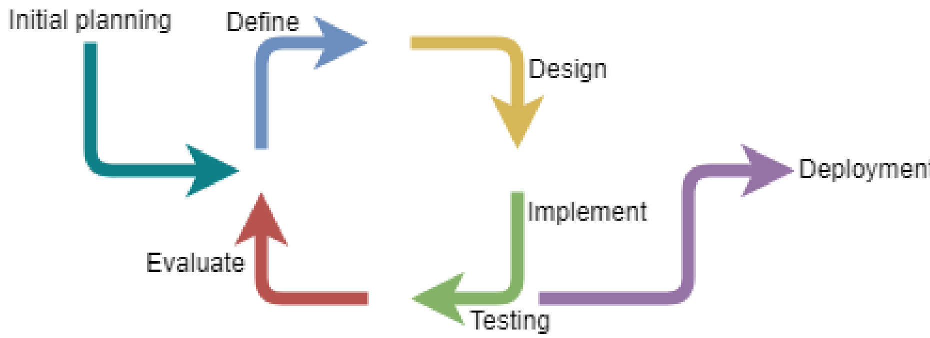


Figure 3.6: The iterative cycle

For this project, the Iterative and Incremental programming model was used. As opposed to the Waterfall model, where we add and perfect functionality step by step, the Iterative and Incremental model doesn't start with a complete requirement list. Instead, the requirements are gradually adjusted to better fit the final product. The Iterative process is a repeating cycle where we improve a part of the software until it works at a bare minimum (see figure 3.6) and with incremental we mean that we complete the software in various small steps, adding single features in each step [24][28]. Combining these two we get a process where we repeatedly cycle over the various parts of the software to ensure functionality. The cycles are also not bound to start only after the last cycle has ended. Multiple cycles can run in parallel and improve each other over time. The advantage here is that with each iterative cycle we gain experience in the software's application which can lead to new discoveries and ideas. This creates an environment where we can find problems in the early development stages of features that can be corrected before their deployment. As a result, the solution evolves over time, becoming more suited for the final application as time goes on.

Typically does a cycle consist of 4 phases (definitions based on [28]):

- **Defining** phase where we define the bare minimum requirements for the software or feature.
- **Designing** phase where we propose a solution for the requirements. Here we usually define a new solution or an extension to a previously completed piece of code.

- **Implementation and testing** phase where we code, integrate, and test our proposal.
- **Evaluation** phase where the implementation is reviewed and changes to the requirements are proposed.

The iterative and Incremental model was chosen because the end result of the software was uncertain at the start of the thesis. We know beforehand that we want to solve a crane scenario problem with AR, meaning that we need some knowledge on how to implement software for AR technology. By making this model a part of the research on AR, we can use the iterative cycle to come up with ideas that are possible with today's technology. Looking at each phase, we can start with defining our minimum requirements. Even though we are uncertain at what the final solutions will be, an iterative cycle can be started early for finding how to utilize the HoloLens. With this initial cycle, we can learn the software environment and bring forth potential ideas. The continuous feedback from supervisors could also be used for the evaluation phase of the cycle.

A larger part of the thesis's research period was used for researching how to use the HoloLens. Due to the HoloLens only being available for two years and its high cost, there have been little resources available for non-generic use cases. Microsoft also typically writes their API documentation explicitly for professionals making it harder to understand for entry-level programmers [12]. There was also a lot of miss-information floating around on the internet after some larger updates removed features that were previously available. All of this resulted in a longer than expected programming period on the thesis.

3.9 Collecting data

To answer the research questions we need to know how useful AR is in crane operations. This can be seen in two separate ways. One was we test if AR, in general, is a viable technology for use in offshore cranes, and two were we look at specific solutions created with AR technology and their usefulness. We can find the general usefulness of AR by researching the status of today's technology, how it performs and how it is implemented. Testing the usefulness of the solutions, on the other hand, requires insight from crane-operators to be able to draw conclusions. For this, we can collect both quantitative and qualitative data, but quantitative data requires ways of measuring improvements by using AR. Several ways to collect quantitative data were considered such as eye-tracking and measuring joystick usage. These measuring solutions require either expensive hardware or advanced algorithms that are able to detect potentially small improvements. This was out of the scope of this thesis and the focus was instead to gather subjective data from the crane operators. The main way to gather this was through the use of a digital survey.

3.9.1 Gathering test subjects

The test subjects for the thesis were crane operators. Gathering these test subjects involved getting in contact with the crane operators directly or with people who had crane operator contacts. Considering that OSC had built the crane-simulator we were testing with, they initially became the primary source for getting in contact with crane operators. They did have some crane operator contacts but most where either reluctant or to busy to participate. Initially, the idea was to gather data by interviewing the test subjects and have them try out the solutions. But because of the difficulties of getting in contact with crane operators because of their tight schedule, a digital survey was made instead. This meant that the survey could be sent out to the various test subjects for them to try out whenever they had time. Yet many of the crane operators that were contacted were still reluctant to participate, mainly because of the time it would take to complete the survey. Nonetheless, after the survey was posted on a Facebook group for crane operators, I was contacted by several crane operators willing to participate. Without asking the participants directly, it is uncertain if they are in contact with each other leading to potential biases. As the data will be gathered from people who are members of a Facebook group consisting of people of the entire nation, we can assume that every participant is unbiased. At this point in time, there are 5 participants that have completed the survey.

3.9.2 Survey structure

The survey was a 3 part (4 parts if you include the introduction to AR) questionnaire consisting of 35 questions that in total takes about 30 minutes to complete (See figure 3.7 for the parts).

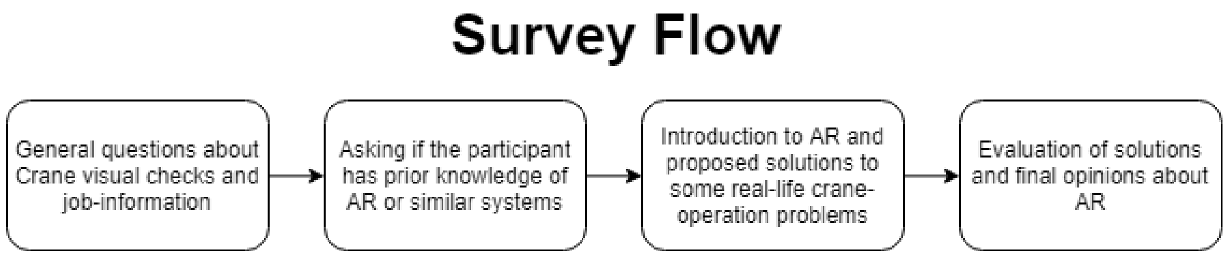


Figure 3.7: Flow of the survey

It was based on the idea that we want to show AR as a new product for crane operations. This meant that we could base it off product evaluation surveys structures. Inspiration was drawn from the "before and after the product has been used" strategy. The survey was built with a "before" section looking into what the participant thought that crane-operations lacked information while in the "after" section we asked if AR could solve these problems. The first section of the survey was intentionally generalized so that the participant can be creative and give potential further ideas that AR might be able to solve. This meant that many of the answers in the first part are given as open paragraph answers which have to be summarized. To reduce the potential of the participant being biased by prior experience with AR or the opposite, being blown away by a cool new technology, a section that looks into their stance on AR was included before the "after" section. The "after" section gave an introduction to the solutions and what AR is all about and would then go on to ask the user of their evaluation. As opposed to the first part many of the questions in the second part are multiple choice questions that strictly ask if they like it or not. These questions are accommodated with optional fields for arguments if the user feels like they have more to say about the subject. The general usefulness is rated on scales of 1-5 based on the keywords: Not at all, Slightly, Moderately, Very and Extremely.

3.9.3 Analysing the data

Most of the data consists of qualitative data in the form of small texts. For small amounts of data, we can directly address some of the answers and discuss their meaning for the project. If sufficient data is gathered the answers can be summarized to reveal the essence. The questions themselves have to be analyzed in correspondence with the answers to get a feel of what the participant thought when he/she answered the survey. The mindset of the user at the time of taking the survey can reveal some insight into their understanding and interest of the survey, which can strengthen their answers or invalidate them. Oral feedback by both crane experts and users who aren't directly involved should also be taken into consideration. This will give some insight into the general interests on the topic along with insight from people who think differently. The use of a digital survey can also be seen as limiting because of language barriers which should be considered. The value of their answers on the other hand also relies on their creativity and willingness to answer the questions. As part of the GPDR message included, the data gathered would be deleted after they had been analyzed. This is put in place to potentially appeal to more people but will result in the analysis only consisting of summarizing and little actual data to be shown in this thesis.

Chapter 4

Results

4.1 Problem descriptions and the AR solutions

Through literature reviews and suggestions by OSC, three crane situation problems were addressed. The first problem was adapted from the initial project by OSC. OSC's goal is to have a HoloLens application which can display the predicted position of the load on a crane ahead of time based on the current operator's input. Because there is a delay between input and reaction of the crane this solution could show the operator ahead of time where the load will end up. Predicting the position of the load is outside of the scope of this thesis and will be handled by OSC's specialized crane simulator software. This leaves us with only making an HoloLens visualization solution for the simulator. Instead, the idea of predicting the load's position was adapted to fit another common problem with crane operation.

4.1.1 Hologram on load

The first solution was based on the problem that the load hanging in the crane can be in a blind spot of the crane. Blind spots do as the name entails, render the user blind on what is happening. These usually occur when the load has to be placed behind a structure, on another ship or underwater. Generally to avert this problem a Camera is added on either the hook or the boom of the crane to give a better overview. For underwater operations, the solution is instead to use Remotely Operated Vehicles (ROVs) as cameras. This is called a CCTV system. The crane oper-

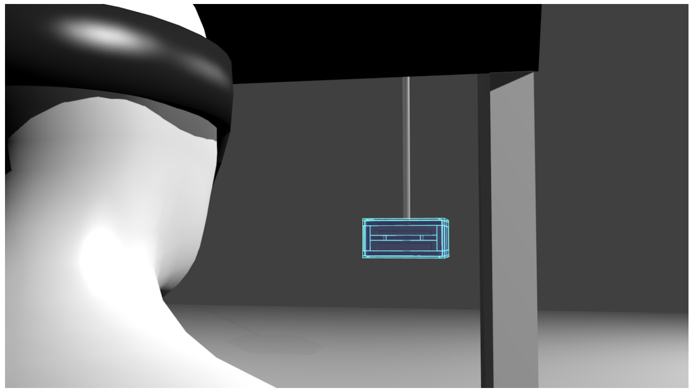


Figure 4.1: Concept of wearing an AR headset and seeing a blue hologram "hanging" on the load in the crane. Created with Blender

ator can see the camera feed on a small screen usually located in a corner of the main window of the cabin. Working with this screen doesn't give the operator a natural feel of how the load is positioned. A proposed solution is to make an AR indicator, or preferably an exact model of the load, to project the position and rotation of the load directly on the real world. By using an AR headset we can always show where the load is located by moving a projection of the load to where the actual load is. For instance, if the load happens to be behind some containers, we would still see the load's AR projection.

4.1.2 AR information widget

Another problem that was addressed was the accessibility to information. Any system or scenario information is usually situated around the control chair of the operator in the form of screens. In order for the crane operator to read this information, he or she needs to avert his or her focus towards these screens. This can be seen as a minor interaction but the attention on the load is lost during this process. The simple solution for this is to move the information we need towards the load. Using AR we can attach the information as a "widget" directly to the load so it is always in view. The widget can consist of a small 2D window that floats after the load. This widget does have to float because the load will most likely not be stationary. Making the widget float after it instead makes it more stable. Microsoft calls this function for "Billboarding" and is usually incorporated into menus to make them more stable in view [14]. As opposed to Microsoft's solution where the info is attached to our view, we want to attach it to the load. As for the information we might want to show there are a lot of options. There are 3 major things we can focus on, static information about the load and crane, dynamic information about the load and crane, and information about the scenario. The information gathered here is based on the input from OSC and the results from the survey described in 4.7.

- Static crane information examples:
  - Weight of the load, regardless of if its an internal or external lift
  - Maximum wire strain allowed
  - Boundary limits and area information, such as maximum allowed height or danger zones
  - Balance point of the load
  - Load contents
  - Connection point (Main or auxiliary hook)
- Dynamic crane information examples:
  - Current safe working load value (SWL)
  - Strain on the wire
  - Alarm systems
  - Crane sensor status
  - Velocity and acceleration of the crane or load
- Scenario information examples:
  - Wind and weather conditions
  - Distance from load to destination
  - Boat sway
  - Wave height

The suggested widget solution consists of only a small screen meaning that we cant show all information at once. The information given should be divided into several sections for the user to go through. An idea here is to use voice commands to bring up the needed information. It is better to use voice commands in this case as gestures or other kinds of interactions would result in the operator needing to stop interacting with the crane controls. Further research on which information is more important than others and what can help the operator with his/her job has to be done.

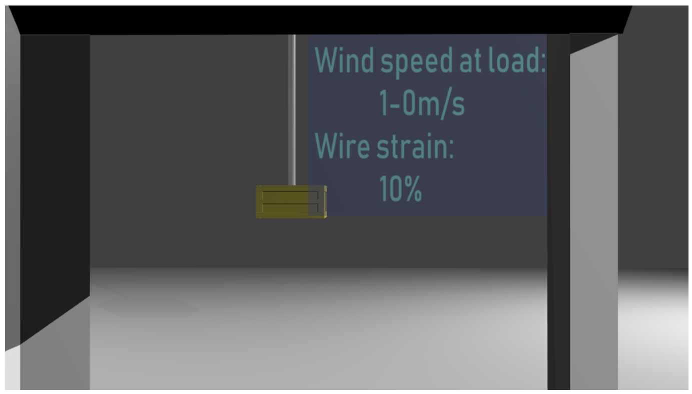


Figure 4.2: Concept image of Information Widget, it can basically be seen as a floating display.

4.1.3 Miniature boat

The third and last problem addressed was that it is sometimes hard to get a good perspective of the scenario. Sitting in a crane's cabin usually gives the operator a poor perspective of what is happening. Looking straight forward we lose the sense of how far away the load really is. If there are no direct reference points to look at we are forced to guess where how far away the load is located. As previously explained in 4.1.2 we also know that whenever the load disappears in a blind spot we use CCTV to compensate for the lack of vision. These cameras are usually pointing downwards giving the crane operator a top-down view. The problem here is that a top-down view only gives the user a 2D perspective. As it is a top-down view it could be hard for the operator to guess how far up the load is hanging in the crane. To circumvent these problems another solution was proposed. By adding a virtual miniature AR model of the entire ship, or at least the crane and the objects surrounding it, to the crane's cabin we can give the crane operator a better overview. The miniature model of the ship will be updated in real time on

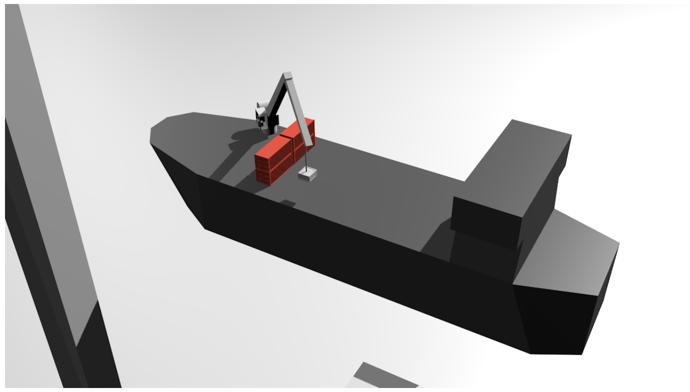


Figure 4.3: Concept of a miniature boat to show perspective. Here the load is behind some containers.

the whereabouts of the load and the crane's position. This essentially gives the crane operator a birds-eye perspective of the scenario when he/she looks at the miniature model. This idea can also be extended to showing the destination of the load in 3D. Considering a scenario where we might need to move the load from ship-to-ship it could be a good idea to see how the other destination ship is swaying. Here we can use the same trick and for instance show 2 boat models where our boat is stationary and the other boat sways. For ease of access can this 3D overview be placed near or replace CCTV systems already installed in the cabin. By the use of gestures might it also be possible to zoom in and move the virtual AR object to where the crane operator isn't distracted by it but can still use it.

