Evaluating approaches to improve upon a Leap Motion-based hand-gesture recognition system

Stephen Ricardo Chase

MSc by Research (IRAC)

4/9/2020

Evaluating approaches to improve upon a Leap Motion-based hand-gesture recognition system

Author: Stephen Ricardo Chase

Supervision Team: David Jazani | Renxi Qui

Word Count: 13,259/20,000

A thesis submitted to the University of Bedfordshire, in fulfilment of the requirements for the degree of MSc by Research

Author's Declaration

“I, Stephen Chase, declare that this thesis and the work presented in it are my own and has been generated by me as the result of my own original research.

Evaluating approaches to improve upon a Leap Motion-based hand-gesture recognition system

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;

2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;

3. Where I have cited the published work of others, this is always clearly attributed;

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

5. I have acknowledged all main sources of help;

6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;

7. none of this work has been published before submission

Abstract

The research proposed in this thesis aims to utilise feedback gained from user testing to evaluate and present viable approaches to improve the self-made Marionette Project while maintaining its core principles. These improvements should primarily allow for users to identify hand gestures efficiently while minimising the number of incorrect hand gestures performed.

Analysing various studies throughout the period of the initial Marionette Projects conception and implementation. It was noted that research in human machine and human computer interaction focused on a few prevalent topics, the most common being sign language, and generic gesture recognition. The implementation of these predominate topics focused typically on one of two implementation types; vision or image-based approaches as seen in Cho et al.'s research which looked into implementing a low-cost vision-based gesture recognition system based on the FPGA approach (Cho et al. 2012) or the device-based approaches utilised in Khambaty et al.'s conference paper into "Cost Effective portable system for sign language gesture recognition".

While both studies look at implementing "low-cost" or "cost-effective" solutions they both tackle it from different standpoints. Khambaty et al.'s study looks at it from a monetary perspective presenting the developed system as a cheaper means of providing daily communication as opposed to the cost of hiring an interpreter (Khambaty et al. 2008), which while successful, still puts the product out of range of the general consumer. While Cho et al.'s study focuses on the reduction of computation costs by process the recognition of gestures through the use of the FPGA approach (Cho et al. 2012).

This focus on one section of "cost" lowering in implementations has left room for research that provides both a monetary reduction allowing for the implementation to be consumer-friendly and a computational reduction allowing for faster and quicker recognition while still maintain accuracy. Additionally, with a larger focus being placed on visual based implementation, but solely in the realms of sign language and generic gesture recognition as a means for human to human communication it provides a gap in the field to test the plausibility of these implementation types for other uses, like machine control.

As such, the initial Marionette Project aimed to find a cost-effective means of producing firstly, an effective but innovate hand gesture recognition system that could be utilised to control a range of robotics but in particular a mechatronic hand. The Leap Motion controller was implemented into the project, to test the viability of a low-cost consumer-grade product as a means to manipulate robotics. Utilising the Leap Motion controller also provided notable innovation as most published studies

incorporating the Leap Motion Controller focused almost exclusively on the identification of various forms of sign language.

In the thesis, three crucial feedback points garnered from the external testing process in the original Marionette Project, and are presented and utilised to shape the work implemented throughout this research in the form of minimum viable requirements listed below:

1) improving upon the accuracy of the hand gestures recognised by users through the use of real-time gesture confirmation system.
2) mitigating the amount of incorrect hand gestures performed when stopping the system.
3) Allowing for the support of more dexterous robotics though more complexed gestures

In the thesis four approaches main approaches are presented, the first of which looks into the first minimum viable required, while the second, third and fourth approaches are created and evaluated as a means to fulfill the second and third minimum requirement points.

By utilising the Spiral methodology, the implementation of each approach primarily followed the pattern of: Planning, Risk Analysis, Engineering and finally an Evaluation phase. The planning phase looked at the original feedback provided by the user as well as any relevant iterations implemented prior, to detail aims and requirements for the current iteration/spiral. After which, additional research was then carried out into hardware, software components as well as, additional published research papers.

After this stage, the implementation or engineering phase would then be carried out. This primarily would look to implement each identified requirement for the iteration. Once these requirements were implemented the evaluation step would then be performed. The evaluation process consisted of two parts; an internal evaluation and then an external evaluation.

Internal evaluations focused on developer testing, and consisted of standard logic, user flow and selected edge case testing. If no issues were found in this testing process the second stage of evaluations, external evaluations would then take place. External evaluations saw the implemented work tested by volunteer users under given scenarios, to generate more user results and feedback. However, in the event that problems were found during the internal evaluation process depending on the severity of the problem an additional mini-iteration could be added to an existing iteration as seen in section 4.3.4 or in the case were larger problems are identified and the approach needs to be altered an entirely new iteration with the listed spiral approach steps would be carried out as seen with approaches, three and four in the thesis.

From the approaches detailed in the thesis, the first approach provided a viable means to improve upon the accuracy of hand gestures recognised by users, though the incorporation of real-time textual confirmation appearing onscreen while the user interacts with the system. The incorporation of this implementation showed an average increase of 28% in gestures recognised and identified by the user. Additionally, the fourth approach detailed in the thesis provided a means to improve upon limiting the amount of incorrect gestures performed by the user showing an overall 20% decrease in the number of incorrect reported hand gestures.

It was concluded that the first approach presented an ideal implementation for clear improvements for the recognition of performed hand gestures and showed a 28% average increase in hand gesture recognised and identified by users. Additionally, the fourth approach was seemingly well-received by testing participants as a means to limit the number of incorrect gestures being performed when users removed their hand from the Leap Motions field of vision lowering reports of incorrect gesturing cause by the system by 20%.

However, while both approaches implemented within this research was viable for the completion of the first two feedback points provided by users from initial testing, the third point was not achieved. As such, it would be ideal to provide extra research to find a plausible and novel solution.

Table of Contents

Table of Tables

Table of Figures

1. Introduction

Due to the continuous growth in technology and its role in quotidian life, there have been vast improvements within the field of human-machine and human-computer interaction; HMI and HCI, respectively. Industries, for example, the video game industry, show a lot of these improvements with the introduction of hardware like the Xbox Kinect, which allows for full-body motion tracking, and the Nintendo Wii, which uses infrared technology alongside other technologies to create a more interactive experience when playing games.

These changes attempt to approach the interaction between users and games more naturally. With these continual advancements, traditional methods of interaction with keyboards, mice, and more recently apps, for example, to manipulate and control robots and other interfaces have begun to show their limitations in the form of cumbersome or overly complicated controls (Wang et al. 2016). As a result of this, HCI related work now endeavours to achieve a more natural and intuitive manner of interaction, similar to that of Human-human interaction.

For HCI research to achieve this standard of interaction, there has been an increase in studies exploring gesture recognition (Chen et al. 2015), to create a more natural means of interaction between humans and computers/machines. The Oxford Dictionary defines a gesture as “a movement of a part of the body, especially the hand or head to express an idea or meaning’ (Oxford Dictionary, 2002). Gestures themselves are thought to be spontaneous and produced without conscious awareness (Pine et al. 2004) this idea is best exemplified in representational gestures where a person usually subconsciously utilises gestures with the desire to enhance communication (Martha W. Alibali & Sotaro Kita. 2017).

To this end, it is evident that gestures are ingrained into human society, potentially making it one of the most natural input methods to utilise for HCI research. In solidarity with this conclusion, the research presented within this thesis will look into improving upon a previously self-built hand gesture recognition system, called the Marionette Project.

This research will specifically look at finding and evaluating approaches to improve the Marionette Projects implemented hand gesture recognition system to rectify some of the noted shortcomings discovered during testing and to add a varying level of complexity and innovation in an attempt to support complex interfaces while maintaining the projects core principles. The thesis will contain the following sections:

1. Aims

This section of the thesis will be used to outline the purpose and goal of the research.

2. Objectives

This section will identify a series of tasks and goals the project aimed to accomplish the completion of these deliverables will help determine the success of the project.

3. Literature Review

The literature review will contain two sections, the first of which will detail and evaluate areas of the previously developed Marionette Project. While the second section will draw a comparison between the Leap Motion Controller and other popular consumer-grade gesture recognition products on the market.

4. Work Plan

The work plan will delve into the Gantt Chart timeline, developed as a means to help manage the deadlines that came with this research and thesis.

5. Methodology

This section of the thesis will be used to detail the method implemented within this research, explaining the reasoning for the chosen method, risk analysis and management strategies crafted from past experiences and the current aims and objectives of the investigation, and finally any ethical considerations.

6. Approaches

This section will outline approaches derived from initial feedback and results from the Marionette Project, as well as additional investigations carried out into both the implemented hardware and software and published studies and research done within the field of Human-computer Interaction and Human-machine Interaction.

7. Conclusion

The conclusion will be used to evaluate the overall project, determining the project success or failure to achieve the desired aims and objectives laid out for it.

1.1 Aim

The proposed research aims to utilised feedback from previous tests to identify and implement viable approaches to improve upon the Marionette Project. These, the approaches should endeavour to allow for users to be able to identify performed hand gestures and for minimising the number of incorrect hand gestures performed when attempting to stopping the system. Additionally, the improvement also aims to support more dextrous robotics, preferably in an innovative way, while keeping to the core principles of the project.

1.2 Objectives

- Evaluate feedback from the previous testing carried out on the Marionette project to identify approaches to improve the system
- Evaluate the Leap Motions position in regards to being the primary input device for the project.
- Implement a method to provide feedback to users about currently performed hand gestures
- Implement a method to reduce the number of incorrect hand gestures performed when interacting with the system
- Find and implement a viable method to allow for support of more dextrous robotics

2. Literature Review

The literature review will consist of two sections: the primary section of the literature review will be used to provide insight into the previously self-developed Marionette Project. This primary section will provide an overview of the aims and objectives, implementation, results and conclusions of the project to demonstrate how those factors have helped to contribute to the proposed research.

The second section of the literature review will evaluate the current implemented consumer-grade hand gesture recognition device; the Leap Motion Controller against other available popular consumer-grade devices. This evaluation will look at both popular low-cost micro-electro-mechanical (MEMS) and computer vision consumer-grade products like the Myo Armband, Xbox Kinect.

2.1 Previously Implemented Work

2.1.1 Introduction

Common trends within human-computer and human-machine interaction (HCI and HMI) research during 2015-2016 focused on making interactions between humans, machines and or computers more natural. To this end, there was a significant focus on systems incorporating gesture recognition devices as a primary control source.

Most documented research and studies made use of computer vision and or microelectromechanical approaches. However, these implementations ordinarily employed more expensive gesture input devices or well-established consumer-grade products. Additionally, a vast majority of the developed system required detailed training manuals or development knowledge of the system to interact with fully.

The Marionette Project aimed to utilise trends in human-computer and human-machine interaction to develop a cost-effective and innovative gesture recognition system. While trying to ensure an easy interface for users to interact with that required little to no explanation.

2.1.2 Hardware and Software Analysis

1. Leap Motion Controller

Figure 1- Visualisation of the Leap Motion controllers' field of vision

The Leap Motion controller is a low-cost consumer-grade hand gesture recognition system, developed by Leap Motion in 2013 (Weichert et al. 2013). The Leap Motion controller is capable of capturing the user's forearm, wrist and hands without the use of additional markers; this is due to the two cameras and three infrared LEDs located inside of the controller that allow it to motion capture in both infrared and stereo vision (Smeragliuolo et al. 2016).

The Leap Motion is a 13mm x 13mm x 76mm rectangular device that weighs up to 45g (Smeragliuolo et al. 2016), which when compared to other devices, for instance, the Xbox Kinect which has a dimension of 279.4mm x 63.5mm x 38.1mm (for the main component) and an overall weight of 453g, is significantly smaller than other camera-based devices, however, due to the wide-angle lenses within the device it provides an interactive space of eight cubic feet, which takes the shape of an inverted pyramid as seen in figure 1 (Colgan. 2014).

The Leap Motion is capable of performing live feedback motion capture of both hands and can stream data at variable acquisition rates of up to 120 Hz (Smeragliuolo et al. 2016). At present, the devices USB controller reads the sensor data into its local memory and performs any necessary resolution adjustments before streaming it via USB to the tracking software or Image API (Colgan. 2014).

The Processing programming language was chosen to implement the various positional hand gestures needed for the Marionette projects completion; Processing allows for the interaction and manipulation of the Leap Motions output data through the use of various created libraries, this manipulation will be explained in-depth in the implementation section.

2. Arduino Microcontroller

The Arduino is an open-source electronics platform that was developed by the Ivrea Interaction Design Institute as a method to produce fast and easy prototypes. The developed boards are capable of accepting and reading various inputs, for example, 'light on a sensor, a finger on a button, or a twitter message' (Arduino. 2017). It can then turn the received input data into output data. These processes can all be performed through the use of the Arduino programming language (based on Java) and the Arduino software (IDE) (Arduino. 2017). Due to these factors, the Arduino as well as the Leap Motion controller which incorporates the Processing language to interact with its output data, as it was easy to implement interaction between both given hardware and software components' similarities.

