CODEN:LUTEDX/(TEIE-5440)/1-98(2020)

Industrial Electrical Engineering and Automation

# Development of a Device to Move Pan-Tilt-Zoom Cameras Using Hand Gestures



**Irene Sempere Díaz**
**Amelie Bäck**

Division of Industrial Electrical Engineering and Automation
Faculty of Engineering, Lund University

# Development of a device to move Pan-Tilt-Zoom cameras using hand gestures

Irene Sempere Díaz
Amelie Bäck

LUND UNIVERSITY

Division of Industrial Electrical Engineering and Automation

MSc Thesis
CODEN:LUTEDX/(TEIE-5440)/1-98(2020)

Division of Industrial Electrical Engineering and Automation
Lund University
Box 118
SE-221 00 LUND
Sweden


Printed in Sweden by Tryckeriet i E-huset.
Lund 2020

# Abstract

Nowadays, many industries are investing a lot of resources in the development of new technologies that interpret hand gestures and use them as a communication interface between the user and the system. However, the surveillance industries have remained on the sidelines of this technological development. Instead, the joystick and the mouse are the most common devices found in control rooms. Using them together with a Video Management System, the operators are able to drive the cameras and navigate between different views. Both devices fulfill their functions successfully but some operators report persistent pain after a demanding work day. Based on the points made above, the purpose of this master thesis is to investigate the possibility of using hand gestures as an interface to control surveillance cameras and design the hand gesture vocabulary to be used. To verify the validity of the idea a prototype will be developed to control Pan-Tilt-Zoom cameras, with special emphasis on having an intuitive and ergonomic design.

Throughout the project different solutions to solve the touch-free control of the cameras have been discussed and investigated. The most suitable concept found was to attach infrared sensors to a pyramid shaped case, in such a way that several sensors could be used to create a sensing field. By measuring the exact position of the hand at all times, the gestures made by the user could be interpreted by the system. The infrared sensors have proven to be very good at sensing the distance to soft materials such as hands with an accuracy of one millimeter.

The results obtained from the usability test made in this thesis show that the device is intuitive for the users and the potential of the device is also remarked for surveillance applications. In addition, the results show that the readings made by the sensors are very precise and that the interpretation of the achieved data into gestures works satisfyingly. However, even though the given results were mainly positive, it shows that there is still work left to do in order for the device to be competing with the joystick.

# Acknowledgement

We would like to express our gratitude to everyone who has contributed to this thesis. To our academic supervisor Gunnar Lindstedt, for believing in and supporting us the entire way through this thesis.

We wish to thank all our colleagues at Axis Communications in Lund who have contributed to the results and offered their knowledge to help us. Also, thank you all very much for making our time at Axis Communications very enjoyable. Above all, we would like to offer our greatest appreciation to our industrial supervisor, Simon Preutz, for always offering help and contributing to discussions that made the work move forward. A special thank you to Johanna Christensson, who helped us throughout the design of the mechanical parts, your help has been vital for us in order to bring our ideas to life. We would also like to give a special thanks to Alexander Löfqvist, who contributed with great help and inputs regarding the software development of the thesis. We wish to express our great appreciation to Benjamin Heymann, for giving us valuable information regarding the PTZ cameras that lead to the results of the thesis. To Madelene Svartberg, a special thank you for helping us to design the usability tests. Without your discussions, the achieved results would not be as clear as they are today. In addition, we would like to thank everyone who took part in our tests and willingly shared their precious time with us.

Lastly, we would like to show our gratitude to our friends and family for supporting us throughout the development of this thesis.

# Acronyms

**ACK** Acknowledgment. 19

**CGI** Common Gateway Interface. 30, 39

**FoV** Field-of-View. 29–31, 76

**GPIO** General Purpose Input Output. 36

**GUI** Graphical User Interface. 42

**HMI** Human Machine Interface. 11, 12

**HTTP** Hypertext Transfer Protocol. 20, 30, 39

**I2C** Inter-Integrated Circuit. 19, 20, 30, 34, 35, 38, 56

**IR** Infrared. 16, 17, 19, 28–30, 32, 68

**LCD** Liquid Crystal Display. 43, 44, 53, 59, 64

**MEMS** Micro-ElectroMechanical Systems. 18

**NACK** Not Acknowledgment. 19

**PTZ** Pan-Tilt-Zoom. 1–7, 14, 15, 26, 40, 41, 47, 48, 63, 65, 68

**SCL** Serial Clock. 19, 34, 35

**SDA** Serial Data. 19, 34, 35

**SSH** Secure Shell. 39

**ToF** Time-of-Flight. 28

**UX** User Experience. 14

**VMS** Video Management Software. 6, 64, 67

# Contents

# 1

# Introduction

This Master Thesis was performed at Axis Communications, a company founded 1984 in Sweden. Axis Communications originally started out as an IT company selling print servers, but after applying their knowledge in networks and embedded computing to develop network cameras for surveillance, they became one of the world leading companies in the industry. [1]

Axis is constantly expanding and it has today more than 3600 employees worldwide in more than 50 countries [2]. The Axis network cameras are installed in many places around the world, such as airports, grocery stores, prisons, casinos and wherever surveillance cameras are needed. Almost all of the research and development is made at the headquarters in Lund, Sweden, which is also where this thesis was performed.

Axis product catalog contains a wide variety of network cameras specialized for different scenarios, such as explosion-protected cameras or panoramic cameras among others. Apart from the network cameras Axis is developing a lot of accessories for the cameras as well as video encoders, audio network systems and solutions for access control, to mention a few. This thesis has been performed with a Pan-Tilt-Zoom (PTZ) camera, one of the most common network cameras in Axis portfolio. The PTZ cameras use their pan, tilt and zoom functions to provide both wide-area coverage and great detail with a single camera. The image quality and the ability to zoom in makes it possible to verify detected events.

## 1.1 Problem formulation

- Investigate the possibility of controlling PTZ cameras using a hand gesture interface.
- Implement the Pan, Tilt and Zoom functionalities to be used with hand gestures.
- Find a vocabulary for the hand gestures that is as ergonomically correct as possible.

## 1.2 Purpose

The main objective of this Master Thesis is to develop a hand gesture control device for PTZ cameras. As of today the surveillance cameras are controlled mainly with a joystick, and the goal is therefore to be able to solve as many as possible of the functionalities included in the joystick but with a hand gesture device. There are no current known similar solutions that could be found out on the market that controls surveillance cameras with help from gestures. Therefore, with the results of this thesis it will be possible to decide if the technology is good enough to further invest in and see if it is a competitive alternative to the current control devices.

The idea that it can be good to control surveillance cameras with hand gestures comes from many different causes. One of them is the fact that several other industries are trying to explore the field of hand gestures and how they can be adapted into technical solutions. Therefore, it would be interesting to see if it is possible to do the same in the surveillance camera industry and how many of the functionalities from the joystick that can be adapted to the hand gesture interface.

In addition, communicating to a machine by using the hands as the only interface is very intuitive for the user. For instance, to move, point or swipe with the hand in the desired direction and the machine will move that way.

From the ergonomics point of view it is also interesting to develop a new device to control the surveillance cameras. Compared to the joystick which puts a heavy load on the wrist or the keyboard which could be hard for the joints in the fingers, a gesture vocabulary could be developed so that the physical effort required and the discomfort level are as small as possible.

## 1.3 Previous work

No research has previously been made at Axis Communications about topics related to this master thesis. Therefore this work has not been founded on any already existing work.

With regards to research and work done outside Axis Communications none has influenced the thesis enough to be credited as a key reference.

## 1.4 Division of work

The main part of the work in this thesis as well as the report has been performed together by Irene Sempere Díaz and Amelie Bäck.

Irene Sempere Díaz was in charge for the mechanical design and drawings, while Amelie Bäck had the main responsibility for implementing the communication with the camera.

## 1.5 Outline

The rest of this thesis report is structured as follows:

Chapter 2 is the background chapter which begins by discussing what Pan-Tilt-Zoom cameras there are, what they are used for and how they work, together with the current control devices there are to control them. In addition there is also a general discussion about hand gestures which is both containing the daily use of hand gestures as well as their current applications in new technologies.

Chapter 3 contains the literature study that was made for this thesis. Here different approaches to design a hand gesture vocabulary as well as their limitations are presented, together with technologies to use when detecting the hand gestures. All this is used to present the concepts and ideas that later are evaluated and rated. In addition, there is a section that is describing the communication that will be used to communicate between the device and the camera.

Chapter 4 describes the methodology used in the thesis. The methodology is divided into several parts, where the first one discusses the requirements that has to be fulfilled concept- and electronic-wise. The second one focuses on the concept and sensor generation and evaluation. Afterwards there is the development process of the first prototype explained, followed by a section that describes how results were gathered. Finally comes the explanation of how the second prototype was developed.

Chapter 5 presents the results gathered in Chapter 4. Chapter 6 discusses the results found in Chapter 5. Some thoughts on future applications and development for the device are also presented here.

Chapter 7 summarizes the conclusions that are drawn from the results of this master thesis.

# 2

# Background

In the following chapter the background of this thesis is described. It begins with introducing the Pan-Tilt-Zoom cameras and their specifications, as well as the current control devices that are used with them. This is followed by a brief explanation about how the control devices are used in different surveillance applications.

The chapter ends with an introduction to hand gestures and some technical applications where they are used today.

## 2.1 Pan-Tilt-Zoom cameras

The Pan-Tilt-Zoom (PTZ) cameras are capable of moving the lens around different axes in order to cover a wider field of view in comparison with other fixed cameras. Pan means that the camera can move within its horizontal plane (left/right), i.e. it can rotate around its fixed vertical axis (see Figure 2.1a). The tilt movement implies that the camera can move in its vertical plane (up/down), i.e. it can rotate around its fixed horizontal axis (see Figure 2.1b). The zoom option consist in a variation of the focal length (distance between the lens and the image sensor) so that the image observed is closer and bigger (zoom in) or further and smaller (zoom out).

The development of the technology has made it possible to improve the quality of the PTZ cameras. This, along with the fact that they can be controlled from a remote location, either by obeying user commands in real-time or following the commands of a pre-defined route, have enabled these types of cameras to be found in many different scenarios. Nowadays it is possible to find them in television productions such as reality shows, in surveillance of indoor and outdoor spaces, in education to record lectures and in sports to cover better field points of view, among other applications [4].

After this brief and general introduction of the cameras, it should be noted that this project is especially focused on applications where the cameras are used for surveillance, since it is the main activity of Axis Communications. One of their

(a) Pan movement, rotation around the vertical axis.

(b) Tilt movement, rotation around the horizontal axis.

Figure 2.1: Explanation of the pan (a) and tilt (b) movements of the camera. Figures adapted from [3].

cameras, the model **AXIS P5655-E** [5] showed in Figure 2.2, is used to carry out all the tests throughout the design and development of the device. However, all PTZ cameras have the same configuration and it could be used with any other camera.

Figure 2.2: PTZ camera, model AXIS P5655-E, used throughout this project [5].

## Current control devices

Nowadays there are several devices used to control the PTZ cameras in all the applications previously mentioned. In this section, a brief review is presented about the products offered not only by Axis Communications but also the ones provided by its biggest competitors. Here it is assumed that these control devices are just used for surveillance purposes, although they can be used for other applications as well.

Axis Communications has three different control modules that can be used individually or combined to create a multi-functional control board [6]. The most commonly used module is the joystick presented in Figure 2.3a, which is mainly used to drive the camera to the user's desired position (Pan-Tilt) and with the right focus (Zoom). Besides that, it has several customized buttons implemented that can be used to define preset positions of the camera, to go to the next/previous view if the user can monitor several cameras on the screen or to replace the mouse right and left click buttons. The other two modules are the keypad and the jog dial. The first one, see Figure 2.3b, has only buttons that allow the user to easily change from one workspace to another and navigate between cameras, views and PTZ preset positions. The last one, see Figure 2.3c, has a small wheel and some customized buttons. It is mainly used to precisely navigate through recorded video. All the modules have USB ports to connect to the other modules and to the computer.

In addition to those devices, the company also provides a Video Management Software (VMS) to get the most out of cameras and control devices. This interface also allows the user to control the camera by using the mouse on the PC, which might be the best solution in small stores and offices.

(a) Joystick module. (b) Keypad module. (c) Jog dial module.

Figure 2.3: Three control devices provided by Axis Communications that are used to steer the PTZ cameras and navigate through different views and recorded videos [6].

Regarding the competitor's products, some quick searches on their web pages are enough to determine that their control devices can perform the same functions as the ones provided by Axis Communications. It is a fact that they are not aesthetically similar and they do not have the same module configuration. One example, the joystick system controller provided by Panasonic [7], see Figure 2.4a, which has the joystick in one gadget and a bigger keypad with a small screen in a separate panel. Another example is the all-in-one device provided by Hanwha Techwin [8]. In this case all the modules are built-in a unique platform and it has a bigger screen, as shown in Figure 2.4b. The only product that varies a little from those already mentioned is the one provided by HIKVISION [9], which has a touch screen instead of the traditional buttons (see Figure 2.4c). Despite these small differences, the most common PTZ camera control devices use the same technology and the same principles, specially when it comes to the Pan-Tilt-Zoom movements due to the fact that all the devices found operate the camera using a joystick.

(a) Ethernet 3D Joystick System Controller by Panasonic [7].

(b) SPC-7000 System Control Keyboard by Hanwha Techwin [8].

(c) DS-1600KI Network Keyboard by HIKVISION [9].

Figure 2.4: Three PTZ camera control devices provided by Axis competitors.

### Different use of control devices in surveillance applications

The fact that this thesis is only focused on the PTZ cameras and their controllers in applications related to the surveillance, does not mean that this devices are always used in the same way. The scope of the surveillance can be really broad, from private houses and small stores up to big companies, bridges or cities. That means that in each surveillance application the PTZ cameras and control devices will be used in a different way, i.e. the frequency of use of these devices will depend on the purpose for which this technology has been installed and on the user responsible for controlling them.

It is very difficult to know exactly how the control devices are used in each of the applications, to do so it is necessary to be close to the users, observe them and listen to their comments. An Axis Communications employee that has experience in this topic is *Benjamin Heymann*, *Product Specialist*. He knows how these control devices are used in the cameras installed by the company and he states the following:

First of all, there is no standard desk set up, whether there is a big room with different operators or a small room with just one operator, any of the users will most likely have a different set up on their desk. As a clarification, here *desk set up* refers to how many devices the operators have (keyboard, mouse, joystick module, keypad module, jog dial module, etc.) and how they are distributed in their workspace.

Another thought is that there is no rule neither on how the operators use the joystick, i.e. how many hands are in contact with the joystick or if they use their dominant hand (right hand if they are right-handed or left hand if they are left-handed) to steer the camera.

Apart from that, the frequency of use of the control devices, i.e. how often and how long operators work with them, varies to a large extent depending on where the cameras are installed. On the one hand, casinos have very extensive surveillance networks, there are cameras placed both on the outside of the buildings and on the inside so that no blind spot is left unprotected. As an example, Figure 2.5 shows one of the rooms of the *Majestar Casino* in Jeju island South Korea [10], in just one picture it is possible to count up to 10 surveillance cameras (see red circles in the image). Therefore, the operators working in the casino's control rooms are responsible for monitoring a great amount of network cameras. They have to be very active, navigating from one view to another very quickly and also steering the camera and zooming in/out very precisely in order to follow suspicious individuals or make sure that all games are played fairly (continuous monitoring). That means they spend most of their time using joysticks and keypads. On the other hand, the surveillance systems installed in bridges or around the cities does not require operators to be as active as in the previous example. In this case, the operators do not spend as much time monitoring the cameras, they only use them to make routine checks from time to time or in case an incident has occurred (event based monitoring). Therefore, the use of control devices is much less in these situations. In between these two applications there are many others where the use of the control devices are adapted to the desired function of the camera system.

Figure 2.5: Example of large surveillance network in *Majestar Casino*, Jeju island South Korea [10].

Finally, the joystick has not a very ergonomic design, it requires the user to do some movements that can generate pain, especially in the wrist, when they are repeated continuously. This is the case of the aforementioned example: the casinos.

## 2.2 Hand gestures

A hand gesture is a movement that is made with the hands to express emotion or information [11]. Therefore, in this thesis a hand gesture is considered as every movement performed by a hand or part of it.

When developing an interface using hand gestures it is important to know that the interface should be seen as an alternative to existing interfaces, such as a joystick or a mouse and a keyboard. Therefore, when developing gestures for an interface, the goal should not be to find the optimal gestures to control all future gesture-based devices. Instead, the goal is to find the optimal gestures for this specific device and thereby the objective is also to develop a more efficient interface than the existing one to the given application [12].

Consequently, the development of the hand gesture vocabulary is a matter of what the desired outcome of the hand gestures is.

### Daily use of hand gestures

By using hand gestures instead of talking, humans can understand each other. An example of this is the handshake or high five. When discussing an object, hand gestures can be used in the conversation to explain the size or velocity of the actual object. Furthermore, gestures can be used to indicate whether a biker intends to make a left or right turn in a street crossing.

From the examples described above it is clear that humans use hand gestures every day to communicate with each other. Moreover, due to the many and widely used application scenarios of hand gestures it is obvious that they are considered as intuitive to understand.

That gestures are a good way of expressing things is also confirmed from sign language. To deaf people or people with speech impediments sign language and gestures are a good complement or the main communication channel between themselves and other people, which makes the importance of the gestures even bigger.

### Application of hand gestures in new technologies

Today hand gestures are also used in various technological applications, some of them are simple and others very complex. Google announced in 2015 their *Project Soli* [13], which is a small radar chip to be integrated in smart devices in order to be able to detect slight human hand gestures and motions. Some of the gestures it responds to are those used for selecting, manipulating and navigating content.

BMW has integrated an in-air gesture control system in their cars. With hand gestures in front of the control panel some of the functions the driver can use are to turn up and down the volume, set the GPS to navigate home and answer a call. The functionality is described as smart and a great solution to the future void of buttons [14].

Another application of hand gestures is the *Leap Motion Controller* from Ultraleap [15]. The device is an optical hand tracking module that captures the movement of hands with unparallelled accuracy. The software provided by Ultraleap generates a virtual model of a hand based on the detected gestures and movements. In addition, the software models the joints and bones inside the hand which means that it can interpret any gesture and display it on the screen.

All application devices using hand gestures are not as complex as the previous described ones. A lot of every-day products such as automatic soap dispensers use simple gestures to activate and distribute the right amount of soap [16].

# 3

# Literature Study

In the following chapter the theory that was needed to complete this master thesis is explained in order to lay the foundation of the remaining report. The theory explains several areas that includes hand gesture vocabularies, different sensor types and communication protocols among other things.

## 3.1 Limitations on the use of hand gestures

Even though the use of hand gestures seems to be the most natural and intuitive way for Human Machine Interface (HMI), there are several limitations that must be taken into account in order to design a successful interface. This section is specially focused in touchless hand gestures used to give orders to the machine or device.

### Technical limitations

The two basic technical limitations are the sensors used to capture the gestures performed by humans and the algorithms implemented to recognize and interpret those gestures. The precision or reliability of the gesture recognition are parameters that depend to a great extent on these limiting factors. As these parameters are key for the customer to buy the product, the manufacturers mainly focus on designing devices capable of detecting hand gestures with very little margin or error, i.e. they use the state-of-the-art technology in order to overcome the mentioned technical limitations.

### Human limitations

If technical limitations are always taken into account in the design of devices that use gestures to interact with the user, human limitations are the great forgotten ones in these cases. There are three main human limitations.

The first one is that the hand is a very complex body part. This whole recognition process could be much easier if the goal was to detect any other part of the body. However, the hand is a limb and has so many joints, it can adopt many different gestures, especially when considering both static and moving gestures.

The second one is given because each person is unique and acts in a certain way. As individuals, our behavior depends on our culture, our education and the society around us, in addition to many other physical and psychological factors that can affect us. Therefore, a gesture can mean different things to two different people and the same thing can be expressed with different gestures depending on the person. In short, humans are very complex and there is no unique classification of hand gestures that is valid for everyone.

The last limitation is the physical effort, both fatigue and discomfort, specially if the gestures are performed very often and for long time periods. Here the sign language interpreters are the most experienced people. They are used to making many hand gestures (simple, complex and at high rates) throughout their work day. One study [17] claims that between 25% and 70% of the sign language interpreters reports persistent hand pain. But not only that, they warn that in the future hand gestures will replace the computer keyboard and mouse, in which case anyone would spend more hours making hand gestures than the sign language interpreters do nowadays. This means that if these HMI systems are not designed taking into account the physical effort, hand pain can become a serious problem. As a conclusion, they mention some useful recommendations in order to avoid hand, elbow and shoulder pain when doing hand gestures. The following list is a selection of the recommendations that can be applied to this thesis:

- When doing hand gestures, the hands should be located close to the height of the elbows, near to the lowest chest area and close to the torso. In this way, the shoulder muscles are in rest position.
- Gestures performed with just finger movements are more uncomfortable than gestures that require relatively small movements of the elbow and the shoulder instead.
- Gestures are comfortable when the wrist is fixed between its neutral position and 45 degrees of pronation. Gestures involving excessive flexion-extension, pronation-supination or ulnar-radial deviation of the wrist tend to generate hand pain after several repetitions. See Figures 3.1 and 3.2 for an explanation of the mentioned wrist positions.

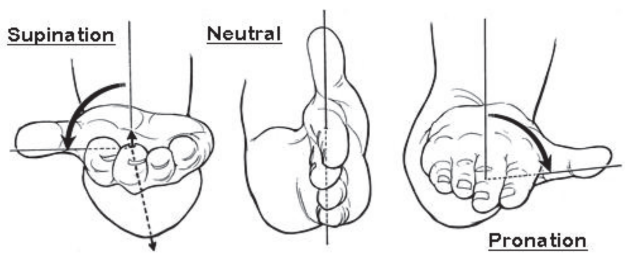



Figure 3.1: Wrist positions: supination, neutral and pronation [18].

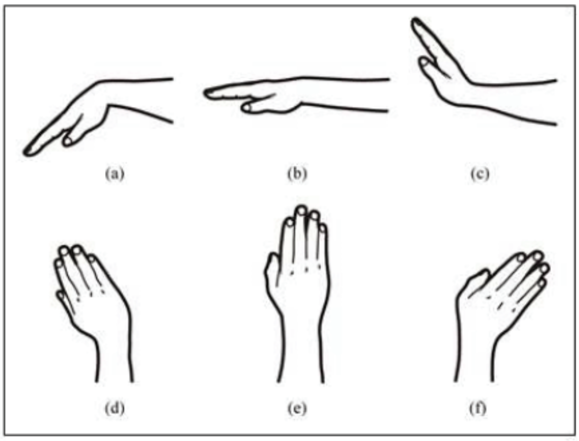



Figure 3.2: Other wrist positions: a) flexion; b) neutral; c) extension; d) radial deviation; e) neutral; f) ulnar deviation [19].