4.2 Initial programming cycles

To test out the solutions, a prototype application for the Microsoft HoloLens was made. As OSC's simulator would be the main data source for our solutions, the first focus was to find a way to transfer the data from it to the HoloLens. OSC had previously tried to run their visualization client directly on the HoloLens, but because of how this client loads all 3D models upon startup this wasn't a viable option. Loading all models would've slowed the HoloLens significantly and we want to only display a few models/and or positions anyway. Another thing is that the models we showcase as AR objects in the HoloLens don't necessarily need to be the exact same models as used in the simulator. An example could be that we just want a simple 2D indicator attached to a real object instead of an exact 3D replica of the real object. This was the first topic that was explored through the use of an iterative programming cycle. Through iterations, three options were explored on how to provide the simulator data to the HoloLens. Unity's app streaming feature known as "Holographic Remoting", Streaming over USB, and TCP/IP streaming. It's important to note here that for Holographic remoting and for USB streaming the idea was to send prerendered display data and with TCP/IP the idea was changed to stream only positional data.

4.2.1 Holographic Remoting over WiFi

Unity's Holographic Remoting feature enables us to stream the behavior of an application directly to the receiving device, omitting the need to build and upload our program. This feature is built with streaming display data over WiFi in mind. While this feature is mainly used for prototyping it can also be used to run heavy load software on a powerful external computer while the visualization device only has to display the received video stream. This idea was initially chosen because this made for simple prototyping and the resulting software could work on virtually any AR device, not only the HoloLens. Another advantage was that we could potentially use the visualization client from the simulator directly because all the heavy loading was on an external computer. Holographic Remoting works with two-way communication where the HoloLens sends sensory and control data to the external computer who then processes it and sends back image data [44]. Testing of this feature resulted in mixed results. Due to processing times and network delays, there were some noticeable delays in the response time of the HoloLens. This made it considerably more nauseating to use the headset. Whenever a delay occurred it seemed that either the AR objects had their position "stick" relative to the headset or drifted away from the user. By "sticking" we mean that when you move your head the AR objects would follow and keep their relative offset. As the delays only occurred periodically this would mess with your sense of balance because your brain inadvertently uses the AR object as a visual reference. This creates a nauseating experience similarly to seasickness.

It can be noted that the main router used was connected to the internet which can induce extra delays. This should not have mattered though because of Holographic Remoting being a LAN only solution through an unused port. By using the diagnostic mode of the HoloLens Remoting app and a quick visual inspection we could see that the latency was mostly stable at 20ms although dipping towards 40-50ms at times. This is considered to be within the acceptable delay of 50ms for the HoloLens [54]. It was noted on the other hand that whenever a large delay occurred, the diagnostic mode itself hanged, meaning that we never got to see what the latency was at that period of time. At the time of writing there isn't a way to digitally log the latency over time with the HoloLens. Because of the instability of the image, this solution was considered to be too nauseating and rejected.

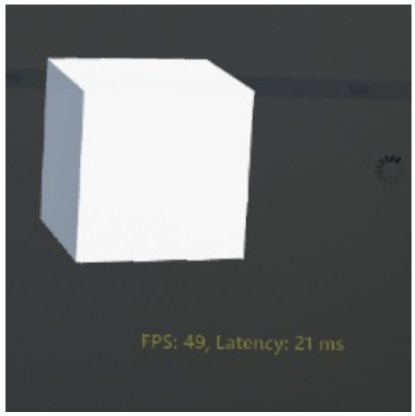


Figure 4.4: An AR object being projected with Holographic Remoting.

4.2.2 Holographic Remoting with IPoverUSB

For the next iteration, the idea was to use Holographic Remoting to stream directly over a USB cable for increased stability. This idea ended up largely consisting of many attempts, or cycles, at implementing this solution. The Holographic Remoting over USB feature was a feature that was described in many forum posts dating from 2017 [41][17][23]. It wasn't a direct problem for this thesis if the headset was wireless or not and because of the cable connection of the HoloLens being on the back, it wasn't really in the way. Using a cable with the HoloLens would essentially result in using USB as a wired cable network also known as IPoverUSB. This could also mean that we could use the USB cable for potential stable TCP communication. It was thoroughly tested if it was possible to use a cable connection but it never worked. Methods that were tested:

- Changing network port and IP address. From the forum posts, it seemed that we needed to use LocalHost (address 127.0.0.1) and port 8000 or 10080 as an entry point to connect with the HoloLens. It was later discovered that in the source code of the HoloLens that it potentially searches for port 8001 instead [42].

- Because it supposedly worked through the IPoverUSB software that is integrated into Windows 10, an attempt was made to make this work outside of Unity. Based on the questions asked by forum user bc3tech [11] and suggestions from [33] the IPoverUSB was ran through command prompt. Settings for this attempt were set in the Windows 10 registry.

- Forum user Lukasz in [11] also suggested using USB type 2.0 instead of 3.0.

- Several USB cables (to be certain that at least one worked)

None of the above-mentioned solutions worked to gain contact with the HoloLens and after some research, it is thought that the USB-cable feature was removed in its entirety as part of the security update 14393.448 [33]. This change wasn't documented in the HoloLens's documentation but is neither explicitly mentioned as a feature at the current time of writing.

4.3 Final data transfer solution: Custom TCP/IP server

The previous attempts at showing data on the HoloLens were all based on letting an external device compute the display data and send it to the HoloLens. Instead, the next idea was to create a custom TCP/IP client and server solution to let the HoloLens itself manage and render the positional data (see figure 4.5 for a connection diagram). The server would be based on the visualization program for the simulator, so that it had direct access to all data from the simulation. A client on the HoloLens would connect and receive the data and process it. After a connection to the server has been requested and accepted, the server will start to send positional data continuously. As opposed to Holographic Remoting this meant that the solution created would be specific for the HoloLens because we need to create a specific client for it. Another consideration we had to take is that because we are counting on the HoloLens being a standalone application, the HoloLens needs to be capable of rendering the 3D models we want to use. On the other hand, because we are sending purely positional data instead of display data we have way less data that has to be sent over the network.

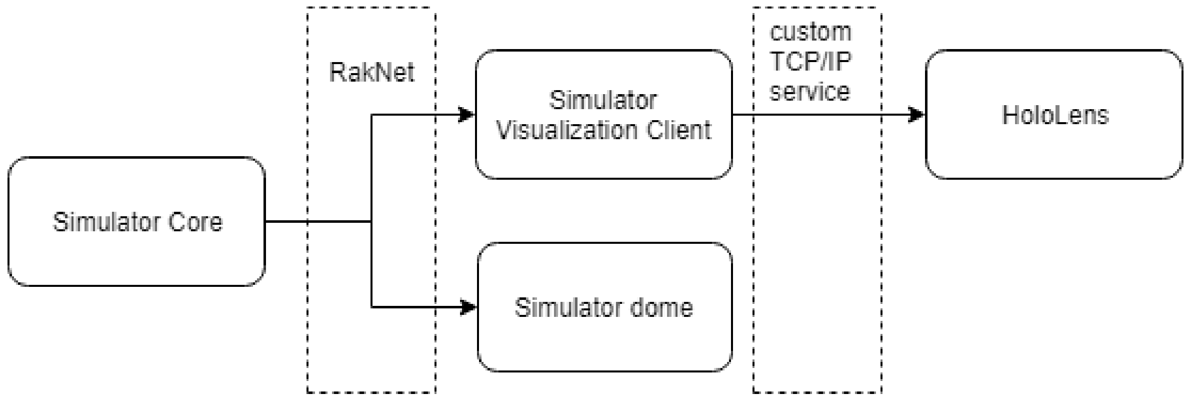


Figure 4.5: Network overview of the solution

4.3.1 Message protocol

The main type of data that we want to receive from the simulation are the positions and rotations from objects in the scene. Each object in the scene has a 3D position, 4D rotation (quaternion) and 3D scale. Because of how Unity sets up objects in a scene, all of these values had to be considered before data could be sent. More on this in section 4.4.2 but the resulting data that had to send mostly consisted of 3 float values for x,y,z coordinates and 4 additional floats for rotation. These 7 floats are the basic data for a single object. To save time on writing the message protocol, the data was sent as a string. The string was build by concating all the float values together and adding "/" as a separator between values. This was initially only intended for the proof of concept but ended up being the final solution because of time constraints. By using equation 2.4 and data from 2.1.1 we can calculate the maximum bytes the data from a single object can take:

$$7*(7+1+1)*2 = 126 bytes \tag{4.1}$$

Adding separators:

$$126 + (6*2) = 142 bytes \tag{4.2}$$

This means that in the worst case scenario to update the position of a single object we send a message of 142bytes in size. If we needed to know the position of several objects at the same time the data from those were stacked to form one large string. This also adds an additional separator after all the data from a single object making each object 144bytes. The default TCP-

Client settings provide an upper message size limit of 8Kbytes (message buffer size), meaning at most we can send positional data of 56 objects at the same time. As we only send a long string of floats, it was expected that the client knew which float belonged to which object and be able to separate the string.

4.3.2 Threading and network buffers

Client connections had to be handled asynchronously by the server so that any client could connect at any time. The main thread in Unity is used to calculate all the physics and can't be halted, so an additional asynchronous thread was required. A common thread-safe message buffer was set into place to be able to share data between these two threads. Unity's main thread would acquire the necessary data from the simulator core and continuously pass it on to the buffer. The buffer would then be periodically checked by the server and send the data to a client. Considering that we always want to send the newest positional data to not lag behind, the data buffer was a LIFO (Last In First Out) or stack type buffer. Upon sending the newest data the server would clear the buffer to prevent old data from being sent. The buffer was also cleared every 30 iterations to prevent further overflow if no clients are connected. To ensure thread safety ConcurrentStack objects that have built-in thread safety were used. As for receiving messages from clients a FIFO (First In First Out) or queue buffer was used to ensure that every message is read. These are made thread safe by using a ConcurrentQueue object. The idea was that messages from the client would be seen as commands for the server which always should be executed in order. This was instead used for sending debug data back to the server from the HoloLens. Implementation of the Buffers and their thread-safe functions are as shown:

```
//Received data can be FIFO so we get everything in order
private ConcurrentQueue<string> receiveMessages;
public string GetAnswer()
{
 string item;
 if (receiveMessages.TryDequeue(out item))
 {
  return item;
 }
 return null;
}

//data to send must be LIFO so we always send newest positional data
private ConcurrentStack<string> sendMessages;
```

```
private string ReadMessageToSend()
{
 string item;
 if (sendMessages.TryPop(out item))
 {
  sendMessages.Clear();
  return item;
 }
 return null;
}
```

Listing 4.1: FIFO and LIFO buffer implementation seen in the source code server-side 8.3, line 13-36

On the HoloLens the client needed to read the data send from the server asynchronously so it also needed a separate thread. In the same way as the server, a LIFO buffer was used to ensure we always use the newest data for setting the positions. A downside of always using the newest data is that the positions don't move linearly when displayed. This could cause the AR objects to have somewhat jittery motions. This can be mellowed out by linearly interpolating the movements but that could induce some delay. It is important to note that this is not the same as the delays described with Holographic Remoting. With Holographic Remoting it was the AR objects who stuck to the headset's movements due to delays in processing. With the TCP/IP solution, a delay results in the AR objects staying in their respective 3D positions and teleporting to a new location after a delay ends.

The entire code for the server and client part can be found in Appendix 8.3 and Appendix 8.4.

4.4 Spatial awareness of the HoloLens

To put a hologram, or AR object, in a specific point in 3D space we need to know several things. The first is that we need to know where we are, or more specifically, the HoloLens needs to know what we are using as a reference point. All AR solutions are meant to be seen from within a crane-cabin so there had to be a way for the HoloLens to know what represents a cabin. The second thing we need to know is where those AR objects should be placed relative to the reference points. How the positional data is gathered and processed is described further in section 4.4.2. There are several options available for the HoloLens to know where it is located, and the final solution ended up being based on image recognition. The HoloLens has built-in spatial awareness hardware that automatically generates a 3D map of its surroundings (see 3.2.1). It is possible to use this to recognize previously scanned areas so that its available as a reference point. This, on the other hand, works poorly in low light scenarios and also means that the solution has to be made specifically for a specific room. The final idea is that the AR solutions can be used with any kind of crane cabin so this isn't viable. Instead, we can use a reference point that is either already found in typical crane cabins, or can easily be added. The spatial mapping will still be used in this case to stabilize the image, but only the reference point will indicate that we are in a cabin. For this project, it was chosen to use a visual reference point and use the Vuforia library (see 3.4) to detect it. Both Unity and the HoloLens have native support for Vuforia so this could easily be implemented.

4.4.1 Using Vuforia

Because all AR solutions are based around the idea that they are going to be used in a crane cabin, the idea was to make a virtual crane cabin that would overlay the real world cabin as an AR object. Using this method we can find out how the scaling of the virtual corresponds to real-life objects. To make Vuforia work, the reference image had to be the same size in both the virtual and physical world. Setting this up helped with getting an overview on how to build the virtual scene

For the reference image a semi-organic and bright image was needed, so a colorful cover from a CD jewel case was used (see figure). A CD cover always has a specific known size (12x12cm) and finding a good digitized version of these images is usually simple. This image was placed in a visible place above the cabin's seat in both the virtual and real world. The user would look at this image and Vuforia would calibrate the AR objects to be positioned accordingly. The reason Vuforia ended up being used only for calibration is that it was found that if the reference image was constantly in view, Vuforia would try to readjust the AR-objects continuously. Due to varying light and head movements of the user, this could result in the AR objects shifting making the image unstable. It was better to use the spatial awareness of the HoloLens as sole image stabilizer after the objects have been calibrated. Calibrating the AR objects means that the reference image in the virtual world get attached to the HoloLens's spatial map where Vuforia thinks the reference image is located. This means that any 3D object that is a child of the reference im-

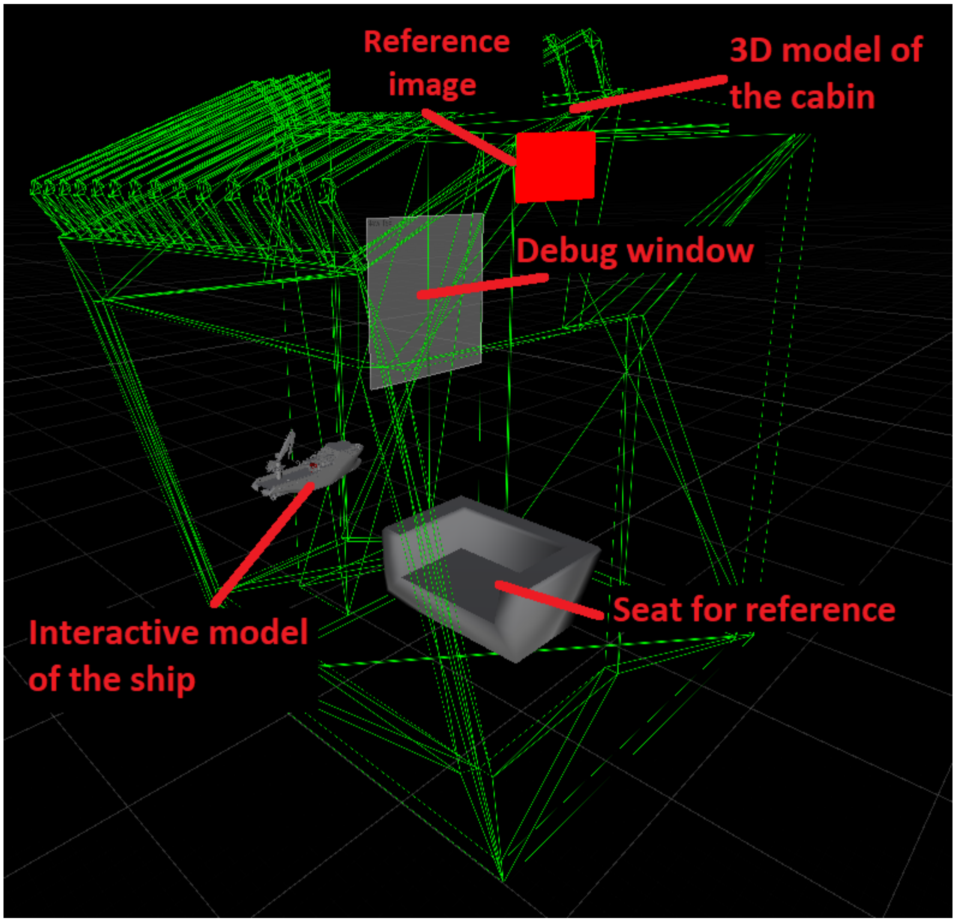


Figure 4.6: The virtual scene that on the HoloLens, the virtual crane cabin is shown as a green wire mesh.

age in the virtual scene gets rotated and positioned along with it. It is important to note here that this means that all virtual objects are moved in virtual space away from the origin. Vuforia essentially creates a new pseudo-origin for our virtual scene.

