HANDS ON DESIGN

Integrating haptic interaction and feedback in virtual environments for enhanced immersive experiences in design practice.

D. CAMACHO,
University of New South Wales, Australia
d.camacho@unsw.edu.au

Abstract. The usability of virtual reality (VR) controller interfaces are complex and difficult for first time or unfamiliar users. Most controllers provide minimal feedback systems which demotes the potential for heightened interaction and feedback virtual experiences. This research explores how haptic technology systems partnered with VR can deliver immersive interactions between user and virtual environment (VE). This research involves the outcome of a haptic glove interface that incorporates a force feedback and vibrotactile feedback system. This research focuses on determining a workflow that communicates user interaction and environmental feedback using Unreal Engine and the produced haptic glove system. Targeting early established and existing haptic technology systems as a case study for the research, the progress and limitations will be explored and the use of these systems within the Built Environment Industry in particular. Accordingly, this research project adopts an action-based research methodology to frame a two-stage iterative process with evaluation. Each iteration includes integration of a haptic system to be used within VE's. Hand interaction is described as a way to "greatly enhance our exploration and use of virtual worlds" (Biocca and Levy, 1995). This statement justifies the removal in the complexity behind current interaction and controller interfaces, heavily used in industry standard VR systems. The outcome for this research is a VR interface that enables a more enhanced VR experience by engaging a range of haptic senses and connecting these with an interactive feedback system. The evaluation of its use in industry unfolds new technical knowledge for haptic technologies in collaboration with VR to review the usability and interaction standards for VR users in the design process.

Keywords. Virtual Environments; Haptic Technologies; Feedback; Interaction; Usability.

1. Introduction: Research Aims and Motivations

Virtual Reality (VR) has provided new and evolving opportunities to a wide range of industries in their design process. Within the architectural industry VR is most commonly used for visualisation and to present concepts, models and final designs through the exploration of scale and space. The Architecture, Engineering and Construction industry (AEC) over last two decades have integrated to an extent how VR can aid it practices and design processes through steady adoption of VR tools. These include collaborative workflows, employee training and showcasing experiences of final design concepts.

VR has been introduced for showcasing architectural model and concepts, allowing clients, designers and architects to perceive the design. There has been recent research and development of the use of VR partnered with sensory elements (reference).

This research aims to determine how haptic feedback and interaction can aid and mediate the communication between the user and VE's. It also outlines a methodology on allowing integration of haptic technologies in a VE.

This research project is in collaboration with PTW Architects and encompasses Aron Sheldon's research "Putting the AR into Architecture: Integrating voice recognition and gesture control for augmented reality interaction to enhance design practice" who focused on the usability and manipulation of architectural models within AR (Augmented Reality).

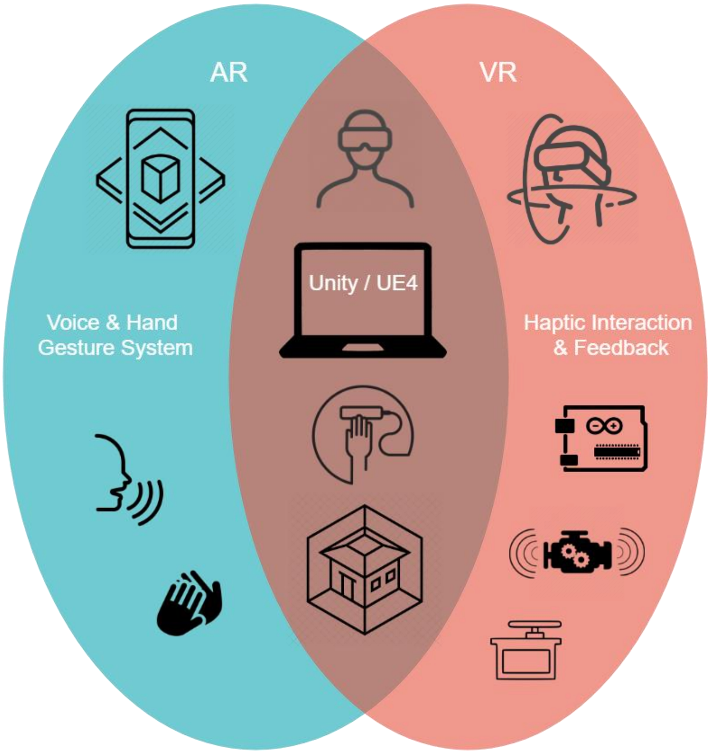


Figure 1. Projects Scope in collaboration with PTW and Aron Sheldon's research.

