Heriot-Watt University

School of Engineering and Physical Sciences

Course: B81EZ Critical Analysis and research preparation (CARP)

Project Title: Gesture Controlled Underwater Manipulator

MSc programme: Robotics, Autonomous and Interactive Systems with a specialism in Marine Robotics

Student's name: Roshenac Mitchell

Matriculation Number: H00074473

Supervisor: David Lane and Matthew Dunnigan

Date: 23 March 2016

Final Mark: ……

Table of Contents

Project proposal

Context

Teleoperation of underwater manipulators allow operators to control the arm and end-effector remotely from above water. There are currently a number of different styles of controllers, for example, joysticks, teach pendants, radio remote controllers and game controllers (Zubrycki and Granosik, 2015). However, without extensive training, these are often hard to operate well.

One alternative that has been of growing interest is the use of gesture controllers. Research into gesture control began in 1980 (Bhuiyan and Picking, 2009) and since then the technology and research have dramatically advanced. With the development of game controllers such as the Wii and the Kinect, many people are now starting to have direct experience using these technologies. The expectation is that by providing the operator with a more intuitive control method the manipulator will be easier to use and operators will feel more confident using the technology.

Novelty

While there has been a lot of research on using gesture recognition technology to control a manipulator, it is often the case that only a single method of gesture recognition is used. This research aims to demonstrate that by concatenating multiple gesture-controlled technologies, the operators can become more precise with their movements when controlling the manipulator and the end-effector.

Objectives

The aim of this project is to create a controller that allows the HDT Adroit-M Undersea Manipulator to mimic the arm movements of an operator in real time. For example, if the user straightens their arm and pinches their hand the manipulator and gripper perform the same action.

The project can be divided into two parts. The first section of the project is to get the joint angles from the operator's arm which will be read by the Kinect and the joint angles of the operator's hand and fingers which will be detected by the Leap Motion. The second part is to create a controller that scales these human movements on to the HDT Adroit-M Undersea Manipulator.

Work Package

The main work packages are the following:

- Gather relevant information about gesture control, kinematics and control systems
- Literature Review
- Risk Assessment

- Write software to extract information from the Leap Motion and Kinect
- Combine information from the Leap Motion and Kinect
- Get the manipulator running in simulation
- Investigate the kinematics of human arm and the manipulator
- Control the manipulator using simulation
- Connect the gesture control input to manipulator output in simulation
- Develop a test plan
- Check the simulated software on the HDT Adroit-M Undersea Manipulator
- Analyse and validate the data
- Prepare project report and poster.
- Submit final dissertation.
- Presentation

Expected Outcomes

The expected outcomes are as follows:

- The development of a working simulation that shows the Leap Motion and Kinect controlling the motion of the underwater manipulator.
- Real life control of the HDT Adroit-M Undersea Manipulator controlled by mimicking human gestures captured by the Leap Motion and Kinect sensors.

Associated Risks

There are a number of potential risks during the software development phase. For example, there is a risk that working long hours on the computer may result in repetitive strain injury. To mitigate this risk, I should take regular breaks from the computer. Additionally, there is a risk that as the software that is being used is unfamiliar, it may take longer to understand than anticipated. This will be minimised by familiarising myself with the software at the start of my project and speaking to others that have used similar software before.

There are also a number of hardware risks that we need to be made aware of. The manipulator that is located in the Ocean Systems Lab is a heavy piece of machinery. There is always a risk of injury by being hit by the manipulator or dropping it on yourself. To mitigate this risk, the Ocean Systems Lab will have certain rules in place that need to be followed regarding this. Furthermore, it is expected that the lab and the manipulator will only be used near the end of the project after it has been thoroughly tested in simulation. There is also the risk of the robot being damaged. The HDT arm already has a fault in which the thumb, which should have two degrees of freedom, only has one. At some point this needs to be repaired and it is hoped that this repair can be done in the lab or after the project is complete. Finally, there is the risk that others may need to use the HDT manipulator around the same time as I need to use it. To avoid this, I have booked a slot on the lab's schedule so I will have enough time to do my experiments.

Due to time pressures, the project has a very aggressive delivery plan and there is a risk that not all the activities can be complete within the available time frame. If this risk turns into an issue the project scope will be reduced and the project will deliver a simulation of the final outcome.

Resources

This project will involve a mixture of hardware and simulation software.