So the steps to finding out where the HoloLens is located are as follows:

1. The HoloLens creates a new crude spatial map of the current room we are in and starts a virtual scene with AR objects hidden.

2. Vuforia starts to look for a reference image.

3. When a reference image has been found, Vuforia will attach the virtual reference image to the corresponding placement on the spatial map. All child objects of the image will follow.

4. When all objects have been moved (calibrated) they will be displayed.

Figure 4.7: CD cover from the CD "My Hero Academia Original Soundtrack" by Yuki Hayashi released July 20, 2016

5. We either look away from the reference image or the Vuforia library gets shut down, leaving the HoloLens to try to stabilize the AR image by aligning and updating the spatial map.

4.4.2 Calculating positions

In the visualization software, which the server is based on, are all objects loaded as Unity GameObjects. This means that we can search for the GameObject that represents the object we need and copy its position. Although the idea is straightforward, implementing it is another matter. The virtual scene loaded by the server has a different origin then what the HoloLens uses for its virtual scene. All positions are related to the global origin in their specific scene, so copying positions from one virtual scene to another will result in different locations. To counteract this we can use a common reference point in both the server's and the HoloLens's virtual worlds. Because we already use a reference image for detecting a room with Vuforia, we can use the same image as a reference point in the server to HoloLens communication. As we know where the image located in a real-world cabin we can add the same image to the crane cabin in the server's virtual world. By drawing a vector from this reference point towards the object we want to know the location of we get the relative position of the object. Using the vector from two points equation 2.5 we can find a vector that points from the reference point to the object. This vector gives us the distance between these points, but the orientation is initially wrong. The reference point could have some rotation in both the HoloLens and server virtual worlds. This rotation isn't considered when calculating the vector. If we applied the vector directly in the HoloLens's virtual world the vector could point to literally anywhere if the HoloLens's reference point doesn't have the same rotation as in the server's virtual world. To fix this we can rotate the vector by the inverse rotation of the reference point, essentially rotating the reference point back until the reference point has no more rotation. An example can be seen in the Hologram on Load solution in code-section 4.2. By adding our object as child GameObject to the reference point in the HoloLens's world we can use this vector as a position for our object that always will be relative to the default position and rotation of this reference point.

```
//Hologram on load

 //Find a vector from our reference point to load
Vector3 TrackerToLoad = f.Crane.transform.position -
   f.CabinTracker.transform.position;
 //The vector holds rotation information, we want a vector from
    a reference point that isn't rotated
TrackerToLoad =
   Quaternion.Inverse(f.CabinTracker.transform.rotation) *
   TrackerToLoad;
 //Rotate the rotation of the load so we get its localRotation
Quaternion rot =
   Quaternion.Inverse(f.CabinTracker.transform.rotation) *
   f.Crane.transform.rotation;


string message = TrackerToLoad.x + "/" + TrackerToLoad.y + "/" +
   TrackerToLoad.z;
message += "/" + rot.x + "/" + rot.y + "/" + rot.z + "/" + rot.w;
```

Listing 4.2: Calculating the positional vector by resetting its orientation, from server-side code 8.3, line 144-155

Similarly to calculating the position, the rotation of the object also has to be rotated back. Multiplying the rotation by the inverse quaternion rotation will reset the rotation of the object to the original local rotation it had if the reference point wasn't rotated. After the position and rotation were processed and send, the HoloLens could just apply these values directly as local position and rotation of the object. The default unit size in the server's virtual world is based on real-life meters. So one unit in the server is 1m in real life. This means that there is no scaling used in the server's virtual world. On the HoloLens however, are several objects scaled to fit their applications. This means that the positional units received from the server have to be scaled accordingly. An example is the miniature boat solution where the boat is only 0.005 the original size (see 4.3).

```
    //For the miniature boat model

    //In this scenario, the model of the load itself has been scaled
       down, so the movement it has to take also needs to be scaled
       down.
    miniTarget.transform.localPosition = new Vector3(floats[7],
       floats[8], floats[9]) * miniTarget.transform.localScale.x;
```

Listing 4.3: Scaling the received positions for the Miniature boat solution, from client-side code 8.4, line 171-174

4.5 Implementation of the solutions

After the implementation of the server/client solution, we could start working on implementing the actual AR solutions. The solutions were researched at the same time as development for the HoloLens began. The first solution, Hologram on Load, was already well defined after the data transfer solution was implemented. This meant that this solution also worked as a basis for the creation of the network message protocol (see 4.3.1). Because of this did the cycle of implementing the Hologram on load solution start roughly at the same time as the custom TCP/IP cycle. These cycles would then run parallel and improve each other over time. The specific scenario that was to be used for the Hologram on load solution was to lower a CPDU-module down to the bottom of the sea from the stern of the ship. A CPDU module is a large unwieldy structure that can sway significantly and lowering it to the sea bottom will result in losing sight of the object. This made a perfect scenario for testing out the solution. This scenario was already built as a scene for the simulator and was ready to use. For this solution do we only need to know where to place the hologram that represents the load. This meant that with the solution from 4.4.2 we can easily receive the positions and rotation of the load. It was vital here to include rotation because the center point of the 3D model can be far of the geometry center point. This means that pivoting on the center can result in unexpected movements that wouldn't be shown if we didn't have rotation. See figure 4.8 for an example of what a difference the rotation makes for the object's position.

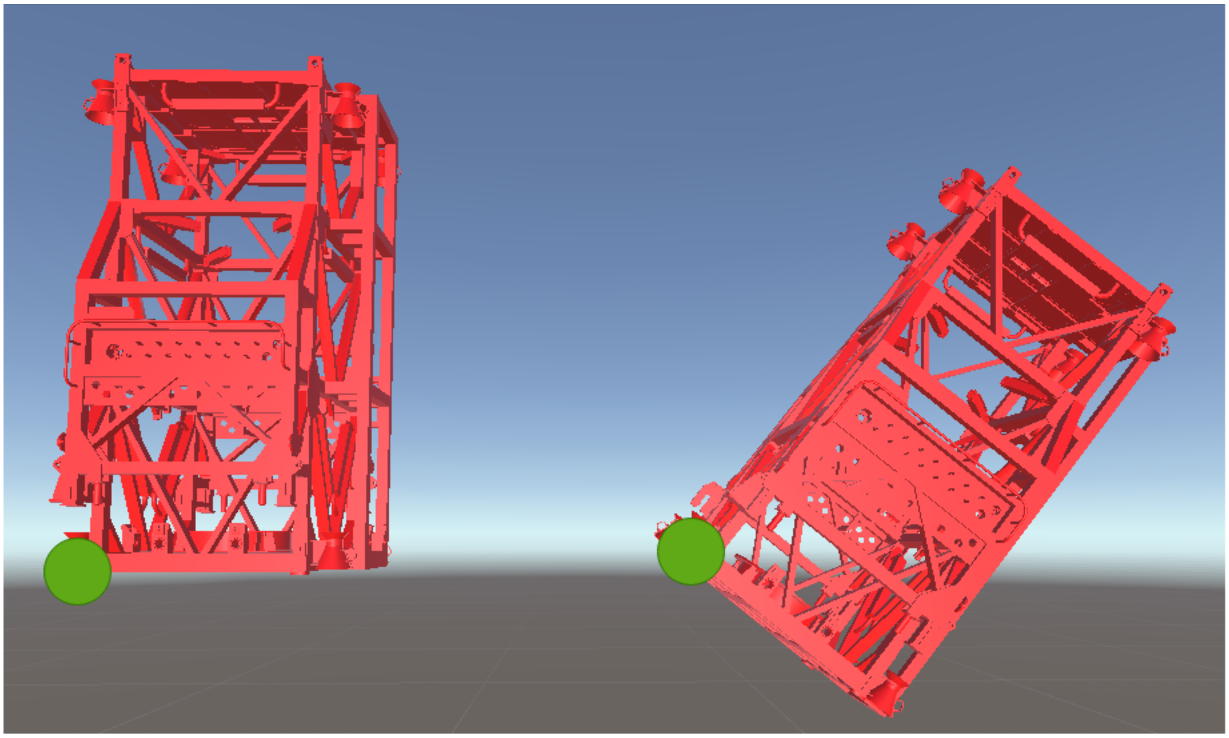


Figure 4.8: Non-Rotated (Left) and rotated (Right) CPDU module if the pivot point (in green) was located in the lower left corner

Miniature boat solution

The second solution to be implemented was the miniature boat model. This solution is essentially the same as the Hologram on Load solution programming wise only that the end result is a scaled down version. Using the slimmed-down model for the boat in the simulation we could create a main overview in the HoloLens (see figure 4.9). From showcasing this to several people a suggestion was made that it would give a better overview if you also saw the crane moving according to the movements of the actual crane. Another thing that was noted was that because the load on the crane was now so small, rotations were insignificant. To fix these two problems we instead of sending rotational data of the load send out the current local rotations of the crane booms and base. Because the rotations of the booms and base are local rotations that are independent of the global origin we can just send the rotations directly (see the code in 4.4). Even though we could have a crane cabin on the ships model, the size of the reference image would so small that it wouldn't make sense to use it as a reference point. Instead, we needed a new bigger reference point. As we use the entire ship in both the server's and the HoloLens's virtual world we can use the hull as our main reference point.

```
    //Miniature boat

     //Same as with hologram on load, only we use the ship's hull as
        reference point
    Vector3 HullToLoad = (f.Crane.transform.position -
       f.Hull.transform.position);
     //Reset the rotation that the Hull might've had.
    HullToLoad = Quaternion.Inverse(f.Hull.transform.rotation) *
       HullToLoad;


    message += "/" + HullToLoad.x + "/" + HullToLoad.y + "/" +
       HullToLoad.z;
     //For the miniature boat it isn't nescassary to include the
        rotation for the load, so we send rotation data on how the
        crane behaves instead
    message += "/" +
       f.CraneBase.transform.localRotation.eulerAngles.y + "/" +
       f.Boom1.transform.localRotation.eulerAngles.x + "/" +
       f.Boom2.transform.localRotation.eulerAngles.x;
```

Listing 4.4: The message definition for the Miniature boat solution, from server-side code 8.3, line 161-171

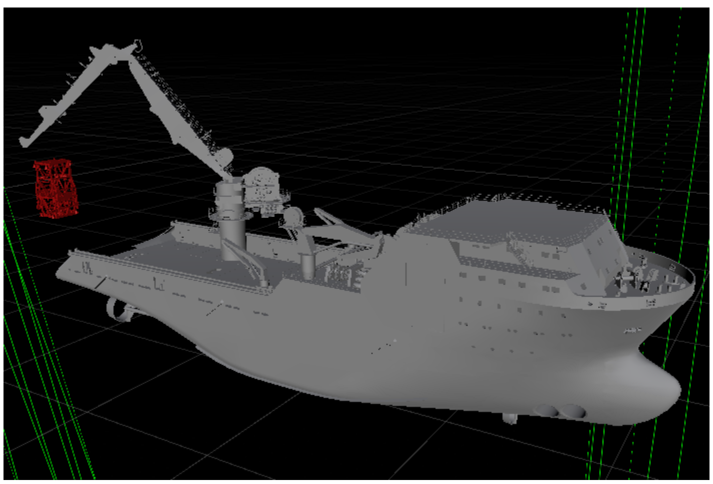


Figure 4.9: Caption of the miniature boat model on the HoloLens. Look at image 4.6 to get a sense of scale

.

AR widget solution

The Ar widget solution wasn't implemented. The main reason for this was because the server was based only on the visualization client of the simulator. To access any data other than positional or rotational data a separate specialized client had to be used. This meant that the server didn't have access to for instance weather data, sea levels, system statuses and etc. Considering how the message protocol built, it could easily be re-purposed to send any situational data we needed. A mock-up could also have been build to show try out its usage in the simulator dome, but the idea of this solution was made at the very end of the programming period. At this point, the focus was already to use a video to showcase the usage of the solutions so this idea could be faked with CGI instead.

4.6 Video

An explanatory video was made to showcase the application made for this thesis. The main purpose of this video was to show crane-operators what AR is and what is possible with it. This also made it possible for crane operators to be interviewed remotely without them needing to specifically come to the simulator to test out the application. For this purpose, a digital survey was made that included the video (more on this in section 4.7). The video consists of two parts, one section in CG made with Blender and a section that showcases the implementation in real life. The first section functions as an introduction to AR and explains what features were made with the HoloLens. The next section gives an introduction to the three solutions and their scenarios. The last section of the video was filmed on-site at OSC in their crane-simulator. Both the load-hologram and the miniature boat were implemented with the HoloLens, so they could be shown in this section of the video. The entire video was also narrated to explain what is happening in detail. As it was uncertain if the crane-operators were well versed in English, the video was dubbed in both Norwegian and English. Both the English and the Norwegian video were uploaded on YouTube so it is publicly accessible and could be easily integrated into the digital survey. It was considered to make the video hidden so that only the people taking the survey could access it, but with an agreement with OSC, the video was made public [29].

4.6.1 Creating the video

The main tool for creating the video was Blender. Most of the 3D models and their materials used in the video were directly made with Blender. Some of the more advanced models such as the model of the rigged virtual user or the boat-containers were acquired from turbosquid.com

[1]. The animating and rendering process was also done with Blender with the Blender render engine. No initial dubbing script was made beforehand because inspiration was drawn from how the video turned out. The dubbing also depended on what was actually possible to create in the short amount of time available to create the video. This meant that it was impossible to know beforehand how long the speech part was going to be. A solution to this problem was to render the video in several small parts, so the video could easily be extended with repeating scenes if longer dubbing was required. This also saved us from re-rendering scenes that were already completed, saving precious rendering time.

The first part of the video would give an introduction to what Augmented Reality is. Considering that AR has been increasingly popularized through for instance apps for phones(see 2.3), an idea was to remind the viewer that this is, in fact, AR technology and is what we want to use. To showcase this the first scene in the video shows a virtual user that uses various AR technology. With the chance of the viewer not having seen any of these technologies before, we start the scene by explaining what AR is through the use of a tablet. Other technologies shown are HUDs and HMDs. The rest of the video focuses on HMDs and how it can be applied in the scenarios studied in this thesis.