2. Research Observations and Objectives

It is evident that at this stage in VR's integration in industry, that user interaction and feedback remain limited. Within this area of research, there is a divide in how consumers interpret, and approach VR compared to designers and architects in industry. There is an opportunity for academic research to provide enhanced usability for user groups of commercial and industrial standard VR.

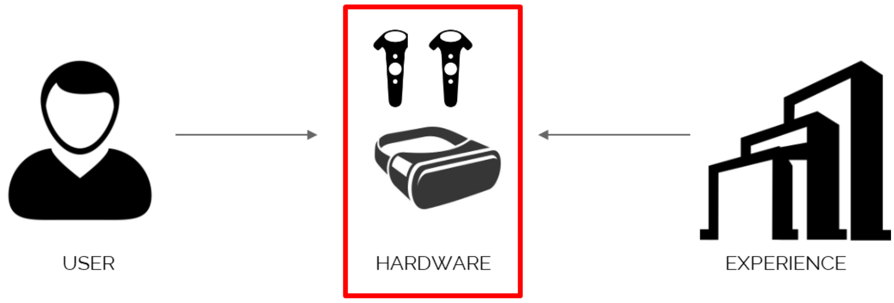


Figure 2. Current barriers of VR systems

This research situates the validity of VR systems, future research and exploration in industry. The proposed methodology will allow designers to integrate haptic interaction and feedback within VR systems. This paper outlines a basic prototype of a haptic glove system that allows users to feel realistic properties of virtual objects. The prototype also allows architects to manipulate their designs in VR and be feedback prefabricated constraints and rules of the design. The main project objective is to create an immersive first-person perspective through the medium of VR for architects, designers and clients.

3. Research Questions

VR and AR have been generally used as visual mediums in architecture for conveying scale and perspective and improve design engagement through interaction. The projects research questions are as follows:

1. In what ways can we make use of haptic technologies to mediate the interaction between virtual objects and the user in virtual environments.

2. How can haptic feedback and interactions in a virtual design enhance and benefit an architect's, designer's or client's virtual experience?

4. Methodology

In order to evaluate the viability of haptic technology implementation within VE' a prototype has been developed to demonstrate enhancement for a user. The following research is framed by a practice-based methodology of action research. This approach links the practical methodologies applied to the research with theoretical findings for controlled testing, evaluation and outcomes of the design stages.

In this project the research will be developed through a series of design iterations which will evaluated and tested to inform reasoning for further research. The research will be developed through a set stage process of plan, action, observe and reflect. This process constructs a foreground for the practical experimentation with situated ideas to guide outcomes and iterative prototyping stages.

The practical-based methodology would include specifically a two-stage framework. Each will develop the outcome of an area the design approaches to situate the significance of research. The iterations stages are depicted below:

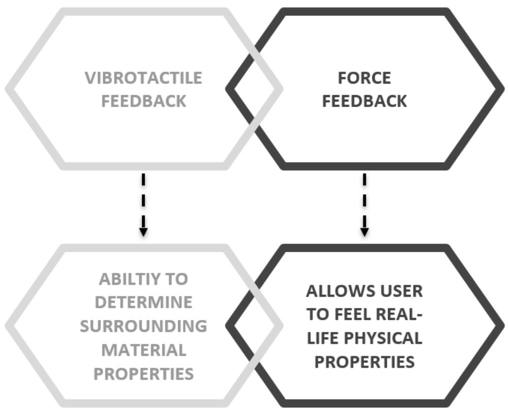


Figure 3. Iteration Stages – Practice Based

Vibrotactile feedback will include actuation of vibration motors to simulate tactile feedback with also the exploration of pin array systems to produce tactility. Force feedback will include the development of a resistive system, that actuate resistance of hand dexterity and closure through a channel mechanism controlled by a servo. These systems are all actuated and controlled through the Arduino microcontroller.