The Arduino can be purchased pre-assembled for less than £40 and is capable of running across various operating systems; most notably Windows, Macintosh OSX and various forms of Linux, allowing for a range of potential implementations to be created using it.

2.1.3 Implementation

The accomplishment of the Marionette system came through the use of the Leap Motion Controller and Arduino microcontroller hardware components, as well as, Processing 3 which utilises the Java programming language and the Arduino IDE, employed to ensure the hardware components could communicate.

The core interaction with the hand gesture recognition system took the form of positional based hand gesturing. The developed hand gestures utilised the Processing IDE and focused on referencing the user's hand location against the Leap Motion controllers' field of vision shown in figure one; this reference came from the use of its own predefined Cartesian coordinate system. For example, if the user placed their hand in the middle of the controller and slowly moved it forward moving it on the z-axis, it would actuate the servos motors, rotating them forward at the same rate resulting in the radio-controlled (RC) car driving forward. This result occurs due to the user's movement resulting in their hand passing a particular threshold value taken from the Cartesian System produced from the interaction field of the Leap Motion, compared against the user's current hand location within the field of vision once the hand position has surpassed that of the pre-determined threshold the code is set to change the value of the serial variable and then pass it back to the Arduino IDE which would then update the information on the Arduino Uno board.

2.1.4 Results

Ten users were selected to test and evaluate the developed system. The sample size consisted of participants who possessed no technical knowledge in regards to gesture-controlled robotics, human-computer or machine interaction. Additionally, the selected users also had no interactions with the developed system before testing.

During the testing session, each participant was brought into a room individually and told the following information:

"The setup that is in front of you contains a Leap Motion Controller, that has been implemented to capture positional hand movements/gestures, which are utilised to affect the circuits attached to the computer, the circuit itself is designed to reflect the basic setup of radio-controlled (RC) car.

This session will be broken up into three sections; you will have two minutes to interact and play with the system while doing so you will be asked to keep a mental note of any hand gestures you believe you have found which I will record.

For the second section of the session, I will ask you to perform a series of tasks using the system, taking notes on how you interact and any issues you find during your interacting with the system. For your peace of mind, I will inform you in advance that the system does not make use of any physical feedback, so if you do perform an incorrect action that is okay.

Once completed, in the final section, you will have an additional 5 minutes to play and interact with the system while answering a few questions, should you choose you can also provide general feedback in this time as well."

After this explanation, an inquiry was made into if they had any questions before beginning the first section of testing, as explained during the "free play" section participants were left to their own devices and allowed to interact with the system as they deemed fit for two minutes. During this period, the tester also made a note of the number of hand gestures performed by each participant. After the first two-minute period, participants were then asked to feedback the hand gestures; they had found, as seen in Appendix B.

After this point, participants were then asked to perform a series of actions using the system which is listed below in table 1.

Table 1- List of actions for users to perform in original Marionette testing session

	Action / Task
1	Attempt to move the RC car forward
2	Attempt to move the RC car backwards
3	Attempt to move the RC car forward left
4	Attempt to move the RC car forward right
5	Attempt to move the RC car backwards left
6	Attempt to move the RC car backwards right
7	Attempt to stop the car from moving

Each participant had three attempts to complete a task if the user could not complete the task within three attempts; they were provided with an explanation and shown the correct hand gesture before moving on to the next task, the results of these test are in Appendix C.

After the completion of the testing process, participants had the opportunity to interact with the system while answering the following questions:

1. On a scale of 1 to 10, with one being bad and ten being good. How easy did you find the system to learn?
2. On a scale of 1 to 10, with one being bad and ten being good. How natural/comfortable did you find the system to use?
3. On a scale of 1 to 10, with one being bad and ten being good. How innovative did you feel the system is, compared to other products on the market?
4. Outside of the implementation using the Radio-Controlled car that you have seen, could you see this system utilised to control more complex robotics?

Additionally, participants were allowed to provide general feedback in regards to the system as well, the results and general feedback shown in Appendix D. Most notably 80% of participants experienced problems of the system performing incorrect gestures when attempting to stop the system from manipulating the radio-controlled car. Additionally, users also reported during the “free play” session that they were unaware of the start of a new hand gesture due to the similarities in movements from the servos.

2.1.5 Conclusions

The initial concept proposed within the Marionette project was implemented successfully and fulfilled all aspects of the desired functional and non-functional requirements identified throughout the studies duration.

From the result acquired through testing, users had difficulties differentiating between hand gestures when given no context of the gesture available to them in the system, from the results table shown in Appendix B participants only correctly reported three hand gestures during the “free play” exercise when they in fact successfully performed up to 5 hand gestures in total, which aids in exemplifying this point. Additionally, a key feedback point given by users was the request for additional confirmation about the currently performed hand gesture, as seen in Appendix D’s results tables section.

However, overall, the system does seem relatively accessible for new users when given context. From the results recorded in Appendix C, in the first task, 50% of participants managed to perform the correct hand gesture first time round, while 40% performed the hand gesture the second time around only 10% of participants required all three attempts to perform the correct hand gesture. For the second requested action, all participants managed to correctly identify the hand gesture within the first attempt, showing a 50% increase from the first task. When asked participants stated that when performing the second task, which required them to move the radio-controlled car backwards, they went with what would make the most sense given what they already knew and proceed to do so with the rest of the given tasks. These results show that the developed system itself presents a more natural means of interaction for the user; as once the participants saw the initial logic with a relatable context, they were able to correctly identify the rest of the hand gestures in the given tasks.

While these results present an excellent initial foundation for the project, during conversations during the final section of testing and general feedback from participants, shown in Appendix D. Participants voiced that while they did find the system easy to learn and utilise, 30% could not see the system utilised to control more complex robotics, and 60% voiced concerns due to the system causing the user to perform an incorrect gesture and their inability to differentiate between some of the implemented hand gestures. Of the 90% negative responses, 60% agreed that providing that there was a means to get confirmation of the currently performed hand gesture as well as, potentially extending the field of vision given by the controller or through the addition of more complex hand gestures, a second hand or another leap motion they could see the system being viable for more dextrous robotics systems.

2.2 Consumer-grade gesture recognition methods

A gesture recognition system can incorporate a multitude of methods to achieve the recognition, hand tracking and classification necessary for successful implementation; however, the two most prominent methods are the microelectromechanical (MEM), and computer vision approaches. Within the section, an outline of both approaches' methods, as well as an evaluation of their respective cost-effective consumer-grade input devices, will be provided.

The MEM approach incorporates the use of a 'MEM system', which is generically comprised of a set of sensors customarily built into chipboards which are then attached to a wristband or glove (Silva et al. 2016). As such, the user is typically required to wear a physical item in order for the gesture recognition and tracking to take place. The primary implementation methods within the microelectromechanical approach are the data glove and the Myo armband.

The computer vision approach takes its implementation method from standard practices within the computer vision field; which aims to provide computers with the means of extracting high-level understanding from digital images and videos (Huang 1996). The primary input devices implemented with this approach is the Xbox Kinect and the Leap Motion. Unlike the MEM approach, the user is ordinarily not required to wear any additional items when using this input method but is usually bound to a specific region to ensure that recognition and tracking can occur.

As noted within section 2.1.3, the Leap Motion controller is already the implemented input devices used to implement the current gesture control system; as a result of this, the following section will be used to evaluate the other viable input devices to ensure the Leap Motion is still fit for purpose.

2.2.1 Microelectromechanical Approach

The data glove is a sensory device made of a flexible material that contains a series of motion sensors such as; accelerometers, gyroscopes, bend sensors and force sensors (Kim et al. 2009). The glove allows for hand and finger movements to be measured, transferring the recorded values from the sensors to a computer system in the form of spatial and temporal information (Wang et al., 2016). Due to the use of sensors the MEM approach is barely affected by the surroundings of its implementation, allowing for it to be used in more complex environments (Kim et al., 2009) for instance in areas with low lighting. This advantage allows for accurate information to be generated from the glove; however, the implementation of this method in an everyday environment can be hard due to the cost of high-quality sensors.

The data glove method traditionally requires a connection between the user and the computer to transmit information; this connection is usually in the form of a cable. This physical connection creates one of the significant disadvantages of the data glove implementation as the cable can become a potential impediment to the freedom of the user (Raheja et al. 2010), resulting in an unnatural experience using the data glove. Newer studies, for example, Kim et al. and Zhang et al. 's research have incorporated Bluetooth/Wireless modules removing the necessity of a cable connection between the input device and the computer (Silva et al., 2016), thus taking away one of the major disadvantages with data glove use.

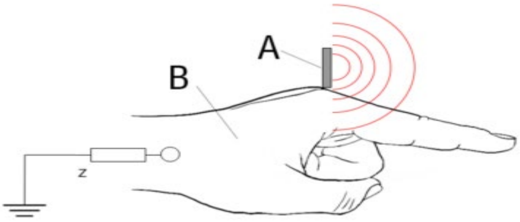


Figure 2 – Visualisation of the implemented used by Silva (Silva et al. 2016)

A more recent study using a data glove approach is Silva et al. 's (2016) study, which aimed to improve the ergonomics of the data glove, as stated from studies before this point the major ergonomic flaw within the data glove design is the cable used to transfer sensory information. To remove this flaw, Silva et al. proposed the use of a Theremin Sensor due to it being less sensitive to luminosity interferences and being cheaper than ultrasonic devices as well as a Bluetooth module to transfer data without the use of a physical cable (Silva et al. 2016). Their implementation made use of four mounted in a plastic case placed in section B, shown in figure 2.

With the primary aim of developing a more ergonomic design with minimal additional costs, the newly implemented version was fully functional, however, with the change in sensors resulting in the system function in a 2-dimensional space rather than a 3-dimensional space; the users' thumbs were unable to be tracked due to the sensor's inability to accurately record the additional degrees of freedom granted to the thumb through the metacarpal phalanx.

From this research, it can be generalised that in order to implement a more ergonomic design for the glove-based approach, perform would need to undergo certain inevitable sacrifices. In the given case, Silva et al. 's study opted not to record the user's thumb data, as their system mapped the flexion and extension of the fingers and their joint angles on a 2D plane (Silva et al. 2016). Furthermore, to reduce the level of interference; they would require more sensors around their outer case which would lead to the secondary ergonomic problem that sensors used in an embedded data glove can result in the glove being quite bulky and impeding the natural movements of the user (Schroder et al. 2012).

In order to correct the disadvantages posed with current MEM technologies like the data glove, the introduction of new technologies like the Myo armband saw release.

Figure 3 – Image showing the Myo armband (Abreu et al. 2016)

The Myo armband shown in figure 3 is an electronic device worn as a bracelet around different locations of the arm that allows for the detection and transformation of energy generated from muscles into computation information through the use of eight Electromyogram (EMG) sensors (Artal-Sevil & Montanes 2016). In Abreu et al. 's (2016) research that the Myo can by default, recognise five hand gestures, which can be used in conjunction with IMU data to control applications, as well as the raw timestamped EMG data which is captured from each sensor (Abreu et al. 2016). Additionally, the armband also contains an accelerometer which allows for the retrieval of inertial measurement unit (IMU) data, as well as a gyroscope and a magnetometer.

From the evaluations and research provided from Abreu et al. (2016), Cabreira & Hwang (2015) and Ploengpit (2016); as well as the timeframe in which their study and research was carried out, coupled with the fact the studies focus on the Myo armband as the primary MEM approach, it can be assumed that the Myo armband, is regarded as the superior implementation ranked above the data glove approach as a method for MEM technologies.

A comparison between the Myo armband and the data gloves implementation demonstrations that the Myo armband does improve the majority of the shortcomings found with the data glove approach, for example in terms of ergonomic design the Myo armband is vastly superior. For instance, the Myo armband is both lighter and more comfortable to wear; furthermore due to the sensors being built into the armband its cost to implement within research is significantly reduced compared to the data glove approach (Abreu et al. 2016). As such, the Myo armband presents the ideal step towards the more

natural interaction sort within the field of HCI research. However, while the Myo armband does improve previous MEM implementations it does come with its shortcomings, research carried out by Abreu et al. (2016) in which implemented the Myo armband to recognise 20 stationary letter gestures from the Brazilian Sign Language (LIBRAS) alphabet helps highlight some of these.
It was discovered that Abreu et al.'s. (2016), approach in the classification of new gestures through the use of the default captured EMG values taken from the Myo armband presented various problems. The initial problem was that an association between the values and the featured classification was not feasible because the captured values depended on the gesture made but also on the position of the Myo armband on the user's arms. Therefore to perform a different gesture, it is plausible that the Myo armband may need to be repositioned on the arm to ensure the values are coming from the correct muscle groups (Abreu et al. 2016).

The second problem found within the study was that EMG data could vary broadly from person to person, while there are many reasons for these variations, for example, the amount of fatty tissue, hair or sweat on the arm. It was noted that the only way to classify gestures made by individuals would be through the collection of vast amounts of training data (Abreu et al. 2016), which would result in a high computational cost.
Additional problems found within the research, was that due to the nature of the EMG readings Abreu et al.'s (2016) concluded that the data provided from the Myo armband would have issues defining subtle finger gestures, to this end it was determined that computer vision techniques or the data glove technique would be more beneficial (Abreu et al. 2016).