A pre-requirement that is expected from the viewer is that he/she knows what a hologram is. The video explains AR objects as hologram objects by giving the AR objects a typical hologram appearance as seen in science-fiction (see figure 4.10). The idea was to replicate a holographic material as seen in e.g. Star Wars by adding fake scan-lines but this idea was omitted in favor of render time. The only solution that did not implement the hologram material is the miniature boat solution. This was chosen because the hologram material is a transparent material and it would make it harder to see what is what. As the boat consists of several parts which we need to be able to distinguish we can't have the color be coherent and transparent.

As Augmented Reality main purpose is to make 3D visuals, for this we move the camera to various locations throughout the video to really give the user a 3D perspective. For instance, in the introduction with the tablet, we move towards the perspective of the virtual user holding the tablet to give a good perspective on how it is used. After the video has explained what AR is, the video continues with a virtual representation of the proposed solutions (see 4.1). The scenarios in which the solutions are applicable are recreated virtually to explain the concept (see figure 4.11). The virtual overview gives an easier way to explain how the solutions work and what they do than what was filmed in real life. The real-life footage from the crane simulator was filmed at some awkward angles because of the lack of space inside the cabin. With the Virtual representation, we could instead give a full overview by recording from outside the crane-cabin.

The real-life footage recorded was initially intended to showcase the usage in a presentation at OSC, but this event was canceled. This footage showed later to be useful for this explanatory video instead. The HoloLens has a built-in recording feature that records both what the user sees and what AR objects are present. So for filming the real-life scenario in the simulator, we could record with both the HoloLens and an external camera to get an overview. One downside we noticed was that recording with the HoloLens proofed to be computationally heavy and a lot of footage was lost because the HoloLens created corrupt files. This meant that some of the HoloLens footage had to be faked to give the user a good indication of what is happening in the video. Another problem was the lighting in the simulator. The simulator chamber is very dark to provide optimal projections on the walls without the projector lights being too intense for the eyes. This, on the other hand, made it difficult to record a video that wasn't too dark.

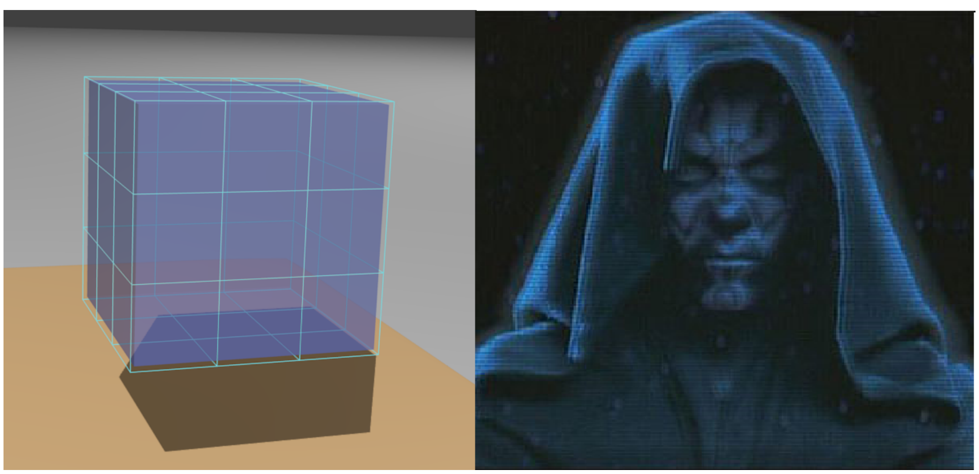


Figure 4.10: Left: Hologram representation made with Blender. Right: A hologram of Darth Maul in the movie Star Wars The Phantom Menace (1999)

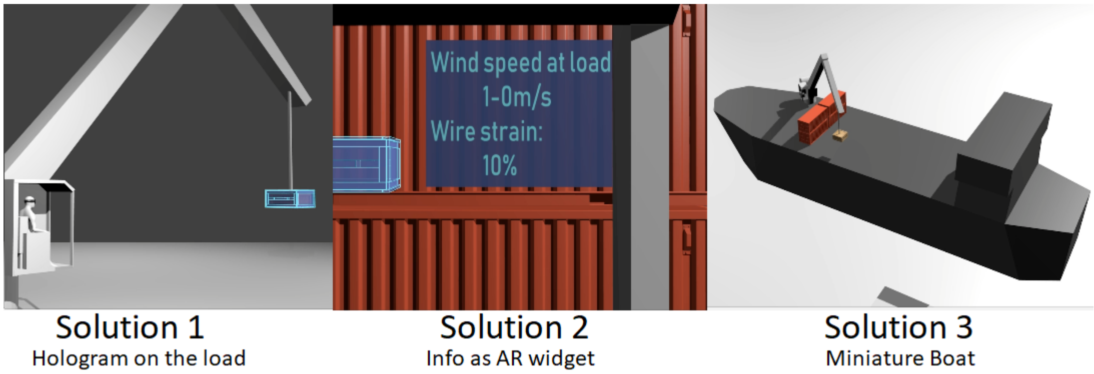


Figure 4.11: All three solutions showcased in the video

4.7 Survey

The survey has a focus on what information is vital for crane operations, what potential information a crane operator lacks and lastly if Augmented Reality can help with crane jobs. Considering that crane-operators might not be well versed in English, the survey was created in English and translated in Norwegian. The test subjects got a choice of both the English and Norwegian survey to answer. From the results, it quickly showed that Norwegian was preferred and no one ended up answering in English. The survey consisted of 35 questions and would take around 30min to complete. It was created with the web-based Google forms service which proved to be an easy to use tool to create surveys. Using Google forms also meant that the survey was accessible for the crane operators at all times. Originally the idea was to have two separate surveys. The test subject would first answer some general questions, then test out the solutions in the simulator, and lastly answer some questions about the experience. But because of difficulties getting crane operators on site, an introductory video was made which replaced the testing phase4.6. This meant that the survey could be compacted as a single survey where the video would play after some questions were answered. A major focus was to keep the survey anonymous and store the data according to GDPR rules. For this reason, a small text about the usage of private data was added to the beginning of the survey. For an easier overview, the survey was divided into 4 topics.

- The first part of the survey included general anonymous information about the participant along with what the participant thinks is useful information with crane lifting jobs.

Vital points here are to find out how much experience the participant has with cranes and what they deem important information. This part also briefly asks about the participant's opinion of current crane cabin layouts.

- The next part looks into how much the user knows about AR and VR systems. Considering that most crane-operators probably aren't familiar with AR systems, we can measure their actual experience with these questions. The questions here also included VR systems as its more likely to have experience with VR through e.g gaming applications. VR is also similar enough to understand AR concepts.

- The third part doesn't include any questions but showcases an explanatory video (see 4.6). This gives the user an introduction to AR and the solutions described in section 4.1.

- The last part of the survey asks the participant what they think of the solutions presented in the video. This section starts by asking if the video was understandable and then goes in depth about the solutions. Now that the participant also has some idea of what AR is, we also ask what their opinion is on the technology and if they potentially can think of other solutions.

Most of the questions are based on written feedback as small paragraphs. This will give us qualitative information about the participant's opinion. The reason this was chosen was that prior knowledge and user-opinions of cranes have to be obtained. These type of questions don't have strict answers and will need to be generalized when processed. A focus was to make the questions fit the idea of "before and after" seeing the video, but without the participant having any real prior knowledge of AR there are very few ways to ask for opinions.

4.8 Analysing survey and verbal data

At the end of the testing period, a total of 5 male participants fully answered the survey. Many of the potential crane-operators were shown the survey but didn't have time to answer, but instead gave some oral feedback. All of the participants of the survey are considered to be experts on the field. This can also be proven by looking at the first few questions that ask for their experience in the field (see figure 4.12 for data). From these values, we get an average yearly experience of 16.4 years although the majority is below this value. It can be noted here that there is a large gap between the person with the longest experience at 30 years vs the one with the least experience at 5 years. At an average do we have an experience gap of around 5 years between each participant if their years' experience was sorted in an increasing fashion. This thesis will assume that 5 years of experience is enough to be considered an expert in the field. Looking at what type of experience the participants have in question 5 we can see that all participants have experience with most offshore crane types. From this, we can say that all crane operators have roughly the same type of experience, only that their expertise level is graded by their experience in years.

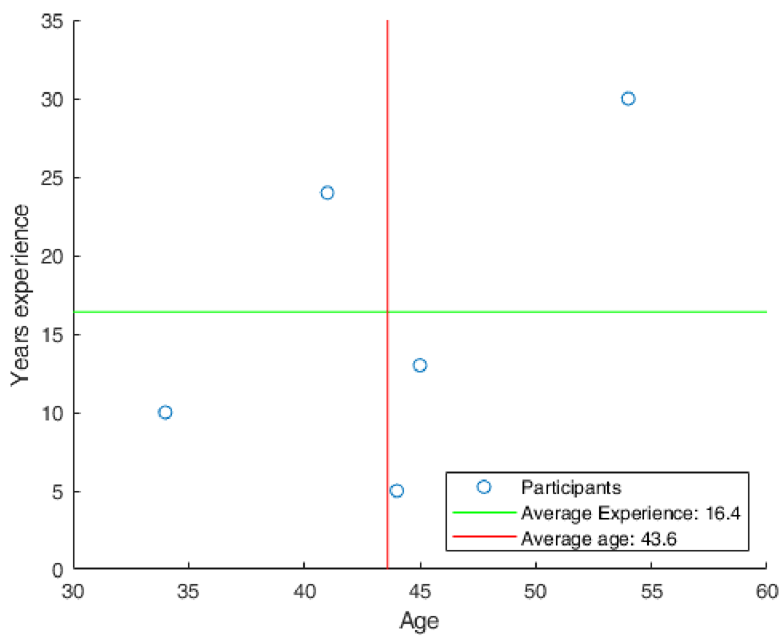


Figure 4.12: Age vs years experience of the participants of the survey

Looking at the first real qualitative data we have a set of questions asking what the participant thinks is vital information on the job. From the initial question which asks what the most challenging task with crane operations we get several answers pointing to working in either bad weather or working blind. A specific scenario that is pointed out is the task of moving supplies towards a secondary boat. These scenarios give us two problems that we can address. The first is visibility and the second is the movement of ship and load. This validates that blind spots of the crane are seen as a real problem and that the idea of the Hologram on Load is on the right track. We can further validate its usefulness by looking at the answers from question 24 and 26 in part 4 of the survey (see figure 3.7 or 8.1 and 8.2 for the various survey parts). Answers from question 24 show unanimous support for the Hologram on Load solution especially for subsea operations. It can be said that question 24 only asks for what the participant liked the most, their opinion on which AR solution is most useful is answered in question 26. Question 26 shows that 3/5 participants think the Hologram on Load is actually useful (see figure 4.13). As this result differs from question 24, this could indicate that there might be some bias created by the wow-factor of the Hologram on Load solution and it might potentially not be needed technology. But if we look back at answers from question 7-14 we can see that instead, the two who voted in favor of the Information Widget solution are in general more concerned with how information is displayed. One person, in particular, did not experience many blind lifts in their job and was really passionate to have all information for an operation in a single location. In comparison, the three who voted for the Hologram on Load solution work with cranes that have specific blind-zones. Considering each of the participants' work scenarios it is more or less self-explanatory which solution they will prefer. This gives us at least insight that even though all participants are working with offshore cranes in general, their work scenario might call for other AR-solutions. Going back to the first two problems addressed, we also have movement of ship and load as a problem. This can be caused by bad weather or in general by working at sea. The proposed solutions weren't specifically designed to fix this problem but from survey feedback, we can see interest in using some of the solutions to counteract this. A suggestion that can be extracted from the answers is to show the relative movement of the crane or load VS the destination. This can be either displayed directly by expanding on the Hologram of the Load solution or what many participants suggested is to show values of this through the Information Widget solution. Suggestions from people who didn't partake in the survey also included showing the entire scenario and its movements as part of the Miniature Boat solution. An example would be to show both the current vessel and the destination vessel at the same time in rig-to-ship crane operations.

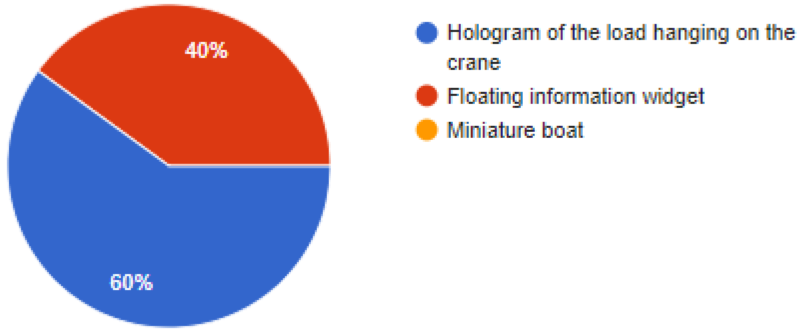


Figure 4.13: Question 26: Asking which AR solution is the most useful.

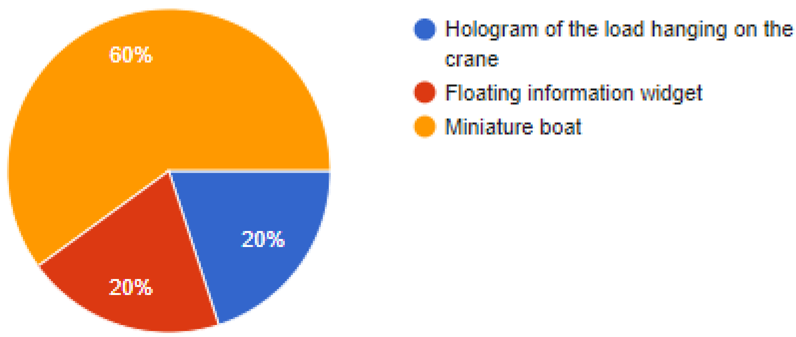


Figure 4.14: Question 27: Asking which AR solution is the least useful.

Analyzing the questions 7-13 further give us more insight into what could be potential information that could be shown with the Information Widget solution. Some of the highlighted suggestions which most participants mention are:

- Weight of the load, regardless of if its an internal or external lift
- Boundary limits such as height or unsafe areas
- What equipment is in use, such as Aux hook or Main hook

A full list of suggestions found through both the survey and research can be found in section 4.1.2. Looking at question 13 specifically we find that there are a lot of blind zones for a crane. All participants of the survey answered with that the cranes they've worked with all had blind zones, but in different scenarios. Highlighted blind zone areas are as follows:

1. Right below the crane's cabin.
2. Behind the crane itself.

3. The opposite side of the pedestal, considering the crane-cabin being mounted on the side.

4. Behind containers or other objects on top of the ship.

Considering how 1 and 3 of these blind-zones don't directly apply to the load itself but rather to how the crane itself is positioned, these can't be solved with the Hologram on Load solution. Questions 7-12 aren't directly associated with the Information Widget solution, but where meant to carefully introduce the participant to thinking of potential information that can be displayed with it. The way these questions were worded ended up being somewhat confusing for some people. The intention was to let people voice their own opinion on what information was lacking on the job and maybe be creative and suggest new things. Instead did many perceive it as your typical exam to get certified and answered with the default answers found in crane operation regulations. One crane operator even refused to partake in the survey because quote: "most questions can be answered by reading the rule-book". This answer is of course not wrong and crane operation regulations were considered in making the final AR solutions, but this doesn't provide a measure of crane-operator opinions. The goal, after all, is to find out if AR can be implemented alongside the current regulations. It can be seen in the answers that default routines are certainly present, but luckily did most participants see the intention and add personal twists. It is also important to note that questions 8 and 10 ask about missing information specifically, and the lack of non-default answers here can indicate that there is little more information needed to complete their job. Instead, we can analyze questions 15 and 28-29 to give us suggestions on which current information can be shown with AR. The immediate problem that is highlighted here is that current display technology used in cabins don't have a default mounting position. Information of the job, CCTV cameras, Alarm systems and etc are all displayed by their own screen and mounted around the view of the user. Many participants suggest collecting all these systems as one, regardless of if it's displayed with AR or not. Even though, many suggest that AR can be a great tool to function as such a collective system, especially if it can be used as an additional system as opposed to a replacing system. Also through answers from question 7-12, it is said that having information on small screens where it is required to scroll through is both a tedious and long job that halters efficiency. Expanding on this idea we can modify the Information Widget to potentially work as a common lookup system where the crane operators can look through and add information they think they require for their job. This is as opposed to a strict system where we as designers of the software decide what the user sees on the Information Widget. Some suggestions from oral feedback also included that it might be possible to use speech recognition to easily access the data one wants.