The process of communicating between a VE and a user can be broken down into several sections. First, the communication between a game engine, Unreal Engine and electronic microcontroller, Arduino will be explained, as this is crucial for the understanding of data being reconstructed and processed between software.

This paper will then outline how user inputs will be translated into outputs and be feedback to the user, this is the core framework of the constructed workflow. This process is situated with an Unreal Engine plugin, “UE4Duino”, specifically made for the purpose of communicating between Unreal and Arduino IDE. The data received by the workflow will determine the actuation and amount of feedback given to the user in the VE. An evaluation of this method and suggestions for further research and development will be provided.

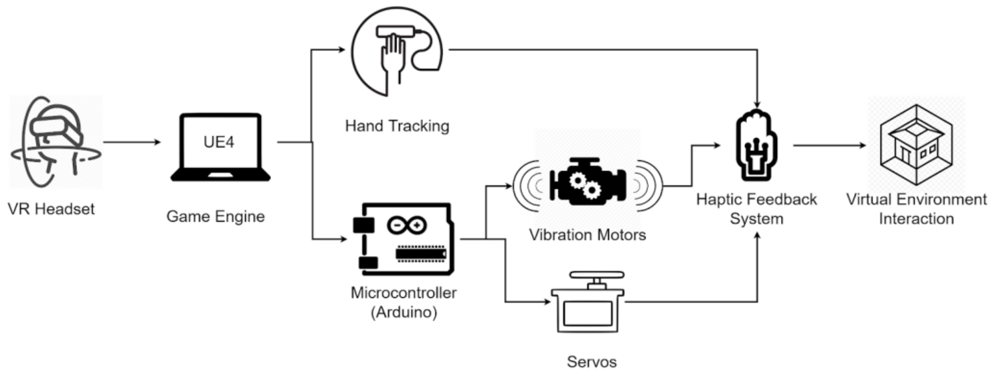


Figure 4. Prototyping Workflow Diagram

5. Background Research

5.1 HAPTIC TECHNOLOGIES

VE systems engage only the visual and auditory senses of the user are limited in their capability to interact with the user (Srinivasan, 2002). Haptic interfaces are a relatively new technology, with increased use for human interaction with VE’s since the early 1990s (O’Malley & Gupta, 2008). Commercial application of haptic and tactile devices have been simple and inexpensive, such as vibrations of a mobile, or force feedback joysticks common in gaming. These have improved usability and interaction of devices making our sense of touch “haptic” an important part of everyday life. The sensation of touch is one of the most informative senses and allows us to relate to the world around us through perception. Haptic interfaces are employed for tasks that are usually performed using hands in the real world, such as manipulation and manual exploration of virtual objects (Srinivasan, 2002).

The importance of our sense of touch is heavily depicted by Srinivasan, 2002 as it allows us to relate to the world around us and is greatly reliant in how we interact with the environment with sight as being the next major sense of humans. The word haptic originates from the Greek verb hapto - “to touch” which refers to the ability to touch and manipulate objects. Researchers such as Srinivasan from the Massachusetts Institute of Technology define three sub-areas within haptics. Human haptics, machine haptics and computer haptics all have wide varieties of applications emerged and span many areas of human needs such as product design, medical trainers and rehabilitation.

5.1.1 Haptic interaction and feedback

For review purposes, (Shanthosh and Bhavani, 2017) have published a paper explaining what the potentials in haptic technology with VR and what haptic systems are currently used. The review defines that sufficient research and development has been done in providing the audio and visual feedback in VE’s but presents the potentials in the implementation of haptic feedback systems with other sensory feedback such as taste and smell.

Later the review outlines that the ‘sense of touch’ diminishes the gap between the virtual and real-world experience. Giving the ability to physically feel the virtual objects opens up a new world of opportunities.” This provides the framework of this research to be explored in, proving that a gap exists, and opportunities are there. The review establishes the sense of touch into two elements, Cutaneous (sensed by skin) and Kinesthetic feedback (sensed by muscles, joints and hands, etc.). The underlying principle of haptics is force and pressure. These go hand in hand with determining what is possible for the scope of implementing haptic technologies with VR (Figure 1). Further on this research discusses current and emerging haptic devices that have been made or developed with their target market and use described.