Due to these findings, a fair conclusion is that while the armband is generally superior to previous MEM technologies, it still needs improvement The current drawbacks noted within the technology are at present to substantial to consider it improve the proposed project, as such, a computer vision approach would be superior to the proposed MEM approaches.

2.2.2 Computer Vision Approaches

The Kinect sensor is a Microsoft product that contains a colour VGA video camera, which aids in facial recognition as well as the detection of other features through the use of three colour component detection, a depth sensor which is created through the use of an infrared (IR) projector and a monochrome CMOS sensor allowing the Kinect to have a 3D view of its given environment, and finally, a Multi-array microphone, which is comprised of four microphones to isolate the voice of the user from background noise within the given space (Crawford, 2010).

As highlighted within Ratul et al.'s (2016) study, the Kinect can be used to track up to 6 people, including the 20 skeleton joints of two people (Ratul et al. 2016). This feature allows for the Kinect sensor to identify and locate specific joints and limbs within the Kinect's field of vision, through the use of its depth and infrared cameras and the Kinect SDK software processes information drawn from each joints Cartesian coordinates procured from the camera.

However, while the Kinect does remove the fundamental issues that made the Myo armband unsuitable for use within the proposed study, by allowing the incorporation of the positional recognition system implemented within the Marionette project, there are still some limitations.

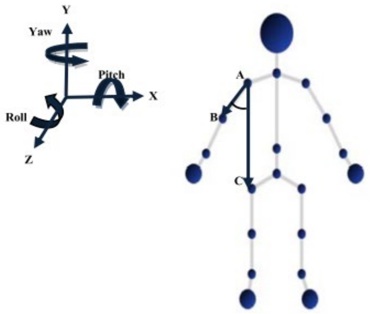


Figure 4 – An example of the skeletal structure used by the Kinect (Ratul et al. 2016)

The first of which can be identified through the use of figure 4, which shows the skeletal image gained by using the Kinect alongside the Kinect SDK software within Ratul et al.'s (2016) study. Within the image, we can see the identification of essential joints within the human body, for example, the shoulders, elbows and hips. With the data provided by the Kinect's skeletal tracking, it allows for the ability to generate more complex gestures based on the introduction of other limbs. However, the previous and proposed study still aims at present to focus on precise hand tracking, using the hand and fingers to define gestures. As such, the addition of additional limbs is redundant within this proposed study.

In addition to this, Cabreira & Hwang (2015) study shows that gestures involving fingers and hand movements are less commonly used in apps for Microsoft Kinect due to its current limitations on hand-tracking (Cabreira & Hwang 2015). Studies have included additional components like the colour marker gloves to offset the Kinects inability to identify precise hand and finger movements. However, while this does counter at the initial problem, complications like occlusion can still occur. Furthermore, a densely populated database would be required to ensure that postures and movements can be computed without temporarily losing control of the artefact (Schroder et al. 2012).

2.2.3 Conclusion

From the analysis presented above regarding the data glove and Myo armband MEM approaches

In conclusion from the analysis presented above regarding the data glove and Myo armband MEM approaches, while both implementation methods are viable in their own right, they are not fit for purpose in regards to the direction of the following core principles of the Marionette Project.:

- The development of a hand gesture (focused) recognition system, following trends within HCI research.
- The manipulation of a variety of artefacts, while still adhering to HCI trends, where possible.

This first bullet point can be split into two key areas:

1. The development of a hand gesture recognition system
2. The aim to follow current HCI trends

From the analysis above it can be noted that both implementation methods can be used to implement a hand gesture recognition system, however, due to the method of implementation used within the Marionette project, both methods are not suitable for use within the project, due in part to their inability to generically provide positional and orientation information (Schroder et al. 2012), which is the primary control method used within the Marionette project.

In the endeavour to seek out and follow trends within HCI research, a notable trend which has a clear correlation amongst researchers is the development of natural interactions between the user and the implemented input device; this is evident in Chen et al. 's and Artal-Sevil & Montanes's studies which took place in 2015 and 2016 respectively. Taking this trend into account both the data glove and Myo armband have significant drawbacks in achieving this sort of natural interaction, primarily due to the data glove being generically cumbersome and bulky, thus impeding the natural movements of the user (Schroder et al. 2012). Moreover, while the Myo armband, which as stated above corrects a large amount of issues presented by the data gloves typical implementation; mostly the ergonomic drawbacks, it still presents its issues owing to its incapability of providing the means of performing a variety of gestures without the potential of the removal or repositioning of the armband to give it access to the correct muscle groups for the correct data to be processed (Abreu et al. 2016). This potential of repositioning directs to concluding that while the Myo might be feasible in providing a natural experience for the control of rudimentary robots when it comes to the control of a variety of more complex artefacts, it will not be able to provide a natural control method, which goes against another aim present within the Marionette project. As such, the use of MEM technology will not be viable to use to improve upon the research already presented within the Marionette project.

Moving onto the Kinect sensor, in regards to the first point, the Kinect sensor can be used to recognise gestures, due to that being the primary use for the system outside of personalised or custom project develops. Additionally, the Kinect also follows the primary trend of HCI research by allowing the user to perform gestures without the addition of any other device. Moreover, the Kinect sensor also removes the limitations identified by both the Myo armband and the data glove approaches due to it providing positional data through the use of its depth camera.

As such, when compared to the previous MEM implementation methods, the Kinect has clear advantages. However, when compared to the Leap Motion, there are several areas that the Kinect fails to achieve, these areas are as follows:

Firstly, due the Kinect sensors ability to both detect and recognise the users whole body as seen within figure 4 (above) to perform the recognition of areas like the users hands, which is the primary focus with the Marionette system, the researcher is required to extract the required information through the use of colour extraction or other feature extraction methods, as such the implementation of the Kinect would require a higher computational cost as opposed to the Leap Motion-based implementation (Marin et al. 2014), as the Leap Motion is designed to target hand gesture recognition. Secondly, while the Kinect does provide a more substantial amount of information due to the limited information produced from the Leap Motion it generically provides more accurate data which is processed and analysed by the Controller itself, finally it was also noted that the Kinect sensor fails to identify subtle motor movements that can be performed by the human hand (Moldovan et al. 2017), this aspect is a fundamental aspect of the Marionette project which is unachievable by the Kinect sensor unless additional items such as a colour marker glove is worn by the user, which would defeat the purpose of implementing the Kinect sensor rather than a MEM approach.

To conclude the Leap Motion controller was still the ideal hand gesture input device for the proposed research due to it accommodating all the aspects required for the project while remaining cost-effective.

3. Project Plan

The project plan will consist of three sections; the primary section will outline the methodology utilised in the proposed research as well as any additional details and features surrounding the methodology. The second section will explore the key ethical considerations to adhere to within engineering research, evaluating and solving these problems were required to ensure the project remains up to standard.

Finally, the work plan will provide a detailed project flow estimation in the form of the Gantt chart which will be used to detail key points within the research and how they will all be managed to complete the project within the predefined timeframes.

3.1 Methodology

This section will outline the method implemented in the research to achieve the proposed aims and objectives.

3.1.1 Chosen Methodology

The methodology implemented in this research was the Spiral methodology. The spiral method presents an iterative approach to the development process, emphasising risk analysis, prototyping, and documentation. Through the use of multiple “spirals” or iterations, the model allows for each new iteration to build upon the findings of the previous spiral to update objectives and aims based on testing which is a critical aspect of engineering research.

While generically utilised in more extensive projects, the Spiral methodology can still be viable in smaller and shorter projects, additionally, in terms of engineering research the spiral model sees more use than other methods like the waterfall model, v-model due to their inflexibility which leaves little room for requirement or conceptual changes (istqbexamcertification, 2017)

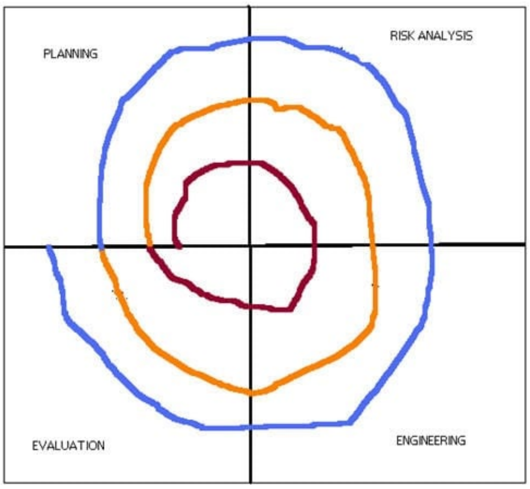


Figure 5 - Pictorial Representation of SDLC Spiral Model. (softwaretestinghelp. 2019)

As seen in figure 5, there are four stages to the spiral method. The planning phase is the starting point. Within this phase, research helps to develop the initial idea, specifications, and objectives for the sprint. The second phase is the risk analysis where the specifications are then analysed to identify any potential risks that could occur as a result of the specifications; upon completion of this identification process, ideally, risk mitigation should be put in place to offset as many of the identified risks as possible.

The third phase is the engineering phase, which then looks at the design and implementation process to achieve the specifications set for the sprint; as well as any additional testing. Finally, the fourth phase is evaluative, were the implementation and results are compared against the specification to determine if the various specification points have been achieved, depending on the outcome of this evaluation depends on if any more iterations need to take place in the research.

3.1.2 Risk Management

As stated, the Spiral methodology places a significant focus on risk analysis. As such, the proposed research identifies risks during each iteration of the concept development states. Alongside this, however, a risk analysis will also be carried out on the project, on the whole, to identify and mitigate any substantial risks that could affect the scope of the project.

The main risks identified from this analysis were: Time management and scope creep. Time management refers to the allocation of time for activities throughout the project (Wrike, 2019). Its reason for being listed amongst potential project-wide risks comes from the nature of the research aiming to be carried out. Due to a large amount of analytical research required in order to produce the concepts presented in the research, there is a necessity for equilibrium between the time spent on implementation and analytical analysis. If this goes unachieved, it could result in a large amount of conceptual material with little, to no testing, which would result in a lack of validity in the researches conclusions.

On the other hand, should prototypes contain a lack of insight into prevailing research, the proposed concepts could conceivably present a solution that has known problems highlighted in other HCI research. As such, while the solution may improve the Marionette project at face value, it would then produce further work in the future to answer the new problem. The implementation of a work plan should help mitigate this risk, as it will help present an estimation on the scope of the research as well as highlight predetermined critical deadlines like the Mini Progress Point and course completion date that also need consideration.

Scoop creep, also referred to as requirement creep or feature creep; refers to a general increase in requirements across a project's duration (Wrike, 2019). Scope creep was a potential risk to the project-wide scope once again due to the nature of the research. As explained above, the research makes use of analytical analysis of existing projects and concepts to help make informed decisions about the approaches investigated throughout the project duration. As such, it is easy for the project's requirements to be altered and shifted to overcome either known issues within the field or to gain more innovation.

To counter this risk within the research will aim to construct requirements for each iteration using a minimum viable product approach (MVP). While the developed system has no intention of being released to the general public at this stage, using MVP allows for an explicit restriction between what needs to be delivered verse, what would be nice to deliver.

3.2 Ethical Considerations

The research detailed in this thesis focuses on the field of human-computer interaction (HCI). As described by John M. Carroll, HCI research is primarily comprised of computer science embracing cognitive science and human factors engineering (John M. Carroll et al. 2014). As such, the research presented within the thesis must remain in line with the base ethical considerations presented in the field of engineering and robotics research.

As discussed by Philip Brey & Philip Jansen in their paper titled, ‘Ethics Assessment in Different Fields – Engineering Sciences.’ In engineering, there are two categories of ethics to be taken into consideration when looking at the impact of technological improvements.

These considerations are hard impacts, which looks at the physical effects on the environment, health and safety (Philip Brey & Philip Jansen. 2015) and soft impacts which looks at social realities and ideals such as justice, equality and individual rights (Philip Brey & Philip Jansen. 2015). Additionally, there are also more tailored ethical issues detailed for ‘emerging’ technologies, which include consideration on potential military applications and safety.

The remainder of this section will be used to outline and discuss the various ethical considerations that are appropriate for the research while providing an evaluation about how each issue if required, is addressed.

3.2.1 Safety

When discussing safety in terms of technological advances, according to Philip Brey & Philip Jansen, the main concerns are derived from the potential damage that could occur, be it in the form of physical injury and death to economic, social or political loss (Philip Brey & Philip Jansen. 2015). While there are several points under this ethical concern, safety in regards to physical injury or death is the main point that applies to the proposed research. The main reasoning for this lies in the goals of the project itself, which aims for the developed system to be able to control primarily, a mechatronic hand but also a range of unspecified robotics.

Currently, the presently developed Marionette System utilises no haptic feedback to provide information to the user to this end from interacting and controlling the system the user is in no immediate harm. Additionally, the user is also not physically attached to any part of the current system. So, if for any reason, they would need to evacuate the area of use, they would not need to spend time being disconnected from anything.