The aforementioned recommendations will just be achieved if the workstation set up is appropriate. The article [20] explains how the elements of daily use in offices affect the posture of workers and the damaging effect they have if their design is not ergonomic or if employees misuse it (not knowing how to adapt it to their needs). The measures explained are very important, however, it is the responsibility of the company to provide the necessary equipment (monitor, keyboard, mouse, desk, chair, etc.) and train their employees so that they can accomplish them and take care of their health.

The part of office ergonomics that can be most directly applied to the present project is the definition of the recommended work areas on the desktop, since the device to be designed is a desktop device which operators will use when they are in front of the computer monitor. The Figure 3.3 shows in black the primary working area. Here the most commonly used devices should be placed because it is the most comfortable and requires the least effort from the user. In the same image but now in grey, the secondary working area is represented. In this case, just the objects of occasional use should be located.

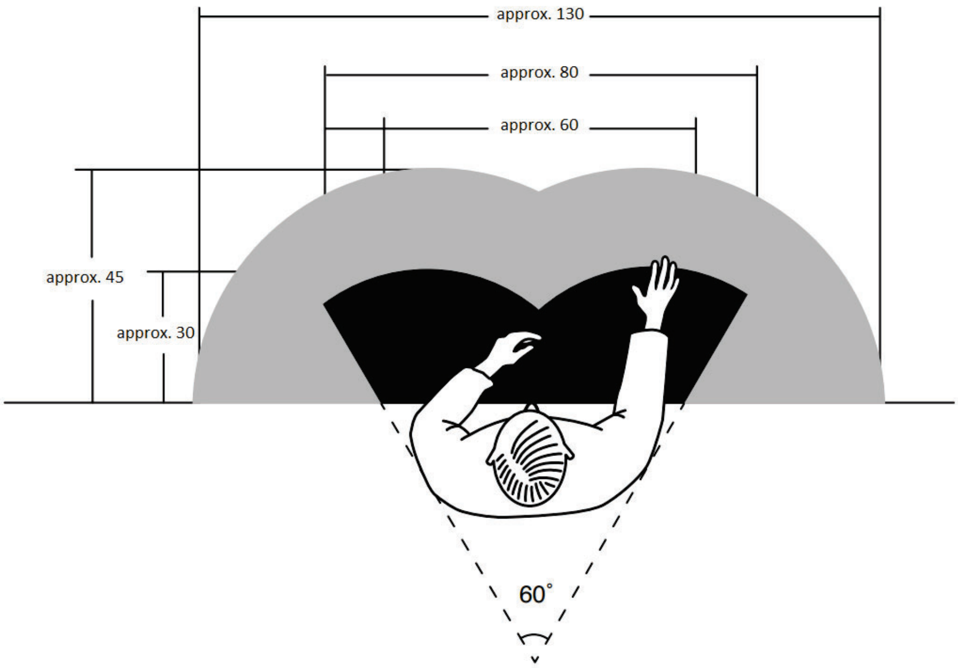



Figure 3.3: Desk working areas, distances in cm. The black area is the primary working area and the gray area is the secondary working area. Adapted from Arbetsmiljöverkets föreskrifter och allmänna råd om belastningsergonomi [21].

## 3.2 Methods for hand gesture vocabulary design

Every application where gestures are used to interact with the system must have a hand gesture vocabulary where it is indicated which gesture should be used to do a certain action. To design this type of vocabulary is not an easy task due to the fact that both the technical and the human limitations mentioned in the previous section must be taken into account. To do that, one can use several methodologies. *Madelene Svartberg*, *User Experience (UX) Designer PTZ Cameras Firmware* at Axis Communications, recommended the following three simple methods, any of them could be applied to evaluate the usability of the present project.

The first one is the *Thinking Aloud* method [22]. It is a very simple test that can be applied at any stage of the project, from the first drawings of the concept to fully working prototypes. The point is to provide the users with the idea, concept or device that they need to evaluate, give them specific tasks to perform and write down every comment made by the users as they try to accomplish the tasks assigned to them. To obtain valid results, it is important not to try to guide the user in any way, only talk to ask them to think out loud and not to hold back any thoughts, no matter how irrelevant they may seem. This is a cheap method, quite robust and flexible. However, the results obtained should not be blindly trusted since the user is subjected to an unnatural situation and the feedback received could be biased.

The second method is *I Like, I Wish, What If* [23]. The purpose here is to obtain just positive feedback from the user. Once the users know what is the topic to be evaluated, they have some time to individually write *I like...* statements i.e. things that they appreciate about the concept or prototype, *I wish...* statements i.e. new features or improvements that could be included, and *What if...* statements i.e. complete new concept or combination of ideas that could be further developed to cover the same function as the evaluated concept or device. Finally, the users have to justify each of the statements and the person in charge of the experiment has to analyze the feedback obtained in order to keep just the feasible input provided by the users.

The last one is the *Wizard-of-Oz approach* [24]. This methodology is meant to be applied at an early stage of the project, usually when there is no working prototype available. The key concept here is that the interaction between the user and the machine is not implemented yet so it will have to be simulated in real-time so that the user can see a proper response to their actions. For example in this case, if one wants the user to move a PTZ camera using hand gestures, every time the user gives a new command there must be someone hidden interpreting the command and moving a joystick in the corresponding direction so that the user can see on the screen that the camera has actually moved to the intended position.

Apart from the aforementioned methods, more advanced studies prove that it is possible to develop an analytical procedure to find an optimal hand gesture vocabulary. The article *Optimal Hand Gesture Vocabulary Design Using Psycho-Physiological and Technical Factors* [25] presents a mathematical model that solves a multi-objective decision problem: maximize the gesture recognition accuracy (solve the technical limitation) and maximize the comfort and intuitiveness (solve human limitation).

## 3.3 Hand detection approaches

There are several approaches to use when detecting hand gestures. The methods can be divided into two groups: wearable and non-wearable devices.

### Wearable

Wearable approaches of detecting hand gestures include all kinds of physical devices that can be worn. For instance a glove or an armband, these devices then contains the sensors needed to be able to detect what gestures are performed. However, there are some problems with this type of devices, since they have to be worn they can easily be perceived to be inconvenient or not handy to use. [26]

### Non-wearable

For the non-wearable devices there is no physical equipment that need to be used. Instead, the solutions can be vision based. These devices often contain a camera which in some way detects the hand gestures with help from for instance the temperature or color of the skin, or by background subtraction. By using background subtraction the background is learned at the initial stage and then subtracted from the frames in the video so that only the hands are observed. [26]

Apart from the vision based approaches using cameras to detect gestures there are other non-wearable approaches that instead use a mixture of sensors such as IR, laser and ultrasound. The devices using these approaches can use both touch-free as well as touchable technologies.

## 3.4 Sensors that can detect hand gestures

A motion sensor is an electronic device that is designed to detect and measure movement. Motion sensors are mainly used in home and business security systems, but are also widely used in phones, paper towel dispensers and virtual reality systems. Motion sensors are often seen as embedded systems containing three major components: the sensor unit, a microcontroller and some additional hardware or mechanical components. The three parts can vary in size and configuration as the sensors can be customized to perform highly specific functions [27].

When it comes to sensors to use for detecting hand gestures there are several options to choose from.

### Ultrasound

Ultrasonic sensors work by emitting sound waves at a frequency which is too high for humans to hear. By then waiting for the emitted sound to bounce back and afterwards calculating the time for this, it can interpret the distance the waves were transported until they hit an obstacle.

Some ultrasonic sensors use a separate transmitter and receiver and others use a combined version of these. The combined version can be manufactured in a smaller package which is convenient when the size of the components are important.

Ultrasonic sensors have no problem in sensing through plastic materials that are both transparent or colored. However, if an object is made out of a material that absorbs sound or is shaped so that the sound waves are reflected away from the receiver, the readings will be impossible to interpret. [28]

### Infrared (IR)

In the electromagnetic spectrum the IR portion is divided into three regions: near, mid and far IR region. The frequency of IR light is higher than microwave and lower than visible light.

An IR sensor emits and/or detects IR radiation to sense its surroundings. It transmits an IR signal which bounces from the surface of an object and the signal is then received at the IR receiver. Usually there are five components used in a typical IR detection system: an IR source, a transmission medium, an optical component, IR receivers and signal processing. IR lasers and LED's of specific wavelengths can be used as IR sources. The transmission medium usually consists of vacuum, the atmosphere or optical fibres. The optical components can be used to focus the IR radiation or to limit the spectral response. Some important specifications of IR receivers are photosensitivity, detectivity and noise equivalent power. [29]

There are many advantages with IR sensors. Firstly, IR motion sensors can detect motions both when it is dark and when direct sunlight is facing the sensor. Secondly, the ir sensors does not require any direct contact with the obstacle to be sensed, and thirdly, IR sensors can measure the distance to soft materials such as hands which may be hard to detect with other technologies. In addition, they provide good stability over time. Besides all the advantages for IR sensors there are some disadvantages as well, such as that IR waves of high frequencies can damage eyes if not careful and that it supports lower data rate transmission compared to wired transmission. [30]

### Radar

Radar sensors use radio waves to determine where in the front field the object is located. The Doppler effect is used to measure speed in radar sensors. When the fixed-frequency radio wave sent from the sender continuously strikes an object that is moving towards or away from the sender, the frequency of the reflected radio wave will be changed. This frequency shift is known as the Doppler effect. The presence and the speed of the moving object can be determined from the difference in frequency between the transmitted and the reflected radio waves. [31]

### Resistive

Sensors that varies their resistance as a result of the surrounding environment are known as resistive sensors. They are able to measure various physical quantities such as temperature, pressure, vibration and force among others. This quantities can be hard to measure but due to that the sensor converts the physical quantities into resistance and gives it as an output, it is easier to get a result. [32] [33]

A common example of a resistive sensor is the photo resistor. It measures the absence or presence of light, or the light intensity. When it is dark the output resistance from the sensor is very high, about $1M\Omega$ and when it is light the resistance decreases. Photo resistors have a sensitivity that varies with the wavelength of the light that is applied. [34]

Another example of a common resistive sensor is the strain gauge, which is a sensor that responds to the extraction or contraction of a material. A strain gauge consists of a long thin piece of metal which gets longer or shorter while the material that it is fastened to extracts or contracts, which changes the resistance of the metal. The voltage output of the sensor then corresponds to the change in resistance of the metal wire. [35]

### Capacitive

Capacitive sensors are widely used in different human interface applications such as track pads, touch screen monitors and proximity detectors. In traditional capacitive human-interface applications, the user initiates contact with the sensor electrodes, typically by finger touch [36].

Capacitive proximity sensors work based on the capacitor principle, since it uses the technology where two electrical plate conductors are separated by a nonconducting dielectric. A common dielectric used in capacitors is air, which is what also can be used in capacitive sensors. The sensor has one plate inside it which acts as one plate of the capacitor and the user acts as the other plate, while they are separated by air which is the dielectric in between the two plates. As the user comes closer to the sensor the capacitance increases in the sensor and as the user moves away the capacitance decreases. One of the disadvantages with this type of sensors is that they are sensitive to environmental conditions such as temperature, humidity or dust [37].

### Inductive

Inductive sensors measure the presence or absence of objects using electromagnetic fields, they detect conductive materials, ideally steel thicker than one millimeter. The inductive sensors contain an oscillator which creates symmetrical, oscillating magnetic field that radiates from a ferrite core and coil. When a ferrous target enters the magnetic field, current is induced in the metal's surface. This in turn changes the reluctance of the magnetic circuit, which reduces the oscillation amplitude. As more of the metallic material enters and comes closer to the magnetic field, the oscillation amplitude decreases, and when it is removed the amplitude increases again and the sensor returns to its previous output. [38]

### Gyroscope and accelerometer sensors

The gyroscope and the accelerometer sensors belong to the Micro-ElectroMechanical Systems (MEMS) technology. Both sensors complement each other, that is why they are used together in many applications, such as smartphones, industrial robots or automotive industry, among many others. The gyroscope sensor [39] measures the orientation and angular velocity of the object and the accelerometer sensor [40] measures the linear change of velocity of the object, i.e. its linear acceleration.

Although there are different types of gyroscope and accelerometer sensors, both have a similar working principle for this kind of applications (motion sensing). They have an internal structure that vibrates with the movement of the object and produces a small change in the electrical charge. This signal is then amplified and analyzed in order to identify the desired movement.

### Photodiode

The photodiode is a sensor that detects visible and IR light and uses its energy to generate an electric current or voltage. These sensors are semiconductors with PN-junctions and reverse polarization. Their lens concentrates the light that falls upon the sensor, its energy excites the electrons in the PN-junction and a current is generated. The more illuminance incident on the sensor, the more current or voltage will be obtained. As the current value obtained is very small (of the order of $\mu$A), this type of sensor requires a subsequent amplification stage [41].

## 3.5 I2C - Communication

Inter-Integrated Circuit (I2C) was invented in 1982 by Philips Semiconductor (now NXP Semiconductors). It uses master/slave communication which is a model of asymmetric communication where one device controls one or more other devices and serves as their communication hub, as can be seen in Figure 3.4. [42]

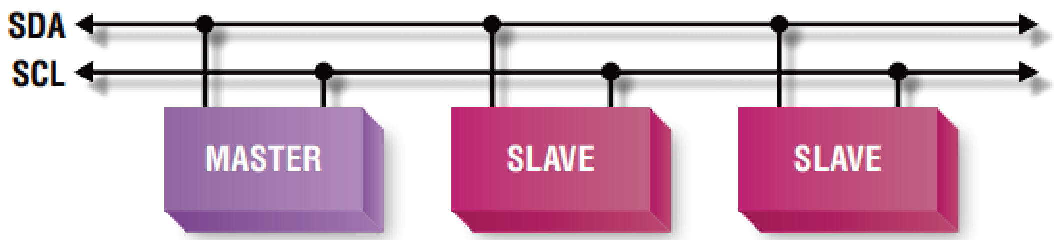



Figure 3.4: Master-Slave connection to the I2C channel [43].

I2C is a common used communication protocol for communications between on-board peripherals to transfer low to medium speed data. Communication with I2C is often used in various controllers, sensors and integrated circuits. However, the protocol does not have a central server to resolve the data conflicts, instead they are resolved by the Serial Data (SDA) and Serial Clock (SCL) signals. Moreover, the acknowledgment signal is sent by the receiver when every byte is transferred, which prevents the data loss of the SDA signal.

With I2C protocols it is possible to communicate with many devices in the same bus network due to the fact that each device is recognized by its unique address. The data is transmitted between the devices by the two wires SDA and SCL, where a start and stop signal is sent in order to communicate that the transfer has started and ended, this can be seen in Figure 3.5. The negative edge and the positive edge of SDA in the high level of SCL, as marked in Figure 3.5 represent the start and stop signal. The signal of SDA would not reverse in the high level of SCL when the other signals are transmitted. An Acknowledgment (ACK) bit or a Not Acknowledgment (NACK) bit must be transmitted after transmitting the data of one byte in the I2C protocol [44].

In I2C each transaction that is sent begins with a START and ends with a STOP. The transactions are in different formats but can contain a single message where a master writes or reads data from a slave. Combinations of these do also occur, that is when a master issues at least two reads or writes to one or more slaves. In a combined transaction, each read or write begins with a START flag and the unique address of the specific slave that it wants to communicate with. When the STOP flag is sent by the master and received by the slave, the slave knows that received actions from the message should take effect. [42]

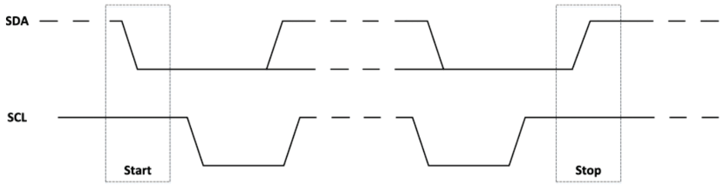



Figure 3.5: Start and stop signals for I2C protocol [44].

## 3.6 HTTP - Communication

HTTP stands for Hypertext Transfer Protocol and is a set of protocols designed to enable communication between clients and servers. The client could be the web browser and the server an application on a computer that hosts the web site. To request a response from a server there are mainly two methods, **GET**, which is used to request data from the server. And, **POST**, which is used to submit data to be processed by the server. To make HTTP requests in python, there are several libraries to use, but one of the most common one is *requests* [45]. Figure 3.6 illustrates a diagram which explains the basic concept of the GET and POST methods in HTTP. It can be seen that with the GET request, data is requested from the server and then sent back to the module. With the POST request data is instead uploaded to the server. [46]

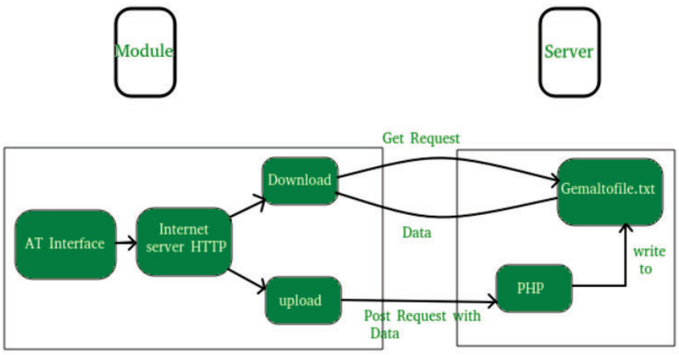



Figure 3.6: The basic concept for GET and POST commands using HTTP [46].

# 4

# Methodology

This chapter explains the methodology used to perform this master thesis and what has been done in order to achieve the results that can later be read.

First there is an explanation of the requirements that the final result had to meet. Following this there is the concept brainstorming, in what way it was done and what resulting concepts were found are discussed. Then there is a description of how the evaluation of the concepts were done and finally the three ideas with highest scores are presented.

Following this there is a similar section of the brainstorming process and evaluation but for the sensors, which resulted in two winning ideas.

This is continued by a final explanation of how the final idea was found with the winning concept and sensor.

After this, the development process of the first prototype is described together with the evaluation of it. Finally, there is the development process of the second prototype which is based on the evaluation that was made.

## 4.1 Requirements

First of all the requirements of the device were evaluated. To be able to develop the necessary functionalities it was desired to know if there were any limitations on the final concept that needed to be taken into account. The requirements were divided into eight different categories named *Appearance*, *Functionality*, *Ease of use*, *Content Master Thesis*, *Economy*, *Limitations*, *Electronics* and *Other*. The first four subgroups contained criteria concerning the physical concept and the last four the electronics inside it. Table 4.1 states the subgroups and their requirements regarding the physical concept and Table 4.2 states the subgroups and their requirements regarding the electronics inside the concept.

Table 4.1: Categories for requirements reagarding concepts.

| Appearance | Functionality | Ease of Use | Content Master Thesis |
|---|---|---|---|
| Size & Weight | Smooth Movement | Safety | Use of electronics |
| Aesthetics | Protected from Environment | Intuitive | Not software based |
| | | Portable | Complexity |
| | | USB | Innovative |
| | | Ergonomic | Touch-Free |
| | | | Use of Gestures |

Table 4.2: Categories for requirements regarding electronics.

| Economy | Limitations | Electronics | Other |
|---|---|---|---|
| Price | External conditions | Range | Software requirements |
| | Temperature | Accuracy | Ability to further develop |
| | Detection of any surface | Beam Angle | |

### Requirements regarding the physical concept

From discussions with the supervisor it was concluded that regarding the *Appearance*, the size and weight were highly prioritized since the final device needed to be able to move around from desk to desk as well as to be brought to different presentations to show. The size and weight of the device were also important since it needed to be able to fit on a desk together with other office equipment such as a monitor, keyboard and mouse. Therefore, a fairly small and light weight device was desired. Regarding the aesthetics no special requirements had to be met. For the *Functionality* section, the requirement of smooth movement was not a very important criterion, since the first requirement of the final concept was to only be able to perform a Pan, Tilt and Zoom movement with the camera. But in the future this would be of bigger importance in order to further develop the device. The device had a requirement that it needed to work inside normal office conditions, which meant indoors, around 20 degrees Celsius with a normal indoor humidity. However, it might need to be protected from dust or direct sunlight for the sensors to work properly, depending on what electronics were to be used. Concerning the subsection of requirements regarding *Ease of Use*, the safety was of course important, meaning that no user should be able to hurt itself when using the device. The implemented hand gestures should be intuitive to the user to use, which meant that the user should easily understand how to move the camera with the device. The device also had to be portable in order for it to be possible to move the device from desk to desk, but also to be brought to different presentations and workshops. Therefore, the requirement of being portable was highly prioritized. Another requirement in this subcategory was regarding power supply since the device was planned to be run from a USB connection to a computer. The last requirement from the *Ease of Use* was that the device needed to be ergonomically correct and therefore fit well in the primary or secondary working areas on a desk, as can be seen in Figure 3.3 and explained in Section 3.1. Finally, the requirements in the subsection *Content Master Thesis* were discussed and evaluated. Since this master thesis was made in collaboration with one of the electronics departments at Axis Communications it was a requirement from them that the final concept needed to use electronics and that the main idea behind the concept would not be mainly software based. Another requirement was that the complexity of the final concept needed to be on a good and proper level. There was no need to make the device more (or less) complex than was desired for the necessary functionalities to be implemented. The next requirement concerned that the device needed to be innovative in the sense that it should not look or work in the same way as any of the products that already existed in Axis Communications. Finally, two very important requirements were added. The first one was that the concept had to be touch-free, which meant that the user could not move or touch the actual device in order to move the camera. And the other one was concerning the use of hand gestures, which meant that some type of hand gestures actually had to be used.

### Requirements regarding electronics

Next up was the requirements regarding the electronics inside the actual concept. From the *Economy* subsection it was concluded that about the price, no limitations needed to be applied. Of course, it is always better to make a product as cost efficient as possible but that was not the main goal here. The *Limitations* category processed the requirements that the sensors could have. Independence from both external conditions and the temperature were taken into account, this meant whether the behavior of the sensor was affected by for instance dust, dirt or smoke, but also if the sensor needed any extra electronics to compensate for temperature changes inside the sensor. The criteria of being able to detect any surface, was added in order to be able to determine if the sensor could sense any material and shape. The *Electronics* subsection describes the electronic requirements. By the range, accuracy and beam angle criteria it was desired to find what sensor was the most optimal one to use when taken into account how far the sensor can detect an object, at what accuracy it will detect the distance to the object and around what angle it can be detected. The subsection named *Other* contains the last requirements of the electronics. The software requirement was added as a way to find technologies that would be able to implement with the chosen concept. Finally, the last requirement of ability to further development was added in order to find electronics that could be further developed with more functionalities than just the basic requirements of the master thesis. A more detailed description of the electronics assessment criteria can be found in Appendix A.