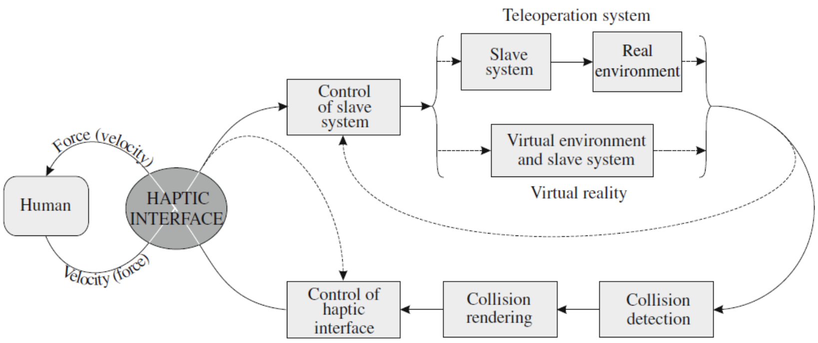


Figure 5: Haptic system: interaction between a human and the haptic interface represents a bidirectional exchange of information (Mihelj, 2014)

5.1.2 The sensory spectrum and hyperrealism

Current VE systems only engage the visual and auditory senses of the user are limited in their capability to interact with the user. This enables a greater gap between environment and the user, making it an area of research that is poised for rapid growth (Srinivasan, 2002). The communication of a design intent or proof of concept through a VE can be further enhanced through a higher level of interaction. Research related for haptic technologies for VR simulations have been of great interest to many researchers since the early times using visual displays. Haptic technologies are heavily recognized with idea of 'hyperrealism' or known as full immersion, where the user interacts with the VE replicating and rendering real-world physics and feelings.

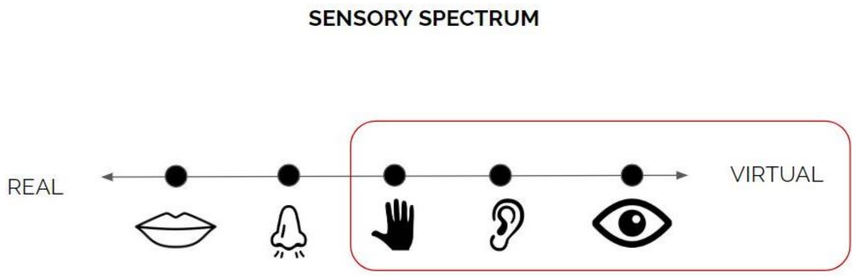


Figure 6: Sensory Spectrum

Many studies depict that the level of realism currently in VE's have extremely progressed in the past decade and predict that as technology advancements come, the level of realism comes closer and closer. However (O'Malley and Gupta, 2008) believe that haptic VE's will always be at a rendered level referring to an implemented way of a desired representation of a design. This presents that there is great divide between viewpoints and opinions on the future progression of 'hyperrealism' or 'full immersion'.

More related research is found, within the domain of gaming and simulation experience development (O'Malley & Gupta, 2008), specifically for areas of research that aim to reach a level hyperrealism. Adopting these study methods and research provides the key components of framework behind the process of designing and evaluation in testing and puts into context that haptic interfaces and technology systems is a major implementation that could be coming to consumer VR in the near future.

5.1.3 Cutaneous wearable devices

Work by O'Malley and Goldfarb (2001, 2005) and O'Malley and Upperman (2006) have extended on study between real-world and virtual haptic interaction to demonstrate that users can still perceive differences in quality of the rendered VE around them in terms of the perceived hardness of surfaces. Such studies give an indication of the extent to which a device and its accompanying rendered VE can mimic real-world actions or scenarios for humans to perceive a realistic and natural world (Srinivasan, 2002). These studies provide and outline the future challenges and opportunities in haptic technology integration.

5.2 VIRTUAL REALITY

The Uncanny Valley

The uncanny valley is a hypothesized relationship between the degree of an object's resemblance to a human being and the emotional response to such an object. Better understanding how realistic touch sensations can break the VR illusion may help developers create more engaging VE's for games and VR therapy. People feel moderately immersed in a VR experience when their handheld controllers provide no touch sensation or an even buzzing.