On the other hand, while the user is not required to be physically attached to the system, taking into consideration that a goal for the system is to be potentially compatible and deployable with a range of robotics. The current shortcomings identified in the Marionette project could potentially cause harm depending on the type of robotics under the users' control, should they attempt to move away with haste.

In order to tackle the potential concerns about safety, the proposed research intends to find a means to add more complexity to the current positional hand gesture system used within the Marionette Project. This complexity will allow for separation between the commands required to manipulate a given robotic artefact.

3.2.2 Dual Use

In regards to ethical considerations, the term 'dual-use' covers the possibility where a technological product can provide a secondary use, outside of its intended purpose, be it for either 'good' or 'evil.' This outcome is most common in civilian technologies that can additionally be used for military purposes (Philip Brey & Philip Jansen. 2015).

This consideration, while not precisely relevant at this stage of the research, due to the systems current implementations optimisation for a radio-controlled car within the Marionette Project, has not been issued an appropriate objective within the proposed research. It has been highlighted, however as it is potentially something that will need to be heavily considered depending on how far the research into this project goes.

3.2.3 Conclusions

From the analysis into the vital ethical areas in engineering, that was deemed appropriate in regards to the proposed research, the implementation of one objective was deemed necessary in order to counteract one potential ethical concern with the evaluated of the research.

3.3 Work Plan

This section of the thesis will detail how the proposed research was systematised to ensure the completion of the project, in line with the given predefined deadlines. There will be an explanation given for crucial points and predefined deadlines in the project to detail their importance and how those points alongside the methodology implemented in the research helped shape the estimated workflow.

3.3.1 Gantt Chart

A Gantt Chart is a project management tool which utilises a timeline to illustrate how a given project will run (Brett Anderson, 2018). The Gantt Chart provides an estimation for the project as a whole but also allows for the tracking of individual tasks or subtasks to help identify their scope within the project (Brett Anderson, 2018).

The Gantts Chart developed for the proposed study, provided under Appendix K, details various points of the project in line with the predefined completion date set for October 2017. With essential tasks, in the Gantt Chart presented in both with various subtasks provided underneath. The rest of this section will be used to highlight a few selected vital tasks that helped shape the general structure of the project's workflow, for instance, the Mini Progress Point (MiPP).

The MiPP is one of two predefined tasks allocated within the proposed research and is required to be submitted six months into the course. The submission revolves around two principal sections, the first of which is a report which needs to contain a literature review, work plan and any ethical considerations. The latter part of this submission takes place in the form of a presentation, where the information submitted within the report and initial concepts require in-depth clarification and evaluation.

4. Approaches

This section will provide various approaches to improve the previously developed Marionette projects hand gesture recognition system through the evaluation of user feedback from initial testing. Each feedback point will be evaluated and prioritised before then be used to fuel an investigation into how the implementation of the respective feedback point can improve the system from both a performance and, or an innovative standpoint; each implemented iteration will undergo either only developmental/internal evaluation and testing or both developmental/internal and external user testing to determine if its implementation has provided a positive effect in compared to the previously recorded results.

4.1 User Feedback Evaluation

In this section, an initial evaluation of user feedback against the core principles of the Marionette project and the currently proposed project will be detailed to identify the minimum viable feedback points that can be implemented to improve the system.

The core principles of the proposed study are as follows:

- To follow trends in current Human-computer and machine interaction research, HCI and HMI respectively
- To identify approaches to improve the previously developed Marionette project's system.
- To help facilitate a means to control more complex robotics

The first listed core principle revolves around making the interactions between humans and machines or computers innovative, while still feeling natural to the user, while the second principle revolves around identifying clear-cut improvements that when implemented will cause little to no additional problems or complications once implemented.

From these points, the user feedback listed in Appendix D was separated into two sections: Practical improvements and innovative improvements, as seen below in table 2.

Table 2 - user feedback showing practical and innovative improvements

	Practical Improvements
1	Visual confirmation, for current hand gesture performed by a user
	Innovative Improvements
1	Addition of more complex hand gestures
2	Add a second leap motion controller

From the table above, the feedback listed under practical improvements were determined to be the minimum viable ideas that could be implemented to improve the previously developed Marionette system. As such, the first iteration(s) in the implementation stage will be used to implement and test that initial point to see if it can provide definite improvement to certain aspects of the system.

In regards to the feedback listed under innovative improvements; the most exciting prospect would be the introduction of a second leap motion controller. However, while this would be the most innovative approach, it could be redundant if the incorporation of more complex hand gestures provides no drawbacks. As such, determining if the use of two leap motions is required would be preferable regardless of the amount of innovation it could bring.

4.2 Approach One – Gesture Confirmation Feedback

4.2.1 Planning

This initial approach looks into the first point listed within the practical implementation, which details the need for visual representation and confirmation of hand gestures. This feedback originated from the ‘free play’ testing users were asked to perform before they were asked to perform specific tasks using the system.

The ‘free play’ section asked participants to interact with the system while physically or mentally taking note of any implemented hand gestures they believe they found during their time interacting with the system. As shown within the results table shown in Appendix B, on average participants identified three hand gestures during the ‘free play’ exercise. However, the tester noted that on average participants correctly performed at least five hand gestures during their time interacting with their system.

Taking both the participant feedback and tester observations into account, it does seem that users were not aware that the actions they were performing resulting in new functionality presumably due to the similarity of actuation from the servo motors. The introduction of this hand gesture feedback system should provide users with more details about how they are directly affecting the system as well as give them confirmation for any hand gestures they are performing as well as the real-time data visualisation of their hand on the screen.

4.2.2 Risk Analysis

The primary risk identified within this iteration comes from the presentation of information to the user. The user at present already is shown a real-time data visualisation of their hand while interacting with the system as the user moves their hand over the controller the hand on the screen will also move, as such it could be a problem to find an ideal spot to place this new hand gesture confirmation information, so it does not overlap with the current data visualisation.

To avoid overlapping with the hand visualisation having text appear in different places depending on the hand gesture performed would be a definite option, however, for new users, this could be quite jarring as it would lead to an inconsistent experience as well as potentially cause distractions from the task at hand. An alternative method to solve this potential risk could be to find a position on the screen that the user's hand should ideally never be in to provide the required hand gesture confirmation updates.

4.2.3 Implementation

Through additional research carried out into the processing3 IDE, text can appear on the screen through the use of a built-in library. The fully implemented code shown in Appendix F declares and implements this new text instance within the conditional statements already found within the code. So, when one of the conditional statements return true text detailing the performed hand gesture also appears.

4.2.4 Testing and Evaluation

The free play session was recreated to provide an evaluation for the work implemented in this iteration. Ten participants were chosen to test the work implemented in this iteration, identical to the initial testing process the chosen users had no previous experience with the original or improved system and were all brought into a room individually to test the new system.

At the start of the session, each participant had the following information explained to them:

"The setup that is in front of you contains a Leap Motion Controller, that captures your positional hand movements/gestures and affects the circuits attached to the computer, the circuit itself is designed to reflect the basic setup of radio-controlled (RC) car.

In this session, I will give you will have two minutes to interact and play with the system. While doing so, I will ask you to keep a mental or physical note of any gestures you believe you find and record them afterwards."

With this iteration of the system, participants correctly identified an average of five out of the seven potential hand gestures in the two-minute timeframe. In comparison to the three out of seven hand gestures identified by users during the original Marionette external testing process. Converting these scores into percentages to allow for a numerical analysis, five out of seven would become 71%, and three out of seven would become 43% rounding to the nearest whole number. With these values we can see that there is a 28% increase in the amount of gestures correctly identified with the implementation of the gesture confirmation system.

This percentage increase in correctly identified gestures allows for the assumption that the implementation presented in this iteration was successful in enabling participants to identify and confirm how their actions directly affect the system in real-time. Additionally, due to the implementation of visual confirmation in the system it was noted that users were more accurately able to determine the outer bounds of the controller's field of vision, as well as allowing them to determine the minimum average distance they were required to move their hands for each new action.

From these concluded points, the work implemented in this iteration provided clear improvement which makes it a worthwhile additional implementation into future iterations of the Marionette system. Moreover, for future iterations of the project potentially outside of the current presented research the identification and subsequent implementation of visual confirmation in other areas of the system could potentially be utilised to see a further increase in performance.

4.3 Approach Two – Addition of Complex Gestures

4.3.1 Planning

The second approach is the second implementation iteration done within the proposed project and looked into the implementation of more complexed hand gestures into the previously developed system. The hand gesture recommended by participants was the circle gesture. Upon the successful implementation of this gesture, it could be utilised to switch the mode of control, resulting in a change of hand gestures available to the user. From further research into both the Leap Motion controller and Processing3 IDE, a pre-built class named "Gesture" allowed for the incorporation of specific movement patterns to be identified by the Leap Motion Controller (Leap Motion. 2017).

Additional investigations into the Processing IDE provided insight into various available tutorials available on the official processing website. One such tutorial written by Casey Reas and Ben Fry covered the fundamentals of interactively within processing programs. The tutorial focused mainly on the introduction of mouse inputs as a means to implement interactivity into a developed program, utilising mouse clicks to drawn or change the colour of various objects. In order to achieve this implementation, the tutorial made use of nested conditional statements which would change the colour of a rectangle depending on the interaction inputted by the user.

4.3.2 Implementation

The initial investigation provided insight into a means to provide additional interactivity into the Marionette system, and the initial iteration looked to implement a variation of the investigated tutorial to test the viability of the circle gestures implementation.

In order to achieve this implementation, the introduction of several changes to the original tutorial code seen in appendix G came about. These changes, shown in Appendix H, shows the initial introduction and initialisation of a new variable using the statement "leap = new LeapMotion(this). allowGesture ();". In addition to the method ". allowGesture ()" which is also passed during the instantiation, allowing the project to recognise and utilise the class.

Presumably due to copyright the full algorithm used to calculate and recognise the gesture is not detailed within the tutorial, provided by the Leap Motion company or Darius Morawiec; however, it is assumed that the data identified and recorded by the code, for example, the finger the user uses, the direction, radius and progression of the finger as well as the duration the finger is in the air and the time it takes to complete its movements provide values used within the algorithm to identify and associate a ID for the given gesture.

4.3.3 Evaluation

While the hand gesture recommended by participants are not innately innovative and would not be unique if implemented in the Marionette system, it is still considered sophisticated in comparison to the currently implemented positional based system and aimed to add a method to support depth in controlling specific robotics.

Testing the code implemented in this iteration found that the implemented code did not perform as expected. While the code did allow for the recognition of gestures through the use of the built-in class, it did not allow for the persistent change of the colour of the rectangle which was the base aim for the iteration's implementation.

Upon further inspection and testing the deduction was made that the logic error presented in the implementation was the result of the "draw ()" functionality in processing projects. The draw () function allows for lines of code to be continuously executed, acting as a looping mechanism to update the screen (Fry and Reas. 2017 (2)). To achieve aspects of the implement marionette project, the "draw ()" function was implemented and used to reset both the background and redraw the user's hand position, which allows for the user to see their hand and gesture confirmation text in real-time. To that end, the removal of the "draw ()" block is not a viable option instead a miniature sprint was created to find a potential alternative solution.

4.3.4 Approach Two – Second Iteration

The iteration in this miniature sprint aimed to manipulate the functionality of the "draw ()" method without removing it entirely. Through additional investigation, the implementation of the method "noLoop ()" and "redraw ()" aimed to stop the continuous execution of the code in the "draw ()" method and instead redraw or update the screen when required. From various testing and implementations, this solution was deemed not viable as the implementation of "noLoop ()" and "redraw ()" causes a noticeable delay in the data visualisation of the user's hand on the screen compared against the user's actual movements.

The use of threading was also reviewed as a means to rectify the problems found in Approach two. Threading allows for various sequences of code to be run in parallel alongside the main thread (the animation thread), which is created via the setup () and the draw () functions in Processing3. The idea behind the incorporation of the thread method was to attempt to create a constant variable in the background once a hand gesture is performed within the code; this constant variable would run independently of the animation thread which aimed to make it exempt from the updates caused from the draw() function.

The implementation of this idea proved rather challenging due to several factors, primarily the first challenge was the fact that due to new threads being independent of the animation thread, it could not directly affect variables within the draw() function, as such it was devised that additional variables needed to be created to confirm the start and completion of a threaded task that could be checked within the animation thread and then used to update the draw() function. The second issue was the limited resources available for the implementation of the project. It was noted due to the computation time and resource usage of multiple threading that would be required in order to reach the level of complexity required to complete the project. Moreover, it was also noted by Fry and Reas (2017) that due to the code not being synchronised to the animation thread, its threading could cause strange and or inconsistent results (Fry & Reas. 2017 (3)).

As a result of both the main and mini iteration failing to fulfil the desired outcome, external testing was not carried out, and it was determined that another iteration utilising a new approach would need to be developed to provide a viable solution.

4.4 Approach Three – Introduction of Second Hand

4.4.1 Planning

From the evaluation of parts of the second approach, implementing the standard circle gesture to initiate a swap in gestures or control modes was deemed unfit; as the single performance of the gesture did not result in a persistent change due to the implemented "draw" method which is employed to provide real-time hand visualisations and gesture confirmation text.