## 4.2 Concept phase

During the first part of the concept phase, a generation of ideas was made by brainstorming in order to come up with different ideas to be applied in a concept. Afterwards the concepts were evaluated according to the criteria described in Section 4.1.

### Concept generation

In the beginning of the concept phase time was spent to come up with different solutions that could be implemented so that a hand gesture recognition system would be possible.

***Sensor Box*** The first generated concept was called the *Sensor Box*. The idea was to create a box-shaped base with one of the six sides open for the hand to enter. Inside the box would then be several sensors mounted on the remaining five walls in order to measure the position of the hand in three different dimensions, as shown in Figure 4.1.

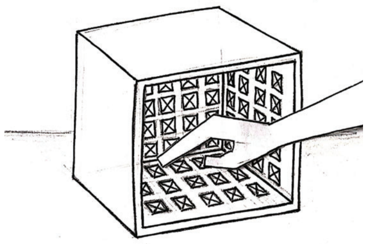


Figure 4.1: Sensor Box.

***Sensor Platform*** The idea behind the *Sensor Platform* was similar to the above described *Sensor Box*, but with the difference that this would only be a flat surface where all sensors would be attached, see Figure 4.2. By using the sensors to calculate the distance to the hand this would work as a matrix where one all the time could find out the exact position of the hand.

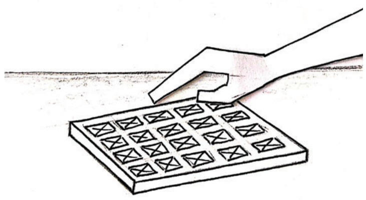


Figure 4.2: Sensor Platform.

***Sensor Stick*** The next concept that was brainstormed was the *Sensor Stick* in Figure 4.3, a concept similar to the already existing joystick but in this case it would use sensors to determine whether it is tilted or not and in what direction. Also, this concept would be wireless in contrast to the existing joysticks.

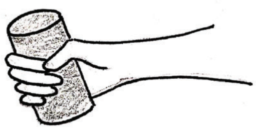


Figure 4.3: Sensor Stick.

***Sensor Glove*** Another concept that was generated by the brainstorming was a *Sensor Glove*. This concept was based on a glove which the operator would use, see Figure 4.4. Attaching sensors to the glove, the movement of the hand could be measured and then interpreted as gestures.

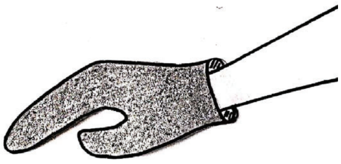


Figure 4.4: Sensor Glove.

***Sensor Field*** The *Sensor Field* would be a similar concept to the already mentioned *Sensor Platform* with the difference that this concept would use one single sensor to create a field instead of the matrix of sensors in the platform, as shown in Figure 4.5.

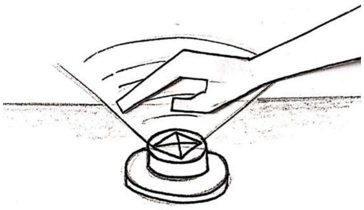


Figure 4.5: Sensor Field.

***Sensor Cube*** In contrast to the *Sensor Box*, this concept would attach the sensors to the outside of the box rather than to the inside, see Figure 4.6. By then moving the hand around the sides of the box, and thus activating different sensors, the camera would move based on this.

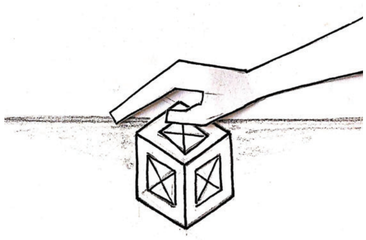


Figure 4.6: Sensor Cube.

***Sensor Dome*** The concept of the *Sensor Dome* is somewhat similar to a combination of the *Sensor Field* and *Sensor Cube*. The idea was to 3D-print a holder for mounting sensors so that it has the shape of a dome camera, see Figure 4.7. With this solution different sensors could easily be combined in order to perform more advanced gestures like a combination of pan and tilt at the same time. The dome shape would not only look nice together with other Axis Communication products, but would make it easier for the user not to activate any sensors by mistake since they are more spread out compared to the platform.

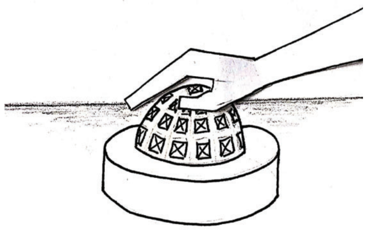


Figure 4.7: Sensor Dome.

***Camera*** The last concept that was generated through brainstorming was the one just called *Camera*. This concept would use a camera to record the hand gestures and by this interpret how the user wanted the PTZ camera to move, see Figure 4.8. This concept would require a lot more heavy software development to be able to perform some kind of computer vision together with machine learning in order to work.

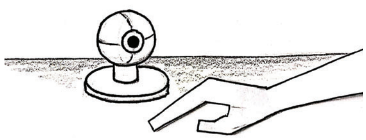


Figure 4.8: Camera concept.

## Concept evaluation

After all the concepts were generated they needed to be evaluated in order to find the most optimal one for the purpose. This was made by creating a scoring matrix where all the requirements mentioned in Table 4.1 were stated on one axis together with the concepts on the other axis of the matrix. From the discussions with the Axis Communications supervisor regarding what requirements were the most important it was possible to give a weighted value (1-5), where 5 meant that it was a very important requirement and 1 that it was not very highly prioritized, to each of the requirements. When all the requirements had received a grade corresponding to their importance for the final product, a similar scoring was done with the concepts. This meant that for each concept, all the requirements were tested and graded with respect to how well the specified concept could stand up to the specified requirement. This scoring was done similar to the one before where a 5 meant that the concept could meet the requirement very well and a 1 meant that it did not meet the requirement well. A summarized version of the results are presented in Table 4.3 and the entire version with the weights can be seen in Table B.1 in Appendix B. The concepts were compared out of their total scores in the grading and the three that achieved the highest scores were chosen to be continued with until the final decision.

The scoring gave that the two concepts with the highest scores were *Sensor Field* and *Sensor Dome*. Those had sufficiently higher scores than the other concepts and it was therefore decided to continue and investigate them further, together with the possible sensors.

Table 4.3: Concept grading matrix.

| Attribute | Sensor Box | Sensor Platform | Sensor Stick | Sensor Glove | Sensor Field | Sensor Cube | Sensor Dome | Camera |
|---|---|---|---|---|---|---|---|---|
| **Appearance** | 8 | 35 | 37 | 34 | 40 | 35 | 35 | 40 |
| **Functionality** | 29 | 23 | 31 | 27 | 23 | 14 | 23 | 35 |
| **Ease of Use** | 88 | 111 | 110 | 100 | 110 | 102 | 115 | 120 |
| **Content** | 140 | 140 | 88 | 115 | 140 | 135 | 140 | 85 |
| **Total score** | 265 | 309 | 266 | 276 | 313 | 286 | 313 | 280 |

## 4.3 Sensor evaluation

In this section a discussion is presented about the sensors that can be used in this thesis. The fact that two device concepts had already been chosen slightly simplified the process, since the sensors to be evaluated had to be consistent and applicable to the selected concepts. Therefore, certain sensors were directly discarded. The Appendix C presents a brief explanation about which sensors were discarded and why.

### Sensors to evaluate and evaluation criteria

The sensors that fit the selected concepts according to the requirements were the Infrared (IR), the radar, the ultrasound and the photodiode sensors. In this section all of them are evaluated in order to determine their potential when they are used in this thesis. To do so, a new scoring matrix was used. This matrix was constructed following the same methodology as explained in the previous section, but this time the vertical axis contains the assessment criteria for the electronics and the horizontal axis contains the four sensors to be evaluated.

The assessment criteria and their weights have been deduced from the electronic requirements discussed in previous paragraphs, which are also shown in Table 4.2.

In order to grade the sensors accurately, an intermediate step was necessary here. A commercial model was chosen from each of the four sensor types and a table was created with values and recommendations obtained from their data sheets and other sources, the result can be seen in Table B.2 in Appendix B. The sensors chosen as examples are presented below:

- *Photodiode sensor SFH 2701* [47].
- *A111 – Pulsed Coherent Radar (PCR)*[48].
- *Time-of-Flight (ToF) VL53L1X sensor* [49].
- *Ultrasonic Time-of-Flight CH-101 sensor* [50].

Finally, the last step was to translate all the data gathered in Table B.2 into a 1-5 scale (1 if the sensor does not meet the requirement and 5 if the sensor fully meets it). The result summary of the sensor scoring can be seen in Table 4.4. The entire result of this procedure is presented in Table B.3 in Appendix B. From the tables it can be seen that two of the sensors evaluated obtained a lower score than the other two. These sensors were the photodiode and the ultrasound sensors and, therefore, they were directly discarded. The main downsides of the photodiode were that it is dependent on external factors such as dust, dirt, ambient light and temperature. As well as its accuracy depends on the amplification applied to the current generated by the sensor proportional to the light and it does not generate enough data in order to recognize more complex hand gestures in the future, if the project is further developed. The ultrasound sensor, in turn, was rejected due to its price, its dependence on the temperature (need of extra electronics to compensate the temperature variations) and software implementation, i.e. the signal processing required to recognize the gestures.

Table 4.4: Sensor grading matrix.

| Attribute | Photodiode | Radar | IR | Ultrasound |
|---|---|---|---|---|
| **Economy** | 10 | 4 | 8 | 2 |
| **Limitations** | 21 | 28 | 33 | 35 |
| **Electronics** | 36 | 60 | 50 | 55 |
| **Other** | 20 | 26 | 28 | 22 |
| **Total score** | 87 | 118 | 119 | 114 |

The **IR and the radar sensors** were the two evaluated sensors that obtained a higher and very similar score, both of them were therefore taken into account in the thesis until there was enough information to dismiss one of them.

See Section 3.4 in the chapter 3. *Literature Study* for more information about the sensors mentioned throughout this section.

## 4.4 Final decision

Based on the scoring matrices, the two concepts and two sensors that best fit the needs of this project were preselected. In order to be able to make a final decision, it was decided to further investigate the operation and performance of the sensors, focusing especially on their behavior when detecting human hands.

Following that procedure, the radar sensor was finally discarded after attending a demonstration at a local radar company that designs and manufactures this type of sensors.

The radar is capable to precisely detect the position and speed of metal objects, but its performance is decreased when it comes to detecting the hand, as it is a surface that does not reflect signals effectively. The detection range from 0 cm to 15 cm was not recommended, the object is too close to the sensor that the signals sent are directly reflected and the receiver does not have enough capacity to interpret them. Even if the FoV of radars is quite wide, they can just detect gestures in one direction (1D). That would imply that several radar sensors should be used to cover the desired functionality in this project, a fact that increases the possibility of radars interfering with and disturbing each other. The last reasons was that the signal processing and the application of deep learning to recognize gestures was going to consume a lot of time in this project and there was a risk of not being able to show any working prototype at the presentation.

The fact that the radar was discarded as a sensor led to determine that the IR sensor was the most appropriate for this project. Another consequence was the rejection of the concept called *Sensor Field* due to the fact that, of the two sensors preselected, the radar was the only one with a sufficiently wide FoV to allow the design of a device with a single sensor.

The last decision was to choose the *Sensor Dome* as the final concept of this project. This concept allows the FoV of each sensor to be oriented so that the overall FoV of the device covers the desired 180º, while preventing individual sensor FoV from interfering with each other. This facilitates the recognition of hand gestures as it reduces the possibility of covering some sensors by mistake. Besides that, it is aesthetically more appealing and allows to follow the design of the Axis dome cameras, in which this concept was inspired. It is a more compact design, which takes up less space on the desktop and also differs from other track- and touch-pads used for other applications.

In summary, in this section the ***IR sensor*** and ***Sensor Dome*** concept have been chosen as the best options for the final design of the device to be developed in this thesis.

## 4.5 Development of the first prototype

### General schematic

Figure 4.9 shows the block diagram of how all parts were connected in the first prototype. Both the electronics and microcontroller were placed in and covered by the mechanics. Through one common I2C channel the sensors communicated with the microcontroller and sent their measurement data, which was processed by the microcontroller and sent to the PC through serial communication. Arduino MEGA 2560 was the microcontroller used in this thesis. The PC then again processed the data and translated it to gestures before it was sent as CGI commands through HTTP to the camera, which executed the desired action.

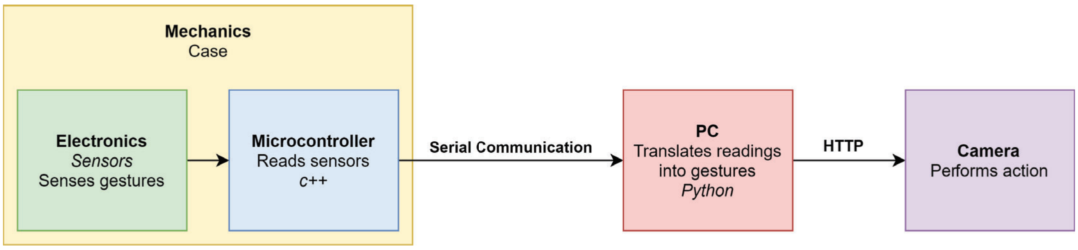



Figure 4.9: General schematic for the first prototype.

## Mechanical design

The mechanical design of the device was an important part in this thesis from several perspectives. The first idea to make the device in the shape of a dome camera was for the reason that it would look nice together with the other products in Axis portfolio. In addition, this would give the sensors a better FoV, but it would also be easier to perform the adequate gesture without interfering with other sensors while still keeping the design fairly compact.

In order to recognize the gestures for pan, tilt and zoom, as well as to have the possibility to further develop the device with other functionalities, nine sensors were considered a suitable amount. In this way it would be possible to combine the readings from the sensors and thereby recognize multiple gesture commands.

However, the round shape caused problems concerning the attachment of the sensors which is why it was decided to alter the hemispherical shape into a truncated octagonal pyramid, as can be seen in Figure 4.10. This way all sensors could easily be attached to a flat surface and therefore their FoV to be the best possible. When the drawings of the prototype were finished, it was 3D printed and afterwards its functionality got tested.

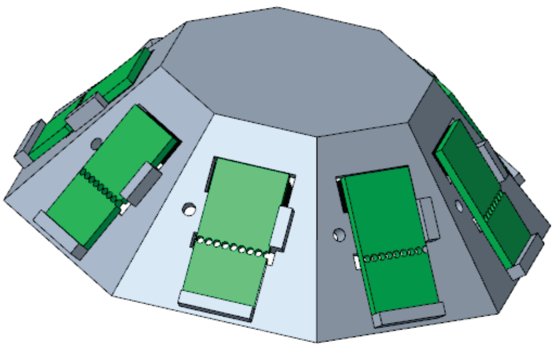
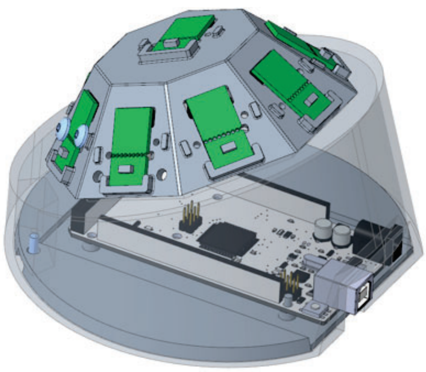


Figure 4.10: The pyramid to attach the sensors to.

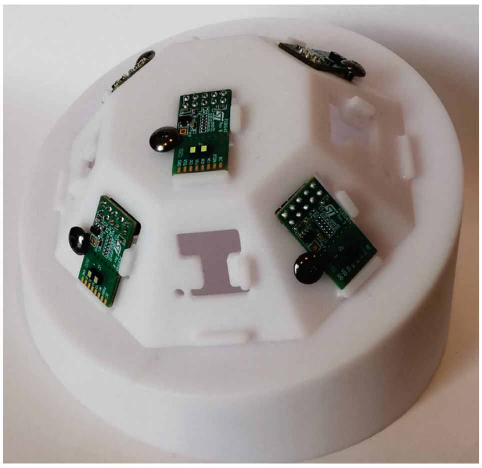
While testing the new pyramid it was discovered that the attachment of the sensors was not good enough, since they could easily fall out while moving the device around. Therefore, a new design was tried that gave a better result. It had two screws and another mounting to fasten the sensors.

When the design of the pyramid was finished, a case to hide the microcontroller was needed that also could work as a holder for the pyramid. The holder had to be big enough to hide all the additional electronics that was needed, but small enough to fit in a desk and still be comfortable to use. For ergonomic reasons as well as to ease the use of all the sensors, it was decided to design the holder in a tilted shape, this put the hand in a more natural position while using the device. Figure 4.11a illustrates both the design of the second pyramid as well as of the holder. The real 3D-printed model can be seen in Figure 4.11.

Another reason to this design was that it would make it possible to be used with both left- and right-hand without any modifications required. This was a result from that the device is designed completely without buttons or moving parts, as well as that the compact design makes it easy to move it from side to side on the desk.



(a) 3D model of the first prototype.



(b) Real printed model of the first prototype.

Figure 4.11: Design of the first prototype.

## Development of hardware

While building the prototype of the concept, testing was made continuously in order to find out if something did not work as intended.

In the very beginning a test board was built in order to connect and test the ordered sensors.

***Comparison of sensors*** When the final decision was made that the prototype would use IR sensors, the market was searched for suitable options. Two sensors were found, the first one, VL53L1X from *ST Microelectronics* [49], which is the same model as the one used in the sensor evaluation in Section 4.3, and the second one, APDS9960 from *SparkFun Electronics* [51]. Both of them were ordered to be able to compare them to each other in action.

The criteria used to compare the sensors were the following:

- Ability to detect up/down and right/left movements
- Ability to measure distance accurate
- How far and close can distance be measured accurately
- Ability to detect finger movements (e.g. not the entire hand but just a finger)
- Ability to react upon the velocity of the hand (e.g. give a faster response if the hand moves faster above the sensor)
- Existence of libraries to use
- Need of extra electronics such as level shifter in order to work

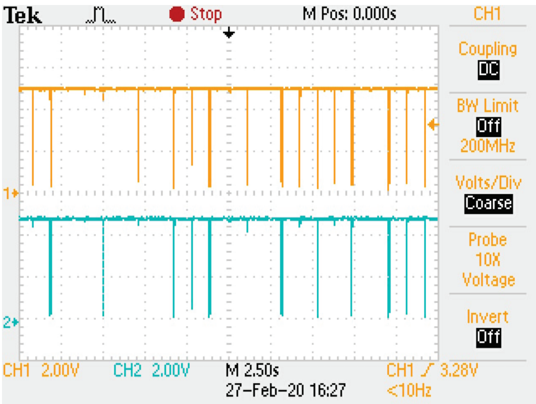
When connecting and testing both sensors using the criteria above it was concluded that both of them could detect up/down and left/right gestures. However, APDS9960 was able to do it with only one sensor but the VL53L1X needed a combination of sensors to detect this type of movements. Moreover, both sensors were also able to measure distance above them accurately, but VL53L1X could do it in a wider span, from about 2 cm above the sensor up to 400 cm. This was to be compared with APDS9960 that measured from about 6 cm above the sensor up to 24 cm. Furthermore, both sensors were able to detect both finger movements and movements of different velocity. When suitable libraries to use together with the sensors were investigated, only one fairly simple was found to be used together with APDS9960, while several and more complex were found that could be used with VL53L1X. Finally, in order to connect APDS9960 to the microcontroller, a level shifter was needed. This was due to that the microcontroller board uses an output voltage of 5 V while this sensor could not handle more than 3.3 V. In addition, no extra electronics were needed to connect the VL53L1X to the microcontroller.

From the results of comparing the two ordered sensors, the conclusion was drawn that in this case the VL53L1X would be a better option for the purpose. This was mainly decided due to the wider measurement span, but also due to the more complex libraries to use when implementing the software.

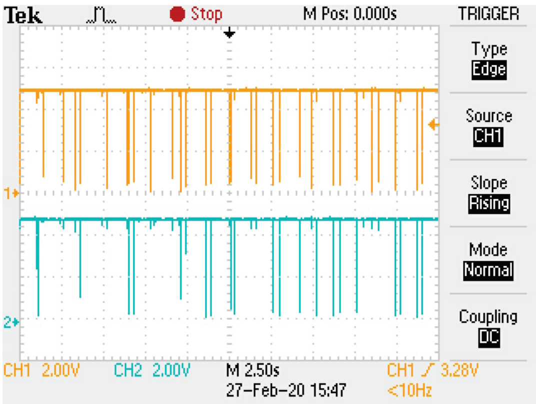
The basic characteristics of the sensor chosen are the following:

- Measurement range: up to 4 m
- Beam angle: 27°
- Resolution: 1 mm

***I2C Communication test*** To confirm that the VL53L1X from *ST Microelectronics* was a good sensor communication-wise for the project to continue with, some I2C tests had to be made. This was done by using an oscilloscope to observe the SDA and SCL signals in the I2C channel while the sensors were collecting measurement data. In Figure 4.12 the results can be seen. In Figure 4.12a the load of the I2C channel while one slave (sensor) was connected can be seen, and in Figure 4.12b the load of the I2C channel with three slaves connected can be seen. Channel 1 measured the SDA signal and channel 2 measured the SCL signal. From the oscilloscope plots the conclusion could be drawn that two more slaves slightly affected the I2C channel. This was due to the more crowded SDA signal, which could be seen from the more frequent spikes in channel 1 in Figure 4.12b compared to Figure 4.12a. However, since the channel did not get entirely full from the extra slaves, it was assumed that more sensors could be successfully connected to the I2C channel.




(a) I2C channel with one sensor connected.




(b) I2C channel with three sensors connected.

Figure 4.12: I2C signals with one and three sensors connected. Channel 1 is the SDA signal and channel 2 is the SCL signal.

***Pin configuration*** The schematics for the VL53L1X breakout board can be seen in Figure 4.13. Male pins were connected to **J1** in order to connect several of the sensors together. The pins marked with 2-6 were used to connect the sensors. Pin 2 and 4 are the SCL and SDA which were handling the I2C communication. Pin 5 is the power supply which was connected to 5 V and pin 6 was connected to ground. Pin 3 is the XSDN which is a digital input pin used to enable the sensor.