Technology review

Prior to the development of the research project a technology review would be conducted to subjectively understand the potential of VR and haptic integration technologies (appendix A).

5.2 GAME ENGINES

Game Engines such as Unity and Unreal play a critical role in providing capabilities for more than what they were designed for, game building. One major capability includes the content creation and programming for VR systems. This is because game engines have the power of fast rendering, real-time interactions, the ability to be programmed and the ability to be compatible with a wide range of file types. Making it a perfect utility for VR systems to be used with whilst enabling users the ability to create highly dynamic, custom and interactive visual experiences. Majority of industry standard headsets such as the Oculus Rift or HTC Vive are compatible with leading game engines as they provide VR/AR templates and examples that have been pre-programmed for the game engine.

5.2.1 Collision detection

VRE's can simulate a physical environment in such as way that humans can readily visualize, explore and interact with virtual objects. Since physical environments are inherently geometric based, algorithms and physics-based engines are built within Gaming Engines and VR programs enable collisions (reference). One important problem with physics-based environments is the ability for real-time interactive collision detection.

5.2.2 Photogrammetry mapping

Photogrammetry mapping is the science of making measurements from photographs. The output is typically a map, drawing, measurement, or a 3D model of a real-world object or scene. The fundamentals of the technique are based on camera location and triangulations which are overlaid to create data points that can be outputted as a mesh or 3D object to visually represent form and shape digitally. Current applications involve drone or aerial mapping of large sites as well as 3d scanning physical objects to have them digitally represented. As VR/AR systems become more proficient in the AEC industry where multiusers environments, virtual objects and avatar creations are involved (reference).

5.2.3 Motion and hand tracking

The Leap Motion system hand tracking system is a most common development tool to use finger dexterity and fidelity. It has been used through applications from gaming, VR/AR, robotics, training, interfaces to rehabilitation applications. The Leap Motion uses infrared technology that sense the movement of your hands above the sensor. The device captures image data and processes it into a 3D virtual representation tracking a user's hands in real time. The Leap Motion's SDK is compatible with a wide range of software's and gaming engines. The SDK allows for developers to incorporate a range of interfaces, rigged hand characters and custom hand animations

6. Case Study

This case study outlines a prototype for integration of haptic technologies to allow immersive and interactive experiences in VE's. The prototype is aimed at allowing clients to interact with virtual objects and allows architects or designers to manipulate their design according to pre-defined constrains in VR. This study involves exposing the methodology for users to interact and manipulate architectural designs within the Unreal Engine VR application. Interaction data from the users within the VE is processed and sent to a serial communication system. The data is to be read by an Arduino microcontroller and output actuated haptic feedback to the user.

6. 1 LEAP MOTION HANDTRACKING

This prototype incorporated methods of tracking hands, finger movements, dexterity and gestures to decide what feedback sent to the user using the VR system. The Leap Motion allows a wide range of SDK’s designed for multiple development software’s, from game engines to scripting languages making it a versatile system to use within VR or AR systems. It is beneficial to start by understanding the capabilities and API systems behind the Leap Motion and its use in game engines particularly for Unreal Engine and Unity.

6.1.1 Rigged Hands Systems and Interfaces

The Leap Motion plugin for Unreal Engine 4 comes with pre-set fabrication assets that allows developers to understand how particular elements work in detail. In Unreal, rigged hands are determined by having mesh geometries being constantly updated inside the engine and are feed 3D location data from a user’s tracked hands in front of the sensor. The plugin consists of prebuilt character blueprints accompanies with rigged animations. Each has a fabrication and material assigned to the chosen character blueprint. This determines the look of the user’s virtual hands on screen and can be assigned in the Map and Modes settings in Unreal. With each rigged hand character in the game engine, physics and collisions can be enabled with also the VR scale adjustment. In testing, the floating hands rigging character was selected as it best portrayed realistic finger movements and provided a high-poly mesh along with real-time collisions with other virtual objects.