- Hardware
    - HDT Adroit-M Undersea Manipulator – in Ocean Systems Lab
    - Laptop
    - Leap Motion
    - Kinect

- Software
    - Uw Sim software - UnderWater SIMulator for marine robotics research and development
    - Rviz - 3D visualizer for displaying sensor data and state information from ROS
    - Integrated development environment (IDE)

The HDT Adroit-M Undersea Manipulator is currently located in the Ocean Systems Lab at Heriot-Watt University. Therefore, I will need access to this lab when doing tests that involve the manipulator.

Beneficiaries

It is hoped that this project will benefit the Ocean Systems Lab as it will give them an alternative way of controlling the HDT Adroit-M Undersea Manipulator. This project will also benefit myself as it will increase my understanding of gesture control, control system and kinematics.

Literature review

1. Introduction

Many underwater vehicles are now equipped with manipulators. These allow them to carry out more advanced tasks such as drilling or inspections (Shim et al., 2010). One example of an underwater arm is the HDT Adroit-M Undersea Manipulator.

Currently, the HDT Adroit-M Undersea Manipulator uses a Hardened Operator Control Unit (HDT Global, 2016). While this method is sufficient, it tends to require an experienced operator to control it. This paper presents an alternative method of control using gesture-controlled technology. The use of gesture control should allow a more intuitive way of control and improve ease of use.

The HDT Adroit-M Undersea Manipulator that is going to be explored is currently located in Heriot-Watt's Ocean Systems Lab. The manipulator's end effector has two fingers and a thumb. Each finger has one degree of freedom. The thumb usually has two degrees of freedom but due to a current problem with the thumb's movement it is assumed that the thumb has only one degree of freedom as well.

The remainder of the paper is organised as follows. Section 2 discusses control inputs and devices. Alternative methods of gesture control are detailed in Section 3, and Section 4 describes different ways of motion tracking. This is then followed by concluding remarks in Section 5.

2. Control input

2.1. Operator Control Unit

One of the most common methods of controlling a manipulator is the operator control unit *(Fig. 1)* (Zubrycki and Granosik, 2015; Bassily et al., 2014). An operator control unit usually consists of a joystick and a computer. The joystick is used to control the position and velocity of the end-effector while the computer is used to calculate the necessary joint angles of the arm using kinematics in order to move the end effector to the desired location (Shim et al., 2010).

Figure 1: HDT Adroit Hardened Operator Control Unit (HDT Global, 2016)

As discussed on the HDT manufacturer site (HDT Global, 2016) Currently, the HDT Adroit-M Manipulator uses a Hardened Operator Control Unit *(Fig.1)*., having a joystick allows the operator to rest their hand comfortably when they need to take a break. However, it also has its disadvantages. Using the joystick often requires hand and arm movements that may be awkward and unintuitive. Additionally, it takes a skilled and experienced operator to successfully use an Operator Control Unit (Shim et al., 2010). However, it is often the case that skilled operators are not available (Zubrycki and Granosik, 2015). In this case, an alternative method of operation, which is more intuitive to use, is needed. Two of the possible alternative control methods are discussed below.

2.2. Voice recognition

Voice recognition allows an alternative approach to controlling a manipulator compared to operator control units. In "A FRIEND for assisting handicapped people" (Martens et al., 2001), speech recognition software allows spoken commands to be compared to a set of allowed commands, which in turn carries out the required movement. Voice control relies on a set of pre-programmed gestures. This allows simple tasks to be performed. However, it does not give any scope for advancing these gestures or using them in more complex environments. Additionally, for safety reasons, these spoken commands are often long and complicated. By having complex sentence commands, this prevents background noise being confused as commands. As a result, to carry out a string of tasks, multiple sentences would need to spoken in turn.

2.3. Gesture Control

In the paper "Gesture-controlled user interfaces, what have we done and what's next?", (Bhuiyan and Picking, 2009), development of gesture-controlled technology over a period of 30 years is explored. Gestures allow the user to control devices using an alternative method to the standard input. This is found to be particularly beneficial to users that struggle with more conventional input i.e. mouse and keyboard.

Gestures are a primary form of communication for humans and are often found instinctively in babies before they can speak. It is defined in the paper as a "non-verbal communication made with a part of the body" (Bhuiyan and Picking, 2009) . With recent advances in technological affordability, it is now possible to control electronic devices using these gestures as an input. Currently, there are many different methods of gesture sensing and, as a result, nearly all parts of the body can be sensed.

Gestures can be gathered using several different methods such as accelerometers, wearables, gloves and cameras. These various technologies are often used in cooperation with one another which allows them to achieve the desired task. Cameras are now being used more than sensors as they are easier to use. This has resulted in more household products such as laptops and TVs adding the possibility of gesture controlled interaction.