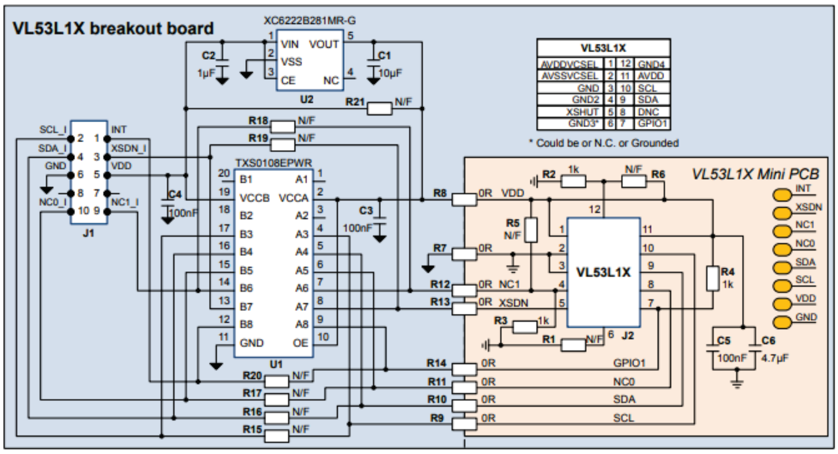



Figure 4.13: Schematics for VL53L1X breakout board [52].

***Connecting more sensors to the I2C channel*** More sensors were ordered and connected to a new testing board. To the new testing board the sensors were connected with wires from the sensor and then to the board. All of the sensors were connected to the same SDA, SCL, ground and power channel, while their digital inputs were handled from separate outputs in the microcontroller. As a result, several new problems occurred. One was that the connection from the wires were bad so the signals were often interrupted or lost.

When connecting the sensors one by one, the I2C communication stopped working after the seventh sensor was connected to the channel. A possible cause that was taken into account when trying to solve the I2C problems was the length of the wires used. Therefore, shortening the wires from the testing board was attempted but without success. A new way of connecting the sensors was also found which instead connected all the sensors in one wire after one another, rather than having them connected to the same board with several shorter wires. With the new wire the previous described problems with bad connections were solved but the I2C communication channel still had some unsolved problems.

Another cause that was taken into account was the possibility of the channel getting full when connecting more sensors. This could be the case if the network do not support more devices. The number of I2C devices that could be connected to a single I2C channel is limited by the maximum bus capacitance (400 pF) and the address space [43].

After several attempts to solve the problems with the communication that occurred when more than six sensors were connected at the same time to the channel, it was decided to continue anyway. Five sensors were considered sufficient to be able to recognize the pan, tilt and zoom movements, which was the reasoning behind the decision. The idea was to continue with this and start to implement the functioning in these sensors first to get a working prototype, and later go back and further investigate the problems with the I2C channel. Figure 4.14 illustrates the schematics that were used to connect all the sensors together and to the microcontroller. In Figure

4.15 the schematics for the relevant used parts of the microcontroller are shown. All the GPIO pins are digital pins.

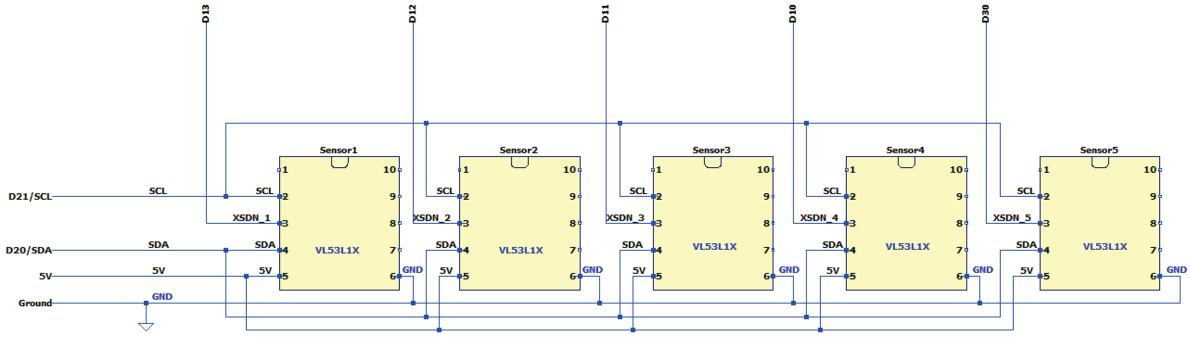



Figure 4.14: Schematics for how the sensors were connected to each other and to the microcontroller.

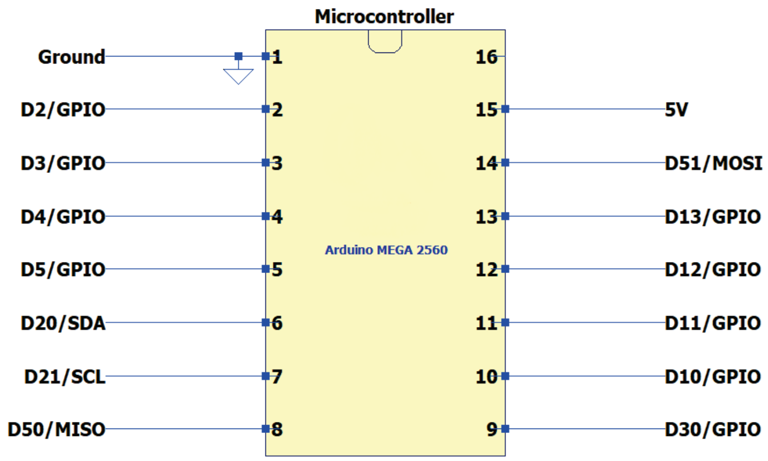



Figure 4.15: The schematic illustrates what pins were used on the Arduino MEGA 2560 microcontroller to connect the electronics.

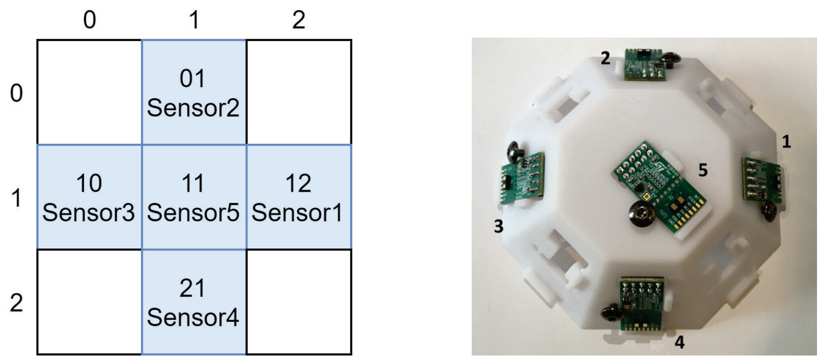
## Functions

To ease the implementation of the device and get it to respond to different hand gestures, the sensors were seen as being put into a 3x3-matrix, as can be seen in Figure 4.16a. The blue squares indicates the five sensors that were used to detect gestures. Figure 4.16b shows how the sensors were placed and numbered on the real device. The gestures that were decided to use were:

- **PAN:** Place the hand above sensor 1 to pan right and above sensor 3 to pan left.
- **TILT:** To tilt it up, place the hand above sensor 2, and to tilt down place the hand above sensor 4. This will send the desired tilt commands to the camera.

- **ZOOM:** Move the hand above the top sensor, sensor 5. When being closer than 3 cm from the sensor, the device will tell the camera to zoom in. When instead moving the hand more than 3 cm away from the sensor, the device will tell the camera to zoom out.

Apart from these gestures, it was decided that it would be good to have a function to enable and disable the device, which meant to put it into sleep mode. This was considered useful since by this it would be possible to keep the device on the desk and move around it, without having it to change the direction of the camera. It was decided that to enable the device, the hand should be placed closer than 6 cm from sensor 5 for 0,5 seconds. Instead, to disable the device both hands should be placed above both sensor 1 and sensor 3 at the same time. Since this gesture required two hands to be performed, it was considered safe to use as disabling gesture since it was desired to not risk to turn the device off accidentally.




(a) The matrix of sensors.

(b) Placement of sensors.

Figure 4.16: Placement of the sensors both in the model and reality.

By observing what sensors could be activated apart from the desired ones while performing the desired gestures, it was possible to put the combinations into functions for each gesture that was called in order to send the desired command to the camera. As an example, to tell the camera to tilt up, the sensor 2 has to be activated. However, when activating sensor 2, sensor 4 and sensor 5 are also easily activated and therefore this was also established as a possible combination to use for sending the tilt up command to the camera. A detailed description of what other combinations of the sensors were programmed into what function can be seen in Table D.1 in Appendix D.

### Development of software

Once the necessary hardware was in place, software had to be developed in order to get the device to work as intended. When it was decided how the sensor values were to be used as a matrix and how the readings from them should be interpreted, the necessary functionalities had to be developed so that the camera would get instructions with how to move.

***Reading of sensors*** As mentioned earlier, the sensors were divided over a matrix. By continuously reading all the sensor values in the matrix with very short breaks between each reading, it was possible to detect where the hand was positioned and when. The program implemented to read the sensors is written in C++ and executed in the microcontroller.

The sensors communicate with the microcontroller through one I2C channel. All the sensors are initialized at the beginning of the program, where they get assigned a unique address used to recognize them on the I2C bus. Next step is to wait until the measurements from the sensors are ready. Right after, the values obtained are processed and stored into a 3x3-matrix. Finally, that matrix is sent through the serial communication port to the PC and the program goes back to the waiting state. The aforementioned steps are represented in the state diagram in Figure 4.17.

A library called *vl53l1_api* has been used in order to read the sensors. It is an implementation of the ST Microelectronics VL53L1X API for the Arduino platform made by Pololu [53].

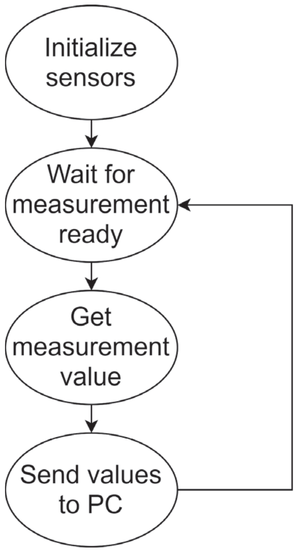



Figure 4.17: Simplified diagram of the algorithm used to read the measurement values from the sensors. Implemented using C++ programming language.

***Gesture recognition*** At this point in the project, it was necessary to implement a program capable of interpreting the data obtained by the sensors and translate it into gestures. In addition, the program needed to communicate with both the microcontroller and the camera. To do this, it was decided that the program will be executed in the PC and Python will be used for the implementation since previous projects at Axis Communications, that also sent commands to the camera, used this programming language.

The serial communication protocol was used to establish the communication microcontroller-PC and a library called *pySerial* [54] was imported. Through this the incoming data on the serial port could be read and used in the Python program.

In order to control the camera from the PC, the first idea was to use communication through SSH and with that send commands to tell the camera how to move. Therefore, *paramiko* [55], a library for SSH communication in Python, was imported. However, after some implementation it was realized that not all the necessary commands could be sent to the camera by this way of communication. For instance, there were problems in how to handle the zoom-function. Therefore, a new way to communicate to the camera had to be found. It was decided to rather try communication through HTTP, and due to this *requests* [45], a library that instead handles communication with HTTP was imported. Hence, the control to the camera was sent as CGI commands using HTTP GET requests. In this way, all the necessary commands such as pan, tilt and zoom could be sent. In addition, it was possible to set the speed of the camera while performing them, and also control when and where to stop the camera and prevent it from moving further.

The state diagram in Figure 4.18 represents a simplified version of the algorithm used for the gesture recognition and the communication, both microcontroller-PC and PC-camera. The first thing it does is to configure the necessary parameters for serial port and HTTP communication. Then, the program continuously reads and stores the values captured by the sensors into a matrix, until the device is enabled. After that, the program interprets the information received, i.e. it matches the activated sensors with one of the pre-defined combinations of sensors mentioned in the previous subsection. This way, the program recognizes the gesture made by the user and sends the corresponding command to the camera. The camera will move continuously until a different or no gesture is detected. When that happens, the program sends a stop command to the camera, reads new sensor values and repeats the same steps again. There is only one exception to that loop. When the gesture implemented to disable the device is detected, the program will go back to the continuous readings without performing any other action until it is enabled again.

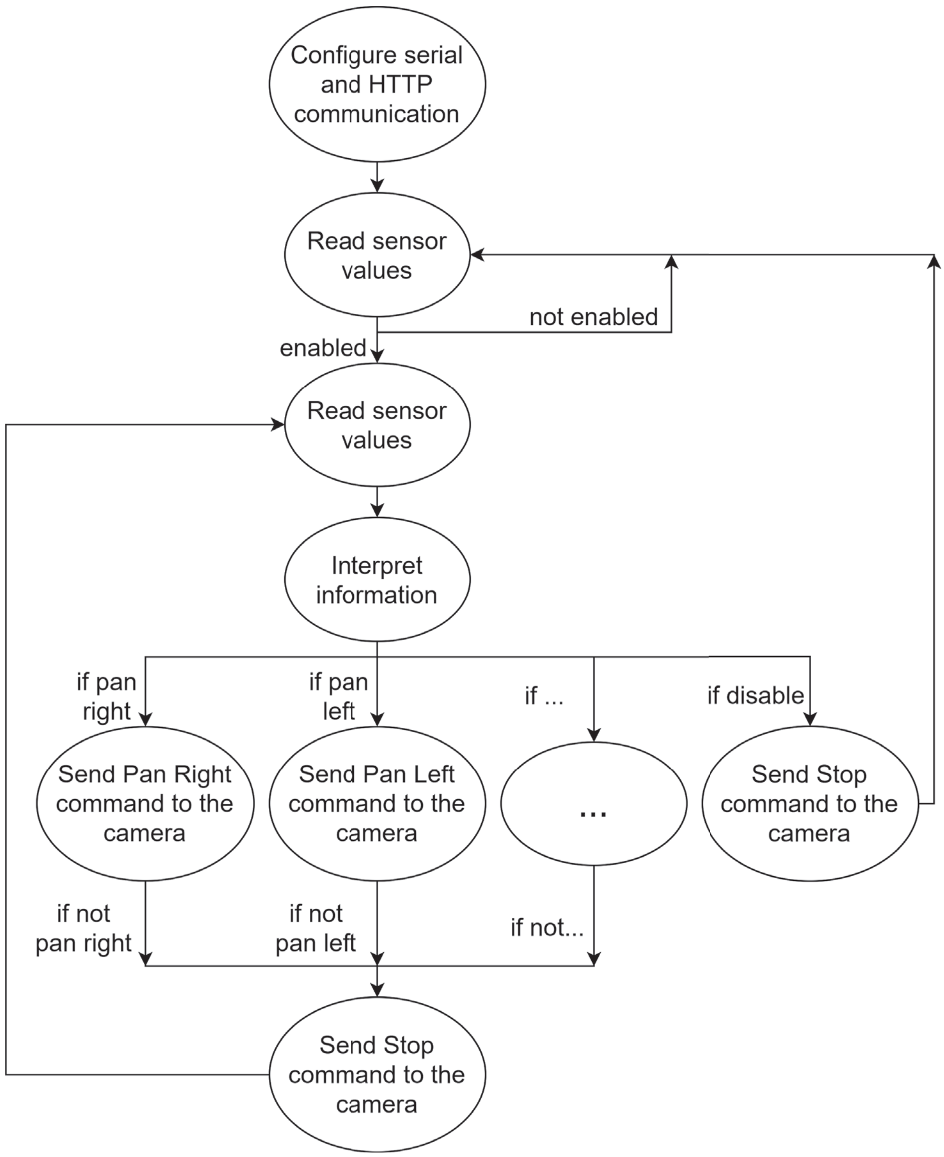



Figure 4.18: Simplified diagram of the algorithm used to recognize the hand gestures and send the commands to the PTZ camera. Implemented using Python programming language.

## 4.6 Gathering results

In order to get the results out of the first prototype a usability test was made that was designed to clarify at what extent the prototype met the requirements related to the usability of the device. The purpose of the test was to find the answer to the following questions.

- What are the difficulties users meet when interacting with the device?
- Which gestures are less intuitive for the users? Which gestures seem to be less comfortable or ergonomic?
- What gesture do the users do to pan, tilt and zoom?

- Do the users use one or two hands when interacting with the device? In case of one, is it their dominant hand or not?

- What posture do the users adopt when using the device? Do they place the device in the center in front of them or to either left or right of themselves? If they place it in one, does it vary if they are right-or left-handed?

- Are there possible solutions to the difficulties found? Did the users suggest good ideas to improve the current functionality? Did the users suggest other functionalities to add in future prototypes?

The test was designed so that the users were given tasks to perform, meanwhile their comments, thoughts and behavior related to the device were observed. This way it was also possible to detect what were the most common errors when using the device. Afterwards, the idea was to use the conclusions that were obtained from this test to improve the current functionality of the device.

During the test it was decided to use the *Thinking Aloud* method described in section 3.2. To get good and reliable results two groups of people with five persons in each were used for the test. All the persons that were selected for the tests were employees of Axis. The first group consisted of people that were used to control and work with PTZ cameras from before, and the participants in the other group were engineers without any major PTZ knowledge. However, none of the test persons in both groups had either seen, tested or received any information about the control device before the test started.

For the test a PTZ camera was mounted and connected to a computer and monitor. In addition, a keyboard, mouse and the prototype under test were connected and placed on a desk in order to get the feeling of a control room. To be able to afterwards analyze the thoughts and behavior of the test persons a second camera was mounted that recorded the entire test. Also, the view of the computer screen was recorded during the test.

The test was designed with three tasks, in the first one the test persons got to see the device and were asked to tell how they would assume that the device is used to do the pan, tilt and zoom movements. In the second task they were asked to tell how they thought the device could be enabled from a sleep mode, and then again put back into sleep. For the final task, the enabled device was given to the test persons and they got introduced to three different objects that could be seen on the screen in front of them. Afterwards, their task was to control the camera so that the object got approximately in the center of the screen, and then zoom in on the object, zoom out, wait for some seconds to do nothing and then continue to the next object and repeat the process.

When all the tasks were finished the test persons were asked some questions about their experience with the device. Those were:

1. What was your experience using the device?
2. What are the best things with this device?
3. What are the biggest frustrations you met from using the device?
4. What was your feeling of using the device, did you feel tired during the test?
5. Are you left or right handed?
6. Is there something you would like to add, or how can we improve the device?

When both test groups were finished with the tests, the recordings were gone through and analyzed. From the results of this it was possible to detect both what was the most common errors made by the users, but also what was the most common frustrations and suggested improvements for the next prototype. From the post test questions it was also found out what the users especially liked with the device. Most important of all, it was found whether the users found it intuitive or not to pan, tilt and zoom by using this technology. The findings were put together and can be seen in Section 5.

## 4.7 Development of second prototype

When the result gathering was finished and it was found what was the most desired changes to be made, the development of a second prototype started.

One of the first things that the test persons mentioned was that they wanted to get feedback from the device. This feedback could include whether or not the device is enabled, how much the camera is zoomed in or out and if the camera is panning up or down, among other things. The best way to include this type of feedback would be on the screen next to the camera view, where the operator could easily see it. Unfortunately, this type of changes to the GUI were out of the scope of this thesis and it was therefore decided to implement a small display that was attached to the device. Figure 4.19 shows what the system were to look like with the feedback display included.

The market was searched for suitable displays that could be implemented together with the already existing hardware and software, and it was decided to continue with a display from *Newhaven Display* [56]. It could show 32 characters in total by the format of 16x2, meaning two rows with 16 characters in each.

To get the display to show feedback to the user the main programs both in the microcontroller and PC had to be modified. Now the program running in the PC also has to specify whether it wants to read the sensor values or print some information on the display. The Figure 4.20 shows the modified state diagram of the program executed in the microcontroller. Here one can see how the schematic has been adapted in order to determine which of the actions should be performed after interpreting

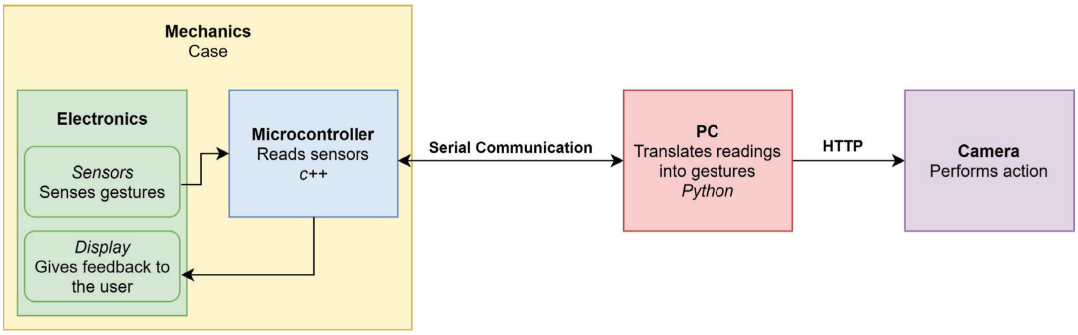



Figure 4.19: The general schematic of the second prototype.

the information received: Send values to PC or Print text in LCD display. It was also necessary to do some modifications in the Python program in order to get it to send the correct information at the correct time about, for instance, at what zoom value the camera was.

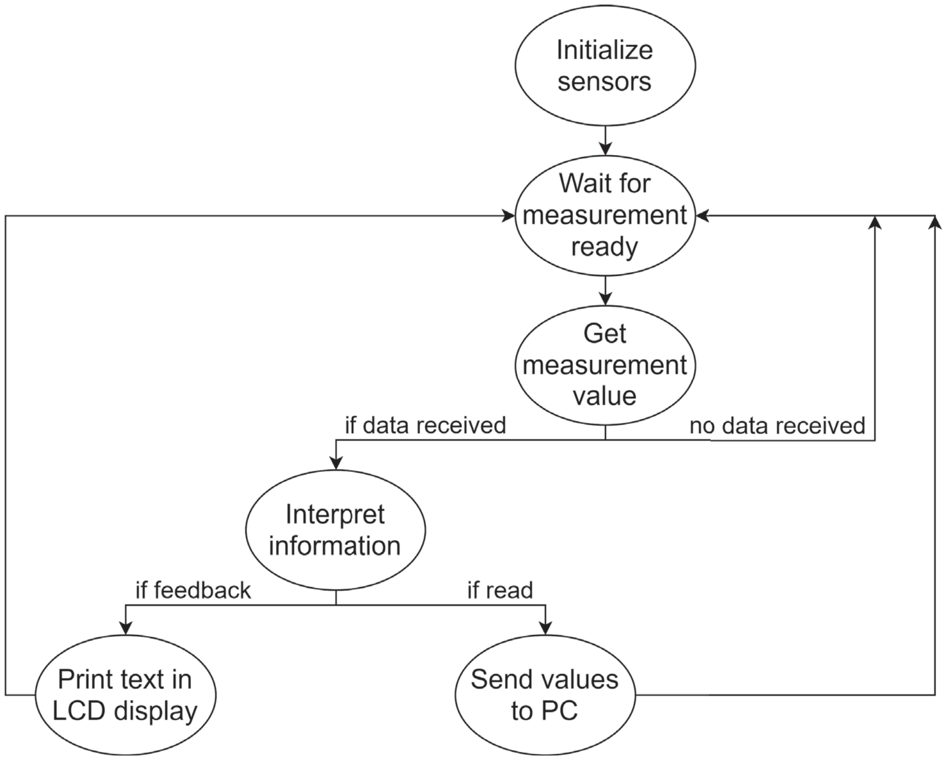



Figure 4.20: Simplified diagram of the algorithm used to read the measurement values from the sensors and print the information on the display. C++ programming language.