6.1.2 Leap Motion Grasping and Gestures

With prefabricated grasping Unreal blueprint editing events, testing was done in allowing a user to trigger an output such as a print string component that allows the user to read on screen when a grasping event has occurred or has stopped. This setup the framework for a Leap Motion event to send data to an Arduino microcontroller through Unreal.

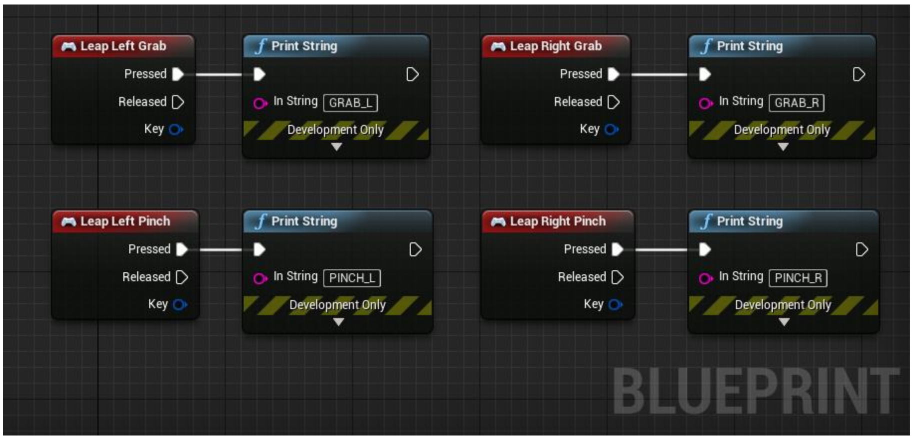


Figure 6. Leap Motion guesture commands

For gestures to be read by Unreal they needed to be enabled which allows the game engine to search for movements of the virtual hands. This allows for gestures to be customized by developers and was used to send data from gesture event components for an actuation of feedback to the user. The actuation would be dependent on data such as pinch distance, bone, finger, hand or arm location, hands or finger widths, direction of members, state of gestures and count of fingers pointing, etc.

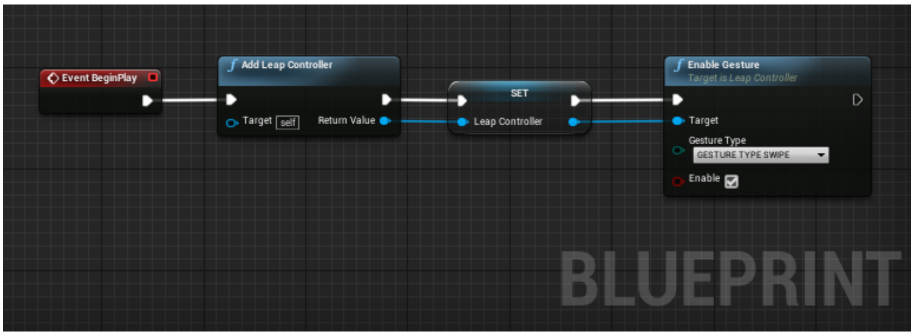


Figure 7. Gesture enable component in blueprint editor

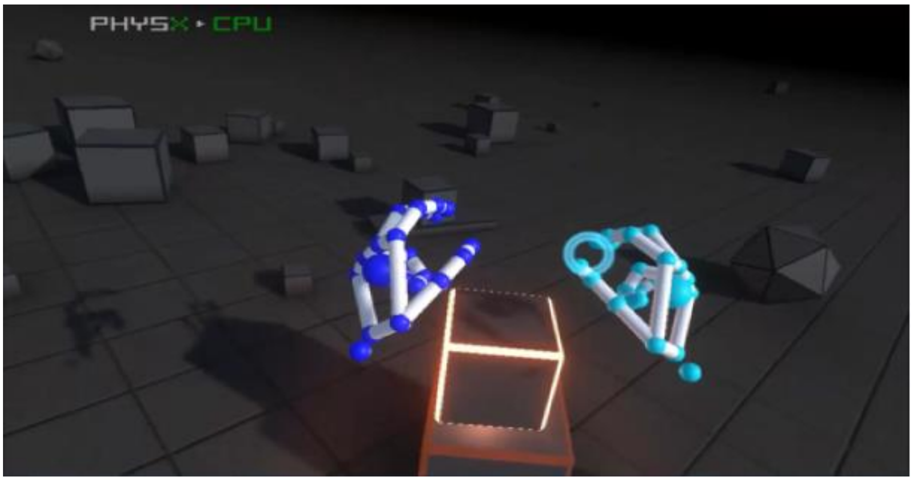


Figure 8. Testing Leap Motions capabilities through Blocks application.