The majority of the research that has been conducted to date has been on hand gestures. This has mainly been achieved using a glove which was connected up to a microcontroller. Additional focus has also been spent on head gesture recognition used alongside speech however, this has not been as predominant. Researchers are now moving away from the likes of gloves sensing and moving towards image processing software.

The games industry is one of the main users of gesture control. However, it can also be found in industries such as entertainment, AI, simulation, training and education as well as assistive learning. Consoles such as the Wii and Kinect which were created for entertainment purposes use gesture control. However, the Wii is also being used as a rehabilitation aid for individuals.

It is expected that gesture control may become more mainstream due to the decrease in cost and the fact that it is relatively non-intrusive. These new technologies are gradually becoming more intuitive and natural to use.

3. Gesture Controlled Technologies

3.1. Controller-based gestures

Within gesture-controlled technologies, there are a wide range of products that operate by direct contact with the operators. These include the Wii, tracking suits and exoskeletons as well as haptic controllers. Technologies such as the WiiMote are embedded with multiple sensors such as accelerometers that allow them to interpret the user's gestures (Guna et al., 2014). Another alternative method is with vision-base experiments which often use markers attached to the body. These markers allow the cameras to focus on specific areas of the user's body (Du and Zhang, 2014; Kofman et al., 2005).

By attaching the technology directly to the user's body, the detection of movement is more reliable than just using computer vision. As a result, the joint angle measurement is a lot more exact (Breazeal and Scassellati, 2002). However, it does have its limitations. Technologies such as exoskeletons are cumbersome and difficult to transport (Goncalves et al., 1995). Additionally, having markers attached to the user's body may hinder and limit the operator's movements (Du and Zhang, 2014).

3.2. Wired gloves

Wired gloves allow the operator's finger joint angles to be accurately measured. This allows the operator to control the end effector gripper in an intuitive way and allows the gripper to mimic the operator's hand gestures. The use of the glove is found to be particularly effective when the gripper is of similar structure to a human hand. (Zubrycki and Granosik, 2015). As discussed in (Zubrycki and Granosik, 2014), because the glove is directly attached to the hand there is no effect from external environmental conditions such as light, that can affect the measurements.

Wired gloves are mechanical and electrical and as a result they come with added issues such as wearing out. They need to be calibrated regularly as the resistance of the sensors can vary greatly. Additionally, as the operator's hand is in the glove, this hand can not be used to perform any other task (Zubrycki and Granosik, 2014).

3.3. Single camera

As previously discussed, many gesture-controlled technologies are now using cameras and computer vision. Single cameras or monocular cameras can be found on most mobile phones. As examined in (Goncalves et al., 1995), it is possible to use an estimated image, compared to the visual image, to get a sense of depth. This allows gestures to be recognised without the use of body markers, allowing the operator to have a wider range of movements.

A disadvantage of this method is the fact that depth has to be relative. A stereo camera is unable to gauge the depth of an isolated point. This results in a low level of accuracy.

3.4. Depth-aware cameras

The Kinect is a great example of a depth-aware camera. It contains three cameras: two infrared red cameras, which are used for depth direction and a standard camera which can be used for visual recognition (Du and Zhang, 2014). It can detect fully body gestures (Guna et al., 2014) and, as stated in (Afthoni, Rizal and Susanto, 2013), it can detect over 15 user joints *(Fig. 2)*. These include the head, neck, torso, shoulders, elbows, hands, knees, hips and feet.

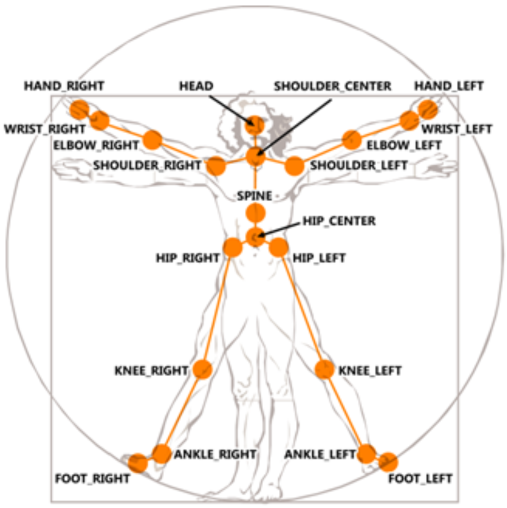


Figure 2. Kinect detected points (Microsoft, 2012)

Kinect is fairly accurate when detecting the user's joints, with errors around a few millimetres. It can also recognise a number of set gesture which can be further added to by the user (Moe and Schjolberg, 2013). Due to the distance being detected with infrared sensors, the Kinect is not affected by changes in the environment lighting (Du and Zhang, 2014). The Kinect allows the operator to connect with the manipulator in a more natural and intuitive way. As with the single camera, it is non-contacting, and therefore removes the need for markers, sensors and cables that may hinder the operator's movement (Du and Zhang, 2014).