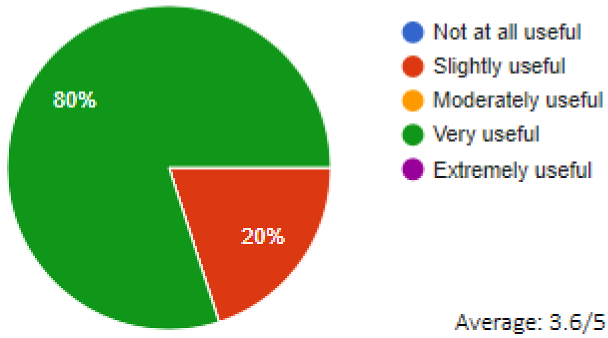


Figure 4.15: Question 34: The rating of AR given by the participants

Section 2 of the survey was intended to find out the participants prior experience with AR and to find their opinion on the matter before and after seeing the AR solutions. A person that has never used AR before could potentially be more impressed than a person who has tried it often. The answers show that basically none of the participants have used AR or VR technologies before, making it impossible to have a "before and after" evaluation of their opinion on AR. One participant had tried AR before but never considered it in a serious gaming scenario. On the other hand, we now know that a wow-factor can be present as a bias. From analyzing the way the participants have answered the survey it was concluded that the answers are unbiased by the wow-factor. This conclusion was drawn by the way the participants elaborated their answer in question 34, even with the above average rating seen in question 34 (see figure 4.15). Question 34 asks the participants if they have any comments about AR technology. Answers to this question directly express that the participants find AR an interesting technology and that it can be used for many things, but that it is uncertain that today's technology will provide the solutions they need. Some concern was shown to using AR headsets as main AR solution for the suggested problems as the HoloLens is large and looks uncomfortable to wear. This is, for the most part, true and a genuine problem with the HoloLens. This concern can also be seen in explicitly in question 25 and elaborated in question 31. With question 25 we ask specifically what they didn't like and with question 31 we ask if the participant thinks that AR will be distracting. With

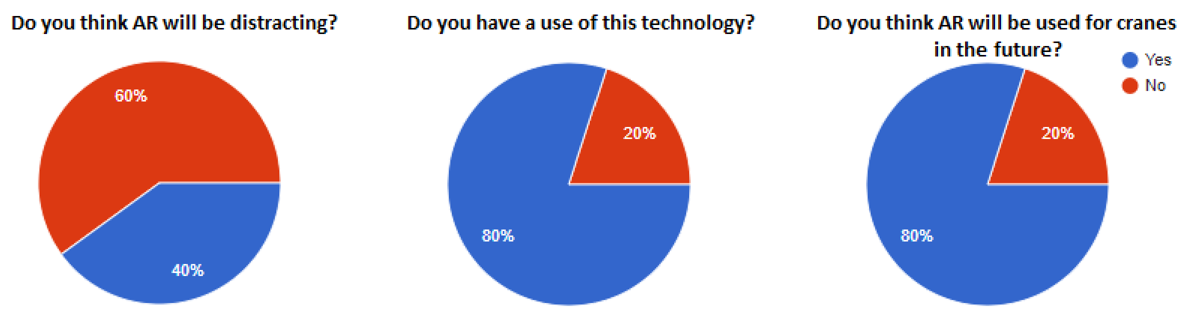


Figure 4.16: Results from questions 31-33, going left to right.

question 31 two out of 5 answered with yes (see figure 4.16). Then again, from results of question 33 that asks if this technology will be useful in the future, we can see that the majority is agreeing on that its a real possibility. From question 32 we can also see that this is a technology that the participants think is a tool they can use if the technology is expanded on.

If we were to grade the actual solutions proposed in the explanatory video, we first have to make sure that the participants understood it. This is answered in question 20, where the result shows that all participants understood the video and the application of the solutions. The second important part in grading the solutions is to check if the problems addressed are actual relevant problems for crane operations. From question 21 we can see that 4/5 answered yes to this question. As a general scoring system for the solutions, we have question 22 that asks if the solutions shown would be able to improve situational awareness. Similarly to question 21 this gets a score of 4/5. It can be noted that it was the same person that answered no to both question 21 and 22. In hindsight, this makes sense as the questions are closely related. This person was also the one with the most experience in the field which could suggest that these problems don't directly apply because the solutions to these problems have become intuitive. He, on the other hand, suggests that AR might be better suited for showing CCTV footage, relative movement between boom and ship and lastly to indicate where people walk on deck. People from outside the survey also suggested that AR could work as a good tool for learning purposes. AR tools could help new crane operators gain that sense of intuition quicker which this person might agree with more. Comparing the AR solutions together we get Hologram on Load as a clear winner along with Miniature Boat in the last place. One could argue that it's hard to grasp how the Miniature Boat solution actually works in real life scenarios. Because in contrast, most people who tried the AR solution briefly in real life liked the idea of the miniature boat almost more than the Hologram on Load. It was noted on the other hand that users tended to look longer at the miniature boat model then what was in front of them, which means that it is rather more distracting than helpful. In total wasn't there much interest shown towards the Miniature Boat model solution in the survey which resulted in little data. It was briefly mentioned in correspondence with subsea operations but the result is nonetheless that this isn't seen as a very useful tool by the participants.

Chapter 5

Discussion

Through researching, if Augmented Reality is applicable for offshore crane scenarios, three common problems in crane operation were addressed. Looking at the first problem to be addressed we have that in some crane operations the crane operators are blindsided by various constructions on the ship. The initial proof of this is that typical crane cabins have a limited field of views as described in section 2.5. This is, of course, a limiting factor but if we consider the fact that the crane is able to rotate on a pedestal this problem becomes less prominent. The load hanging on the crane will essentially always hang in front of the crane, where the 60 degrees of the horizontal field of view might be enough. Where the problem really occurs is when the load disappears behind another structure or because of weather conditions. Research shows that this happens in various common scenarios. The survey points to the scenarios where we want to move the load from ship-to-ship or rig-to-ship, subsea operations, and in the when there is a lot of loads to be moved on deck such as with a container-ship. This is when the crane operator is really blindsided and need help from other people on deck or CCTV camera. The second problem that was addressed was focused around the idea that vital scenario information is usually found away from the general area one has to look for operating the crane. Information such as alarms, the weight of the load, Height limits, unsafe areas, lifting equipment and etc are either handed as a written note or on screens situated around in the crane cabin. This means that the crane-operator has to deliberately move their focus towards these information screens in question to get updated on the current state of the operation. While being a minor interaction, this could delay operations. The third and final problem that was addressed was the problem of the crane operators losing the 3D perspective when working. Sitting in a crane-cabin essentially only gives you a forward view which can be considered a 2D perspective. This view can result in troubles with distance perception and general orientation. This problem especially occurs with a similar scenario as in the first problem addressed, where we lose sight of the load entirely. This problem is usually compensated for by using CCTV cameras that are mounted on the boom. But this results in the same problem, where the crane operator has a 2D perspective of what is happening. Seeing the load from the boom also results in a top-down view. This in general results in lacking a sense of height.

5.1 Usefulness of the solutions

In this study, there were initially three Augmented Reality methods theorized to work as solutions for the problems addressed. Two of the theorized solutions were implemented with the Microsoft HoloLens to proof that it is possible with today's technology. All solutions were further validated by conducting a survey with crane-operators. A total of 5 operators participated in the survey, giving us subjective feedback on the solutions. Going by the test results we have that 4/5 operators think that AR could be a potential tool to use on the job, but the circumstance of their job results in the need for various AR solutions. A concern expressed by 2/5 of the participants was that the technology at hand isn't directly useful because the equipment needed and AR itself can be distracting. This was directly related to the use of a large headset such as the HoloLens and can be seen as a technological limitation. Further survey results show that there is little new information that the crane-operators can think of they need for the job other than what is currently available, but AR can help as a tool to gather scenario critical information in a single location.

5.1.1 Hologram on load

As a proposed solution to the first problem, we want to use AR to overlay an image of the load at the exact same location as the original load. By AR being a local projection for the crane operator, this would effectively result in continuous visuals of the load. This would give the crane operator an impression on what is happening to the load while it's behind a structure. This is one of the two suggestions that were implemented with the HoloLens. Using the Vuforia image recognition library, we could calibrate the HoloLens to the current crane cabin we're in and start displaying the load. Positional data of the load was collected from the simulator by checking the position of the load relative to a reference point in the simulator's virtual world. The same reference point was used in the HoloLens's virtual world so the relative position found in the simulator could be directly applied in the HoloLens. The result was an application which could be run for a subsea scenario where the load was visible at all time through a red AR representation.

To validate the usefulness of this solution we first had to validate how important the problem was. The survey shows that all crane participants experienced blind zones when operating cranes. The blind zones appear on various occasions but those most relevant for this solution are subsea, moving of containers and ship-to-ship or rig-to-ship operations. There was unanimous support for the Hologram on Load solution in the use case of subsea, but this scenario was only relevant to 3/5 crane operators. On the other hand, there were some blind zone scenarios that this solution didn't fix. The backside of the crane and the opposite side of the pedestal are major blind zones, and because this AR solution only focuses on the load this isn't applicable. This is considering that any load hanging in the crane will always be in front of the cabin. Theoretically, it is possible to still see the load after the crane has rotated away, but that is only applicable in scenarios where a tracker is permanently mounted on the load.

On the other hand, it shows that this solution might not be directly useful for experienced crane operators. With experience comes intuition of how the load is positioned and by the help of radio communication there might not be a need to use AR. Nonetheless, in the scenarios where the load vanishes entirely from the crane operators visibility, this solution is used to give the crane operator more insight on the scenario besides taking strict commands from the radio. By analyzing the entirety of the survey, this solution is seen as the clear winner above all other solutions. The 2/5 people who have voted the Hologram on Load solution as second and third best do agree that this is a highly usable solution, but not for their specific work environment. There are on the other hand suggestions for improvements and a potential to use this for training purposes.

5.1.2 AR information widget

The next proposed solution was the idea of showing information to the crane operator directly as a floating window. This was the only proposed solution out of the three that wasn't implemented, but the concept was explained in the explanatory video. The idea sprung up from the potential to show dynamic data such as weather data or alarm systems and portray it directly into the view of the operator. This would mean that the crane operator no longer needs to look at the additional displays situated in the crane's cabin to get information. Being a new technology, this also opens up possibilities for giving more information than what operators have currently access to. That said, the survey results show that the crane operators can't think of new information needed for the job. Most required information to finish the job is already given today as either a note or on displays. Adding more information can also have the opposite effect where the total amount of information is too much. Instead, the survey shows that this solution can be used for listing the critical information needed to perform the job at hand. 2/5 crane operators see a direct use in this solution for gathering all information in one place and preferably nearby. This resolves an issue that the operator has to look through all notes/displays which results in less efficiency. A suggestion here was also that speech recognition could be used to bring up information on the fly. The problem of not having all information in one spot is, on the other hand, a direct problem of the work site. AR doesn't necessarily have to the solution to this problem, but AR can provide a common interaction hub where all the information can easily be stored and retrieved. Results from the survey show that the most vital information for a scenario that AR can show are as follows:

- Weight of the load, regardless of if its an internal or external lift
- Boundary limits such as height or unsafe areas
- What equipment is in use, such as Aux hook or Main hook

5.1.3 Miniature boat

The last solution that was explored was to implement a miniature version of the entire scenario. In particular, we wanted to show a miniature version of the boat we are currently on and indicate where the load is relative to the large structures on board. This is the second solution that was implemented, where we took both the model of the boat and the crane and miniaturized them. Using the same data transfer protocol as used with the Hologram on Load solution we could send the positional data of the load along with showing how the crane moved. The idea was that this could give us a birds-eye-view perspective of the scenario so that we can look at the load from multiple angles. results from the survey show that this was the least interesting solution of the three, where 3/5 operators voted it as the least useful. On the other hand did the people who got to try it in real life like the idea almost more than the Hologram on Load solution. But some drawbacks were also noted here. Most people who got to try it were more interested in looking at the boat model as opposed to the view outside the cabin. This could indicate that the solution is rather distracting than useful. This could, on the other hand, be a factor of them being interested in the technology in general, not indicating usefulness.

5.1.4 Further suggestions

The results from the survey gave many suggestions to changes along with some new potentials for AR. For the Hologram on Load it was mentioned that it would be a good idea to rather then showing the load, show the people walking on deck. This could provide for a safer environment, especially for the crane operators who already have enough experience to predict where the load will end up. It could also be possible to further indicate the relative movement between crane and ship if the ship is a lot in motion. For the Information Widget, it was suggested to utilize potential speech recognition to swiftly bring up the information you want. The HoloLens has this built-in so this is definitely a possibility. Considering that the widget can be seen as a floating display, it was also suggested to show the CCTV footage with this instead of an extra display in the crane's cabin. For the miniature boat solution, it was suggested that it would've been better to indicate how the destination of the load moves. For instance, in a rig-to-ship scenario, it would've been a good idea to show the secondary ship and how it swayed relatively to the ship you work from. This solution, on the other hand, requires more tracking equipment then what was is required from e.g the Hologram on Load solution and is, therefore, more expensive to implement in real life.

5.2 Gathering qualitative data vs quantitative data of AR solutions

Augmented Reality is fundamentally a visual tool for humans to interact with. This means that there are two ways we can measure the usefulness of AR. One is to measure if a human performs a set task more efficient with and without the aid of AR (quantitative) and the second is based on their opinion of the tool (qualitative). Depending on the crane-scenario and the AR solutions qualitative measuring can have several drawbacks. Considering how crane-operations fundamentally only entails the task of lifting an object from A to B, a well-defined crane scenario can still be approached in many different ways. Crane operators can perceive the same scenario differently and perform a better solution as other test subjects, making it hard to distinguish if AR had any factor in this. Some users might not even see any use of AR based on their approach to the problem. Restricting a scenario to only a small set of instructions will also restrict what we can measure about AR.

This brings us to the second type of measuring, gathering qualitative data. If we see the crane operators as experts we can have them evaluate the solutions itself independent of scenarios. This removes the before mentioned restrictions and gives the test subject an incentive to think of where the solution is applicable. Unless a solution fixes a one-off-case this can be used to get a more generalized opinion of AR solutions. There are on the other hand some drawbacks with this, such as we expect the test subjects to be creative and to understand the technology. The application for the AR solutions can also be too vague to the point where the test subject won't understand its use without a strict example.

5.3 HoloLens restrictions

Augmented Reality is a wide area of expertise and can be implemented in many different ways. Considering the HoloLens or other spatial aware headsets gives us a set of limitations that have to be addressed. It was noticed through testing that the rooms the headset was used in had to have some distinct features or else the headset would lose its tracking. Simple things like looking down on a flat floor or wall would make the HoloLens lose its tracking. Another factor is that because the HoloLens's depth vision is made of IR light, it can easily pass through windows or other transparent objects as well. Considering that the crane operator sits in front of a large window where the only distinct spatial features are the cabin's sidewalls and objects in the distance, this isn't sufficient tracking for the HoloLens. As the HoloLens was mainly used within a Dome Simulator it always had the inside walls of the dome to look at. This was sufficient for testing purposes but not viable in real life scenarios. These problems can be omitted by several solutions. The first solution is to add reference points for the HoloLens within or outside the crane's cabin. These can just be physical features that the IR camera can detect, geolocators or tracking images. Geolocators do require additional hardware for the HoloLens to be able to use, but tracking images can be found with the built-in camera. Another potential solution is to cover the main window pane with an IR reflective coating. This would enable the HoloLens to see the window as a wall instead. Considering that tracking can be lost by looking at just a flat wall, shapes can be made with the coating to create features for the HoloLens to detect.