As such, the third approach presented in this thesis looks into the first point listed under the innovative improvements section of table 2, which is the implementation of both hands to control the system. The implementation of this should allow for the incorporation of a persistent variable that can remain in the Leap Motion Controllers field of vision, resulting in an approach that could rectify the problems found in the second iteration, making it a plausible solution to create varying states to either restrict the hand gestures that are allowed unless specific actions occur. Additionally, this implementation will allow for an evaluation into firstly, how innovative the incorporation of two hands is in regards to Leap Motion controller developed systems secondly, if a second leap motion could potentially be a practical suggestion to implement to improve the Marionette system.

From additional investigations, the incorporation of two hands when utilising the Leap Motion controller is unexpectedly novel as various studies, for example, Cavalcanti, De Medeiros & Dantas's research into the evaluation of the leap motion controller for multiple hand posture recognition (Cavalcanti et al. 2017), Or Li, Hsieh, Lin & Chu's research looking into "Hand gesture recognition for post-stroke rehabilitation using the leap motion" (Li et al. 2017), alongside additional research papers that utilise the Leap Motion controller focused on implementations that utilise one of the users hands; even though the Leap Motion has been designed and clearly shown to be able to identify and track both of the users hands.

This point is further exemplified within research that makes use of the Leap Motion to recognise Sign Language, the vast of majority of which focus on signing styles or aspects that only require the user to use one hand, as demonstrated in, Chuan, Regina & Guardino's research into American sign language recognition using the Leap Motion (Chuan et al. 2014) and various other studies that make use of the Arabic alphabet using sign language or Japanese fingerspelling method as opposed to the UK sign language alphabet. At present, only a handful of studies have been carried out using Indian Sign Language, which alongside the Sign Language used within the UK uses both hands to portray the alphabet. As such, from an innovation standpoint, the third approach presented is ideal should a viable means of implementing it be found.

4.4.2 Implementation

This iteration of implementation looks at introducing the users second hand, shown in figure 6, as a means of implementing a natural and easily accessible constant variable. It was decided that the implantation would once again utilise the code found within the interactivity tutorial seen in Appendix G. This code was chosen due to allowing for a secure method of developmental testing. As it provides clear and easy results through the colour transitions of the drawn square.

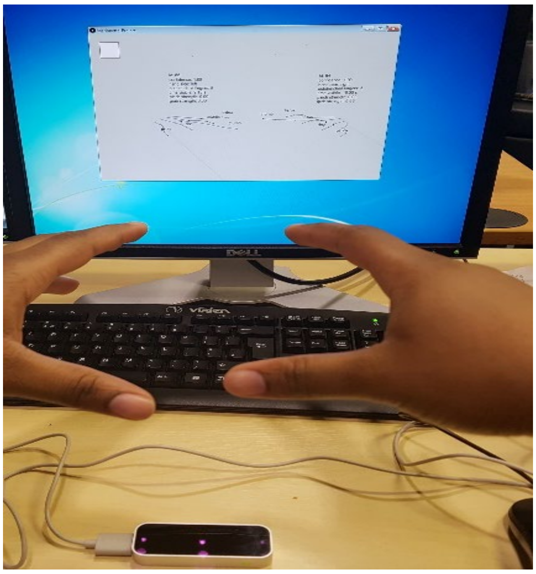


Figure 6 - Leap motion controller recognising both hands

The primary goal was to recognise both of the user's hands when in the controller's field of vision; this was achieved by using the code seen in Appendix I. Through the declaration on two new Boolean variables: handIsLeft and handIsRight, the implementation of conditional statements to dynamically change the fill (colour) of the drawn rectangle was implemented. This change was dependant on the hand presented in the Leap Motions field of vision.

4.4.3 Testing and Evaluation

The initial testing of the implementation which focused on presenting each hand individually to the view controller was found to be successful, as shown in figure 7 and 8 the fill of the square changed dependent on the hand presented in the Leap Motions field of vision. However, through further testing, it was noted that when both hands were presented one after the other and maintained in the field of vision of the Leap Motion, the hand presented first could no longer manipulate the colour of the square even though the Leap Motion still processed and tracked the position and actions of the hand.

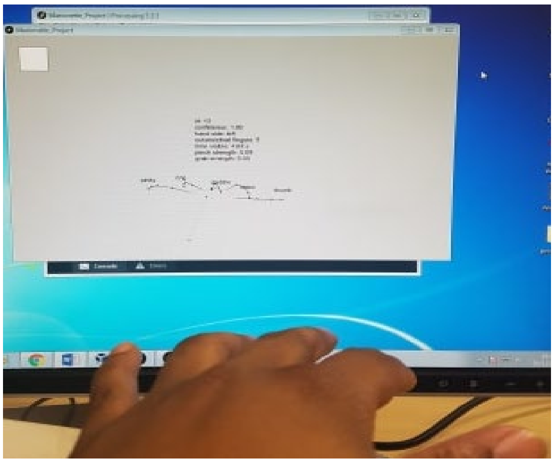


Figure 7 - Left hand presented to Leap Motion controller

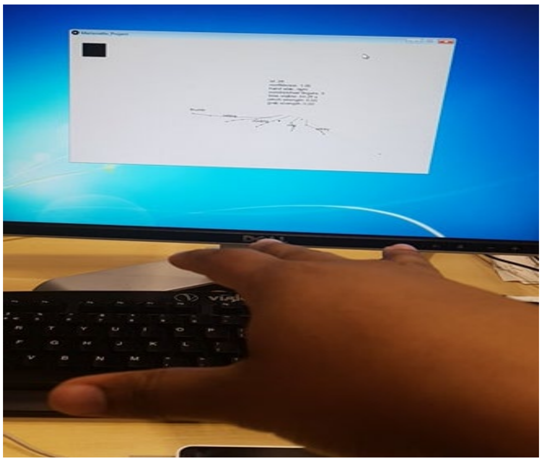


Figure 8 - Right hand presented to Leap Motion controller

Additional testing was carried out to learn the depth of prioritisation given to the hand most recently presented to the Leap Motion, to do so the addition of the pinch gesture was also added in the coding implementation, also shown in Appendix I. The introduction of the pinch gesture was to enable a secondary persistent gesture that could be performed by both hands to attempt to regain the priority of the first hand. Through varying test cases it was found that priority could not be given to a hand that was initially presented to the Leap Motion unless the hand was actively removed from its field of vision, it was determined that this was due to the Boolean variables that were introduced in the iteration, while the controller recognises both hands, only a singular variable is returned to determine which hand in currently ‘present' in the controllers field of vision, as such any conditional checks are done within the implemented code results in a failure if the hand is not currently prioritised.

Once again, due to the implementation not passing the initial internal evaluation phase, external testing was not and it was determined that another iteration utilising a new approach would need to be developed to provide a viable solution.

4.5 Approach Four – Implementation of Pinch Gesture

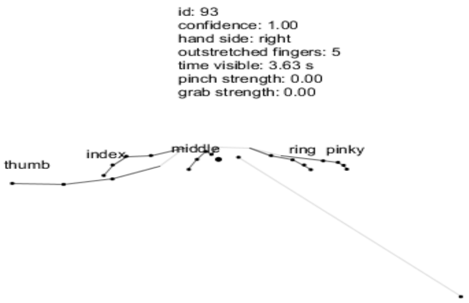
4.5.1 Planning

From the evaluation of the third approach, the introduction of the users second hand as a means to provide a constant variable to determine the activation or change of mode for the current system was not viable. However, from the extended testing done to determine this conclusion, the pinch gesture was implemented.

The pinch gesture originates from the same built-in Leap Motion library that in Approach two, was utilised to implement the circle gesture. During the evaluation of the third approach, it was determined that the pinch gesture could be persisted by the user to allow the constant change of a variable on screen. The fourth approach aims to investigate the viability of the pinch gesture as a means to activate or deactivate the system, to limit the incorrect hand gestures caused when users attempt to stop manipulating the system.

4.5.2 Implementation

The code implemented within this approach focused on the introduction of the pinch gesture to allow the user to stop the system. This implementation aimed to allow for the user to remove their hand from the Leap Motions field of view.



```
id: 93
confidence: 1.00
hand side: right
outstretched fingers: 5
time visible: 3.63 s
pinch strength: 0.00
grab strength: 0.00
```

Figure 9 - Hand visualisation from Marionette System

As seen in Figure 9, the initial data provided the first implemented code provided in Appendix A displays a range of information about the user’s hand while presented to the Leap Motion. From the list shown in Figure 9, the pinch strength is among the list of information recorded by the controller. Thanks to this while the pinch gesture could be implemented through the use of the same gesture library, we can instead use the results recorded directly from the Leap Motion as seen in Appendix I’s implementation.

The pinch strength value ranges from 0 to 1, which refers to the user having their hand spread open as seen in Figure 9, and the user touching their thumb and little finger together. The implementation shown in Appendix J focused on allowing the base six hand gestures: moving forward, forward right and forward left as well as moving backwards, backwards right and backwards left to be enabled if the pinch strength recorded by the user's hand was below the maximum threshold of one. This threshold was decided so that the user would be allowed to have their hand in various states of closeness without affecting the system, and only when the user actively produced the hand gesture should the system stop.

4.5.3 Testing and Evaluation

Tests carried out on the fourth approaches implementation contained a total of 10 participants, six of which were participants that took part in the initial testing and evaluation of the Marionette project. The primary focus of this testing session was to allow users to freely interact with the system while reporting any incorrect gestures that were performed due to the system.

At the start of the testing session participants were told the following:

"In front of you is a new iteration to the Marionette system; this iteration of the system has undergone a few minor tweaks to enable you to start and stop the manipulation of circuitry more dynamically. For your peace of mind, I will inform those of you that are testing this system for the first time that there is no physical feedback implemented in the system, so an incorrect or the wrong gesture is performed there is no need to be concerned for your safety.

This testing session will start with an introduction into the new functionality of the system before you will each individually be given a chance to interact with it on your own during this any feedback you provide will be recorded to aid in the evaluation of the new system."

Table 3 shows feedback that was mentioned by multiple participants throughout their interactions with the system.

Table 3 - Approach Four, user feedback result table

	User Feedback	Number of participants who mentioned this point
1	The pinch gesture is somewhat uncomfortable and frustrating to perform for an extended period	4
2	When stopping the system using the pinch gesture, incorrect gestures would be performed	6

The main feedback from participants revolved around the use of the newly implemented pinch gesture, on average, four out of ten (40%) of users found the pinch gesture uncomfortable when being performed over a long period. While this point presents a negative, it should be noted that only 40%, shared this opinion in regards to the newly implemented gestures. From this deduction, we can establish that from the feedback provided that on average if we were to ask participants the second question shown in the first table listing in Appendix D, the implemented approach would likely maintain its five-point rating.

Additionally, six out of the ten participants (60%) reported that while performing the pinch gesture, additional gestures were still allowed to be performed, this supplementary observations found that while the pinch gesture would be performed correctly by participants, the Leap Motion would not always accurately evaluate the gesture resulting in the pinch strength being below the required threshold value of one. While this is a problem that requires rectifying, when comparing the new testing results to that gathered from the original tests shown in Appendix D table 1 showed that eight out of the ten participants (80%) initially complained. As such there was a 20% decrease in the overall average reports of incorrect gestures being performed.

5. Conclusions

The research presented in this thesis managed to accomplish its primary aim of utilising user-provided feedback to fuel approaches to improve upon the self-developed Marionette system. The first approach presented an ideal implementation for clear improvements for the recognition of performed hand gestures and showed a 28% average increase in hand gesture recognised and identified by users. Additionally, the fourth approach was seemingly well-received by testing participants as a means to limit the number of incorrect gestures being performed when users removed their hand from the Leap Motions field of vision lowering reports of incorrect gesturing cause by the system by 20%.

However, while the overall aim was successfully achieved, some of the listed objectives found in section 1.2 were not. This main objective was the final point that looked to implement a method to allow for more complex controls. Initially, both implementations in approach two and three looked to implement a solution that could provide a reduction in the number of incorrect gestures performed by the user while also showing a viable means of additional hand gestures or functionality that could be implemented to provide additional options for sophisticated controls through the introduction of persistent variables to promote real-time mode changes within the system.

Due to the failure to implement both the second and third approaches, the fourth approach was implemented, which utilised the use of the pinch strength of the user as the backbone of the pinch gesture. As mentioned, the implemented solution did, overall provide positive results. However, the implementation lacked novelty and also presents no avenues for expansion; as a result, the implementation does not make for a viable means to support more dexterous controls for a more complex system.

The next stage for this research, taking into account the current findings, would be to look into the final feedback point provided by testing participants which would look at the introduction of a second Leap Motion controller. From initial research into the plausibility and novelty of a dual Leap Motion implementation, a singular published study was found that correlated to the proposed research. Mohandes et al.'s research incorporates the use of two Leap Motion controllers; it is explained that the second Leap Motion is placed perpendicular (at a 90° angle) to the first as a means to prevent the occurrence of occlusion while performing gestures.
From this initial investigation, it shows that the implementation of a second Leap Motion is firstly viable as Mohandes et al. incorporated it into their study and secondly is novel due to the majority studies only utilising one controller as a gesture input device. As such, this would be an ideal approach to investigate and implement in a future study.