As stated on the Microsoft website (Microsoft, 2012), the Kinect has a sensor range between 0.8 - 4 meters. This distance can be decreased by using the Kinect's 'near mode' which shortens the range to 0.5 - 3 meters. Its depth accuracy has a standard deviation of around 1.5 cm (Bassily et al., 2014). As a result, while the Kinect is good at detecting the arm and body, its accuracy of finger tracking is very low (Guna et al., 2014).

3.5. Stereo cameras

Two good examples of gesture-controlled technology using stereo cameras are the Leap Motion *(Fig. 3)* and 3 Gears (Bassily et al., 2014).

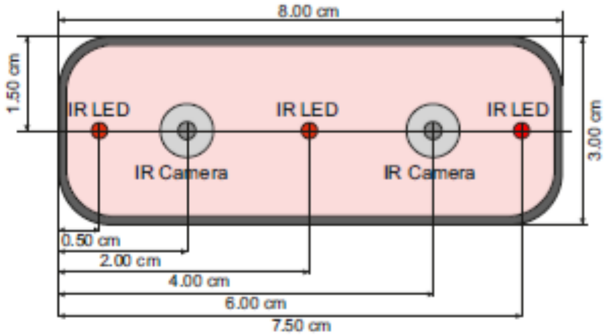


Figure 3: The schematic view of the Leap Motion Controller (Bassily et al., 2014)

Stereo cameras such as the Leap Motion and 3 Gears provide the user with a lot of information regarding the hands and finger joints such as tracking accuracy (Zubrycki and Granosik, 2014). The Leap Motion has sub-millimetre accuracy and can track all ten fingers at once (Bassily et al., 2014). Both the Leap Motion and 3 Gears are specially designed for hand gesture recognition (Zubrycki and Granosik, 2014) and provide position, orientation and joint angles for every visible finger and hand (Zubrycki and Granosik, 2014). It produces a limited amount of data, but the data it produces is a lot more accurate than the data provided by other technologies, such as the Kinect. (Marin, Dominio and Zanuttigh, 2014).

The field of view for the Leap Motion controller is an inverted triangle shape where the point is at the sensors centre (Guna et al., 2014). As the operator's hand is moved further away from the sensor, the accuracy drops (Guna et al., 2014). It is stated on the Leap Motion developer website (Leap Motion, 2016) that the sensor range is between 25-600mm (0.025 – 0.6 meters).

As with all technologies however, it has its disadvantages. Stereo cameras often have an issue with instability and occlusion. The Leap Motion is unable to detect angles when hands are smooth i.e. in gloves and finger tracking is not as accurate when angles are large. As well as this, constant light conditions needed (Zubrycki and Granosik, 2014).

4. Motion Tracking

4.1. Motion primitives

There are a number of ways to drive a manipulator once you have obtained the gestured input. Motion primitives involve saving a set of pre-programmed motions (Shim et al., 2010). The setting of these motion primitives can either be performed by manually moving the arm and saving the joint angles or by manually inputting the required joint angles. (Martens et al., 2001)

Some technologies such as the Leap Motion and the 3 Gears already come with a number of pre set recognised gestures. (Zubrycki and Granosik, 2014). The Leap Motion also allows the operator to set a number of custom gestures (Bassily et al., 2014). These can then be categorised using the cameras on these gesture-controlled technologies which can then give the user feedback on the speed, direction, position and orientation of the gesture performed (Zubrycki and Granosik, 2015).

As mentioned with regards to voice commands, motion primitives are useful when it comes to simple commands. However, when trying to achieve more complex commands or a string of flexible commands, motion primitives can prove to be limiting (Du and Zhang, 2014). Additionally, with fixed allowed gestures, it forces the operator to remember all the gesture commands that can be used. This may be hard to recall under stressful circumstances (Zubrycki and Granosik, 2014).

4.2. End effector tracking

Compared to motion primitives, end effector tracking allows the operator to interact with the manipulator in a more intuitive manner. End-effector tracking relies on the operator's hand to be tracked in free space and moved and orientated in a way that the end-effector mimics it. Inverse kinematics is then used in order to work out the required joint angles for the arm in order to move the end-effector to its required location (Shim et al., 2010; Du and Zhang, 2014). This allows the operator to concentrate on the necessary task without having to remember a set of gestures (Du and Zhang, 2014).