5.4 Other AR technologies

One of the major downsides with current Augmented Reality Headsets is the small field of view. HoloLens 1 has only a screen that covers 30deg by 17.5deg of one's field of view, Magic leap one has a minor improvement with 40deg by 30deg and lastly, the HoloLens 2 has been announced to be 43deg by 29deg. Considering that the human vision has 210deg by 150deg field of view (including eye movements) this is still rather lacking. Heads up displays can solve this by using the crane cabins window as the AR screen. HMD's add the potential of being able to cover the entire view of what the crane operator sees and doesn't require clunky headsets. It can be noted that the main window of the crane's cabin is only 60deg by 50deg and is static. This would mean that we lose the ability to portray AR objects that are in blind spots for the crane. Other factors are cost and visibility. Heads Up displays are in general very costly in comparison to AR headsets because of the requirement of needing large specialized glass. It can be said that there is no need to make the entire window an AR screen, but that would decrease the usage even more. The glass that is used for AR glass, in general, is a type of semi see-through glass. This type of glass is typically much darker as it only allows some of the light to come through. Covering the entire window with this material could reduce the visibility of the crane operator. This is also true for the AR headsets but the headsets have the advantage that you can take them off. Another potential solution can be to use Augmented Virtuality. Through using a Virtual Reality Headset that has built-in cameras we can portray the real environment in the headset and add our AR objects. VR tracking technology is largely based on having external reference devices called "Satellites" which can be easily mounted within a cranes cabin. Considering the top of the line VR headsets of today we have the HTC Vive Pro with its 110deg by 110deg FOV[9] and two cameras, one for each eye, it would certainly seem that that is the better option. The problem is that even though the HTC Vive Pro is capable of Augmented Virtuality it is still lacking in resolution and usability. The cameras of the HTC Vive Pro are only 612x460pixels per camera [21] compared to the screen resolution of 1440x1600 pixels per eye[9]. This will make the visibility quite poor and together with processing delays, will make for a nauseating experience. Another factor is that VR headsets limit the user's field of view to that of the screen. Comparing with Hololens, the Hololens still maintains our horizontal field of view while 30deg have been covered for AR objects. With the HTC Vive pro, we are always limited to seeing only the 110deg of environment because the headset has to be enclosed to block out external light. VR-headsets also require additional hardware to work as an external computer. The solution written in this thesis does utilize the external computer as well, but in AR headsets do have the possibility to be a standalone system. Factoring in the cost of VR headset and the computer needed to run it will make the price similar if not equal to HoloLens 1.

5.5 Getting positional data for our AR solutions

Both the Hologram on Load and the Miniature Boat solution both require that we need to know the exact location of the load at any given time. One solution is to add a positional tracker to the load. Motion Reference Units (MRU) are self-contained tracking units that are widely used in the offshore industry. Typical MRU's give us motion data such as velocity and acceleration in 6 degrees of freedom meaning that if we know the start position of our load, we can calculate the new position relative to the motion data. MRU's have many use cases such as motion compensation systems, ROV orientation tracking, antenna stabilization, Dynamic Position (DP) systems and etc[36]. In our case, we are only interested in its tracking capabilities. One major disadvantage of MRU's is that they can't be used as a wireless solution. MRU's need both power and control cabling to function. In an ideal world would this be integrated into the wire we use for lifting, but this will also reduce the wire strength. Another idea is if we just add a wire along with the main wire, but this adds additional wires to look out for. There is, of course, a possibility to add a battery pack to the MRU, but typical MRU's require a lot of power (around 6W for the smaller units [39][43]) and an additional wireless data transfer solution has to be made. Another solution for tracking the load is to use Radar systems, or Relative Positioning Reference Systems as they are called. These systems are based on that you have multiple radar beacons installed on the ship that function as reference points. The beacons function as reference points and can triangulate the position of our load [37]. A major advantage that these systems have over MRU is that the Radar systems have a large wireless range. MRU's and Radar systems retrieve a lot of positional data meaning that making a wireless solution would be difficult. With Radar systems can we have the base station near the computational unit while the load only has a transponder. With MRU's you would need to add wireless solutions to send over the data as the unit is mounted directly on the load. Current Radar systems also come with a battery option and making for a true wireless tracking system [38]. One major disadvantage with radar systems is that they induce a large delay as the radio waves have to travel back and forth. Another disadvantage is that you need relatively free sight to the load and the radio waves have trouble penetrating the water surface. All of this first and foremost means that the radar beacons have to be mounted near blind spots of the crane if we want to use it for our hologram solution. And secondly, this means that Radar systems can't be used for underwater lifting operations.

5.5.1 Disadvantages by mounting trackers on the load

The first problem we have to address is that in order for either of these solutions to work, we need to mount a tracker on the load. With containers and the like, this wouldn't be a problem but it isn't always said that we can mount something on the load. Building materials, for example, don't tend to have a generalized mounting point. A way to compensate for this problem is to add the tracker to the hook of the crane. This, on the other hand, will not track the rotation of the load, only the position, and the tracker would have to be small in size to fit on the hook.

One of the main disadvantages with both the MRU and the Radar system solution is that we need to compensate for "drifting" if we want to use them for tracking. Drifting occurs when the motion data isn't entirely accurate making the calculated position drift away from the actual position. This is less prominent with Radar systems as we update our position regularly, but with MRU's we mainly get motion data so we need to compensate for it. On the other hand with today's technology and precision, this problem is minimal but with extended use should the actual position be updated by other means.

5.6 Video

This animated section of the video was meant as an introduction to the concept of AR and what the AR solutions and problems they address were. The introduction to AR was based on that the watcher had some previous knowledge of what a hologram was and potentially had used AR devices before. AR has been highly popularized through sci-fi movies and in the gaming industry, but there is always a possibility that the watcher of the video has never heard of AR or holograms before. For instance, the game highlighted in section 2.3, the popular Pokemon Go, is a game for children. Unless specifically interested in the game, you wouldn't have been exposed to this AR technology. The game itself also doesn't explicitly say that it's based on AR, meaning that even though the watcher might've used it, he/she wouldn't have known that this is, in fact, AR technology. This is why the AR introduction focuses on several AR technologies. For the slight possibility that the user has been exposed to AR through either phones/tablets, Heads Up Displays or with AR headsets we want to explain that these technologies are all categorized as AR. The advantage is that this doubles as an introduction to all these various solutions. At the start of the video, we also introduce a blue cube as a hologram, referring back to this color as being an indicator that this is a hologram. The same blue colors are used on the Hologram On Load and the Information Widget introduction. This color was not used on the introduction of Miniature Boat introduction, which could lead to some confusion. With this solution, we want to get an overview of the ships scenario so its vital to be able to distinguish the various objects on board the ship. To showcase this a choice was made to not color the miniature boat as a hologram, but instead, give all the vital objects specific colors so they are distinguishable. Although it was explained through the narration, to get an initial notion that this is, in fact, a hologram might not be understandable. Another downside with how this solution was introduced was that there was no movement on the ship. The visuals that were shown were equal to a static image where we also didn't move the camera around to give a better perspective. This more or less defeated the purpose of the miniature boat, where we wanted to show that we easily can get a better awareness of the scenario if we look through a birds-eye-view perspective. Instead, the Miniature Boat solution was also shown briefly in the recorded section of the video, where care was taken to move around as much as possible to give the notion give different perspectives. The movements of the ship and load, on the other hand, were stationary at this point which didn't give the impression it needed. This conclusion was drawn by the lack of survey feedback received on the Miniature Boat solution. Although a factor could be that the participants of the survey did understand the application of the Miniature Boat solution, but deemed that it wasn't useful for crane operations.

5.7 The survey

Considering that the survey was created as a digital survey, this gave us several advantages. First and foremost we have that the survey was accessible at all times. Through the course of the project, several crane-operators were asked to try out the AR solutions on-site but none had time to travel to OSC to try it. Making the survey digital made it possible for anyone to take the survey without the extra expense of travel times. This also meant that they could take the survey whenever they had time, be it free time or available time at work. The downside is of course that the participants of the survey can't test out the solutions and get a real feel of how it works. But on the other hand not having to test the solutions can also have its advantages. Considering that AR is a fairly new technology, the technologies that are available today can hinder the participant in seeing the true usage of AR. If the focus falls more on the setbacks with the technology then the rating of the AR solutions will be biased. Another downside with a digital survey as opposed to on-site testing is that we can't interview the participant of the survey meaning discussions are impossible. The only feedback we will gain is restricted to the scope of the questions from the survey. It can be debated that it would still be possible to conduct an online interview with the participants but this requires to set up schedules and organize meetings, losing the time saved by just taking an online survey. But, a factor to consider here is that interviews would make the survey lose its anonymous status because we have to be in direct contact with the participant. If we as AR experts don't interact with the participant at all, we can't bias him/her towards liking the technology or solutions either. With the survey being digital we also get the advantage of using digital tools to explain concepts and terminology. For instance, the usage of YouTube to upload an explanatory video which we can include in the survey itself. Another advantage is that we can easily download the responses of the survey and analyze it with digital tools. And as opposed to potential handwritten responses, the text will also always be legible. Here it can be said that these advantages could still be present even if the participants tried the solution in real life. The same digital survey could be used under interviews or before/after the participant had tried the solutions.

5.7.1 Survey structure and questions

The purpose of the survey was twofold, to rate the current AR solutions and to gather new ideas for AR solutions. Considering that we wanted to collect qualitative data, the main way to gather data was through asking open answer questions. The main advantage here is that the participants can elaborate as much as they want on the solution. This also meant that the participants could provide terminology and insight that was earlier unknown for me the creator of the survey. This terminology could then be used further in the research of the thesis. If all questions were in the multi-choice form then the survey wouldn't provide new knowledge, just acknowledge what was already researched. But this also has a setback, we expect the participant to actually be creative and answer to his/her fullest. This isn't always the case. Even when all questions are obligatory to answer, the extent of the participant's answer can be lacking. Another factor is that there is a high risk of the data you get to be less precise then what you require for the application.

5.8 Further work

Through this thesis, many potentials for AR solutions were explored. The implemented solutions are just some of the many possibilities that AR can provide which crane operators can't think of. Solutions created with this thesis took inspiration from crane operators and research of them, resulting in solutions that are based on their ideas. Further research at what is possible with AR both with current and new technology has to be researched. More data can also be collected to check the usefulness of the solutions at hand and testing can be performed in real life scenarios. The solutions proposed in this thesis can also be expanded before it can be considered a complete prototype.

Chapter 6

Conclusions

In this thesis, we look at the potential of using Augmented reality in crane operations. Multiple real-life problems relevant to crane operations were addressed and an Augmented Reality solution was proposed for each. The three problems addressed were blind zones, availability of information for the job, and the perspective in a crane cabin. Two of the problems were directly addressed with implementations which were tested out in a dome simulator. These solutions were implemented using the Microsoft HoloLens AR headset. An explanatory video was made to showcase the solutions and give a general introduction to AR. Using the video, a survey was conducted to get subjective feedback from crane operators.

To solve the blind zones' problem, the suggested solution was to add a marker or a direct model of the load on the load hanging in the crane. This AR model would work as a hologram that followed the actual load in real life. Because AR functions with local projections, the model would overlay the real world and anything that got in the way of the load hanging in the crane. This way the crane operators would be able to see how the load is positioned or rotated at all times. The second problem addressed, availability of information for the job, was addressed by the idea to have a floating display, or "widget", to show information that was relevant to the job. This widget would float in front of the user so that there is no need to "look away" towards systems in the crane cabin. The third and final solution to solve the perspective problem was to make a miniature scene of the scenario. A miniature boat model would float next to the operator's chair and give a birds-eye-view perspective.

At the end of the testing period, a total of 5 crane operators answered the survey. Through the results of the survey, it was shown that the Hologram on Load solution was subjectively the most useful solution out of the three. The information widget came at second place, but it was suggested that this idea could be expanded to be used as a common information hub. The miniature boat idea came third of the solutions, and through testing in the simulator proved to be more distracting than useful. From the final results did 4/5 operators agree that Augmented Reality can be used. Overall does this prove that there is a potential for Augmented Reality in crane operations with today's technology.

Chapter 7

Bibliography

[1] 3d Models for Professionals :: TurboSquid, . URL https://www.turbosquid.com/.

[2] Amazing-Anatomically-correct-Skull-Brain-Polaroid-3d-Hologram-Sticker.jpg (391×400), . URL https://www.picclickimg.com/d/l400/pict/323533722769_/Amazing-Anatomically-correct-Skull-Brain-Polaroid-3D-Hologram-Sticker.jpg.

[3] Offshore Simulator Centre, . URL https://www.offshore-technology.com/contractors/computer/offshore-simulator-centre/.

[4] OSC Homepage, . URL https://osc.no/.

[5] Vuforia Image Targets, . URL https://library.vuforia.com/content/vuforia-library/en/articles/Training/Image-Target-Guide.html.

[6] Vuforia, Developing Vuforia Apps for HoloLens, . URL https://library.vuforia.com/articles/Training/Developing-Vuforia-Apps-for-HoloLens.

[7] Vuforia, Optimizing Target Detection and Tracking Stability, . URL https://library.vuforia.com/articles/Solution/Optimizing-Target-Detection-and-Tracking-Stability.html.

[8] Vuforia Targets, . URL https://library.vuforia.com/articles/Solution/Targets.html.

[9] What Do We Need to Know About HMD - its Visual Features and Problems — ARVI VR, September 2018. URL https://vr.arvilab.com/blog/hmd-visual-features.

[10] Quaternions and spatial rotation, June 2019. URL https://en.wikipedia.org/w/index.php?title=Quaternions_and_spatial_rotation&oldid=900749378. Page Version ID: 900749378.

[11] Mixed reality user bc3tech. Can't get PC to connect to the device, March 2016. URL https://forums.hololens.com/discussion/635/cant-get-pc-to-connect-to-the-device.

[12] Rick Bishop. Why are Microsoft APIs and their documentation so bad? Is .NET to blame? - Quora, March 2019. URL https://www.quora.com/Why-are-Microsoft-APIs-and-their-documentation-so-bad-Is-NET-to-blame.

[13] Nahila Bonfiglio. DeepMind partners with gaming company for AI research. *The Daily Dot*, October 2018. URL https://www.dailydot.com/debug/unity-deempind-ai/.

[14] Brandon Bray and Matt Zeller. Billboarding and tag-along - Mixed Reality, March 2018. URL https://docs.microsoft.com/en-us/windows/mixed-reality/billboarding-and-tag-along.

[15] Dorrene Brown, Jesse McCulloch, Matt Zeller, and Brandon Bray. Environment considerations for HoloLens - Mixed Reality, April 2019. URL https://docs.microsoft.com/en-us/windows/mixed-reality/environment-considerations-for-hololens.

[16] Gianluca Busato. Why you should start using Augmented Reality (AR) and Gamification?, March 2018. URL https://medium.com/enkronos/why-you-should-start-using-augmented-reality-ar-and-gamification-1fdb1e1e57e6.

[17] Andrew C. Latency/Jitter Issue with Holographic Remoting, October 2017. URL https://forum.unity.com/threads/latency-jitter-issue-with-holographic-remoting.466745/.