In addition to the software changes, the mechanics and electronics also had to be modified in order to get the display to fit nicely in the device. The case got re-designed by extending the device from its rear side and adding a hole to fit in the display above the upper sensor, see Figure 4.22, and the microcontroller was com-plemented with a self-designed shield that had the additional necessary electronic components, the design of this shield can be seen in Figure 4.21 .

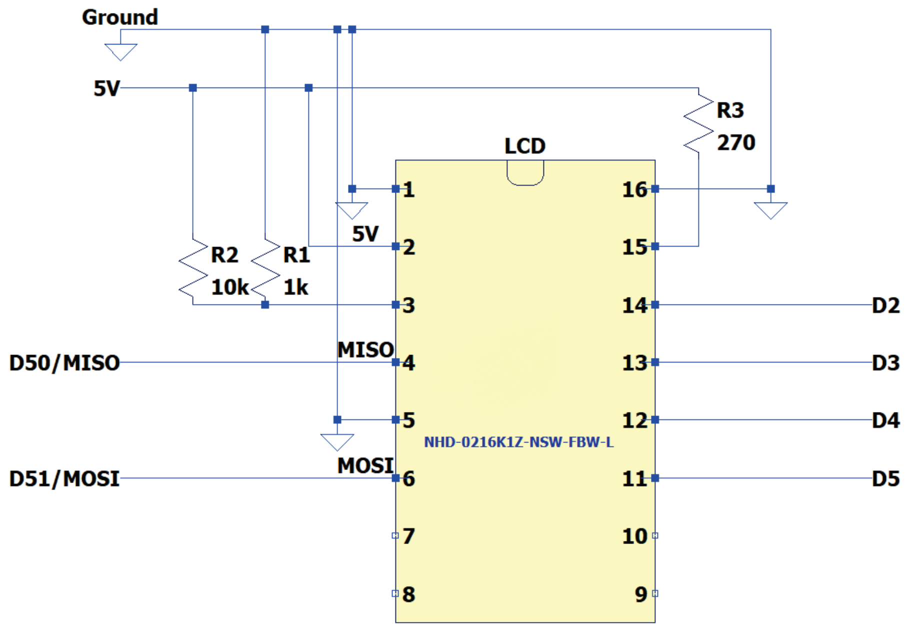
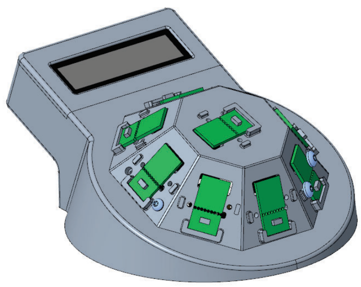
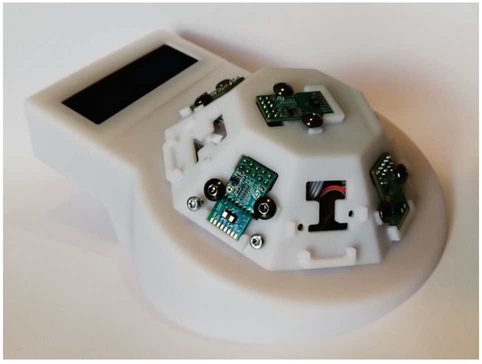



Figure 4.21: The schematics for the connections from the LCD display to the microcontroller.

With the questions after the test, it was found out that many of the users felt tired in the arm, shoulder or wrist after using the device. This was probably due to a combination of reasons including that they used bigger gestures than what was necessary, but also that the device was a bit too big in order for them to easily use it by having the hand and arm resting on the desk. In order to fix this the case was redesigned in a lower version which put the device closer to the desk. This made it easier for the users to rest their wrist on the desk while using the device. The new case can be seen in Figure 4.22.



(a) 3D model of the second prototype.



(b) Real printed model of the second prototype.

Figure 4.22: Design of the second prototype.

From the usability tests it was also suggested by some users to add speed to the pan and tilt movements. In this way, it would be easier to follow a moving object or person with the camera. This was implemented in the software and was working like when the hand activated any of the pan or tilt commands, if moving away from the device the speed got increased and if moving closer to the device the speed decreased.

Other improvements found from the results of the usability test were to lower the distance for zooming out in order to make the required gestures smaller, as well as lowering the distance for the tilt-up sensor. This was to prevent users from accidentally tilting the camera up when they are actually aiming to zoom in.

# 5

# Results

This chapter presents the results from the thesis. Initially there is the results from the usability tests that were made to develop the second prototype of the device. It includes an extensive presentation of the results from the test, these are later reflected upon in the next chapter.

Followed by this there is the results of the second prototype, which includes data about the sensor ranges, frequencies as well as latency in the device. There are also results regarding each gesture and its corresponding sensor signals.

Finally is an economic estimation made of how much the device would cost to produce.

## 5.1 Usability test results

In Section 4.6 it was explained which was the purpose of the usability test and how those tests were performed. Once the tests were finished, all the notes and recordings were looked over and the data was gathered in table format. In the following paragraphs, the results will be presented and commented.

### Task 1

In Task 1 the tests persons explained how they will use the device to pan, tilt and zoom, without having any prior knowledge about how the device actually works. Based on their comments during the task, it was clear to determine that there are two different points of view regarding the use of the device to pan and tilt: the Swipe Mode and the Rotating ball Mode. In the first one, the Swipe Mode, the direction of the hand movement matches with the direction in which the camera should move, i.e. if the hand moves from the left side to the right side of the device, the user expects the camera view to move to the right (pan right). The same applies to the other pan and tilt movements: pan left, tilt up and tilt down. In the second point of view, the Rotating ball Mode, the test persons placed the hand on top of the device and imagined that they were holding an imaginary ball from the sides. Rotating the imaginary ball counterclockwise they expected the camera to move to the left and vice versa, rotating it clockwise the camera should move to the right. Regarding the tilt movement, they imagine that they were taking the ball from the top and rotating it towards them (hand closer to their body) when they wanted to tilt up and towards the computer screen (hand farther from their body) when they wanted to tilt down.

Regarding the zoom function, there were different opinions that could not be classified that easily. However, the two most common points of view were the Touchscreen Mode and the Close-Far Mode. In the first one, the Touchscreen Mode, the users wanted to zoom in/out using the same pinch gestures that they use in their smartphone: pinch the fingers outward in order to zoom in and pinch the fingers together in order to zoom out. In the second mode, the Close-Far Mode, the users moved their hand in a vertical direction, approaching the device (close) to zoom in and moving away from the device (far) to zoom out.

The results of this task showed that 7 out of 10 test persons performed Task 1 using the Swipe Mode, whereas just 3 out of 10 used the Rotating ball Mode instead. At the same time, 5 out of 10 users chose the Close-Far Mode to zoom in/out and 3 out of 10 used the Touchscreen Mode. All the results are available in table format in the appendix (see Table E.1 in Appendix E).

## Task 2

In Task 2 the users had to explain how they would enable the device whenever they want to control the camera and disable it when they do not want to use it. In this task the users contributed with different opinions so only the most common and relevant contributions will be named here (see Table E.2 in Appendix E to see all the results).

Regarding the enable function, 2 out of 10 participants wanted the device to be enabled when it detected any close presence around it, another 2 testers thought it was better to enable it by placing the hand on top, whereas 3 out of 10 preferred to enable the device shaking the hand above the top sensor several times. Focusing now in the disable function, 3 out of 10 test persons wanted the device to be disabled when it does not detect any presence for a specific amount of time, 2 out of 10 would disable the device placing the hand on top of it and 2 out of 10 preferred to shake the hand above the top sensor several times.

Besides that, this task also served to obtain another very interesting input. 2 out of 5 users from Group 1 (with experience in controlling PTZ cameras) stated that in a control room the devices must be active at all times since no time can be wasted activating them when the operators have to pay immediate attention to some event happening, or to start following individuals or vehicles. Meaning that there should not be any enable or disable functions in the device since it should be active at all times.

### Task 3

In Task 3, the participants had to use the device to actually control the camera, find three different objects and zoom in/out on them. This test, unlike the previous ones, provided two different types of data: time spent in solving the task and amount of errors performed during the task.

Regarding time data, it was considered that the most interesting was to observe the time that the participants spent from the beginning of the task until they find the first object (Time 0-1) and also the time interval needed to find the second object, starting to count once the first object has been located (Time 1-2). Thus, it will be possible to determine the learning curve required by the device, i.e. how long it takes for the users to discover the suitable gestures to control the camera and, in addition, if users are able to assimilate the knowledge they had just acquired in order to continue with the task. The complete time table is attached in Table E.3 in Appendix E.

The graph in Figure 5.1 presents the mentioned time intervals for each of the participants and its average values, differentiating the behavior of both groups. The Group 1 (G1, with PTZ experience) contains the test persons from 1 to 5 and it is represented with curves in blue and gray. The Group 2 (G2, no PTZ experience) includes the participants from 6 to 10 and its two corresponding curves are orange and yellow. The first fact that can be observed in the graph is that in both groups the time spent on finding the first object (Time 0-1) is always greater than the time needed to find the second one (Time 1-2). Another thing that can be highlighted from the graph is that, in general, Group 1 with an average time of 00:04:05 needed more time to find the first object than Group 2, whose average time is 00:03:00. Finally, the plot shows that Time 1-2 is quite similar in both groups despite the differences observed in Time 0-1.

Regarding the second type of data gathered, all errors detected were listed and the number of errors made for each participant was quantified by reviewing the videos recorded during the tests. Table E.4 in Appendix E shows the complete list of errors.

The bar chart in Figure 5.2 shows the most common errors detected in both groups, Group 1 in blue and Group 2 in orange. There, one can see that there are three sets of errors. The first one includes all the errors detected when the participant wanted to do a pan or tilt movement. It collects all the cases where the participants wanted to pan or tilt the camera but their hand was too far away from the device so that it was not detected (*Pan far from device* and *Tilt far from device*). Moreover, it counts how many times the users entered the zoom mode when they just wanted to tilt down, *Zoom in/out (Desired tilt down)*. The second set is reserved for errors observed when using the zoom function: *Zoom out far from device* counts how many times the participants left the zoom mode because of having the hand too far from the device, *Tilt up (desired zoom in/out)* represents every time that the users were

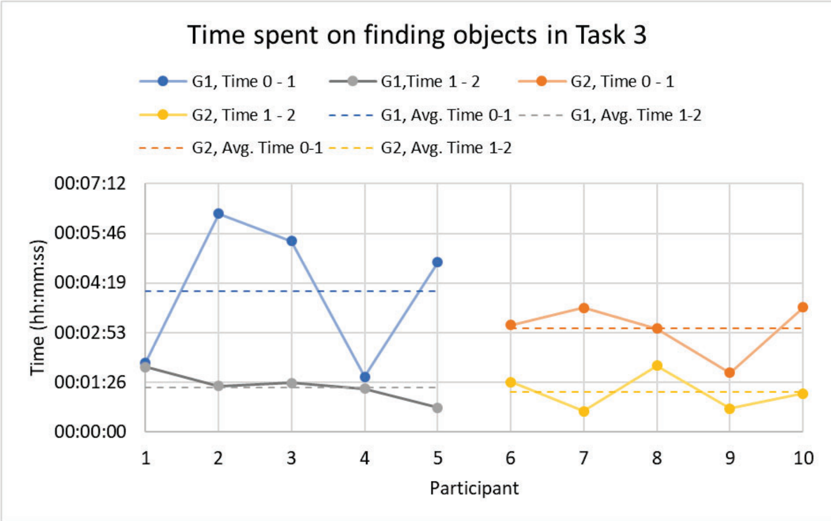



Figure 5.1: This plot represents the time that each of the participants spent in finding the first (Time 0-1) and the second object (Time 1-2). Group 1 corresponds to the blue and gray curves and Group 2 to the orange and yellow ones.

tilting up when they just wanted to enter the zoom mode and *Not entering zoom mode* reflects the failure of participants not entering the zoom mode when they wanted to, usually because they made too fast movements. Last error set represents the amount of times that the users entered the zoom mode when they just wanted to enable the device: *Zoom mode (desired enable device)*.

The most noticeable thing about the graph is the large number of *Tilt up (desired zoom in/out)* errors detected: 28 errors, 14 in each group. Besides that, one can also see that the errors when enabling the device (*Zoom mode (desired enable device)*) are practically negligible compared to the rest.

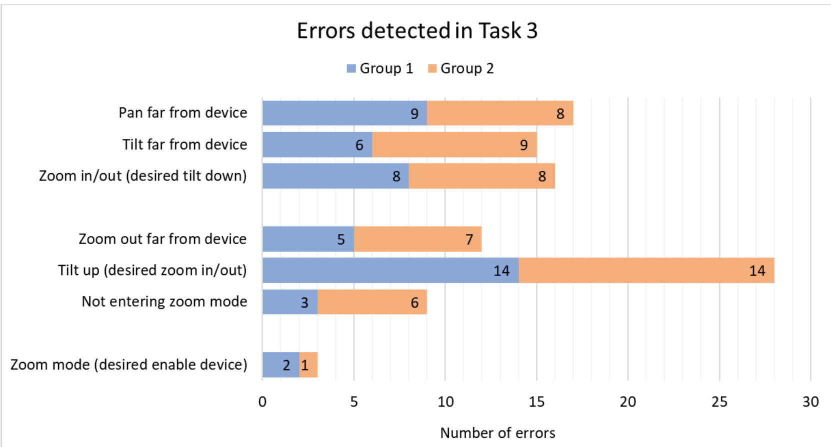



Figure 5.2: Classification of the errors detected in the usability test Task 3. Group 1 represented in blue and Group 2 in orange.

Finally, the bar chart in Figure 5.3 summarizes the errors made in each group as well as the total number of errors. Here it is possible to see that both groups have very similar amount of errors: 52 in Group 1 and 55 in Group 2.

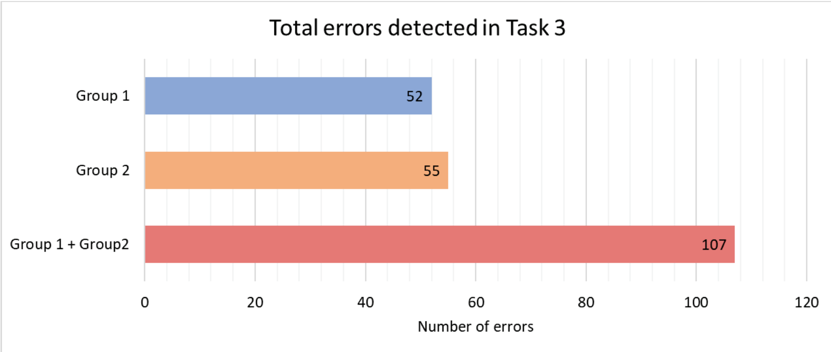



Figure 5.3: Summary of errors detected.

## Final questionnaire

At the end of the tasks, the participants experiences were evaluated through a series of questions. In this section, the results obtained are shown graphically, all the answers can be seen in Table E.5 and Table E.6 in the Appendix E.

The graph in Figure 5.4 shows the main feedback received about the overall experience of the participants. Although one of the participants in Group 1 stated that the device does not respond well to hand gestures and is not suitable for use in surveillance (bar *Bad* in the bar chart), the rest of the test persons gave quite positive comments in general. The topic named *Easy with experience* in the plot shows that 7 out of 10 users said that the device is much easier to use when they have already gained some experience, either by finding out for themselves the most appropriate gestures or by observing a small demonstration of its operation. At the same time, it is possible to see how 9 out of 10 individuals (see bar labelled *Good*) mentioned that, although the current prototype could not be treated as a final product, the device has potential for further improvement and adaptation to the real conditions in control rooms.

The bar chart in Figure 5.5 represents the three biggest frustrations that participants felt throughout the test. In the *Latency* bar it can be seen that 6 out of 10 participants complained about the existing delay between the execution of the hand gesture and the response of the camera. Here the responses of Group 1 should be highlighted, as 4 out of 5 individuals agreed on their opinion about the delay. The next frustration, *Pin-Point* bar, expresses the dissatisfaction of 4 out of 10 participants when trying to locate an object in the center of the screen, just before zooming in. None of them succeeded at first, as they approached the object they all went over the exact position and were forced to go back and make small movements around the object until they achieved their goal. The last one, the *Zoom* bar, represents the frustration felt by some of the participants when they tried to zoom in or out, either

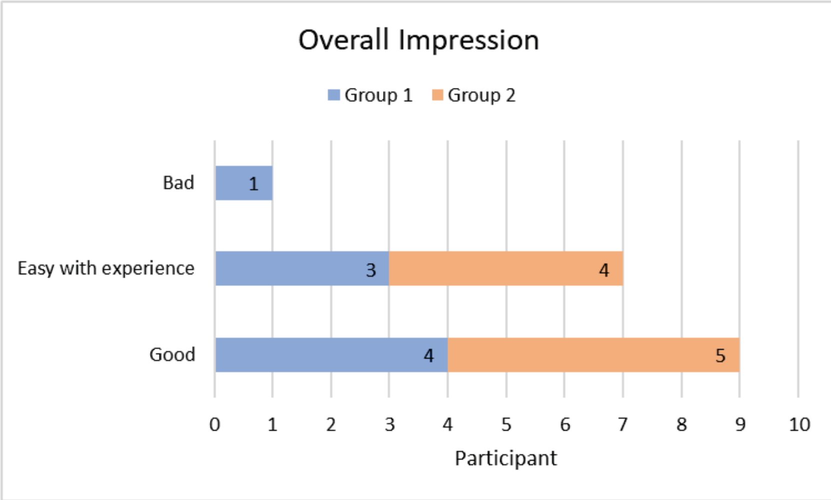



Figure 5.4: Overall impression of the participants, most common comments.

because they could not enter the zoom mode or because the camera started panning and tilting instead of zooming.

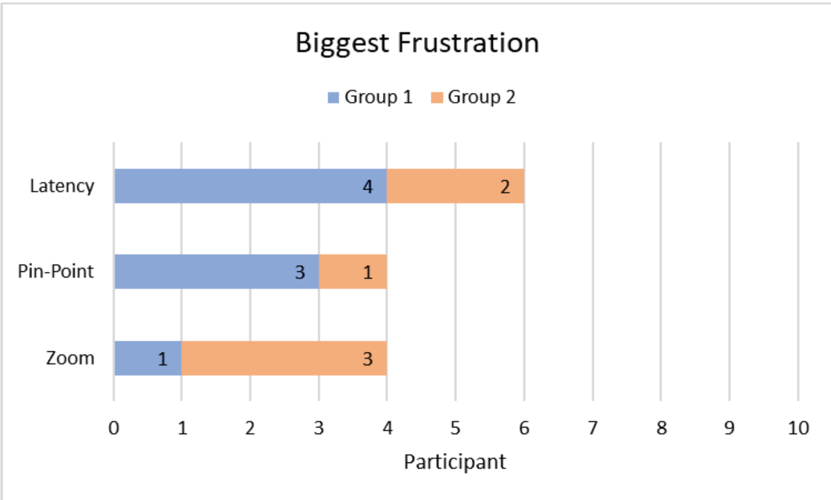



Figure 5.5: Main frustrations that participants have experienced during the tests.

The plot in Figure 5.6 presents what participants consider to be the best of the device. In first place there is the *Sanitary* bar, which has been motivated by the current situation caused by the expansion of the worldwide pandemic of the COVID-19 virus. Here 3 out of 10 participants appreciated the fact that the device is touch-free, as this helps to prevent the spread of diseases. The next bar called *Ambidextrous* shows that 3 out of 10 users, all of them participants of Group 1, pointed out that the device can be used both by left-handed and right-handed operators without modifying its configuration. Finally, 9 out of 10 participants (see *Cool* bar) commented that it is cool to control the camera with hand movements and that it feels very futuristic.

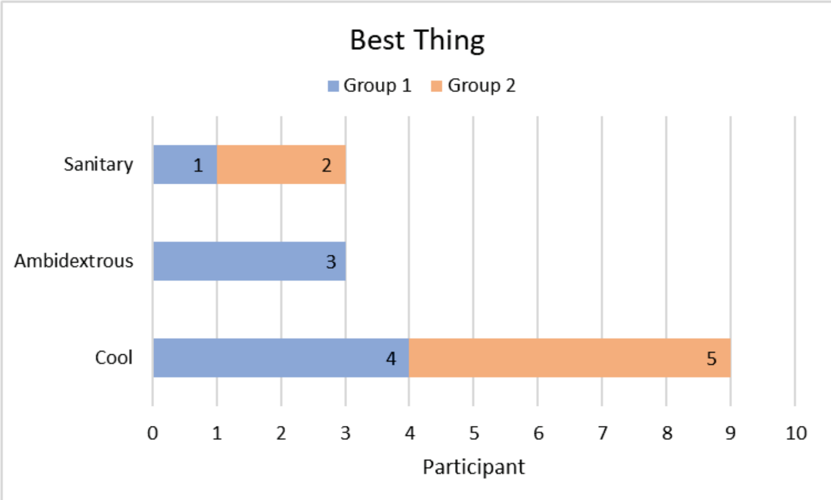



Figure 5.6: Best features of the device.

The last graph, Figure 5.7, presents the most commented suggestions and wishes of the participants that, in their opinion, will help to improve the functioning of the device. 3 out of 10 users said that a smaller device would be more ergonomic and comfortable to use. In *Combined movements* bar some participants from the experienced group stated that the device should be able to perform pan and tilt, or pan, tilt and zoom simultaneously. Besides that, 5 out of 10 individuals expressed the need of modifying the speed of the movements when controlling the camera. Finally, 6 out of 10 thought it would be really helpful to give some feedback to the user, specially telling whether the device is enabled or not and also whether the zoom mode is active or not. The last two bars mentioned are the ones with the higher score in comparison to the others.