As discussed in (Bassily et al., 2014), using technologies such as the Leap Motion allows the operator to get hand and finger readings with given position and angle. Additionally, the use of wrist markers also allows the operator to position the end effector (Kofman et al., 2005).

However, due to the fact that only the position of the end-effector is controlled, the operator has no control of the movement of the arm joints. This lack of control of the arm may become an issue if the arm has singularities or the end effector is attempted to be moved outside the manipulator workspace (Shim et al., 2010). Additionally, hand recognition and orientation can be unstable, making the movements of the end-effector unstable. (Zubrycki and Granosik, 2014).

4.3. Motion retargeting

One of the more complicated ways of controlling a manipulator involves motion retargeting. This form of tracking involves mimicking the skeleton of an operator i.e. hip, shoulder, arm, fingers and getting the joint angles between them. The joint angles can then be mapped directly onto the manipulator (Breazeal and Scassellati, 2002; Zubrycki and Granosik, 2015).

Compared to the end-effector tracking, this does mean that it requires a way of getting all the required joint angles. The joint angles and locations will vary from person to person due to varying height and body sizes, so additional calibration may have to be implemented (Du and Zhang, 2014).

5. Conclusion

Many underwater vehicles are now equipped with manipulators and without extensive training, they are often hard to operate well. This paper explores the idea that the use of gesture control should allow a more intuitive way of controlling underwater manipulator and improve ease of use. In this paper we review a number of alternative control methods for an underwater manipulator, compare different methods of gesture control and look into different possible forms of motion tracking. As discussed in (Marin, Dominio and Zanuttigh, 2014), by combining both the Kinect for the arm and body joint and the Leap Motion for the hand and finger joints *(Fig.4)* this should result in a solution that has better manipulator control accuracy compared to using them both these technologies independently.

Figure 4. Proposed set-up (Marin, Dominio and Zanuttigh, 2014)

There have been several suggestions on how best to achieve the desired control while minimising the strain on the operator. For example, as mentioned in (Kofman et al., 2005), when performing the pinch movement to close the gripper, it is advised to have a threshold. This will allow the operator to relax their fingers a little once they have performed the action, without the gripper opening. Additionally, as seen in (Zubrycki and Granosik, 2014) it is proposed to have a number of states. These states are switched by using gestures recognised by the Leap Motion. Both end-effector and motion tracking methods will be used, these states can be seen in the following table *(Fig.5)*.

States	**Operator interaction**	**Motion tracking**	**Manipulator arm position**	**End-effector gripper**	**Gesture-controlled technology used to acquire joint angles**
Control Arm	Relevant body joint angles and orientation will be used in order to control the manipulator arm	Motion retargeting	Mimicking operator's arm and hips	Fixed	Kinect
Control end-effector	Relevant hand and finger joint angles and orientation will be used in order to control the manipulator end-effector	End-effector tracking	Fixed	Mimicking operator's hands and fingers	Leap Motion
Pause	Allows the operator to step away from the control system	N/A	Fixed	Fixed	N/A

Figure 5: Table showing the different required states

It is anticipated that using Leap Motion and Kinect technology it is possible to develop a solution that can enable an operator to accurately control an underwater manipulator with ease.

6. Bibliography

Afthoni, R., Rizal, A. and Susanto, E. (2013) 'Proportional derivative control based robot arm system using Microsoft Kinect', Robotics, Biomimetics, and Intelligent Computational Systems (ROBIONETICS), 2013 IEEE International Conference, 24-29.

Bassily, D., Georgoulas, C., Güttler, J., Linner, T. and Bock, T. (2014) 'Intuitive and adaptive robotic arm manipulation using the leap motion controller', ISR/Robotik 2014; 41st International Symposium on Robotics, TU München, 1-7.

Bhuiyan, M. and Picking, R. (2009) 'Gesture-controlled user interfaces, what have we done and what's next', Proceedings of the Fifth Collaborative Research Symposium on Security, E-Learning, Internet and Networking (SEIN 2009), Darmstadt, 25-29.

Breazeal, C. and Scassellati, B. (2002) 'Robots that imitate humans', *Trends in cognitive sciences*, vol. 6, no. 11, pp. 481-487.

Du, G. and Zhang, P. (2014) 'Markerless human–robot interface for dual robot manipulators using Kinect sensor', *Robotics and Computer-Integrated Manufacturing*, vol. 30, no. 2, pp. 150-159.