[18] Thomas Caudell and David Mizell. Augmented reality: An application of heads-up display technology to manual manufacturing processes. volume 2, pages 659–669 vol.2, February 1992. ISBN 978-0-8186-2420-9. doi: 10.1109/HICSS.1992.183317.

[19] Tom Caudell. AR at Boeing (1990), 1990. URL http://www.idemployee.id.tue.nl/g.w.m.rauterberg/presentations/hci-history/tsld096.htm.

[20] Seth Colaner. What's Inside Microsoft's HoloLens And How It Works. August 2016. URL https://www.tomshardware.com/news/microsoft-hololens-components-hpu-28nm,32546.html.

[21] Dakenpachi. Vive Pro Camera FPS / Resolution?, July 2018. URL https://community.viveport.com/t5/Vive-SRWorks-SDK/Vive-Pro-Camera-FPS-Resolution/m-p/20731#M439.

[22] Lisa Eadicicco. Microsoft Releasing HoloLens Headset to Developers in March | Time. February 2016. URL https://time.com/4240446/microsoft-hololens-release-date-2016/.

[23] Mixed Reality forum user Ei2kpi. Holographic Remoting over USB Connection, February 2017. URL https://forums.hololens.com/discussion/6649/holographic-remoting-over-usb-connection.

[24] Viktor Farcic. Software Development Models: Iterative and Incremental Development, January 2014. URL https://technologyconversations.com/2014/01/21/software-development-models-iterative-and-incremental-development/.

[25] Beau Folsom, Graham Bury, Matt Zeller, Brandon Bray, and Github user: rwinj. What is a hologram? - Mixed Reality, March 2018. URL https://docs.microsoft.com/en-us/windows/mixed-reality/hologram.

[26] Blender Foundation. blender.org - Home of the Blender project - Free and Open 3d Creation Software, . URL https://www.blender.org/.

[27] Blender Foundation. Introduction — Blender Manual, . URL https://docs.blender.org/manual/en/dev/getting_started/about/introduction.html.

[28] Amir Ghahrai. Iterative Model in Software Development and Testing, November 2008. URL https://www.testingexcellence.com/iterative-model/.

[29] Demy Gielesen. Thesis introduction - YouTube, May 2019. URL https://www.youtube.com/playlist?list=PL9e6xSavMehtPwbf7ugMEqs2-TngZ_1Bn.

[30] Sid Hazra. How does the Microsoft HoloLens work? - Quora, March 2018. URL https://www.quora.com/How-does-the-Microsoft-HoloLens-work.

[31] Holocenter. What is Holography? | holocenter. URL http://holocenter.org/what-is-holography.

[32] Hudway. Head-Up Display that works for any car. URL https://hudwayglass.com/.

[33] Mixed Reality forum user jbienzms. Adding port forwards to Hololens, January 2017. URL https://forums.hololens.com/discussion/2301/adding-port-forwards-to-hololens.

[34] Jiun-Yin Jian, Gerry E. Miller, and Sahil Shah. Preventing Human Error in Crane Operations: A Case Study of Organizational and Design Elements. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 61:1695–1699, September 2017. doi: 10.1177/1541931213601912.

[35] KarlG. Magic Leap: Focus Planes (Too) Are a Dead End, December 2016. URL https://www.kguttag.com/2016/12/16/magic-leap-focus-planes-too-are-a-dead-end/.

[36] Kongsberg. Motion Reference Unit - Kongsberg Maritime, . URL https://www.kongsberg.com/maritime/products/vessel-reference-systems/motion-and-heading-sensors/motion-reference-unit/.

[37] Kongsberg. Relative position reference system, RADius - Kongsberg Maritime, . URL https://www.kongsberg.com/maritime/products/vessel-reference-systems/position-systems/relative-position-reference-system-radius/.

[38] Kongsberg. KONGSBERG RADIUS 700, LONG RANGE BATTERY OPERATED TRANSPONDER, June 2018. URL https://www.kongsberg.com/globalassets/maritime/km-products/product-documents/radius-700---long-range-battery-operated-transponder-.

[39] Kongsberg. KONGSBERG miniMRU, THE COMPACT REFERENCE UNIT, December 2018. URL https://www.kongsberg.com/globalassets/maritime/km-products/product-documents/minimru.

[40] Rick Van Krevelen. Augmented Reality: Technologies, Applications, and Limitations. 2007. doi: 10.13140/rg.2.1.1874.7929. URL http://rgdoi.net/10.13140/RG.2.1.1874.7929.

[41] Unity forum user LostLogic. Device Remote never connects for HoloLens, October 2017. URL https://forum.unity.com/threads/device-remote-never-connects-for-hololens.501905/.

[42] msftgits Microsoft Github user. This is a MixedRealityToolkit style repository for code bits and components that may not run directly on Microsoft HoloLens or immersive headsets but instead pair with them to build experiences.: .., August 2016. URL https://github.com/microsoft/MixedRealityCompanionKit. original-date: 2016-06-28T19:59:44Z.

[43] Norwegian-subsea. Norwegian-subsea,Subsea Motion Sensors, BEST SWaP-C SUBSEA MOTION SENSOR. URL https://norwegian-subsea.no/wp-content/uploads/brochures/NORSUB_Subsea.pdf.

[44] Jason Odom. Holographic Remoting—What It Is & Why It Makes the HoloLens So Much Better. *Next Reality*, November 2016. URL https://next.reality.news/news/holographic-remoting-what-is-why-makes-hololens-so-much-better-0174848/.

[45] Jon Primrose. Glossary of Technical Theatre Terms - Peppers Ghost. URL http://www.theatrecrafts.com/glossary/pages/peppersghost.html.

[46] Lawrence J. Prinzel. NASA / TM-2004-213000 (CORRECTED COPY) Head-Up Displays and Attention Capture. 2004.

[47] Ivan E. Sutherland. A Head-mounted Three Dimensional Display. In *Proceedings of the December 9-11, 1968, Fall Joint Computer Conference, Part I*, AFIPS '68 (Fall, part I), pages 757–764, New York, NY, USA, 1968. ACM. doi: 10.1145/1476589.1476686. URL http://doi.acm.org/10.1145/1476589.1476686. event-place: San Francisco, California.

[48] Matt Swider and Michelle Fitzsimmons. Magic Leap One release date, price and features. *TechRadar*, January 2019. URL https://www.techradar.com/news/magic-leap-one.

[49] W T. Nelson, Merry M. Roe, Robert Bolia, and Rebecca M. Morley. Assessing Simulator Sickness in a See-Through HMD: Effects of Time Delay, Time on Task, and Task Complexity. page 8, November 2012.

[50] Unity. Unity - Multiplatform - Publish your game to over 25 platforms, 2019. URL https://unity3d.com/unity/features/multiplatform.

[51] Rick Van Krevelen and Ronald Poelman. A Survey of Augmented Reality Technologies, Applications and Limitations. *International Journal of Virtual Reality (ISSN 1081-1451)*, 9:1, June 2010.

[52] Robert Workman, LiveScience Contributor | May 23, and 2013 03:48pm ET. What is a Hologram? URL https://www.livescience.com/34652-hologram.html.

[53] Matt Zeller, Graham Bury, Alex Turner, and Brandon Bray. HoloLens hardware details - Mixed Reality, March 2018. URL https://docs.microsoft.com/en-us/windows/mixed-reality/hololens-hardware-details.

[54] Matt Zeller, Jon M. Lyons, Jesse McCulloch, Brandon Bray, Sean Ong, Mike Riches, Alex Turner, and Github user Rwinj. Add holographic remoting - Mixed Reality, February 2019. URL https://docs.microsoft.com/en-us/windows/mixed-reality/add-holographic-remoting.

[55] Yisheng Zhou, Patricio Ricaud, Lucas Lehnert, and Aaron Gokaslan. Tracking and Mapping, April 2018. URL https://cs.brown.edu/courses/csci2951-i/week9/MixedRealityWeek_TrackingAndMapping.pdf.

Chapter 8

Appendix

8.1 Survey in English

Survey on crane operation information

This survey is a part of my Master Thesis "Augmented Reality for Offshore crane Operations". The survey has a focus on what information is vital for crane operations, what potential information a crane operator lacks and lastly if Augmented Reality can help with crane jobs. The survey consists of 35 questions with 1 video and will take around 30min to complete. Thank you for your input!

About privacy in this survey:
All data collected in this survey will be stored on this google forms service. Your google username might be saved by the google forms service but will only be used to remember if you have taken the survey before. When the data will be processed all data will be handled anonymously for this thesis.The information you share will only be used for the master thesis and will be deleted after the data has been analyzed. By taking this survey you accept these terms of use.

* Required

Initial questions

1. **Sex** *
   Mark only one oval.
   - ( ) Female
   - ( ) Male

2. **Age** *

   ____

3. **Years experience with Cranes** *

   ____

4. **Do you currently work as a crane operator?** *
   Mark only one oval.
   - ( ) Yes
   - ( ) No

5. **What type of crane or cranes do you have experience with?** *

   ____
   ____
   ____
   ____
   ____

6. **What do you think is the most challenging task with crane operations?** *

7. **What is the required information BEFORE a crane lifting job?** *

8. **Do you feel there is any information you usually lack BEFORE lifting jobs?** *

9. **What is the required information DURING a crane lifting job?** *

10. **Do you feel there is any information you lack DURING lifting jobs?** *

11. **What information do you think could make a lifting operation safer?** *

12. **Which visual checks do you perform before an lifting operation?** *

13. **In what scenarios do you feel that the crane has blind spots? / Do you ever experience that the crane has blind spots? If you have a specific crane in mind please also define the type in this question.** *

14. **Considering a Crane-cabin, do you think the layout of controls/screens/other systems inside the cabin is optimal?** *
Mark only one oval.
- Yes
- No
- Uncertain

15. **If you could move the controls/screens/other systems in the crane cabin, where would you move them?** *

Augmented Reality

16. **Do you have any experience with AR (Augmented Reality) or VR (Virtual Reality)?** *
Mark only one oval.
- Yes
- No

17. **If so, which system?**

18. **How experienced would you say you are with AR/VR systems?** *

Mark only one oval.

- Never used it
- Tried it once
- Used it on several occasions
- I own either AR or VR systems
- I've tried making games/applications for AR/VR systems

19. **What do you think about AR or VR in games and professional scenarios?**

Mark only one oval.

- Not very useful
- Slightly useful
- Moderately useful
- Very useful
- Extremely useful

Introduction of my thesis

This is a short video I made where I showcase what I've made in my thesis. Please watch it before heading to the next section of the survey. This video also has a narrator explaining the various parts so be sure to turn on sound thanks!

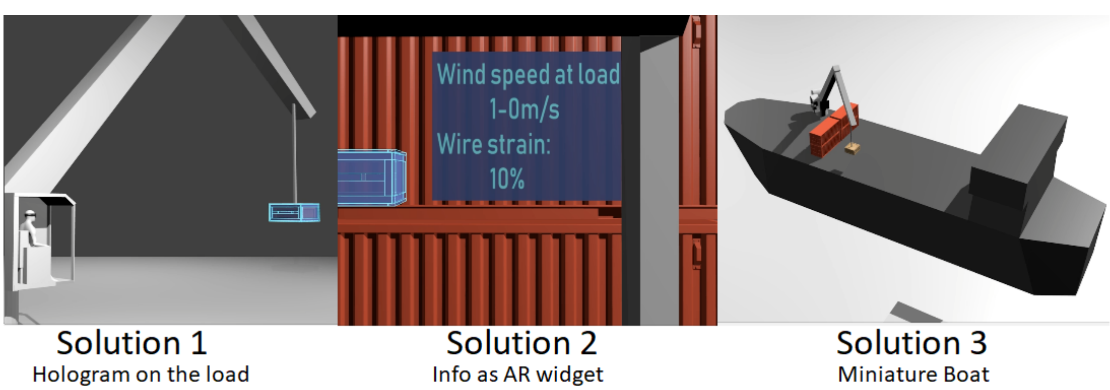
http://youtube.com/watch?v=pe0Il_cfpkE

Questions after seeing the video

20. **Was the video understandable?** *

Mark only one oval.

- Yes
- No



21. **Are the problems addressed in the video relevant for crane-operations?** *
Mark only one oval.

- Yes
- No

22. **What do you think about the solutions, can they improve your overall situational awareness when on the job?** *
Mark only one oval.

- Yes
- No

23. **Any comments on your last answer?**

24. **What did you like about the solutions showcased in the video?** *

25. **What didn't you like about the solutions showcased in the video?** *

26. **Which of the following solutions do you think is most useful:** *
Mark only one oval.

- Hologram of the load hanging on the crane
- Floating information widget
- Miniature boat

27. **Which of the following solutions do you think is least useful:** *
Mark only one oval.

- Hologram of the load hanging on the crane
- Floating information widget
- Miniature boat

28. **What else do you think could be portrayed as AR widgets/Holograms?** *

29. **If you could replace any controls/screens/other systems in the crane cabin with AR widgets, which system would you move where?** *

30. **What do you think about Augmented Reality?** *

Mark only one oval.

- Not at all useful
- Slightly useful
- Moderately useful
- Very useful
- Extremely useful

31. **Do you think that it will be more distracting then useful?** *

Mark only one oval.

- Yes
- No

32. **Do you think you could have a use for this technology in crane-operations today?** *

Mark only one oval.

- Yes
- No

33. **Do you think that AR systems will be used with crane-operations in the future?** *

Mark only one oval.

- Yes
- No

34. **Any comments on Augmented Reality?**

35. **Any comments on this survey?**

Powered by
Google Forms

8.2 Survey in Norwegian

Spørreundersøkelse for kranførere

Denne undersøkelsen er ei del av mi master oppgåve "Augmenterd realitet for offshore-kran operasjoner". Målet er å finne ut hvilken informasjon som er vesentlig for kran-løft operasjoner, hvilken informasjon kranføreren mangler ved kranløft og sist om Augmentert Realitet (AR) kan hjelpe ved kranløft. Undersøkelsen består av 35 spørsmål og ei video, og det hele vil ta ca 30min å fullføre. Takk på forhand for at du deltar i denne undersøkelsen!

Om bruk av privat data:
Informasjonen som du deler i denne undersøkelsen vil bli lagret i google forms sine tjenester. Ditt google brukernavn kan bli lagra automatisk av google forms tjenesten men vil kun bli brukt for å huske om du har besvart tidlegare. Selve stastikk beregninger vil bli foretatt anonymt. Delt informasjon vil kun bli brukt for bruk i master oppgåva for så å bli sletta etter at dataen er analysert. Ved å gjennomføre undersøkelsen aksepterer du desse vilkår.

* Required

Innledende spørsmål

1. **Kjønn** *
   Mark only one oval.

   - Mann
   - Dame

2. **Alder** *

3. **Hvor mange år erfaring har du med kran?** *

4. **Er kranfører din aktive stilling?** *
   Mark only one oval.

   - Ja
   - Nei

5. **Hva slags type kran eller kraner har du erfaring med?** *

6. **Hva vi du si er den mest utfordrende ved kranløft operasjonen?** *

7. **Hvilken informasjon er nødvendig FØR du gjennomfører et løft?** *

8. **Hvilken informasjon mener du at du som regel mangler FØR et løft?** *

9. **Hvilken informasjon er nødvendig NÅR du gjennomfører et løft?** *

10. **Hvilken informasjon mener du at du som regel mangler NÅR du gjennomfører et løft?** *

11. **Hvilken informasjon trur du kan brukast til å forbedre sikkerhet ved ei kranløft?** *

12. **Hvilken visuelle sjekk tar du før du gjennomfører et løft? ***

13. **I hvilken situasjoner merker du at kranen har blindsoner? / Merker du noensinne at kranen har blindsoner? Gjerne skriv type kran også. ***

14. **Viss du tar for deg kran-kabina, vil du si at plasseringen av system/skjermer/kontrollere er optimalt? ***
Mark only one oval.
    - Ja
    - Nei
    - Usikker

15. **Viss du har muligheten å endre på system-oppsettet i kranhuset, hvilken system/skjermer/kontrollere vil du flytte og hvor? ***

Augmentert Realitet

16. **Har du noen erfaring med AR (Augmenterd realitet) eller VR (Virtuel realitet)? ***
Mark only one oval.
    - Ja
    - Nei

17. **Hvis ja, hvilken system eller systemer er du kjent med?**

18. **Hvor erfaren vil du si du er med AR/VR systemer?** *

Mark only one oval.