References

Artal-Sevil, J. S. and Montanes, J. L. (2016) ‘Development of a robotic arm and implementation of a control strategy for gesture recognition through Leap Motion device’, 2016 Technologies Applied to Electronics Teaching (TAEE), pp. 1–9. DOI: 10.1109/TAEE.2016.7528373.

Brett Anderson. (2016) ‘What is the purpose of a Gantt Chart’. Available at: https://www.successfulprojects.com.au/purpose-of-a-gantt-chart/

Cabreira, A. T. and Hwang, F. (2015) ‘An analysis of mid-air gestures used across three platforms’, Proceedings of the 2015 British HCI Conference on - British HCI ’15, pp. 257–258. DOI: 10.1145/2783446.2783599.

Cavalcanti, Medeiros & Dantas. (2017) ‘Evaluate Leap Motion Control for Multiple Hand Posture Recognition’, 2017 19th Symposium on Virtual and Augmented Reality on pp. 341 – 343.

Chen, Y. Et al. (2015) ‘A Real-time Dynamic Hand Gesture Recognition System Using Kinect Sensor *’, pp. 2026–2030.

Cho, Li and Chen. (2012) ‘Implementation of Low-Cost Vision-Based Gesture Recognition Systems Based on FPGA Approach’, 2012 International Symposium on Computer, Consumer and Control on, pp. 329 – 332

Crawford. (2010) ‘How Microsoft Kinect Works’, pp 3. Available at: http://electronics.howstuffworks.com/microsoft-kinect.htm

Colgan. (2014) ‘How Does the Leap Motion Controller Work?’ Available at: http://blog.leapmotion.com/hardware-to-software-how-does-the-leap-motion-controller-work/

Fry, Reas. (2014) ‘Interactivity. Processing: A Programming Handbook for Visual Designers and Artists, Second Edition’. Available at: https://processing.org/tutorials/interactivity/

Fry and Reas. (2017) (2) ‘Draw ()’. Available at: https://processing.org/reference/draw_.html

Fry and Reas. (2017) (3) ‘Thread ()’. Available at: https://processing.org/reference/draw_.html

Huang, T. (1996) ‘Computer Vision: Evolution And Promise’, 19th CERN School of Computing, pp. 21–25. DOI: 10.5170/CERN-1996-008.21.ISTQB Exam Certification, 2017. What is Spiral model- advantages, disadvantage, and when to use it? Available at: http://istqbexamcertification.com/what-is-spiral-model-advantages-disadvantages-and-when-to-use-it/

John M. Carroll et al. (2014) ‘The Encyclopaedia of Human-Computer Interaction, 2nd Ed. Available at: https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed/human-computer-interaction-brief-intro#

Kim, J.-H., Thang, N. D. and Kim, T.-S. (2009) ‘3-D hand motion tracking and gesture recognition using a data glove’, 2009 IEEE International Symposium on Industrial Electronics, (ISIE), pp. 1013–1018. DOI: 10.1109/ISIE.2009.5221998. Leap Motion. 2017. Gestures. Available at: https://developer.leapmotion.com/documentation/csharp/devguide/Leap_Gestures.html

Khambaty et al. (2008) 'Cost effective portable system for sign language gesture recognition', 2008 IEEE International Conference on System of Systems Engineering, pp. 112-118

Li, Hsieh, Lin & Chu. (2017) 'Hand gesture recognition for post-stroke rehabilitation using leap motion', 2017 IEEE International Conference on Applied System Innovation on, pp. 386 - 388

Marin, G., Dominio, F. & Zanuttigh, P. (2014) 'Hand gesture recognition with leap motion and Kinect devices'. 2014 IEEE International Conference on Image Processing, ICIP 2014, pp.1565–1569

Martha W. Alibali & Sotaro Kita. (2017) 'How Do Gestures Influence Thinking and Speaking? The Gesture-for-Conceptualization Hypothesis' pp. 245

Mohandes, M., Aliyu, S. and Deriche, M. (2015) 'Prototype Arabic Sign Language Recognition using Multi-Sensor Data Fusion of Two Leap Motion Controllers', International Multi-Conference on Systems, Signals & Devices, pp. 1–6.

Moldovan, I.M. et al. (2017) 'Virtual rehabilitation programme using the MIRA platform, Kinect and Leap Motion sensors in an 81 years old patient with ischemic stroke.' 2017 E-Health and Bioengineering Conference, EHB 2017, pp.325–328.

Oxford Dictionary. (2019) 'Gesture'. Available at: https://www.oxfordlearnersdictionaries.com/us/definition/english/gesture_2

Philip Brey & Philip Jansen. (2015) 'Ethics Assessment in Different Fields – Engineering Sciences' pp. 11-12

Pine, K. J., Lufkin, N. and Messer, D. (2004) 'More gestures than answers: children learning about balance.', Developmental psychology, 40(6), pp. 1059–67. DOI: 10.1037/0012-1649.40.6.1059.

Ploengpit, Y. (2016) 'Rock-Paper-Scissors with Myo Armband Pose Detection', International Computer Science and Engineering Conference (ICSEC), pp. 1-5.
DOI: 10.1109/ICSEC.2016.7859949

Raheja, J. L. et al. (2010) 'Real-time robotic hand control using hand gestures', ICMLC 2010 - The 2nd International Conference on Machine Learning and Computing, pp. 12–16. DOI: 10.1109/ICMLC.2010.12.

Ratul, A. U., Ali, M. T. and Ahasan, R. (2016) 'GESTURE BASED WIRELESS SHADOW ROBOT', 2016 5th International Conference on Informatics, Electronics and Vision on pp. 351–355.

Schroder, M. et al. (2012) 'Real-time hand tracking with a colour glove for the actuation of anthropomorphic robot hands', IEEE-RAS International Conference on Humanoid Robots, pp. 262–269. DOI: 10.1109/HUMANOIDS.2012.6651530.

Silva, L. et al. (2016) 'PHYS. IO : Wearable Hand Tracking Device', IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA) pp. 1-5. DOI: 10.1109/CIVEMSA.2016.7524320

Smeragliuolo, A. H. et al. (2016) 'Validation of the Leap Motion Controller using markered motion capture technology', Journal of Biomechanics. Elsevier, 49(9), pp. 1742–1750. DOI:

10.1016/j.jbiomech.2016.04.006.

Softwaretestinghelp. (2019) ‘Spiral Model – What Is SDLC Spiral Model?’. Available at: https://www.softwaretestinghelp.com/spiral-model-what-is-sdlc-spiral-model/

Wang, F. et al. (2016) ‘Gesture recognition based on BoF and its application in human-machine interaction of service robot’, pp. 115–120.

Weichert, F. et al. (2013) ‘Analysis of the accuracy and robustness of the Leap Motion Controller’, Sensors (Switzerland), 13(5), pp. 6380–6393. DOI: 10.3390/s130506380.

Wrike. (2019) ‘FAQ: What is Scoop Creep?’. Available at: https://www.wrike.com/project-management-guide/faq/what-is-scope-creep-in-project-management/

Appendices

Appendix A – Original Marionette Code

```
//Importing the Leap Motion Processing and the Serial Libraries to use within the code.
import de.voidplus.leapmotion.*;
import processing.serial.*;

//Creation of a new instance of the Leap Motion Class with a variable called leapController
LeapMotion leapController;

//Creation of two variables, servo angle right (servoAngleR) and servo angle left
//(servoAngleL) which both have the integer datatype
int servoAngleR;
int servoAngleL;

//Creation of a called port
Serial port;
PFont newText;

void setup()
{
  //Setup for the size of the window displayed for the user to see
  //the captured hand data. The window is sized 800 x 500 and has its background colour
  //set to white.
  size(800, 500);
  background(255);

  //Defines variable leapController as a new instance of the Class LeapMotion.
  leapController = new LeapMotion(this);

  //Prints a list of available/usable ports and then assigns the variable port
  //to the first option within the list at a baudRate of 9600
  println(Serial.list());
  port = new Serial(this, Serial.list()[0], 9600);
}

void draw()
```

```
{
  //
  background(255);

  //
  for (Hand : leapController.getHands ())
  {
    //draws the users hand onto the created window, allowing them to see what they are
    //doing in real-time
    hand.draw();

    //A new variable called hand position which has a PVector data type is associated with
    //the method .getPosition() being used on the variable hand.
    PVector handPosition = hand.getPosition();
    //The captured hand position is then printed out for the user to see in the Processing
    //IDE. it is printed out in the format of X, Y and Z values
    println(handPosition);



    //Rotating the servos clockwise ~~~~Moving Forward~~~~
    if (handPosition.z > 60)
    {
      servoAngleR = 92;
      servoAngleL = 92;
    }

    //Rotating the servos clockwise ~~~~Moving Forward and turning right~~~~
    if (handPosition.z > 60 && handPosition.x > 540)
    {
      servoAngleR = 107;
      servoAngleL = 92;
    }

    //Rotating the servos clockwise ~~~~Moving Forward and turning left~~~~
    if (handPosition.z > 60 && handPosition.x < 300)
    {
      servoAngleR = 92;
```

```
    }

    //Stopping the servos rotation ~~~~Stopping Movement~~~~
    if (handPosition.z < 60 && handPosition.z > 40)
    {
      servoAngleR = 104;
      servoAngleL = 104;
    }

    //Rotating the servos anti-clockwise ~~~~~Moving Backwards~~~~~
    if (handPosition.z < 40)
    {
      servoAngleR = 107;
      servoAngleL = 107;
    }

    //Rotating the servos anti-clockwise and turning to the right
    //~~~~~Moving Backwards and turning right~~~~~
    if (handPosition.z < 40 && handPosition.x > 540)
    {
      servoAngleR = 107;
    }

    //Rotating the servos anti-clockwise and turning to the left
    //~~~~~Moving Backwards and turning left~~~~~
    if (handPosition.z < 40 && handPosition.x < 300)
    {
      servoAngleL = 107;
    }

  //port.write(x) is then used to send the information chosen depending on the if
  //statement recognised to the serial port so that i can be picked up and used by the
  //Arduino
   port.write(servoAngleR);
   port.write(servoAngleL);
   }
  }
```

Appendix B – Original Marionette Project 'Free play' testing results

The average number of gestures reported by participants	The average number of gestures performed by users as seen by tester	Total number of gestures in the system
3	5	7

Appendix C – Original Marionette Project task testing results

Action	The correct gesture performed in the first attempt	The correct gesture performed in the second attempt	The correct gesture performed in the third attempt	A correct gesture not performed	The average number of attempts
1	5	4	1	0	1
2	10	0	0	0	1
3	7	3	0	0	1
4	10	0	0	0	1
5	10	0	0	0	1
6	10	0	0	0	1
7	10	0	0	0	1

Appendix D – Original Questionnaire Responses

	Question	Average Score given by participants
1	On a scale of 1 to 10, with one being bad and ten being good. How easy did you find the system to learn?	5
2	On a scale of 1 to 10, with one being bad and ten being good. How natural/comfortable did you find the system to use?	5
3	On a scale of 1 to 10, with one being bad and ten being good. How innovative did you feel the system is, compared to other products on the market?	8

Question	Number of 'Yes' Responses	Number of 'Maybe' Responses	Number of 'No 'Responses

4	Outside of the implementation using the Radio-controlled car that you have seen, could you see this system utilised to control more complex robotics?	1	6	3

	User Feedback	**Number of participants who mentioned this point**
1	Visual confirmation, for current gestures performed by users	10
2	When removing hand / stopping the system, incorrect gestures performed	8
3	Adding a gesture to change modes or activate the system	5
4	The addition of a second hand to help mitigate the amount of control required from one hand	6
5	The addition of a second leap motion controller	5