Goncalves, L., Di Bernardo, E., Ursellaj, E. and Perona, P. (1995) 'Monocular tracking of the human arm in 3D.', Computer Vision, 1995. Proceedings., Fifth International Conference, 764-770.

Guna, J., Jakus, G., Pogačnik, M., Tomažič, S. and Sodnik, J. (2014) 'An analysis of the precision and reliability of the leap motion sensor and its suitability for static and dynamic tracking', *Sensors*, vol. 14, no. 2, pp. 3702-3720.

HDT Global (2016) *Controller*, [Online], Available: http://www.hdtglobal.com/product/controller/ [20 March 2016].

Kofman, J., Wu, X., Luu, T.J. and Verma, S. (2005) 'Teleoperation of a Robot Manipulator Using a Vision-Based Human–Robot Interface', *Industrial Electronics, IEEE Transactions*, vol. 52, no. 5, pp. 1206-1219.

Leap Motion (2016) *API Overview*, [Online], Available: https://developer.leapmotion.com/documentation/csharp/devguide/Leap_Overview.html [23 March 2016].

Marin, G., Dominio, F. and Zanuttigh, P. (2014) 'Hand gesture recognition with leap motion and kinect devices', Image Processing (ICIP), 2014 IEEE International Conference, 565-1569.

Martens, C., Ruchel, N., Lang, O., Ivlev, O. and Gräser, A. (2001) 'A friend for assisting handicapped people.', *Robotics & Automation Magazine*, vol. 8, no. 1, pp. 57-65.

Microsoft (2012) *Kinect Sensor*, [Online], Available: https://msdn.microsoft.com/en-gb/library/hh438998.aspx [23 March 2016].

Moe, S. and Schjolberg, I. (2013) 'Real-Time Hand Guiding of Industrial Manipulator in 5 DOF using Microsoft Kinect and Accelerometer', *RO-MAN, 2013 IEEE. IEEE*, pp. pp. 644-649.

Shim, H., Jun, B.-H., Lee, P.-M., Baek, H. and Lee, J. (2010) 'Workspace control system of underwater tele-operated manipulators on an ROV', *Ocean Engineering*, vol. 37, no. 11, pp. 1036-1047.

Zubrycki, I. and Granosik, G. (2014) 'Using integrated vision systems: three gears and leap motion, to control a 3-finger dexterous gripper.', *Recent Advances in Automation, Robotics and Measuring Techniques*, pp. 553-564.

Zubrycki, I. and Granosik, G. (2015) 'Intuitive User Interfaces for Mobile Manipulation Tasks', *Journal of Automation Mobile Robotics and Intelligent Systems*, vol. 9, no. 1, pp. 41-52.

Gantt Chart

ID		Task Mode	Task Name	Duration	Start	Finish	Predecessors
1			**1 Gesture Controlled Underwater Manipulator**	**126 days**	**Mon 01/02/16**	**Wed 31/08/16**	
2			**1.1 Project received and confirmed with supervisor**	0 days	Mon 01/02/16	Mon 01/02/16	
3			**1.2 Background Research**	**29 days**	**Tue 02/02/16**	**Fri 11/03/16**	
4			1.2.1 Gather relevant information about gesture control, kinematics and control systems	29 days	Tue 02/02/16	Fri 11/03/16	2
5			**1.3 Portfolio**	**10 days**	**Mon 14/03/16**	**Fri 25/03/16**	
6			1.3.1 Project proposal	5 days	Mon 14/03/16	Fri 18/03/16	4
7			1.3.2 Literature review	5 days	Mon 21/03/16	Fri 25/03/16	4
8			1.3.3 Portfolio submitted	0 days	Fri 25/03/16	Fri 25/03/16	6,7
9			**1.4 Risk Assessment**	**1 day**	**Mon 09/05/16**	**Mon 09/05/16**	8
10			**1.5 Get simulation working**	**44 days**	**Tue 10/05/16**	**Fri 08/07/16**	
11			1.5.1 Learn ROS	10 days	Tue 10/05/16	Mon 23/05/16	9
12			1.5.2 Understand how the Kinect works	2 days	Tue 24/05/16	Wed 25/05/16	11
13			1.5.3 Understand how the Leap Motion works	2 days	Thu 26/05/16	Fri 27/05/16	12
14			1.5.4 Combine information from Leap Motion and Kinect	9 days	Mon 30/05/16	Thu 09/06/16	13
15			1.5.5 Get the arm running in simulation	5 days	Fri 10/06/16	Thu 16/06/16	14
16			1.5.6 Investigate the kinematics of a human arm and the manipulator	8 days	Fri 17/06/16	Tue 28/06/16	15
17			1.5.7 Control the manipulator in simulation	3 days	Wed 29/06/16	Fri 01/07/16	16
18			1.5.8 Connect the gesture control input to manipulator output in simulation	5 days	Mon 04/07/16	Fri 08/07/16	17
19			**1.6 Hardware**	**12 days**	**Mon 11/07/16**	**Tue 26/07/16**	
20			1.6.1 Develop a test plan	2 days	Mon 11/07/16	Tue 12/07/16	18