- Aldri brukt det
- Har prøvd det ei gang
- Brukt det ved flere anledninger
- Jeg eiger enten et AR eller VR system
- Jeg har laga spel/applikasjoner til AR/VR systemer

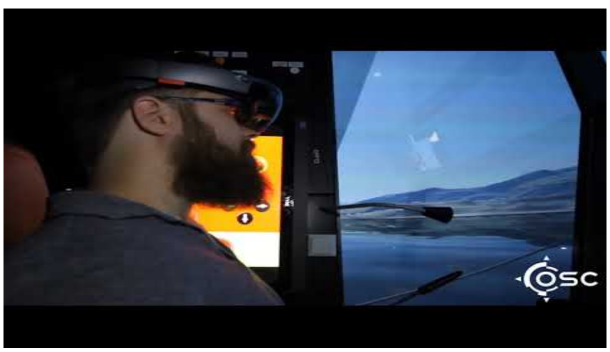
19. **Hva er din mening om bruk av AR/VR i spel eller proffesionelle sammenheng?**

Mark only one oval.

- Ikke brukbar
- Litt brukbar
- Brukbart
- Veldig brukbar
- Ekstremt brukbar

Introduksjon av min master oppgåve

Dette er ei kort video der jeg viser hva jeg har laga som del av min master oppgåve. Vennligst se videoen før du fortsetter videre med spørreundersøkelsen. Videoen har også tale som forklarer dei forskjellige scenene så ver sikker på å skru på lyd Takk!



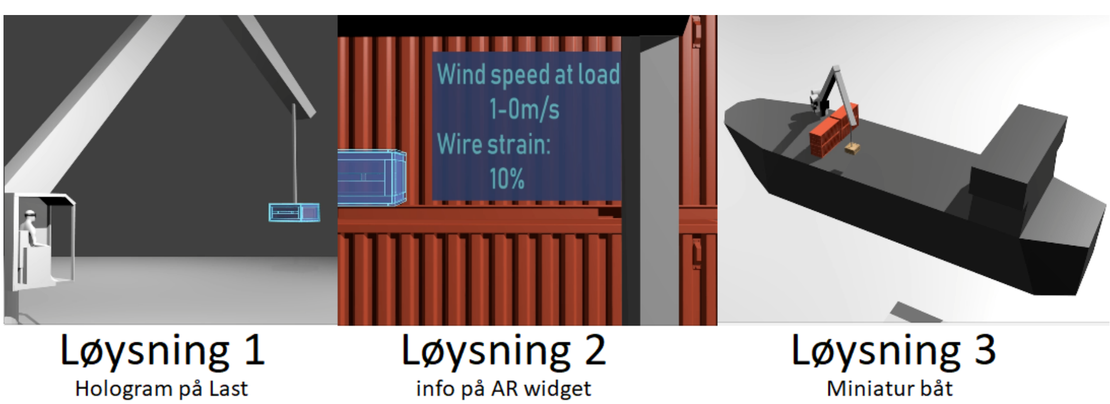
http://youtube.com/watch?v=ztaTBFbqTmw

Spørsmål til etter du har sett videoen

20. **Var videoen forstårleg?** *

Mark only one oval.

- Ja
- Nei



21. **Var problemstillingene som ble vist relevante til kranløft?** *

Mark only one oval.

- Ja
- Nei

22. **Hva syns du om løysningene som ble vist, kan dei øke din bevisthet av situasjonen når du gjennomfører ei kranløft?** *

Mark only one oval.

- Ja
- Nei

23. **Noen kommentar på din siste besvarelse?**

24. **Hva likte du av dei forskjellige løysningene som vart vist i videoen?** *

25. **Hva likte du ikke av dei løysningene som vart vist i videoen?** *

26. **Hvilken løysning i lista nedenfor vil du si var mest brukar?** *

Mark only one oval.

- Hologram av lasten som hang i kranen
- Flyggande informasjon widget
- Miniatur båt

27. **Hvilken løysning i lista nedenfor vil du si var minst brukar?** *

Mark only one oval.

- Hologram av lasten som hang i kranen
- Flyggande informasjon widget
- Miniatur båt

28. **Hva trur du kan ha nytte av å bli vist med AR/som hologram?** *

29. **Viss du har muligheten å erstatte systemer i kranhuset med AR, hvilken system/skjermer/kontrollere vil du flytte og hvor?** *

30. **Hva syns du om AR?** *
Mark only one oval.

- Ikke brukar
- Litt brukbart
- Brukbart
- Veldig brukbar
- Ekstremt brukbar

31. **Trur du at AR vil ver mer distraherende enn brukbar?** *
Mark only one oval.

- Ja
- Nei

32. **Trur du du kan ha bruk for denne teknologien i kranløft idag?** *
Mark only one oval.

- Ja
- Nei

33. **Trur du at AR kommer til å bli brukt for kran-løft i framtida?** *
Mark only one oval.

- Ja
- Nei

34. **Har du noen kommentar angåande AR?**

35. **Har du noen kommentar på undersøkelsen?**

Powered by
Google Forms

8.3 Server

```
using UnityEngine;
using System;
using System.Collections.Generic;
using System.Net;
using System.Threading;
using System.Net.Sockets;
using System.Text;
using System.Collections.Concurrent;

public class TcpNetworkServerManager
{

 //Received data can be FIFO so we get everything in order
 private ConcurrentQueue<string> receiveMessages;
 public string GetAnswer()
 {
  string item;
  if (receiveMessages.TryDequeue(out item))
  {
   return item;
  }
  return null;
 }

 //data to send must be LIFO so we always send newest positional data
 private ConcurrentStack<string> sendMessages;
 private string ReadMessageToSend()
 {
  string item;
  if (sendMessages.TryPop(out item))
  {
   sendMessages.Clear();
   return item;
  }
  return null;
 }
 public void SendMessage(string s)
 {
  if (!string.IsNullOrEmpty(s))
  {
   if(sendMessages != null)
   {
    //if we have filled the stack with 30 entries, clear it so we
       dont have overflow
    if(sendMessages.Count() >= 30){
     sendMessages.Clear();
    }
    sendMessages.Push(s);
   }
  }
 }

 public TcpNetworkServerManager(int port)
 {
  receiveMessages = new ConcurrentQueue<string>();
  sendMessages = new ConcurrentStack<string>();
  TcpListener tcpserver = new TcpListener(IPAddress.Any, port);
```

```
  tcpserver.Start();
  Thread thread = new Thread(() =>
  {
   try
   {
    while (true)
    {
     TcpClient tcpclient = tcpserver.AcceptTcpClient();
     NetworkStream stream = tcpclient.GetStream();
     try
     {
      byte[] bytes = new byte[tcpclient.ReceiveBufferSize];
      stream.Read(bytes, 0, bytes.Length);//It holds here
      receiveMessages.Enqueue(Encoding.ASCII.GetString(bytes, 0, bytes.Length));
      while (true)
      {
       string s = ReadMessageToSend();
       if (!string.IsNullOrEmpty(s))
       {
        bytes = Encoding.ASCII.GetBytes(s + "\n");
        stream.Write(bytes, 0, bytes.Length);
        receiveMessages.Enqueue("Send coords");
       }
      }
     }
     catch (Exception) { }
     stream.Close();
     tcpclient.Close();
    }
   }
   finally
   {
    tcpserver.Stop();
   }

  });
  thread.Start();
 }
}

public class Unity_serv : MonoBehaviour
{
 private TcpNetworkServerManager Tcp;
 private float Counter;

 public Finder f;

 // Start is called before the first frame update
 void Start()
 {
  Counter = 0f;
  startServ = true;
  error = false;
 }

 private string oldPos;
 private bool startServ;
 private bool error;
```

```

 // Update is called once per frame
 void Update()
 {
  if(f.CabinTracker != null && f.Crane != null)
  {
   if (startServ)
   {
    if(!error)
    {
     try
     {
      Tcp = new TcpNetworkServerManager(8080);
      Debug.Log("tcp enabled");
     }
     catch (Exception ex)
     {
      Debug.Log("ex: " + ex);
      error = true;
     }
     if(!error)
     {
      Debug.Log("Server Start");
      startServ = false;
     }
    }

   }else
   {
    //Hologram on load

     //Find a vector from our reference point to load
    Vector3 TrackerToLoad = f.Crane.transform.position - f.CabinTracker.transform.position;
     //The vector holds rotation information, we want a vector from a reference point that isn't rotated
    TrackerToLoad = Quaternion.Inverse(f.CabinTracker.transform.rotation) * TrackerToLoad;
     //Rotate the rotation of the load so we get its localRotation
    Quaternion rot = Quaternion.Inverse(f.CabinTracker.transform.rotation) * f.Crane.transform.rotation;


    string message = TrackerToLoad.x + "/" + TrackerToLoad.y + "/" + TrackerToLoad.z;
    message += "/" + rot.x + "/" + rot.y + "/" + rot.z + "/" + rot.w;





    //Miniature boat

     //Same as with hologram on load, only we use the ship's hull as reference point
    Vector3 HullToLoad = (f.Crane.transform.position - f.Hull.transform.position);
```

```
            //Reset the rotation that the Hull might've had.
            HullToLoad = Quaternion.Inverse(f.Hull.transform.rotation) * HullToLoad;


            message += "/" + HullToLoad.x + "/" + HullToLoad.y + "/" + HullToLoad.z;
            //For the miniature boat it isn't nescassary to include the rotation for the load, so we send rotation data on how the crane behaves instead
            message += "/" + f.CraneBase.transform.localRotation.eulerAngles.y + "/" + f.Boom1.transform.localRotation.eulerAngles.x + "/" + f.Boom2.transform.localRotation.eulerAngles.x;




            if (!message.Equals(oldPos))
            {
             //Only send positions if the positions have actually changed
             if (Tcp != null)
             {
              Tcp.SendMessage(message);
             }
             else
             {
              Debug.Log("TCP remains null");
             }
            }

            if (Counter >= 1f)//Once a second
            {
             if (Tcp != null)
             {
              //Check if any commands need to be executed
              string log = Tcp.GetAnswer();
              if (!string.IsNullOrEmpty(log))
              {
               Debug.Log(log);
              }
             }
             else
             {
              Debug.Log("TCP remains null");
             }
             Counter = 0f;
            }
            Counter += Time.deltaTime;
          }


        }
      }

    }
```

8.4 Client

```
using UnityEngine;
using System;
using System.Collections.Generic;
using System.Collections.Concurrent;
using System.Globalization;
#if !UNITY_EDITOR //HoloLens runs UWP applications, while the Unity editor doesnt
 using System.IO;
 using System.Threading.Tasks;
 using Windows.Networking;
 using Windows.Networking.Sockets;
#endif

public class UWPTcpNetworkClientManager
{
#if !UNITY_EDITOR
 private StreamWriter writer = null;
#endif

 //LIFO buffer to always return the latest data
 private ConcurrentStack<string> cb;
 public string GetLastAdded()
 {
  if(cb != null)
  {
   if (cb.TryPop(out string item))
   {
    cb.Clear(); //all older data can be ignored and deleted
    return item;
   }
  }
  return null;
 }

 public UWPTcpNetworkClientManager(string IP, int port)
 {
  cb = new ConcurrentStack<string>();

#if !UNITY_EDITOR

  Task t = Task.Run(async () => {
   cb.Push("Connecting");
   try
   {
    while(true)
    {
     StreamSocket socket = new StreamSocket();
     await socket.ConnectAsync(new HostName(IP),port.ToString());
     cb.Push("Connected!");

     writer = new StreamWriter(socket.OutputStream.AsStreamForWrite());
     StreamReader reader = new StreamReader(socket.InputStream.AsStreamForRead());

     try
     {
```

```
      while(true){
       try
       {
        string data = await reader.ReadLineAsync();
        cb.Push(data);
       }
       catch (Exception ex2)
       {

        cb.Push(ex2.Message);
       }

      }

     }
     catch (Exception ex) {
      cb.Push(ex.Message);
     }
     writer = null;
    }
   }catch(Exception ex1){
    cb.Push(ex1.Message);
   }
  });
#endif
 }

 public void SendMessage(string data)
 {
#if !UNITY_EDITOR

  if (writer != null) Task.Run(async () =>
  {
   await writer.WriteAsync(data);
   await writer.FlushAsync();
  });
#endif
 }
}

public class UWP_Client : MonoBehaviour
{

 private int Count;
 private UWPTcpNetworkClientManager TcpClient;
 private float Counter = 0f;
 private Quaternion initRot;

 public GameObject target;
 public GameObject reference;
 public GameObject miniTarget;
 public GameObject Hololens;

 public GameObject CraneBase;
 public GameObject Boom1;
 public GameObject Boom2;

 public GameObject Indicator;
 public Material Red;
```

```
// Start is called before the first frame update
void Start()
{
 initRot = reference.transform.rotation;
 Counter = 0f;
 Count = 0;
 TcpClient = new UWPTcpNetworkClientManager("192.168.0.2", 8080);
}


// Update is called once per frame
void Update()
{
 if (Count == 500)
 {
  //Wait arbitrary time for everything to load
  TcpClient.SendMessage("Hello World");
  Debug.Log("Sending");
 }
 else if(Count == 750)
 {
  //Wait some more and turn off vuforia tracking, making the HoloLens take over enterily for tracking
  //This was mainly implemented because the vuforia library takes control of the HoloLens's webcam
  //meaning we cant record video with the HoloLens.
  Indicator.GetComponent<Renderer>().material = Red; //indicate that we no longer use Vuforia
  Vuforia.VuforiaBehaviour.Instance.gameObject.SetActive(false);
 }


 string log = TcpClient.GetLastAdded();
 if (!string.IsNullOrEmpty(log))
 {
  if (log.Contains("/"))
  {
   string[] values = log.Split('/');

   CultureInfo ci = (CultureInfo)CultureInfo.InvariantCulture.Clone();
   ci.NumberFormat.NumberDecimalSeparator = ".";

   float[] floats = new float[values.Length];
   for (int i = 0; i < values.Length; i++)
   {
    floats[i] = float.Parse(values[i], ci);
   }



   //For the Hologram on load

   //Because the load is a child of the reference point, and the reference point has to have a specific scale set on runtime,
   //we need to scale the position to move in the global world, not the scaled world of the reference point.
```

```
      target.transform.localPosition = new Vector3(floats[0],
         floats[1], floats[2]) / reference.transform.localScale.x;
      target.transform.localRotation = new Quaternion(floats[3],
         floats[4], floats[5], floats[6]);



      //For the miniature boat model

      //In this scenario, the model of the load itself has been scaled
         down, so the movement it has to take also needs to be scaled
         down.
      miniTarget.transform.localPosition = new Vector3(floats[7],
         floats[8], floats[9]) * miniTarget.transform.localScale.x;

      //Updating the rotation of the crane's booms.
      //Some values have to be negated because of how the model in the
         simulator differs from the on the HoloLens
      CraneBase.transform.localRotation =
         Quaternion.Euler(0,floats[10],0);
      Boom1.transform.localRotation = Quaternion.Euler(-floats[11], 0,
         0);
      Boom2.transform.localRotation = Quaternion.Euler(-floats[12], 0,
         0);

     }
    }


    if (Count <= 750)
    {
     Count++;
    }
  }
}
```