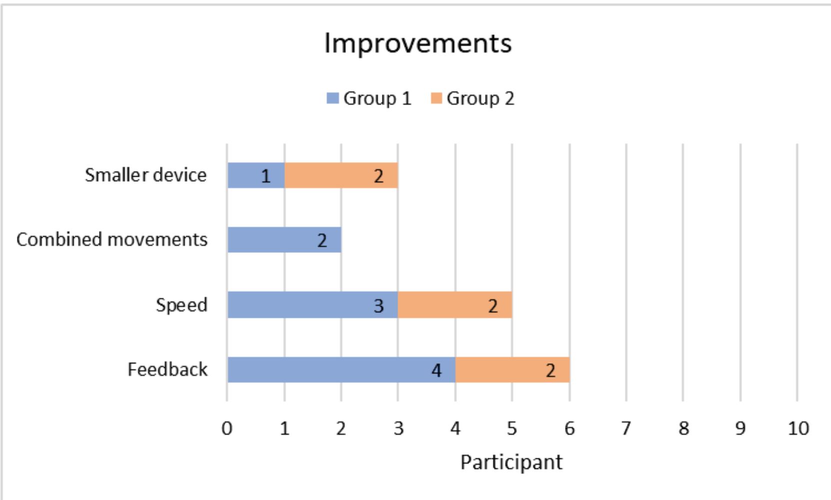



Figure 5.7: Suggestions to improve the current functionality and implement new features.

Apart from all the results shown above in graphic form, during the tests it was possible to observe other interesting behaviors.

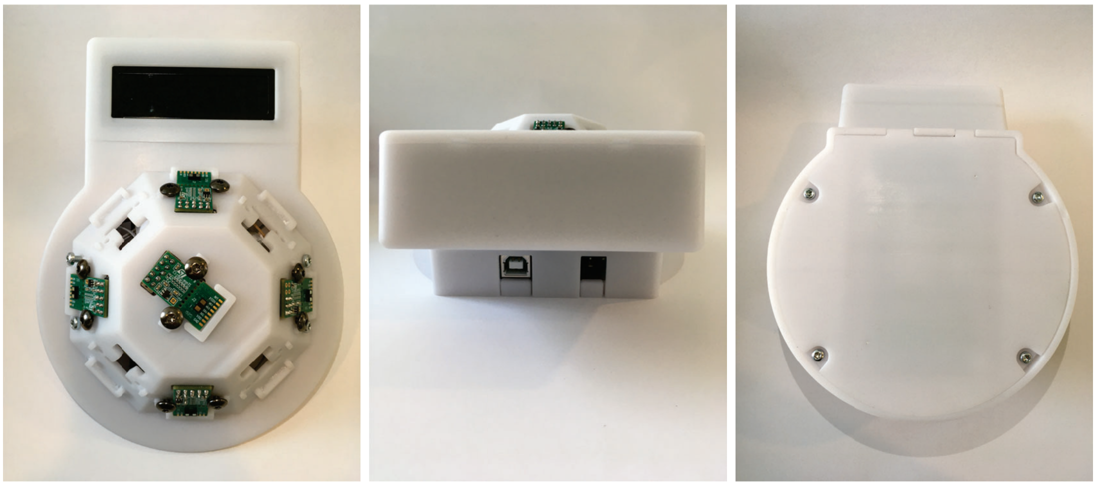
It is remarkable that most participants seemed to have very tense arm muscles while using the device to control the camera. They made strange and uncomfortable gestures, as well as much wider movements than required. All these findings were reflected in the answers obtained when participants were asked if they had felt tired in their arm and wrist during the test. 5 out of 10 participants stated that they felt tired and the rest did not experience fatigue themselves but claimed that they would have begun to feel it if the test had lasted a little longer. In short, all of them agreed that the device was not comfortable enough in order to be used several hours a day.

In addition, 7 of the users are right-handed, 2 are left-handed and 1 is ambidextrous (but uses the mouse of the PC with his right hand). Half of them (5 of 10) used their dominant hand to control the camera, placing the device on the corresponding side: right side if they are right-handed and left side if they are left-handed. 3 other participants used the device with both hands, placing it in the middle, just in front of the body. And the 2 remaining individuals used their non-dominant hand, both are right-handed so they used their left hand to control the camera.

## 5.2 Second prototype results

After implementing all the improvements discussed in Section 4.7, the design of the second prototype was completed. The final result is presented in Figure 5.8. Figure 5.8a shows the front part of the device with the LCD display already integrated in the case and the sensors securely attached. Figure 5.8b shows the connection reserved for the cable that will power the device from the USB port of the computer. And Figure 5.8c shows the bottom part of the device and how it is attached to the rest of the case, providing support for all the electronics which is hidden inside (apart from the sensors). In these images one can see how the height of the device has been reduced considerably from the initial prototype, as suggested by the test participants. In addition, it is possible to observe that it is a compact device, assembled in a robust way, so that it can be moved around without affecting its behavior or its performance.

Apart from the physical aspect of the device, in the second prototype the recognition of gestures has been improved by adjusting the parameters in the new version of the code. Furthermore, some new functionalities have been added, such as the camera speed control.



(a) Front side. (b) Back side. (c) Bottom part.

Figure 5.8: Final design. Real 3D-printed model of the second prototype.

### Sensor ranges

Ideally, the sensors can detect objects up to 400 cm away. However, for the purpose of this thesis a range of 50 cm is more than enough. For this reason, it was decided to set the sensors to the minimum distance mode, Distance Mode Short, which is equivalent to a maximum theoretical detection range of 130 cm. However, a range of 130 cm were considered to be a too long range for the purpose of this device. Since it was going to be placed on a desk and therefore small gestures would be used, a range this wide was not necessary. To limit the sensing range, the program was told that if values greater than 30 cm were detected, they were considered as out of bounds and therefore put to zero.

To calibrate the ranges for each gesture, the values were then adjusted manually. It was decided that to minimize the risk of activating a sensor that was not desired to be activated, the user would have to go close to it first. Afterwards, to keep the sensor and thereby the gesture activated the range for this was increased. The exact numbers for each gesture are written down in Table 5.1. The smaller range to activate the up-command was a result from the first prototype, were it was found that very often users accidentally performed the up-command instead of zoom, due to that they were to close to sensor 2. Regarding the zoom, it was desired to be able to enter the zoom-mode in a wide variety of ranges which is why it was kept with the same range both to activate and keep it activated.

Table 5.1: Main sensor ranges for each gesture.

| Gesture | Activate (mm) | Valid range (mm) |
|---|---|---|
| **Up** | 30 | 200 |
| **Down** | 50 | 200 |
| **Left** | 50 | 200 |
| **Right** | 50 | 200 |
| **Zoom** | 300 | 300 |

## Ranging frequency

According to the VL53L1X sensor data sheet [57] and API User Manual [58], the sensor follows an autonomous ranging sequence whose parameters can be modified by the user. This sequence is presented in Figure 5.9. The first step, *Ranging* with duration *time budget*, is where the sensor performs and reports the measurements. The second step, *Inter. Measurement*, is a programmable delay between ranging operations where the user can get the data from the sensor and clear the interrupt in order to prepare the sensor for the next ranging. The duration of the whole sequence is determined by the parameter called *inter-measurement period*.

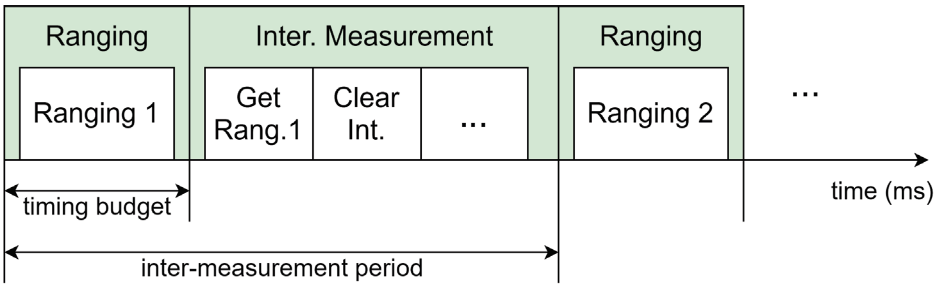



Figure 5.9: Autonomous ranging sequence and timings.

Taking into consideration the API User Manual restriction (minimum value of the *inter-measurement period* must be at least 4 ms larger than the *timing budget*), in this project the following values were chosen:

$$\textit{timing budget} = 20 \text{ ms}$$
$$\textit{inter-measurement period} = 25 \text{ ms}$$

That implies that the **maximum theoretical ranging frequency** of the device designed is:

$$Max\,Theoretical\,Freq. = \frac{1}{25\,ms} \cdot \frac{1000\,ms}{1\,s} = 40\,Hz \tag{5.1}$$

In practice, this frequency value is greatly reduced. The 5 sensors used, altogether, spend 54 ms (on average) to complete the ranging sequence. This implies that there is a new value for the *real inter-measurement period* and, consequently, a new value for the **real ranging frequency**:

$$Real\ Freq. = \frac{1}{54\ ms} \cdot \frac{1000\ ms}{1\ s} = 18.5\ Hz \tag{5.2}$$

The difference between the theoretical and the real value of the frequency is mainly due to the fact that the theoretical values refer to the use of only one sensor, while the *real inter-measurement period* has been measured in the device, where 5 sensors are used at the same time on the same I2C channel.

## Latency

First of all, it must be mentioned that the latency estimated in this section is just the one corresponding to the device developed in this thesis. Any possible delay in the camera is not contemplated.

The communication between the microcontroller, which controls the sensors, and the PC has been implemented in such a way that the latency value is not always constant. The microcontroller reads continuously the value from the sensors and it is only interrupted when the PC communicates with it via the serial port and asks for the sensor measurement values. At that point, the two extreme situations that can occur are the following:

- Best case scenario: the PC requires the values from the sensors when the microcontroller has just obtained them, i.e. it has just finished the ranging sequence. In this case the latency is determined by the time it takes for the PC to communicate with the microcontroller, interpret the information received and send the corresponding command to the camera. On average, the **minimum latency** is **75 ms**.

- Worse case scenario: the PC requires the values from the sensors when the microcontroller has just started a new ranging frequency, i.e. it has to wait until the sequence is finished to obtain the desired values. Here the latency is defined as the time between the start of the ranging sequence and the moment immediately after the command is sent to the camera. In this case the latency is calculating by adding the real inter-measurement period plus the time the PC needs to communicate to the Arduino, interpret the information and send the command to the camera:

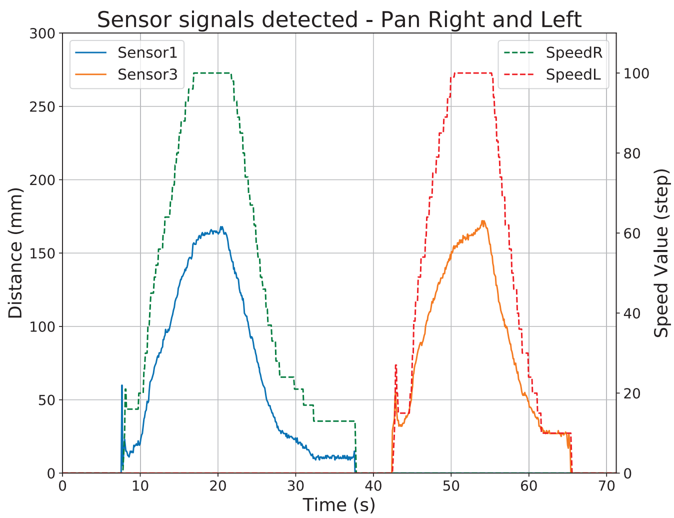
$$\text{Maximum latency} = 54\ ms + 75\ ms = \mathbf{129\ ms}$$

The previous values are calculated for the pan and tilt movements. The zoom is excluded from these calculations due to the fact that it is implemented so that the user must hold his hand above the top sensor for longer than 500 ms (0.5 s) in order to activate the zoom mode. This internal delay causes the total latency to be severely affected. The corresponding best and worse latency values in this case are, on average, 1001 ms and 1055 ms, respectively.

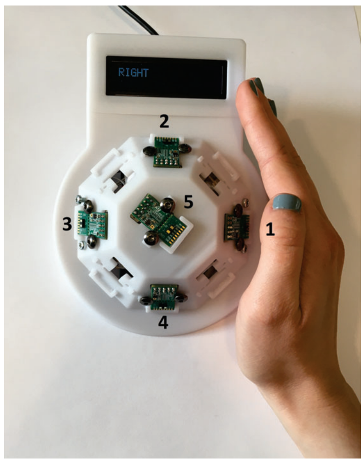
## Gestures and corresponding sensor signals

A series of images and graphs are presented below to explain how the gestures made by the user are read by the sensors in the final prototype of this project.

Firstly, the graph in Figure 5.10a shows the signals provided by the sensors when the user wants to make a pan movement, first to the right and then to the left. The blue curve in the graph shows how the user placed their hand on Sensor 1, as shown in Figure 5.10b, and then moved it slowly away from the sensor, increasing the distance detected (positive gradient in the blue curve between 8 s and 18 s). The gesture just described is used to increase the speed of the camera and this can be observed in the dashed green curve in the plot. The speed increases with the distance detected until it saturates when the maximum is reached (100 steps). When the user moves the hand closer to the device again, the speed decreases until its minimum value is reached (negative gradient between 21 s and 32 s). The exact same behavior is observed when the operator wants to pan left placing the hand close to Sensor 3: the speed (red dashed curve) increases with the distance from the sensor to the hand (orange curve).




(a) Graph representing sensor and speed signals.

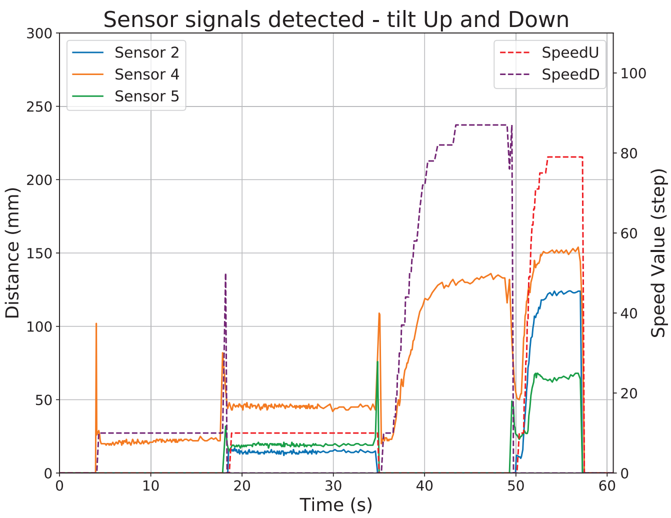



(b) Pan right.

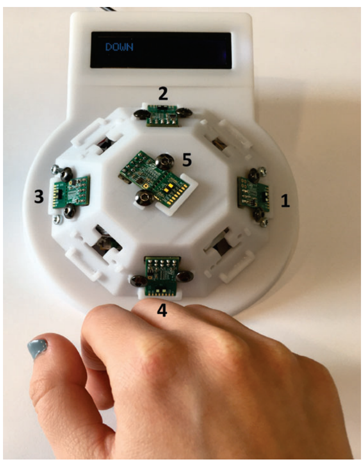
Figure 5.10: Sensor signals detected for pan right and left movements together with the gesture assigned to perform a pan right move.

Secondly, the plot in Figure 5.11a represents a series of tilt gestures detected by the sensors, specifically the sequence is: tilt down (5 s to 19 s), tilt up (19 s to 35 s), tilt down (35 s to 50 s) and tilt up (50 s to 57 s). Looking at the graph, it is easy to differentiate a tilt down from a tilt up movement based on the number of sensors that are activated. In order to tilt down, only the Sensor 4 must be activated (orange curve), as shown in Figure 5.11b. However, in order to tilt up, there are several sensor combinations available. For this example, the most common combination has been chosen, where sensors 2 (blue), 4 (orange) and 5 (green) are activated simultaneously.

If one now pays attention to the speed curves, it is possible to observe how the first two stretches of the sequence are performed at constant speed. The algorithm has been implemented so that small variations in the position of user's hand does not affect the speed value, avoiding to rectify the speed continuously. The next two sections, on the contrary, have been carried out at variable speed. As in the case of pan move, the speed increases with the distance between the hand and the corresponding sensor, and vice versa. However, unlike in the case of pan move, in this case the speed does not saturate at its maximum value (100 steps) but saturates around 88 steps for tilt down (purple dashed curve) and around 79 steps for tilt up (red dashed curve). This happens because there is a maximum tilt up and tilt down position due to physical limitations in the camera. Therefore, the speed saturates when this maximum position has been reached.



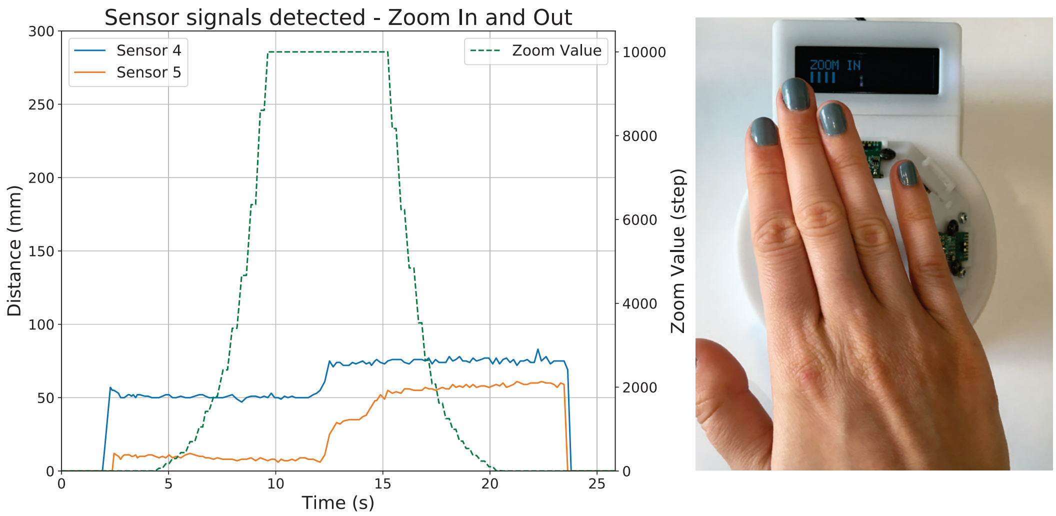

(a) Graph representing sensor and speed signals.




(b) Tilt down.

Figure 5.11: Sensor signals detected for tilt up and down movements together with the gesture assigned to perform a tilt down move.

Finally, the graph in Figure 5.12a shows the signals provided by the sensors when the user wants to zoom, first zooming in and then out. The Sensor 5 is the one in charge of detecting the gestures implemented for the zoom mode. Placing the hand on top of it, the user can enter the zoom mode. Moving the hand closer to the sensor corresponds to zoom in, as shown in Figure 5.12b, and moving the hand away to zoom out. In the graph one can see that in addition to Sensor 5 (orange curve), Sensor 4 has been represented (blue curve). This is because it is the most common sensor combination, since Sensor 4 detects the position of the wrist. When the hand is close to the device and both signal values are small (between 3 s and 12 s), the zoom in mode is activated and the zoom value (green dashed curve) increases until it reaches its maximum (10000 steps, maximum zoom in). Meanwhile, when the signals have larger values (between 12 s and 24 s), the zoom out mode is activated instead until it saturates at 1 step (maximum zoom out).




(a) Graph representing sensor and zoom signals. (b) Zoom in.

Figure 5.12: Sensor signals detected for zoom in and out mode together with the gesture assigned to zoom in.

Throughout the examples discussed in this section, one can see how the LCD display provides feedback to the user by writing down the detected action and the current zoom value (when the zoom mode is activated).

## 5.3 Economy

To estimate the cost of this device has been difficult. However, an attempt was made based on different factors brought up by either supervisors or employees at Axis Communications as well as information from the reseller.

The sensors used were sold in packages of two, meaning that, to build a device with five sensors, three packages of sensors have to be ordered. The price of one package is 197.80 SEK. Regarding the microcontroller that was used, one Arduino MEGA 2560 was needed to control the device and the price for this was 386.90 SEK. Regarding the display, also one was needed to build the device and the price for this was 127.40 SEK. To install the display, three additional resistors were needed and the price for these were about 1.50 SEK. To connect all the electronics together some wires were also used, the price for these were 19.60 SEK for 20 male to male wires.

In addition to the electronics, the price for the material of the device also had to be calculated. This was a bit harder to calculate, since it was a combination of the weight and kilo price of both the printing material used and the supporting material that had to be taken into account. An approximate given estimation of the price of the support used when printing was 311.10 SEK and the cost of the printed parts for the case were 206.70 SEK. This was calculated from the information given in Table 5.2.

Table 5.2: Cost for 3D-printing one case.

| Attribute | Data |
|---|---|
| **Supporting material** | 1571 SEK/kg |
| **Printing material** | 3226 SEK/kg |
| **Weight of the support** | 0.198 kg |
| **Weight of the printed parts** | 0.065 kg |
| **Total price of the support** | 311.10 SEK |
| **Total price of the printed parts** | 206.70 SEK |
| **Total cost** | 517.80 SEK |

When putting all of the above costs together, the total price for one unit would be **1646.60 SEK**. However, if several units of this device were to be manufactured it would be possible to reduce the final price per unit. An example with 100 devices was made to show the difference.

Regarding the sensors, the reseller had no information redarding how much they would be when buying several packages, so here it is assumed they have the same price as when buying only one package. To make 100 devices, 500 sensors are needed. Since they are sold two sensors in each package, 250 packages are needed. This would cost in total $250 * 197.80 = 49\ 445$ SEK. Regarding the Arduino, when buying 100 units, they are instead priced 342.70 SEK/unit, this means that the 100 microcontrollers in total would be $100 * 342.70 = 34\ 270.60$ SEK. The displays are also given a lower price if 100 items are bought at the same time, this would be instead 100.70 SEK/unit, which corresponds to $100 * 100.70 = 10\ 070.50$ SEK. When it comes to the wires and resistors, the approximate cost for these would be 2210 SEK.

The largest costs of this device are the microcontroller and the 3D-printing. Therefore, what has to be taken into account when doing this cost analysis is that if the device were to be manufactured in a larger scale, the Arduino would not be used as the microcontroller. Instead, a cheaper microcontroller would be bought that only has the necessary functionalities, or also possible, a microcontroller board would be designed and manufactured that only had the desired functionality. If producing this device in a larger scale an option would also be to buy only the sensor itself and make the additional electronics required afterwards, instead of buying the sensors together with the breakout boards. Moreover, 3D-printing the cases would be both expensive and time-consuming. Therefore the drawings of the case would be sent to another manufacturer which would produce the cases to a lower price.