Project: GanttChart
Date: Wed 23/03/16

Task	Inactive Summary	External Tasks
Split	Manual Task	External Milestone
Milestone	Duration-only	Deadline
Summary	Manual Summary Rollup	Progress
Project Summary	Manual Summary	Manual Progress
Inactive Task	Start-only	
Inactive Milestone	Finish-only	

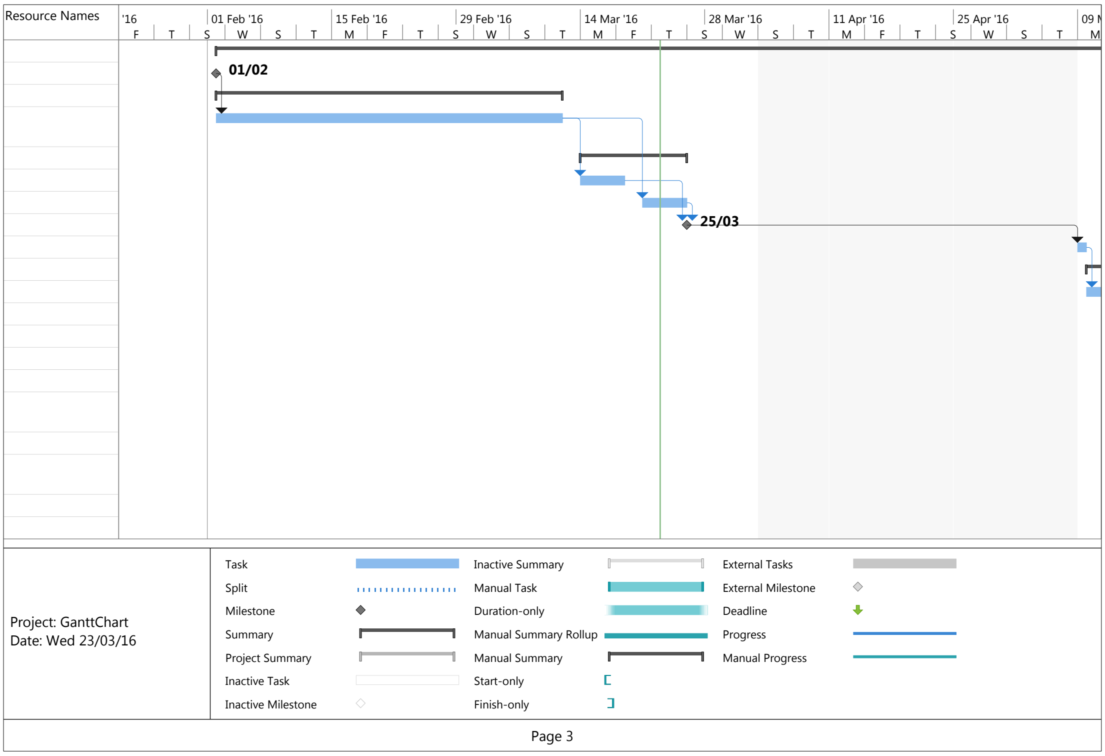
Page 1

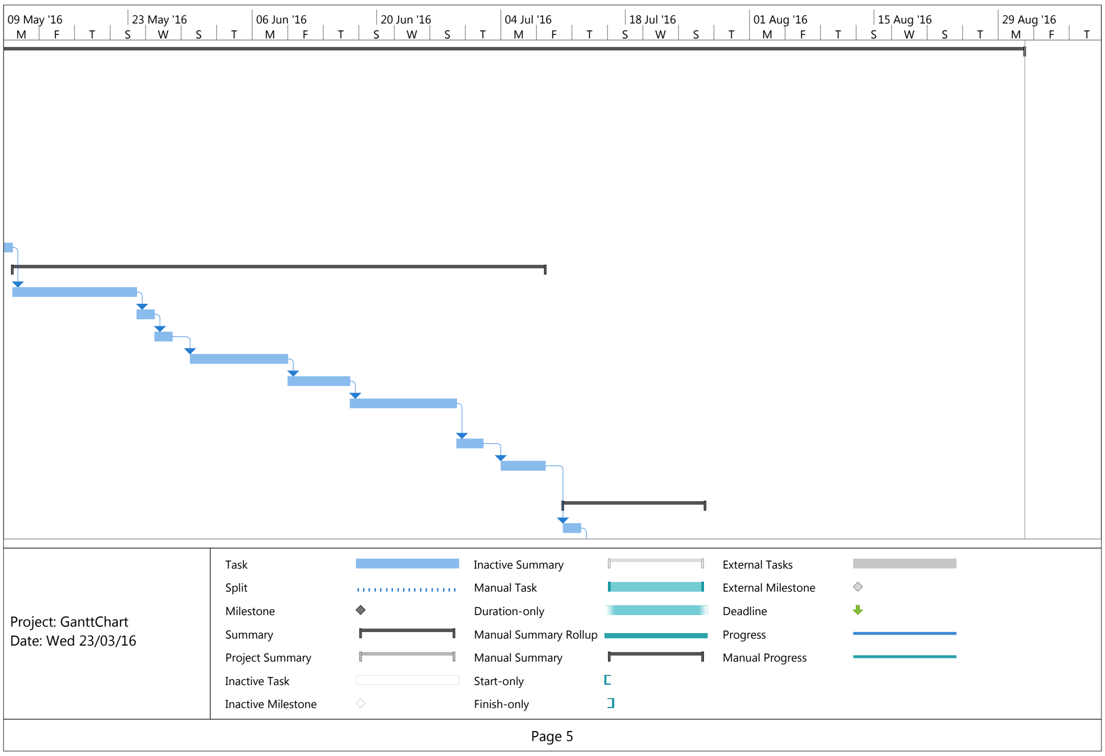
ID		Task Mode	Task Name	Duration	Start	Finish	Predecessors
21			1.6.2 Test the simulated software on the HDT Adroit manipulator	10 days	Wed 13/07/16	Tue 26/07/16	20
22			**1.7 Analyse and validate data**	**2 days**	**Wed 27/07/16**	**Thu 28/07/16**	21
23			**1.8 Dissertation Delivery**	**24 days**	**Fri 29/07/16**	**Wed 31/08/16**	
24			1.8.1 Write final dissertation	19 days	Fri 29/07/16	Wed 24/08/16	22
25			1.8.2 Submit dissertation	0 days	Wed 24/08/16	Wed 24/08/16	24
26			1.8.3 Create poster	5 days	Thu 25/08/16	Wed 31/08/16	25
27			1.8.4 Submit Poster	0 days	Wed 31/08/16	Wed 31/08/16	26

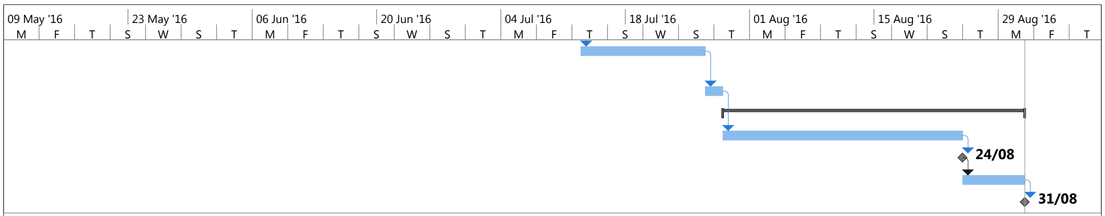
Project: GanttChart
Date: Wed 23/03/16

Task	Inactive Summary	External Tasks
Split	Manual Task	External Milestone
Milestone	Duration-only	Deadline
Summary	Manual Summary Rollup	Progress
Project Summary	Manual Summary	Manual Progress
Inactive Task	Start-only	
Inactive Milestone	Finish-only	

Page 2


Resource Names
'16
01 Feb '16
15 Feb '16
29 Feb '16
14 Mar '16
28 Mar '16
11 Apr '16
25 Apr '16
01/02
25/03
Task
Split
Milestone
Summary
Project Summary
Inactive Task
Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only
External Tasks
External Milestone
Deadline
Progress
Manual Progress
Project: GanttChart
Date: Wed 23/03/16
Page 3


09 May '16
23 May '16
06 Jun '16
20 Jun '16
04 Jul '16
18 Jul '16
01 Aug '16
15 Aug '16
29 Aug '16
Task
Split
Milestone
Summary
Project Summary
Inactive Task
Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only
External Tasks
External Milestone
Deadline
Progress
Manual Progress
Project: GanttChart
Date: Wed 23/03/16
Page 5



Project: GanttChart
Date: Wed 23/03/16