Appendix F – Approach One – Gesture confirmation code

```
//Importing the Leap Motion Processing and the Serial Libraries to use within the code.
import de.voidplus.leapmotion.*;
import processing.serial.*;

//Creation of a new instance of the Leap Motion Class with a variable called leapController
LeapMotion leapController;

//Creation of two variables, servo angle right (servoAngleR) and servo angle left
//(servoAngleL) which both have the integer datatype
int servoAngleR;
int servoAngleL;

//Creation of a called port
Serial port;
PFont newText;

void setup()
{
  //Setup for the size of the window displayed for the user to see
  //the captured hand data. The window is sized 800 x 500 and has its background colour
  //set to white.
  size(800, 500);
  background(255);

  //Defines variable leapController as a new instances of the Class LeapMotion.
  leapController = new LeapMotion(this);
  newText = createFont("Arial", 16, true);

  //Prints a list of available/usable ports and then assigns the variable port
  //to the first option within the list at a baudRate of 9600
  println(Serial.list());
  //port = new Serial(this, Serial.list()[0], 9600);
}

void draw()
```

```
{
 //
 background(255);

 //
 for (Hand hand : leapController.getHands ())
 {
  //draws the users hand onto the created window, allowing them to see what they are
  //doing in real-time
  hand.draw();

  //A new variable called hand position which has a PVector data type is associated with
  //the method .getPosition() being used on the variable hand.
  PVector handPosition = hand.getPosition();
  //The captured hand position is then printed out for the user to see in the Processing
  //IDE. it is printed out in the format of X, Y and Z values
  println(handPosition);



  //Rotating the servos clockwise ~~~~Moving Forward~~~~
  if (handPosition.z > 60)
  {
   text("Moving Forward", 10, 100);
   servoAngleR = 92;
   servoAngleL = 92;
  }

  //Rotating the servos clockwise ~~~~Moving Forward and turning right~~~~
  if (handPosition.z > 60 && handPosition.x > 540)
  {
   text("Moving Forward & Turning Right", 10, 100);
   servoAngleR = 107;
   servoAngleL = 92;
  }

  //Rotating the servos clockwise ~~~~Moving Forward and turning left~~~~
  if (handPosition.z > 60 && handPosition.x < 300)
```

```
{
   text("Moving Forward & Turning Left", 10, 100);
  servoAngleR = 92;
}

//Stopping the servos rotation ~~~~Stopping Movement~~~~
if (handPosition.z < 60 && handPosition.z > 40)
{
   text("System Stopped", 10, 100);
  servoAngleR = 104;
  servoAngleL = 104;
}

//Rotating the servos anti-clockwise ~~~~~Moving Backwards~~~~~
if (handPosition.z < 40)
{
   text("Moving Backwards", 10, 100);
  servoAngleR = 107;
  servoAngleL = 107;
}

//Rotating the servos anti-clockwise and turning to the right
//~~~~~Moving Backwards and turning right~~~~~
if (handPosition.z < 40 && handPosition.x > 540)
{
   text("Moving Backwards & Turning Right", 10, 100);
  servoAngleR = 107;
}

//Rotating the servos anti-clockwise and turning to the left
//~~~~~Moving Backwards and turning left~~~~~
if (handPosition.z < 40 && handPosition.x < 300)
{
   text("Moving Backwards & Turning Left", 10, 100);
  servoAngleL = 107;
}
```

```
//port.write(x) is then used to send the information chosen depending on the if
//statement recognised to the serial port so that i can be picked up and used by the
//Arduino
 port.write(servoAngleR);
 port.write(servoAngleL);
 }
}
```

Appendix G – Reas and Fry Interactivity tutorial code

```
void setup() {
 size(100, 100);
}

void draw() {
 if (mousePressed == true) {
   if (mouseButton == LEFT) {
     fill(0); // Black
   } else if (mouseButton == RIGHT) {
     fill(255); // White
   }
 } else {
   fill(126); // Gray
 }
 rect(25, 25, 50, 50);
}
```

Appendix H – Approach 2, Implementation of gestures

```
import de.voidplus.leapmotion.*;

LeapMotion leap;
boolean gestureUsed = false;

void setup()
{
  size(800,500);
  background(255);
  leap = new LeapMotion(this);
  leap = new LeapMotion(this).allowGestures();
}

void draw()
{
  background(255);
  for (Hand hand : leap.getHands ())
  {
    hand.draw();
    PVector handPosition = hand.getPosition();
    rect (25,25,50,50);
  }
}

void leapOnCircleGesture(CircleGesture g, int state){
  int     id               = g.getId();
  Finger  finger           = g.getFinger();
  PVector positionCenter   = g.getCenter();
  float   radius           = g.getRadius();
  float   progress         = g.getProgress();
  long    duration         = g.getDuration();
  float   durationSeconds  = g.getDurationInSeconds();
  int     direction        = g.getDirection();


 switch(state){
```

```
    case 1: // Start
      break;
    case 2: // Update
      break;
    case 3: // Stop
       if (gestureUsed = true)
  {
    fill(75,98,244,85);
    println("gesture used");
  }

  else
  {
    fill(0,0,0,0);
}
      println("CircleGesture: " + id);
      break;
  }

  switch(direction){
    case 0: // Anticlockwise/Left gesture
      break;
    case 1: // Clockwise/Right gesture
      break;
  }
}
```

Appendix I – Approach 3 Initial Implementing of two-hand control

```
import de.voidplus.leapmotion.*;
LeapMotion leap;

void setup()
{
size(800,500);
 leap = new LeapMotion(this);
}

void draw()
{
 background(204);
 for (Hand hand : leap.getHands())
 {
  hand.draw();
  PVector handPosition = hand.getPosition();

  boolean handIsLeft = hand.isLeft();
  boolean handIsRight = hand.isRight();
  float handPinch = hand.getPinchStrength();
  float handTime = hand.getTimeVisible();

  if (handIsLeft)
  {
   fill(255);
  }



 if(handIsRight)
  {
   fill(0);
  }

  if(handIsLeft && handPinch == 1)
  {
```

```
      fill(255, 240, 39);
    }

    if(handIsRight && handPinch == 1)
    {
      fill(122,233,120);
    }

    rect (25,25,50,50);
  }
}
```

Appendix J – Approach 4 Implementation of pinch gesture

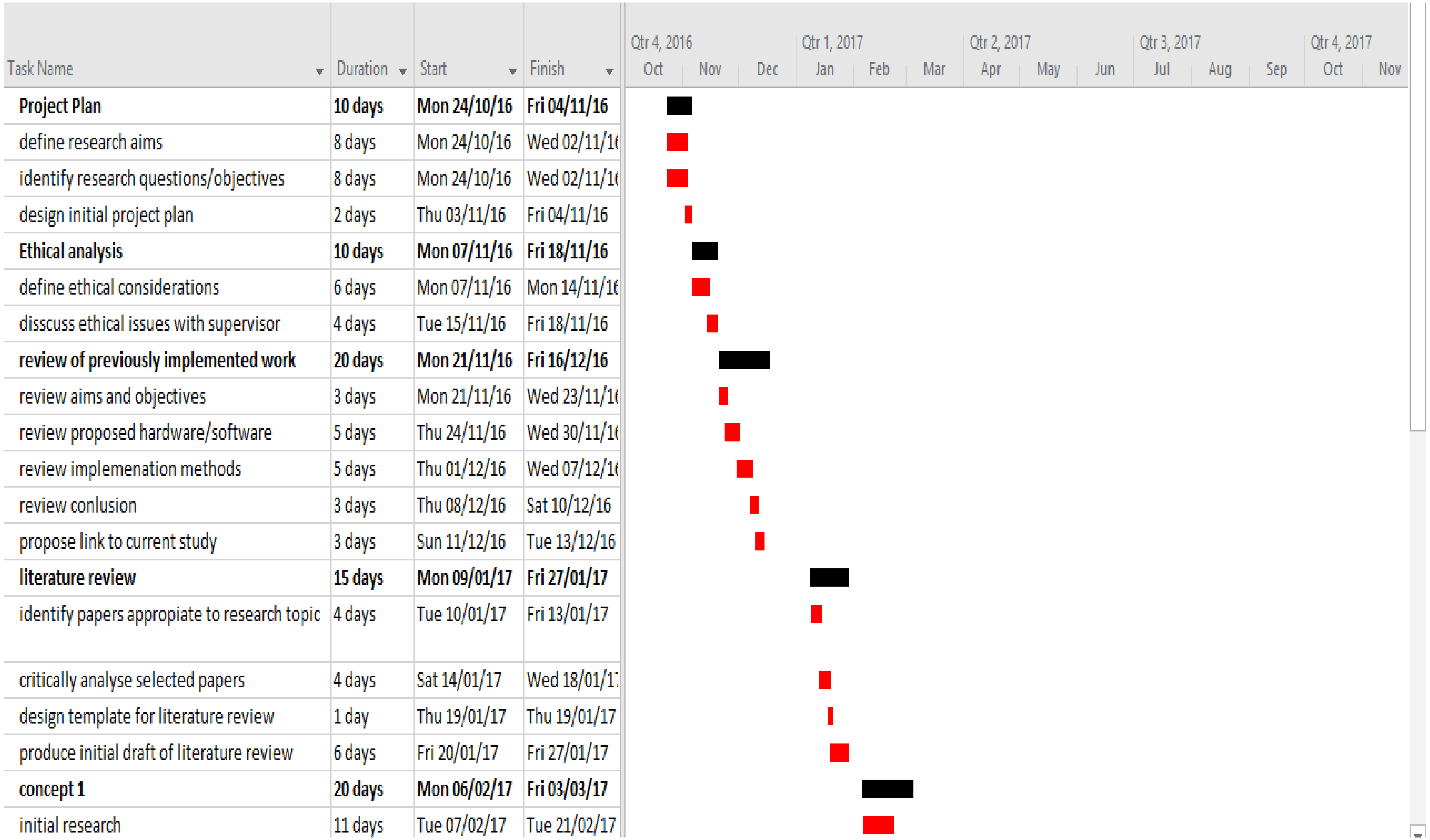
```
//Importing the Leap Motion Processing and the Serial Libraries to use within the code.
import de.voidplus.leapmotion.*;
import processing.serial.*;

//Creation of a new instance of the Leap Motion Class with a variable called leapController
LeapMotion leapController;

//Creation of two variables, servo angle right (servoAngleR) and servo angle left
//(servoAngleL) which both have an integer datatype
int servoAngleR;
int servoAngleL;

//Creation of a called port
Serial port;
PFont newText;

void setup()
{
  //Setup for the size of the window displayed for the user to see
  //the captured hand data. The window is sized 800 x 500 and has its background colour
  //set to white.
  size(800, 500);
  background(255);

  //Defines variable leapController as a new instances of the Class LeapMotion.
  leapController = new LeapMotion(this);
  newText = createFont("Arial", 16, true);

  //Prints a list of available/usable ports and then assigns the variable port
  //to the first option within the list at a baudRate of 9600
  println(Serial.list());
  //port = new Serial(this, Serial.list()[0], 9600);
}

void draw()
{
```

```
//
background(255);

//
for (Hand hand : leapController.getHands ())
{
 //draws the users hand onto the created window, allowing them to see what they are
 //doing in real-time
 hand.draw();



 //A new variable called hand position which has a PVector data type is associated with
 //the method .getPosition() being used on the variable hand.
 PVector handPosition = hand.getPosition();
 //The captured hand position is then printed out for the user to see in the Processing
 //IDE. it is printed out in the format of X, Y and Z values
 println(handPosition);

 float handPinch = hand.getPinchStrength();

 //Rotating the servos clockwise ~~~~Moving Forward~~~~
 if (handPinch < 1 && handPosition.z > 60)
 {
  text("Moving Forward", 10, 100);
  servoAngleR = 92;
  servoAngleL = 92;
 }

 //Rotating the servos clockwise ~~~~Moving Forward and turning right~~~~
 if (handPinch < 1 && handPosition.z > 60 && handPosition.x > 540)
 {
  text("Moving Forward & Turning Right", 10, 100);
  servoAngleR = 107;
  servoAngleL = 92;
 }
```

```
//Rotating the servos clockwise ~~~~Moving Forward and turning left~~~~
if (handPinch < 1 && handPosition.z > 60 && handPosition.x < 300)
{
   text("Moving Forward & Turning Left", 10, 100);
  servoAngleR = 92;
}

//Stopping the servos rotation ~~~~Stopping Movement~~~~
if (handPinch == 1 && handPosition.z < 60 && handPosition.z > 40)
{
   text("System Stopped", 10, 100);
  servoAngleR = 104;
  servoAngleL = 104;
}

//Rotating the servos anti-clockwise ~~~~~Moving Backwards~~~~~
if (handPinch < 1 && handPosition.z < 40)
{
   text("Moving Backwards", 10, 100);
  servoAngleR = 107;
  servoAngleL = 107;
}

//Rotating the servos anti-clockwise and turning to the right
//~~~~~Moving Backwards and turning right~~~~~
if (handPinch < 1 && handPosition.z < 40 && handPosition.x > 540)
{
   text("Moving Backwards & Turning Right", 10, 100);
  servoAngleR = 107;
}

//Rotating the servos anti-clockwise and turning to the left
//~~~~~Moving Backwards and turning left~~~~~
if (handPinch < 1 && handPosition.z < 40 && handPosition.x < 300)
{
   text("Moving Backwards & Turning Left", 10, 100);
  servoAngleL = 107;
```