All the prices used in the calculations are the ones that were valid at the time of the cost analysis. All electronic parts are taken from *Digi-Key* and the information regarding the 3D-printing is provided from employees of Axis Communications.

# 6

# Discussion

The aim of this chapter is to interpret the results that were presented in the previous chapter. The results are discussed and the effects of the results are further debated. There is also a discussion about what future applications the device could apply to, as well as regarding the future work that could be done to the device in order to make it better.

## 6.1 Usability test

In Section 5.1 there is a very detailed explanation of the results obtained from the usability test. The purpose of the following paragraphs is to evaluate those results and draw the conclusions that are used to answer the research questions formulated in the test description, design the second prototype and define the future work.

In Task 1 most of the users performed the task using Swipe Mode to pan and tilt, and Close-Far Mode to zoom in/out. This two modes are based in hand gestures that are very similar to the ones that are already implemented in the first prototype. That leads to conclude that the hand gesture vocabulary chosen is appropriate according to the design of the device and turns out to be intuitive for the users.

From Task 2 it can be concluded that there is no clear agreement on how to enable and disable the device. The vast majority of the participants supported the need to keep this option available on the device but, at the same time, it was decided to take into consideration the important reflection of two of the experienced participants, in which they stated that in the world of surveillance all devices must be constantly active. Therefore, it was decided to keep this function in the device (using the same gestures) but in a way that makes it very easy and fast to enable the device so that the operators lose as little time as possible.

From Task 3 and the time data, several conclusions can be drawn. Firstly, the fact that the participants generally spent much more time to find the first object than the second means that, once they have learned how to operate the device, controlling the camera becomes a much easier task. Furthermore, it shows that the knowledge required for its use can be assimilated very quickly, which could avoid having long training periods for the operators. Secondly, it can be concluded that experience in the control of PTZ cameras does not influence the learning speed of the use of the device, since both groups spent a very similar amount of time when trying to find the second object. Thirdly, an unexpected result was obtained: on average, the experienced group took longer time to find the first object than the inexperienced group. However, after reviewing the recordings again, it is believed that this happened because the members of Group 1 were busy trying to explain how the new device was different from the joystick or mouse, tools they usually use to control the cameras. Meanwhile, the participants in Group 2 were totally focused on performing the task correctly.

Regarding the errors detected in Task 3, the great amount accounted for *Tilt up (desired zoom in/out)* shows the need of improving the software implementation of the zoom and tilt functions. That, together with more experienced users, should be enough to avoid this type of errors or, at least, reduce the frequency of occurrence. Besides that, it was considered that the rest of errors can be easily prevented if the users gain some more experience using the device. For instance, performing the hand gestures far from the device (pan, tilt and zoom out) is an error that just occurs when the user is trying the device by the first time. The same applies to the other errors detected. Therefore, it was decided not to make any change in the implementation of this functions but supporting the idea of including some form of feedback for the user to tell whether the device is enabled or not. Comparing Group 1 and Group 2, it can be said that the experience in PTZ camera control have no influence in the better use of the device since both groups have conducted a very similar number of errors.

The questions asked to the participants at the end of the tasks were useful to gather direct feedback from the users and reinforce some of the conclusions that were already drawn from the tasks. The fast learning on how to use the device and the great improvement of the camera control with the experience are two facts that have been previously commented and the testers realized themselves too. The participants appreciated that the device can be used for both right- and left-handed with no modification required, as it was intended from the design phase; meanwhile the configuration of other tools available, such as the joystick and the mouse, needs to be changed depending on the user's dominant hand. Another feature highlighted was the sanitary condition of the device, which prevents the spreading of diseases. This gives a new benefit and supports the use of touch-less technology in the future. The potential of the prototype was also pointed out, stating that it could be further developed in order to introduce it in a real surveillance environment.

However, it is important to keep in mind that very often people want to be polite with the researchers and the feedback might be biased, i.e. too positive. Despite all these positive comments, it is clear to see that this prototype still needs a lot of development and testing before the reliable use in surveillance can be proved.

The frustrations felt by the participants are somewhat related to the suggested improvements. Including the speed control and the possibility to perform combined movements will make it possible for the user to control the camera more precisely, preventing the pin-point effect observed when trying to locate an object. These modifications will also make the device more competitive with respect to the tools that are currently used. The latency in the device is an important issue detected by most of the testers. Its negative effect can be slightly reduced by adding quick feedback for the user since they will know they are performing the right action and they will just need to wait to see the actual result on the screen. The ideal case would be to give feedback in the VMS (same screen of the camera view) so that the operator does not have to look away from the screen at any time. However, modifying the software platform is out of the scope of this master thesis so it was decided to add an LCD display on the device instead.

Although the results showed that the ergonomic design of the device is not good enough, it is believed that this conclusion is motivated due to the fact that the testers were doing weird, forced and too wide gestures that made them tired when finding out the functioning of the device. Therefore, making the device smaller (reducing its height) and training the users, showing them how to use the device according to the design (wrist resting on the desk), the physical effort required to use the device will be very much reduced.

Finally, the device was used in different ways during the test. That leads to conclude that the design is very flexible and it does not matter whether the operator uses their dominant hand, non-dominant hand or both hands at the same time. However, it has been observed that encouraging the use of the device with the non-dominant hand increases the efficiency of the operator's work as they do not have to move the hand to use the computer mouse, both devices can be used at the same time. This is how the device was used throughout the development and implementation phase and it turned out to be the most comfortable. The gestures are so simple that anyone can perform them with their non-dominant hand.

## 6.2 Second prototype

Although there are still aspects to include and improve in the designed device, as commented in Section 6.4 Future development, the results obtained for the second prototype are very satisfactory. It is a compact and quite complete device that covers the basic functions foreseen and even includes additional functionalities demanded by the test participants. Furthermore, it meets the requirements of portability and flexibility as well as all the other requirements mentioned at the beginning of the project.

The real ranging frequency obtained in the reading of the sensors has been considered adequate for the purpose of this thesis, since taking approximately 18 measurements per second is more than enough to detect the movement of the hand accurately enough.

Regarding the latency observed in the device, it is one of the aspects that can be improved in the future, since it was the greatest frustration experienced by most of the participants. However, the results measured on the second device are not disappointing, especially those obtained for the pan and tilt movements. The tests showed that, despite this delay between the movement and the response of the camera, the device can be used successfully. Once the users are aware of this aspect and they get used to it, the problem becomes less important.

From the readings of the sensor signals, it was possible to detect what sensors were activated when each gesture was performed. This information was of great help when implementing the functions that were to interpret each gesture. From the data given, the most common combinations were observed and then put together in the table in Appendix D, which lay the foundation of the entire translation from sensor data into gestures.

When it comes to the sensor signals that corresponds to each gesture, the results given by the graphs show that the output signals from the sensors are very accurate and without any major outliars or noise. This eased the implementation of the system a lot, since it was easy to identify and read the sensor data, the coding of the corresponding gesture functions was made successfully.

## 6.3 Future applications

The main application for this device is of course to control PTZ surveillance cameras in control rooms, however, it has potential to be used in other scenarios as well. During the usability test one of the participants that was experienced with PTZ cameras gave the suggestion that this device would be nice to have in showrooms to show potential customers what can be done and also to let them try to control the cameras in a fun way.

Apart from this the device could be used in both video games as a controller as well as in cars to control the audio system, blinkers or other features. In hospitals and nursing homes it would be a good control device for the surveillance cameras due to the fact that it is touch free and therefore good for sanitary reasons.

## 6.4 Future development

### Enable and disable function

When the device was developed it was decided to add an enable and disable function which could be used to put the device into stand-by mode while it was not used. But, as a result from the usability tests, some of the participants stated that in a real situation the control devices used in surveillance cameras should always be turned on in case the operator has to act quickly. When the second version of the device was developed the enable and disable function was kept since it was still considered useful. This was due to the fact that even though in some cases operators need to be able to act really fast, in some other cases the operator might want to be able to lock the camera in a certain view without moving if someone comes too close to the device sensors. Afterwards, an idea was that it could be useful to have two modes of the device, one that activates the enable and disable functions and one that deactivates them. From the test it was also concluded that the users had many different ideas on how to enable and disable the device, which is why it could be a good idea to let them customize what gesture they want to use that suits best for their specific use case. This way the operators could decide themselves how they would like to use the device.

### Reduced latency

When operating the camera, the device was considered somewhat inefficient due to the latency that was present when a gesture was performed until the desired command was executed by the camera. Some of the experienced latency was in the camera, which cannot be controlled by the device. But, by writing the code more efficiently, it would most likely be possible to reduce the current delay.

### Second usability test

Based on the results gathered from the usability test, the second prototype was developed. After this, it would be possible to make a second usability test to get more knowledge about how useful the device is as well as if and how it could be further developed. This test would give answers to if the new implementation made the device easier to use and also whether the users managed the tasks better when they already have tried the device once. In addition, from a second test the users might have further suggestions on how to improve the device that could be taken into account when designing the next versions.

The idea from the beginning was to make the second usability tests as well during the thesis. However, it was not possible to proceed with another test round before the due date. One of the main reasons was that preparing the tests, performing them and analyzing the results usually takes a lot of time, especially during the COVID-19 outbreak. It was hard to schedule suitable meetings with the test persons, which made the usability test require more time than expected.

### Follow object

Another suggestion that came from some of the users during the usability test was to add a feature that makes it possible to activate the function in the camera to follow a moving object on the screen. This is in particular useful in the surveillance industry where the camera operator wants to follow a person that appears on the screen. This way, the camera would automatically move to keep the person centered in the image without the operator having to control the camera. This means that the operator could rest his arm and only focus on watching the recording, which would be a great improvement to bring to a future product.

### Extra commands

By using the fact that some of the gesture combinations in Table D.1 in Appendix D are never getting used due to that it is very hard or impossible to perform them while using the device with only one hand, these combinations could instead be used to perform special commands. The joysticks as of today usually have buttons to be used when controlling the camera, for instance make the camera move to a preset position. For this new device an unused sensor combination could instead be used to perform the same command.

Another feature that could be implemented for the unused sensor combinations is for instance to make a screen shot of what is seen in the VMS at any time.

### Design and manufacture PCBs

The microcontroller that was used was the Arduino MEGA 2560 which lived up very well to its expectations in this thesis. However, in order to make the manufacturing of this device cheaper, a suggestion is to design and manufacture a PCB, just including the necessary functionality for this specific device.

### Sensor cover

To make the device more robust and less sensitive to dust and moisture the sensor pyramid could be put inside a cover. This way the electronics will not be exposed to any external damage. What is hard with this is however the fact that when covering all the sensors, their signals will bounce into the material and change values. Most likely the entire signal will not be able to pass through the cover, detect a hand and then get back to the sensor through the cover, without it changing direction. Therefore, adding a cover will require an investigation in what material to use and how to design it, as well as to calibrate the sensors accordingly.

# 7

# Conclusions

The results from this master thesis show that it is possible to control PTZ cameras by using hand gestures. More specifically, it is possible to develop a device that uses IR sensors to sense the placement of the hand and then uses the data to interpret a hand gesture and thereby a desired movement of the camera. Like this, the standard commands of the PTZ camera (pan, tilt and zoom) can be sent and executed successfully.

The results show that not only the standard pan, tilt and zoom commands can be included in this device, it is also possible to change the speed of the pan and tilt movements as well as to enable and disable the device by using simple gestures. From this, a conclusion that can be drawn is that other specific commands could also be used in the future by combining the readings of the sensors. This way the features of the device will be improved and therefore the device could become a competitor to the current joystick.

Another conclusion that is drawn is that the device very well stands up to its expectations about being intuitive to the user. Since the usability tests tells that all of the users reduced the time it took for them to find the second object from the first, this is a clear proof of that they could easily figure out how to use the device and what it could be used for. Also, when the users were asked how they thought that the device worked, several suggested solutions similar to the one that was implemented in the device already.

The tests made it clear that the majority of the users felt either pain or got tired when using the device. However, most of them used a lot bigger gestures than what was necessary and they did not try to place the device so that they could use the desk to rest their arm and wrist on while using it. These are of course reasons why the users felt pain, but another cause to it was the design of the device. By designing a new case for the device which is lower, the intention was that the ergonomics of the device would be improved, since it should be easier to keep the arm resting on the desk while using the device. However, tests have to be made in order to observe how the device affects the operators in the long term.

# Bibliography


[1] *Sdm. Interview with martin gren, inventor of the network camera*. 2011. URL: https://www.sdmmag.com/articles/87054-interview-with-martin-gren-inventor-of-the-network-camera (visited on 2020-03-10).

[2] *Axis communications. About axis*. 2019. URL: https://www.axis.com/sv-se/about-axis (visited on 2020-03-10).

[3] *Axis communications. Axis q60 ptz network camera series*. 2020. URL: https://www.axis.com/products/axis-q60-series (visited on 2020-02-19).

[4] *Redshark. Ptz cameras have grown up: now they do broadcast quality and beyond*. 2020. URL: https://www.redsharknews.com/production/item/4333-ptz-cameras-have-grown-up-now-they-do-broadcast-quality-and-beyond-sponsored (visited on 2020-02-06).

[5] *Axis communications. Axis p5655-e ptz network camera*. 2020. URL: https://www.axis.com/sv-se/products/axis-p5655-e (visited on 2020-02-12).

[6] *Axis communications. Joysticks & control boards*. 2020. URL: https://www.axis.com/sv-se/products/joysticks-and-control-boards (visited on 2020-02-12).

[7] *Panasonic. Wv-cu980 - system controllers*. 2020. URL: https://security.panasonic.com/products_technology/products/wv-cu980/ (visited on 2020-02-12).

[8] *Hanwha techwin. Spc-7000 system control keyboard*. 2020. URL: https://www.hanwha-security.com/en/products/peripherals/controller/controller/SPC-7000/overview/ (visited on 2020-02-12).

[9] *Hikvision. Ds-1600ki network keyboard*. 2020. URL: https://www.hikvision.com/europe/products/Transmission-and-Display-Products/Controllors---Transmission/Keyboards/ds-1600ki-b-/ (visited on 2020-02-12).

[10] *Axis communications. Majestar casino enhances customer security with 24-hour surveillance system*. 2020. URL: https://www.axis.com/customer-story/3670 (visited on 2020-02-20).

[11] *Collins. Definition of hand gesture*. 2020. URL: https://www.collinsdictionary.com/dictionary/english/hand-gesture (visited on 2020-02-12).

[12] M. Nielsen, M. Störring, T. B Moeslund, and E. Granum. "A procedure for developing intuitive and ergonomic gesture interfaces for hci". *Aalborg University, Laboratory of Computer Vision and Media Technology, Denmark* (2003). URL: https://link.springer.com/chapter/10.1007/978-3-540-24598-8_38.

[13] *Google. Google soli, radar chip for motion detection*. 2020. URL: https://atap.google.com/soli/ (visited on 2020-02-12).

[14] *Techcrunch. Bmw's magical gesture control finally makes sense as touchscreens take over cars*. 2019. URL: https://techcrunch.com/2019/11/04/bmws-magical-gesture-control-finally-makes-sense-as-touchscreens-take-over-cars/ (visited on 2020-02-12).

[15] *Ultraleap. Leap motion controller, small. fast. accurate. world-class hand tracking for anyone, anywhere*. 2020. URL: https://www.ultraleap.com/product/leap-motion-controller/ (visited on 2020-02-12).

[16] *Tork. Tork foam skincare automatic dispenser*. 2018. URL: https://www.torkusa.com/product/571608/dispenser/hand-sanitizer (visited on 2020-02-12).

[17] D. Rempel, M. J. Camilleri, and D. L. Lee. "The design of hand gestures for human–computer interaction: lessons from sign language interpreters". *International Journal of Human-Computer Studies* **72**:10 (2014), pp. 728–735. ISSN: 1071-5819. DOI: https://doi.org/10.1016/j.ijhcs.2014.05.003. URL: http://www.sciencedirect.com/science/article/pii/S1071581914000706.

[18] *No more pain ergonomics. Do i need a vertical mouse?* 2020. URL: https://www.nomorepainergonomics.com.au/blogs/no-more-pain-ergonomics/115968197-do-i-need-a-vertical-mouse (visited on 2020-03-02).

[19] P. Y. Loh and S. Muraki. "Effect of wrist deviation on median nerve cross-sectional area at proximal carpal tunnel level". *Iranian Journal of Public Health* **43**:3 (2014), pp. 180–185. ISSN: 2251-6085. URL: https://www.researchgate.net/publication/267903091_Effect_of_

Wrist_Deviation_on_Median_Nerve_Cross-Sectional_Area_at_Proximal_Carpal_Tunnel_Level.

[20] L. Dimberg, J. Goldoni Laestadius, S. Ross, and I. Dimberg. "The changing face of office ergonomics". *The Ergonomics Open Journal* **8** (2015), pp. 38–56. DOI: 10.2174/1875934301508010038. URL: https://www.researchgate.net/publication/277672834_The_Changing_Face_of_Office_Ergonomics.

[21] *Arbetsmiljö verket. Afs 2012:2 belastningsergonomi - arbetsmiljöverkets föreskrifter och allmänna råd om belastningsergonomi*. 2019. URL: https://www.av.se/globalassets/filer/publikationer/foreskrifter/belastningsergonomi-foreskrifter-afs2012-2.pdf (visited on 2020-03-02).

[22] *Nielsen norman group. Thinking aloud: the 1 usability tool*. 2012. URL: https://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/ (visited on 2020-03-02).

[23] *Kimberly crawford. Design thinking toolkit, activity 14 – i like, i wish, what if*. 2018. URL: https://spin.atomicobject.com/2018/09/12/i-like-i-wish-what-if/ (visited on 2020-03-02).

[24] *Usability body of knowledge. Wizard of oz. usability testing with users*. 2012. URL: https://www.usabilitybok.org/wizard-of-oz (visited on 2020-03-02).

[25] H. I. Stern, J. P. Wachs, and Y. Edan. "Optimal hand gesture vocabulary design using psycho-physiological and technical factors". *7th International Conference on Automatic Face and Gesture Recognition (FGR06)* (2006), pp. 257–262. URL: https://ieeexplore.ieee.org/document/1613029/references#references.

[26] S. Gulati and B. Rosepreet Kaur. "Comprehensive review of various hand detection approaches". *2018 international conference on circuits and systems in digital enterprise technology (ICCSDET)* (2018). URL: https://ieeexplore.ieee.org/abstract/document/8821238.

[27] *Fierceelectronics. What is a motion sensor?* 2019. URL: https://www.fierceelectronics.com/sensors/what-a-motion-sensor (visited on 2020-03-09).

[28] *Arrow. Ultrasonic sensors: how they work*. 2018. URL: https://www.arrow.com/en/research-and-events/articles/ultrasonic-sensors-how-they-work-and-how-to-use-them-with-arduino (visited on 2020-03-09).

[29] *Electronicshub. Ir sensor*. 2015. URL: https://www.electronicshub.org/ir-sensor/ (visited on 2020-03-09).

[30] *Rf wireless world. Advantages and disadvantages of infrared sensor*. 2012. URL: https://www.rfwireless-world.com/Terminology/Advantages-and-Disadvantages-of-Infrared-Sensor.html (visited on 2020-03-10).

[31] Y. Jia, L. Guo, and X. Wang. "Transportation cyber-physical systems". *Department of Automotive Engineering, Clemson University, Greenville, SC, United States* (2018), pp. 81–113. URL: https://www.sciencedirect.com/science/article/pii/B9780128142950000046.

[32] *Circuitglobe. Resistive transducer*. 2020. URL: https://circuitglobe.com/resistive-transducer.html (visited on 2020-03-12).

[33] *Watelectronics. What is resistive transducer – working & its applications*. 2020. URL: https://www.watelectronics.com/what-is-resistive-transducer-working-its-applications/ (visited on 2020-03-12).

[34] *Resistorguide. Photo resistor*. 2019. URL: http://www.resistorguide.com/photoresistor/ (visited on 2020-03-12).

[35] J. Christenson. "Sensors and transducers". *Handbook of Biomechatronics* (2019). URL: https://www.sciencedirect.com/science/article/pii/B9780128125397000039.

[36] *Fierceelectronics. Capacitive sensing in human body contact applications*. 2010. URL: https://www.fierceelectronics.com/embedded/capacitance-sensing-human-body-contact-applications (visited on 2020-03-09).

[37] *Rf wireless world. Advantages of capacitive sensor | disadvantages of capacitive sensor*. 2012. URL: https://www.rfwireless-world.com/Terminology/Advantages-and-disadvantages-of-Capacitive-Sensor.html (visited on 2020-05-14).

[38] *Machinedesign. Proximity sensors compared: inductive, capacitive, photoelectric, and ultrasonic*. 2001. URL: https://www.machinedesign.com/automation-iiot/sensors/article/21831577/proximity-sensors-compared-inductive-capacitive-photoelectric-and-ultrasonic (visited on 2020-03-12).

[39] *Elprocus. Gyroscope sensor working and its applications*. 2020. URL: https://www.elprocus.com/gyroscope-sensor/ (visited on 2020-03-12).

[40] *Elprocus. Accelerometer sensor working and applications*. 2020. URL: https://www.elprocus.com/accelerometer-sensor-working-and-applications/ (visited on 2020-03-12).

[41] *Elprocus. Photodiode working principle, characteristics and applications*. 2020. URL: https://www.elprocus.com/photodiode-working-principle-applications/ (visited on 2020-03-30).

[42] J.-M. Irazabal and S. Blozis. "An10216-01 i2c manual". *Philips Semiconductors* (2003). URL: https://www.nxp.com/docs/en/application-note/AN10216.pdf.

[43] *Analog devices. I2c quick-guide*. 2020. URL: https://www.analog.com/media/en/technical-documentation/product-selector-card/i2Cb.pdf (visited on 2020-03-30).

[44] C. Liu, Q. Meng, T. Liao, X. Bao, and C. Xu. "A flexible hardware architecture for slave device of i2c bus". *2019 International Conference on Electronic Engineering and Informatics (EEI)* (2019). DOI: 10.1109/EEI48997.2019.00074. URL: https://ieeexplore.ieee.org/document/8991113.

[45] *Real python. Python's requests library*. 2020. URL: https://realpython.com/python-requests/ (visited on 2020-05-04).