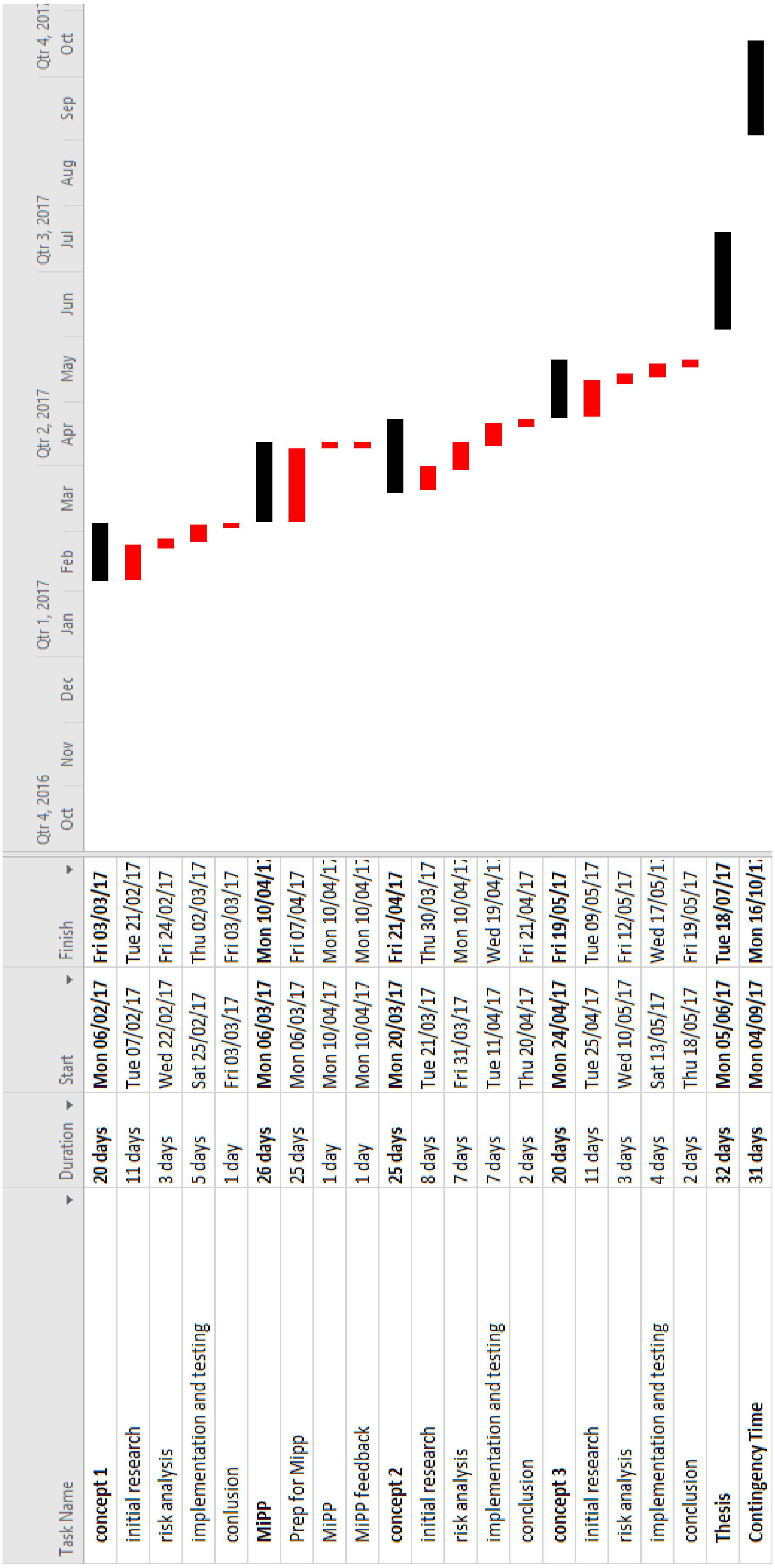
```
   }

//port.write(x) is then used to send the information chosen depending on the if
//statement recognised to the serial port so that i can be picked up and used by the
//Arduino
 port.write(servoAngleR);
 port.write(servoAngleL);
 }}
```

Appendix K.1– Gantt chart



Task Name	Duration	Start	Finish
Project Plan	**10 days**	**Mon 24/10/16**	**Fri 04/11/16**
define research aims	8 days	Mon 24/10/16	Wed 02/11/1
identify research questions/objectives	8 days	Mon 24/10/16	Wed 02/11/1
design initial project plan	2 days	Thu 03/11/16	Fri 04/11/16
Ethical analysis	**10 days**	**Mon 07/11/16**	**Fri 18/11/16**
define ethical considerations	6 days	Mon 07/11/16	Mon 14/11/1
disscuss ethical issues with supervisor	4 days	Tue 15/11/16	Fri 18/11/16
review of previously implemented work	**20 days**	**Mon 21/11/16**	**Fri 16/12/16**
review aims and objectives	3 days	Mon 21/11/16	Wed 23/11/1
review proposed hardware/software	5 days	Thu 24/11/16	Wed 30/11/1
review implemenation methods	5 days	Thu 01/12/16	Wed 07/12/1
review conlusion	3 days	Thu 08/12/16	Sat 10/12/16
propose link to current study	3 days	Sun 11/12/16	Tue 13/12/16
literature review	**15 days**	**Mon 09/01/17**	**Fri 27/01/17**
identify papers appropiate to research topic	4 days	Tue 10/01/17	Fri 13/01/17
critically analyse selected papers	4 days	Sat 14/01/17	Wed 18/01/1
design template for literature review	1 day	Thu 19/01/17	Thu 19/01/17
produce initial draft of literature review	6 days	Fri 20/01/17	Fri 27/01/17
concept 1	**20 days**	**Mon 06/02/17**	**Fri 03/03/17**
initial research	11 days	Tue 07/02/17	Tue 21/02/17

Appendix K.2 – Gantt chart



Task Name	Duration	Start	Finish
concept 1	**20 days**	**Mon 06/02/17**	**Fri 03/03/17**
initial research	11 days	Tue 07/02/17	Tue 21/02/17
risk analysis	3 days	Wed 22/02/17	Fri 24/02/17
implementation and testing	5 days	Sat 25/02/17	Thu 02/03/17
conlusion	1 day	Fri 03/03/17	Fri 03/03/17
MiPP	**26 days**	**Mon 06/03/17**	**Mon 10/04/1**
Prep for Mipp	25 days	Mon 06/03/17	Fri 07/04/17
MiPP	1 day	Mon 10/04/17	Mon 10/04/1
MiPP feedback	1 day	Mon 10/04/17	Mon 10/04/1
concept 2	**25 days**	**Mon 20/03/17**	**Fri 21/04/17**
initial research	8 days	Tue 21/03/17	Thu 30/03/17
risk analysis	7 days	Fri 31/03/17	Mon 10/04/1
implementation and testing	7 days	Tue 11/04/17	Wed 19/04/1
conclusion	2 days	Thu 20/04/17	Fri 21/04/17
concept 3	**20 days**	**Mon 24/04/17**	**Fri 19/05/17**
initial research	11 days	Tue 25/04/17	Tue 09/05/17
risk analysis	3 days	Wed 10/05/17	Fri 12/05/17
implementation and testing	4 days	Sat 13/05/17	Wed 17/05/1
conclusion	2 days	Thu 18/05/17	Fri 19/05/17
Thesis	**32 days**	**Mon 05/06/17**	**Tue 18/07/17**
Contingency Time	**31 days**	**Mon 04/09/17**	**Mon 16/10/1**

Appendix L – Ethics Form

UNIVERSITY OF BEDFORDSHIRE

Research Ethics Scrutiny (Postgraduate Research Students)

When completing this form please ensure that you read and comply with the following:

Researchers must demonstrate clear understanding of an engagement with the following:

1. Integrity - The research has been carried out in a rigorous and professional manner and due credit has been attributed to all parties involved.
2. Plagiarism - Proper acknowledgement has been given to the authorship of data and ideas.
3. Conflicts of Interest - All financial and professional conflicts of interest have been properly identified and declared.
4. Data Handling - The research draws upon effective record keeping, proper storage of date in line with confidentiality, statute and University policy.
5. Ethical Procedures - Proper consideration has been given to all ethical issues and appropriate approval sought and received from all relevant stakeholders. In addition the research should conform to professional codes of conduct where appropriate.
6. Supervision - Effective management and supervision of staff and student for whom the researcher(s) is/are responsible
7. Health and Safety- Proper training on health and safety issues has been received and completed by all involved parties. Health and safety issues have been identified and appropriate assessment and action have been undertaken.

The **Research Institutes** are responsible for ensuring that all researchers abide by the above. It is anticipated that ethical approval will be granted by each Research Institute. Each Research Institute will give guidance and approval on ethical procedures and ensure they conform to the requirements of relevant professional bodies. As such Research Institutes are required to provide the University Research Ethics Committee with details of their procedures for ensuring adherence to relevant ethical requirements. This applies to any research whether it be, or not, likely to raise ethical issues. Research proposals involving vulnerable groups; sensitive topics; groups requiring gatekeeper permission; deception or without full informed consent; use of personal/confidential information; subjects in stress, anxiety, humiliation or intrusive interventions must be referred to the University Research Ethics Committee.

Research projects involving participants in the NHS will be submitted through the NHS National Research Ethics Service (NRES). The University Research Ethics Committee will normally accept the judgement of NRES (it will never approve a proposal that has been rejected by NRES), however NRES approval will need to be verified before research can commence and the nature of the research will need to be verified.

Where work is conducted in collaboration with other institutions ethical approval by the University and the collaborating partner(s) will be required.

The **University Research Ethics Committee** is a sub-committee of the Academic Board and is chaired by a member of the Vice Chancellor's Executive Group, appointed by the Vice-Chancellor and includes members external to the University

Research Misconduct: Allegations of Research Misconduct against staff or post graduate (non-taught) research students should be made to the Director of Research Development.

UNIVERSITY OF BEDFORDSHIRE

Research Ethics Scrutiny (Annex to RS1 form)

SECTION A To be completed by the candidate

Registration No: 1305699

Candidate: Stephen Chase

Degree of: Masters by Research

Research Institute: IRAC

Research Topic: "Testing three concepts to improve upon a previously developed gesture recognition system"

External Funding: N/A

The candidate is required to summarise in the box below the ethical issues involved in the research proposal and how they will be addressed. In any proposal involving human participants the following should be provided:

- clear explanation of how informed consent will be obtained,
- how will confidentiality and anonymity be observed,
- how will the nature of the research, its purpose and the means of dissemination of the outcomes be communicated to participants,
- how personal data will be stored and secured
- if participants are being placed under any form of stress (physical or mental) identify what steps are being taken to minimise risk

If protocols are being used that have already received University Research Ethics Committee (UREC) ethical approval then please specify. Roles of any collaborating institutions should be clearly identified. Reference should be made to the appropriate professional body code of practice.

The research carried out in this thesis requires participants to test the developed approaches. Participants will be given the option to join the testing voluntarily and will be informed about the testing process and what is expected of them. Additionally no physical or otherwise harmful feedback will be used.

Answer the following question by deleting as appropriate:

1. Does the study involve vulnerable participants or those unable to give informed consent (e.g. children, people with learning disabilities, your own students)?

   No

   If **YES**: Have/will Researchers be DBS checked?

   No

2. Will the study require permission of a gatekeeper for access to participants (e.g. schools, self-help groups, residential homes)?

   No

3. Will it be necessary for participants to be involved without consent (e.g. covert observation in non-public places)?

   No

4. Will the study involve sensitive topics (e.g. sexual activity, substance abuse)?

   No

5. Will blood or tissue samples be taken from participants?

   No

6. Will the research involve intrusive interventions (e.g. drugs, hypnosis, physical exercise)?

   No

7. Will financial or other inducements be offered to participants (except reasonable expenses)?

   No

8. Will the research investigate any aspect of illegal activity?

   No

9. Will participants be stressed beyond what is normal for them?

   No

10. Will the study involve participants from the NHS (e.g. patients) or participants who fall under the requirements of the Mental Capacity Act 2005?

No

If you have answered yes to any of the above questions or if you consider that there are other significant ethical issues then details should be included in your summary above. If you have answered yes to Question 1 then a clear justification for the importance of the research must be provided.

*Please note if the answer to Question 10 is yes then the proposal should be submitted through **NHS research ethics approval procedures** to the appropriate **NRES**. The UREC should be informed of the outcome.

Checklist of documents which should be included:

Project proposal (with details of methodology) & source of funding	
Documentation seeking informed consent (if appropriate)	
Information sheet for participants (if appropriate)	
Questionnaire (if appropriate)	

(Tick as appropriate)

Applicant declaration

I understand that I cannot collect any data until the application referred to in this form has been approved by all relevant parties. I agree to carry out the research in the manner specified and comply with the statement of ethical requirements on page 1 of this form. If I make any changes to the approved method I will seek further ethical approval for any changes.

Signature of Applicant: Date: 06/02/18

Signature of Director of Studies Date: 06/02/18

This form together with a copy of the research proposal should be submitted to the Research Institute Director for consideration by the Research Institute Ethics Committee/Panel

Note you cannot commence collection of research data until this form has been approved

SECTION B to be completed by the Research Institute Ethics Committee:

Comments: Can you clarify where and how human gesture information will be captured for your research?

~~Approved~~

Signature Chair of Research Institute Ethics Committee:

Date: 7/2/18

This form should then be filed on the student’s record

If in the judgement of the committee there are significant ethical issues for which there is not agreed practice then further ethical consideration is required before approval can be given and the proposal with the committees comments should be forwarded to the secretary of the UREC for consideration.

There are significant ethical issues which require further guidance

Signature Chair of Research Institute Ethics Committee:

Date:

This form together with the recommendation and a copy of the research proposal should then be submitted to the University Research Ethics Committee