[46] *Geeksforgeeks. Get and post requests using python*. 2020. URL: https://www.geeksforgeeks.org/get-post-requests-using-python/ (visited on 2020-04-02).

[47] *Digi-key. Sensor photodiode sfh 2701. osram opto semiconductors inc*. 2020. URL: https://www.digikey.se/product-detail/en/osram-opto-semiconductors-inc/SFH-2701/475-2967-1-ND/2794398?cur=SEK&lang=en (visited on 2020-03-20).

[48] *Digi-key. A111 – pulsed coherent radar (pcr). acconeer ab*. 2020. URL: https://www.digikey.se/product-detail/en/acconeer-ab/A111-001-T%5C%26R/1891-A111-001-T%5C%26RCT-ND/8040771?utm_adgroup=RF%5C%20Misc%5C%20ICs%5C%20and%5C%20Modules&utm_source=google&utm_medium=cpc&utm_campaign=Google%5C%20Shopping_RF%5C%2FIF%5C%20and%5C%20RFID&utm_term=&productid=8040771&gclid=Cj0KCQjwjcfzBRCHARIsAO-1_OrPGEI0U6vMmBLb6h-onbcvABxVKH67issyYNTHS9rUUzBqZR4r6ZgaAtIoEALw_wcB (visited on 2020-03-20).

[49] *Digi-key. Ir sensor vl53l1x. a new generation, long distance ranging time-of-flight sensor based on st's flightsense™ technology. stmicroelectronics*. 2020. URL: https://www.digikey.se/product-detail/en/stmicroelectronics/VL53L1CXV0FY-1/497-17764-1-ND/8276742?_ga=2.193786669.1416849787.1584537805-804808509.1584537805 (visited on 2020-03-20).

[50] *Digi-key. Ultra-low-power ultrasonic time-of-flight (tof) range sensor ch-101. tdk invensense*. 2020. URL: https://www.digikey.se/product-detail/en/tdk-invensense/MOD_CH101-03-01/1428-MOD_CH101-03-01-ND/10494500 (visited on 2020-03-20).

[51] *Digi-key. Rgb and gesture sensor apds-9960*. 2020. URL: https://www.digikey.se/product-detail/en/sparkfun-electronics/SEN-12787/1568-1131-ND/5623214 (visited on 2020-05-04).

[52] *Stmicroelectronics. Vl53l1x-satel data brief*. 2019. URL: https://www.st.com/content/ccc/resource/technical/document/data_brief/group1/7f/04/01/87/89/4d/4f/99/DM00487170/files/DM00487170.pdf/jcr:content/translations/en.DM00487170.pdf (visited on 2020-04-29).

[53] *Pololu. St vl53l1x api implementation for arduino*. 2019. URL: https://github.com/pololu/vl53l1x-st-api-arduino (visited on 2020-05-08).

[54] *Pyserial. Documentation of pyserial library*. 2020. URL: https://pythonhosted.org/pyserial/ (visited on 2020-05-13).

[55] *Paramiko. A python implementation of sshv2*. 2020. URL: http://www.paramiko.org/ (visited on 2020-05-04).

[56] *Digi-key. Lcd mod 32dig 16x2 transmisv wht*. 2020. URL: https://www.digikey.se/products/en?keywords=NHD-0216K1Z-NSW-FBW-L-ND (visited on 2020-05-04).

[57] *Stmicroelectronics. Vl53l1x - a new generation, long distance ranging time-of-flight sensor based on st's flightsense™ technology*. 2018. URL: https://www.st.com/content/ccc/resource/technical/document/datasheet/group3/7d/85/c8/95/fb/3b/4e/2d/DM00452094/files/DM00452094.pdf/jcr:content/translations/en.DM00452094.pdf (visited on 2020-02-28).

[58] *Stmicroelectronics. Vl53l1x api user manual*. 2018. URL: https://www.st.com/resource/en/user_manual/dm00474730-vl53l1x-api-user-manual-stmicroelectronics.pdf (visited on 2020-02-28).

[59] *Digi-key*. 2020. URL: https://www.digikey.com/ (visited on 2020-03-18).

# A

# Explanation of the assessment criteria

***Explanation of the assessment criteria*** As it has been previously mentioned, the criteria used to evaluate the sensors have been deduced from the requirements set for this project. This section is intended to give a more detailed description of those criteria so that the reader can easily understand the reasoning followed to translate the data in Table B.2 into the scores in Table 4.4 and Table B.3.

- *Price*. This criterion considers only the price of the sensor, without taking into consideration any breakout board or development kit. The higher score corresponds to the lower price, and vice versa. Besides that, to make the comparison as realistic as possible, the prices of all four sensors have been checked in Digi-Key [59], one of the electronic component suppliers for Axis Communications. This is the reason why all the references given for the four selected sensors are from the same web page, there one can find the price per sensor together with a link to its data sheet.

- *Independence from external conditions*. Here it is evaluated whether the behavior of the sensor is affected by the existence of dust, dirt, smoke or moisture. The highest score corresponds to the absence of effects produced by any of those factors, and vice versa.

- *Independence from the temperature*. In this case, it is evaluated whether the sensor needs any extra electronic components/circuit to compensate for the effect of temperature changes on the signal generated from the sensor. The highest score corresponds to the ideal case where the performance of the sensor is not dependent on the temperature.

- *Detection of any surface*. This criterion evaluates whether the sensor is able to detect a hand and any fabric that could be used on or near the hand (the operator could wear gloves, for example). A lower score is given when the sensor has better performance detecting other surfaces that those mentioned.

- *Range*. This parameter represents how far can the sensor detect an object, i.e. the distance between the sensor and the object measured perpendicularly from the sensor surface. It has been decided that the range needed for this project should be the following interval: from 0 cm to 50 cm. Therefore, the highest score is given when the sensor is able to measure within the specified limits.

- *Accuracy*. This criteria evaluates which is the smallest measurement unit in which the sensor can provide data. The highest score corresponds to the smallest measurement unit.

- *Beam angle*. This parameter represents the angle in which the sensor is able to detect an object accurately enough. The ideal FoV is 180º, so the higher score corresponds to angles close to 180º, and vice versa.

- *Software requirements*. This criterion evaluates the complexity of the code that needs to be implemented in order to detect hand gestures in relation to the time available to do the master thesis (taking into account that this is not a software based project). The lower scores are given to the most complex code implementations (such as advanced signal processing).

- *Ability to further develop*. Here it is evaluated whether the data provided by the sensor is sufficient to be able to recognize more complex hand gestures in the future. The highest score corresponds to the greatest capacity for future development.

# B

# Concept and sensor scoring

Table B.1 presents the complete version of Table 4.3, showing the importance of each of the attributes expressed as weights (column *Weight*) and the score given to each of the evaluated concepts. The *Total score* of the best concepts is highlighted in green.

Table B.2 gathers the characteristics of the sensors used to grade them in Table B.3. Here the *Total score* of the best sensors is also highlighted in green.

Table B.1: Concept grading matrix.

| Attribute | Weight | Sensor Box | Sensor Platform | Sensor Stick | Sensor Glove | Sensor Field | Sensor Cube | Sensor Dome | Camera |
|---|---|---|---|---|---|---|---|---|---|
| **Appearance** | | | | | | | | | |
| Size & Weight | 5 | 1 | 4 | 5 | 5 | 5 | 4 | 4 | 5 |
| Aesthetics | 3 | 1 | 5 | 4 | 3 | 5 | 5 | 5 | 5 |
| Sum | - | 8 | 35 | 37 | 34 | 40 | 35 | 35 | 40 |
| **Functionality** | | | | | | | | | |
| Smooth Movement | 3 | 3 | 5 | 5 | 5 | 5 | 2 | 5 | 5 |
| Closed case | 4 | 5 | 2 | 4 | 3 | 2 | 2 | 2 | 5 |
| Sum | - | 29 | 23 | 31 | 27 | 23 | 14 | 23 | 35 |
| **Ease of Use** | | | | | | | | | |
| Safety | 5 | 5 | 5 | 5 | 4 | 5 | 5 | 5 | 5 |
| Intuitive | 5 | 3 | 5 | 5 | 4 | 4 | 4 | 5 | 5 |
| Portable | 5 | 3 | 4 | 5 | 5 | 4 | 4 | 4 | 5 |
| USB connection | 5 | 5 | 5 | 3 | 3 | 5 | 5 | 5 | 5 |
| Ergonomic | 4 | 2 | 4 | 5 | 5 | 5 | 3 | 5 | 5 |
| Sum | - | 88 | 111 | 110 | 100 | 110 | 102 | 115 | 120 |
| **Content** | | | | | | | | | |
| Use of Electronics | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 1 |
| Not Software based | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 1 |
| Complexity | 5 | 5 | 5 | 5 | 4 | 5 | 4 | 5 | 1 |
| Innovative | 4 | 5 | 5 | 2 | 5 | 5 | 5 | 5 | 5 |
| Touch-Free | 5 | 5 | 5 | 1 | 1 | 5 | 5 | 5 | 5 |
| Use of Gestures | 5 | 5 | 5 | 1 | 5 | 5 | 5 | 5 | 5 |
| | - | 140 | 140 | 88 | 115 | 140 | 135 | 140 | 85 |
| **Total score** | - | 265 | 309 | 266 | 276 | 313 | 286 | 313 | 280 |

Table B.2: Parameters and recommendations of four sensor commercial models.

| **Attribute** | Photodiode | Radar | IR | Ultrasound |
|---|---|---|---|---|
| Price per sensor (SEK) | 8,37 | 130,28 | 68,79 | 188,96 |
| External conditions affecting | dirt, dust, light | - | dust, smoke | - |
| Temperature affecting | yes | no | no | yes |
| Detection of any surface | yes | metallic preferred | yes | not soft, thin, curved |
| Range | - | 2 m | 4 m | 4 cm - 1,2 m |
| Accuracy | - | mm | mm | mm |
| Beam angle | Half angle ±60° | 80°(H) & 40°(E) | 27° | 180° |
| Software requirements | simple | ML | simple | ML |

Table B.3: Sensor grading matrix.

| **Attribute** | Weight | Photodiode | Radar | IR | Ultrasound |
|---|---|---|---|---|---|
| **Economy** | | | | | |
| Price per sensor | 2 | 5 | 2 | 4 | 1 |
| Sum | - | 10 | 4 | 8 | 2 |
| **Limitations** | | | | | |
| Independence from external conditions | 1 | 2 | 5 | 2 | 5 |
| Independent from temperature | 3 | 1 | 5 | 5 | 1 |
| Detection of any surface | 4 | 4 | 2 | 4 | 3 |
| Sum | - | 21 | 28 | 33 | 35 |
| **Electronics** | | | | | |
| Range | 5 | 3 | 5 | 5 | 3 |
| Accuracy | 3 | 3 | 5 | 5 | 5 |
| Beam angle | 5 | 4 | 4 | 2 | 5 |
| Sum | - | 36 | 60 | 50 | 55 |
| **Other** | | | | | |
| Software requirements | 3 | 4 | 2 | 4 | 2 |
| Ability to further develop | 4 | 2 | 5 | 4 | 4 |
| | - | 20 | 26 | 28 | 22 |
| **Total score** | - | 87 | 118 | 119 | 114 |

# C

# Discarded sensors

As it has already been mentioned at the beginning of the Section 4.3, some sensors were directly discarded for this project. In the following paragraphs one can find which are the sensors discarded and a brief description explaining why that decision was made.

The gyroscope, the accelerometer and the resistive strain gauge sensors need to be in contact with the moving object, in this case the hand, in order to detect its motion. It was decided not to use any of those three sensor types due to the fact that the two concepts, the *Sensor Field* and the *Sensor Dome*, are devices that can be designed by using touch-free technology, which is one of the requirements to be fulfilled in this project.

The inductive sensor generates an electrical signal that depends on the displacement of a conductive object (ferrous material) with respect to a coil. This working principle applied to this project would imply that either the user have to touch a surface in order to vary the distance between the coil and the ferrous material, or that a wearable made out of conductive material has to be designed so that the user can attach it to their hand and move it over the inductive sensors. None of these solutions were considered suitable for this project since the first one implies the non-fulfilment of the touch-free technology requirement and the second one implies a considerable increase in the complexity of the project (a wearable would also have to be designed) and at the same time a decrease in the comfort of the operators.

Regarding the capacitive sensors, they are generally used to detect proximity, solid and liquid levels or on touch screens. In this case, they were discarded due to the fact that their measuring range is very small (0.05 mm to 40 mm) and the touch screen was not an option.

# D

# Gesture combination table

Table D.1 represents all the possible combinations of activated (1) and deactivated (0) sensors that can be found with the 5 sensors used:

$$2^5 = 32 \text{ possible combinations}$$

In the last column, *Output*, the action corresponding to each combination is indicated. A dash means that either the combination of sensors is considered unfeasible or that no action should be taken.

Table D.1: Possible combinations of activated and deactivated sensors that can be found when using 5 sensors.

| Sensor2 | Sensor3 | Sensor5 | Sensor1 | Sensor4 | Output |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | - |
| 0 | 0 | 0 | 0 | 1 | Down |
| 0 | 0 | 0 | 1 | 0 | Right |
| 0 | 0 | 0 | 1 | 1 | Right |
| 0 | 0 | 1 | 0 | 0 | Zoom |
| 0 | 0 | 1 | 0 | 1 | Zoom |
| 0 | 0 | 1 | 1 | 0 | Right |
| 0 | 0 | 1 | 1 | 1 | Right |
| 0 | 1 | 0 | 0 | 0 | Left |
| 0 | 1 | 0 | 0 | 1 | Left |
| 0 | 1 | 0 | 1 | 0 | Disable |
| 0 | 1 | 0 | 1 | 1 | - |
| 0 | 1 | 1 | 0 | 0 | Left |
| 0 | 1 | 1 | 0 | 1 | Left |
| 0 | 1 | 1 | 1 | 0 | Disable |
| 0 | 1 | 1 | 1 | 1 | - |
| 1 | 0 | 0 | 0 | 0 | Up |
| 1 | 0 | 0 | 0 | 1 | Up |
| 1 | 0 | 0 | 1 | 0 | Up |
| 1 | 0 | 0 | 1 | 1 | Right |
| 1 | 0 | 1 | 0 | 0 | Up |
| 1 | 0 | 1 | 0 | 1 | Up |
| 1 | 0 | 1 | 1 | 0 | Up |
| 1 | 0 | 1 | 1 | 1 | - |
| 1 | 1 | 0 | 0 | 0 | Up |
| 1 | 1 | 0 | 0 | 1 | Left |
| 1 | 1 | 0 | 1 | 0 | - |
| 1 | 1 | 0 | 1 | 1 | - |
| 1 | 1 | 1 | 0 | 0 | Up |
| 1 | 1 | 1 | 0 | 1 | - |
| 1 | 1 | 1 | 1 | 0 | - |
| 1 | 1 | 1 | 1 | 1 | - |

# E

# Usability test results

The following tables present all the results gathered after the usability tests performed in this thesis.

Tables E.1, E.2, E.3 and E.4 are based on the observation of the participants behavior and comments during the proposed tasks. In addition, Tables E.5 and E.6 express the answers obtained in the post-test questionnaire.

Table E.1: Results Task 1. Suggested gestures to pan, tilt and zoom.

| Group | Participant | Pan | Tilt | Zoom |
|---|---|---|---|---|
| **1** | **1** | Swipe Mode | Swipe Mode | Touchscreen Mode |
| **1** | **2** | Swipe Mode | Swipe Mode | scroll up/down |
| **1** | **3** | Swipe Mode | no answer | no answer |
| **1** | **4** | Rotating ball Mode | Rotating ball Mode | Close-Far Mode |
| **1** | **5** | Swipe Mode | Swipe Mode | Close-Far Mode |
| **2** | **6** | Rotating ball Mode | Rotating ball Mode | Touchscreen Mode |
| **2** | **7** | Swipe Mode | Swipe Mode | Touchscreen Mode |
| **2** | **8** | Rotating ball Mode | Rotating ball Mode | Close-Far Mode |
| **2** | **9** | Swipe Mode | Swipe Mode | Close-Far Mode |
| **2** | **10** | Swipe Mode | Swipe Mode | Close-Far Mode |

Table E.2: Results Task 2. Suggested gestures to enable and disable the device.

| Group | Participant | Function needed? | Enable | Disable |
|---|---|---|---|---|
| **1** | **1** | yes | when presence detected | when no presence detected |
| **1** | **2** | yes | placing hand on top | placing hand on top |
| **1** | **3** | no | no answer | no answer |
| **1** | **4** | no | pressing button | pressing button |
| **1** | **5** | yes | placing hand on top | placing hand on top |
| **2** | **6** | yes | when presence detected | when no presence detected |
| **2** | **7** | yes | shaking hand several times | pressing button |
| **2** | **8** | yes | snapping fingers | when no presence detected |
| **2** | **9** | yes | shaking hand several times | shaking hand several times |
| **2** | **10** | yes | shaking hand several times | shaking hand several times |

Table E.3: Results Task 3. Time spent solving the task: Time 0-1 (from start until first object found) and Time 1-2 (from first to second object found).

| Group | Participant | Time 0-1 | Time 1-2 |
|---|---|---|---|
| **1** | **1** | 00:02:01 | 00:01:53 |
| **1** | **2** | 00:06:20 | 00:01:20 |
| **1** | **3** | 00:05:32 | 00:01:25 |
| **1** | **4** | 00:01:36 | 00:01:15 |
| **1** | **5** | 00:04:55 | 00:00:42 |
| **2** | **6** | 00:03:06 | 00:01:27 |
| **2** | **7** | 00:03:36 | 00:00:36 |
| **2** | **8** | 00:03:00 | 00:01:55 |
| **2** | **9** | 00:01:43 | 00:00:41 |
| **2** | **10** | 00:03:37 | 00:01:07 |
| **1** | **Avg. Time** | 00:04:05 | 00:01:17 |
| **2** | **Avg. Time** | 00:03:00 | 00:01:09 |

Table E.4: Results Task 3. Amount of errors detected during the task. The most common ones are distinguished in bold.

| | **Group 1** | | | | | **Group 2** | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Errors detected** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **Total** |
| **Pan far from device** | | 3 | 3 | | 3 | | 2 | 2 | 2 | 2 | 17 |
| **Tilt far from device** | | 2 | 2 | | 2 | 1 | 4 | 3 | | 1 | 15 |
| **Zoom in/out (desired tilt down)** | 1 | 2 | 2 | | 3 | 2 | 2 | 1 | 1 | 2 | 16 |
| Zoom in/out (desired tilt up) | 1 | | | | | | | | 1 | | 2 |
| Tilt up (desired tilt down) | | | | | | | | 1 | | | 1 |
| **Zoom out far from device** | | 1 | 2 | 2 | | 3 | 2 | 1 | | 1 | 12 |
| **Tilt up (desired zoom in/out)** | 2 | 3 | 3 | 3 | 3 | 6 | 5 | 1 | 1 | 1 | 28 |
| **Not entering zoom mode** | | | 2 | 1 | | 2 | | 2 | | 2 | 9 |
| Zoom mode (desired do nothing) | | 1 | | | | | | | | | 1 |
| Zoom in (desired zoom out) | | 1 | 1 | | | | | | | | 2 |
| Zoom out (desired zoom in) | | | 1 | | | | | | | | 1 |
| **Zoom mode (desired enable device)** | 1 | | | | 1 | | 1 | | | | 3 |
| **Total per participant** | 5 | 13 | 16 | 6 | 12 | 14 | 16 | 11 | 5 | 9 | 107 |

Table E.5: Results post-test questionnaire. Answers to Question 1: What was your experience using the device?, Question 2: What are the best things with this device? and Question 3: What are the biggest frustrations you met from using the device?

| Group | Participant | Question 1 | Question 2 | Question 3 |
|---|---|---|---|---|
| **1** | **1** | Easy to use, easier with experience; good, it has potential | Both right- and left-handed (no changes required); it is cool | Latency in the response |
| **1** | **2** | Difficult to control, easier with experience | Good to use in show rooms; it is cool and interesting | Pin-Point effect is not good for a control room |
| **1** | **3** | Bad, not suitable for surveillance; easy with experience | Touch-free prevents spread of diseases; both right- and left-handed (no changes required) | Latency in the response; zoom mode not working properly |
| **1** | **4** | Good follow-up of the hand; hard to control precisely | Both right- and left-handed (no changes required); it is cool, good device | Latency in the response; Pin-Point effect is tiring; cannot control speed |
| **1** | **5** | Good to use new technologies; there is some delay | It is cool to use it | Latency in the response; Pin-Point effect is not good; performs undesired movements; not very comfortable to use |
| **2** | **6** | Easy with experience; good, it has great potential | It is a cool device | Zoom out not always working |
| **2** | **7** | Hard to control first, easier with experience; good response to gestures | It is cool; it has great potential | Zoom mode is hard to understand and use |
| **2** | **8** | Good; pressing imaginary buttons (no use of gestures) | Touch-free prevents spread of diseases; it is cool | Hard to reach an exact position (Pin-Point); it does not give any feedback |
| **2** | **9** | Easy with experience; good | Touch-free prevents spread of diseases; it is cool and futuristic | Latency in the response |
| **2** | **10** | Easy with experience; good | It is cool and intuitive | Latency in the response; it tilts up when trying to zoom |

Table E.6: Results post-test questionnaire. Answers to Question 4: What was your feeling of using the device, did you feel tired during the test?, Question 5: Are you left or right handed? and Question 6: Is there something you would like to add, or how can we improve the device?

| Group | Participant | Question 4 | Question 5 | Question 6 |
|---|---|---|---|---|
| **1** | **1** | Yes, tired in the arm | I am comfortable using both, but I use the mouse with the right hand | Speed control; user feedback; combined movements (pan-tilt) |
| **1** | **2** | Yes, tired in the arm | Right-handed | Speed control; user feedback; combined movements (pan-tilt-zoom) |
| **1** | **3** | No, but I would be after a while | Left-handed | User feedback; cover the sensors with a bubble |
| **1** | **4** | Yes, tired in the wrist | Left-handed | Speed control; user feedback; smaller device |
| **1** | **5** | No, but I would be after a while | Right-handed | Include preset positions of the camera; design dual mode gesture-touch |
| **2** | **6** | Yes, tired in the arm and wrist | Right-handed | Speed control; smaller device |
| **2** | **7** | Yes, tired in the arm and shoulder | Right-handed | User feedback; smaller device |
| **2** | **8** | No, but I would be after a while | Right-handed | User feedback when the device is enabled |
| **2** | **9** | No, but I would be after a while | Right-handed | Point an object on screen so that the camera centers and zooms in on it |
| **2** | **10** | No, but I would be after a while | Right-handed | Speed control; more soft and precise movements |