6.2 CONNECTION OF UNREAL AND ARDUINO

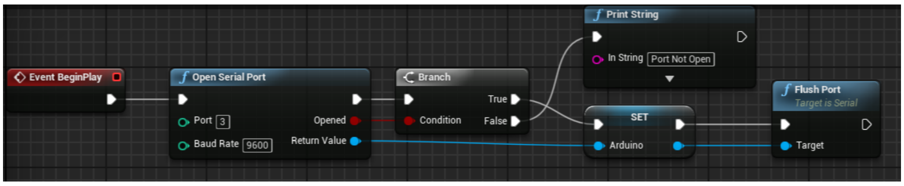
UE4Duino was used to communicate a user's hand interaction in a VE between Unreal Engine and Arduino IDE. UE4duino is a plugin for Arduino COM port communication, this allowed serial data from Unreal Engine to be translated into actuation of haptic feedback to a user via the Arduino COM port.

6.2.1 UNREAL ENGINE UE4Duino scripting

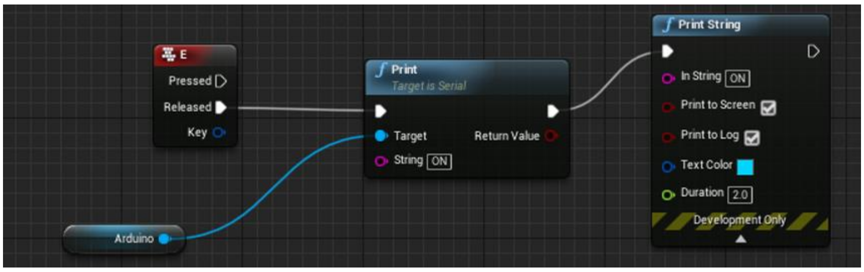
The resources for UE4duino plugin were limited and defined a framework of just opening ports, printing strings and closing ports. This allowed for extensive testing in the process of communication between Unreal and Arduino IDE. Testing the communication involved opening the serial, sending a string to Arduino IDE and closing the port once the scene was stopped in a 3rd person template in Unreal. Opening the serial COM port is suited when the scene begins to play in Unreal and this was tested through the same port as the connected Arduino microcontroller. Printing a string after the serial open component allowed the user to understand if the Arduino is detected. Next was implementing an in-game event to be triggered and to send data through the serial as a string. This involved a simple keyboard input for strings to be sent. Once events were triggered a close serial component allowed the serial port to stop reading data being sent to it once the Unreal scene ended.

The UE4Duino plugin testing contained the following components:

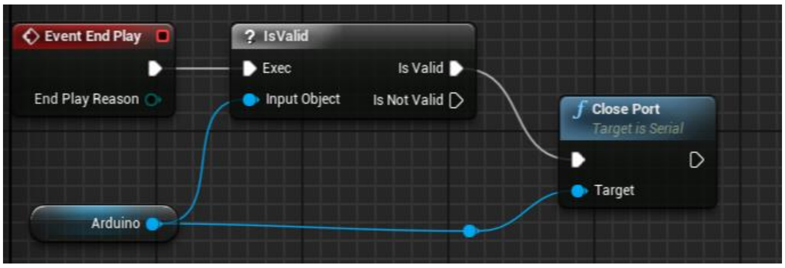
- Open Serial; This component is used to open COM port and was made to trigger a string to print on Unreal scene once play had begun.



- Send and Print Data; This component determined if a keyboard input was triggered and would send a string to the port and print the set serial in-game.



- Close Serial; This component closed the serial and stopped any data to be sent to the port.



6.2.2 Arduino IDE coding

The Arduino IDE was used to allow an output from the data sent from Unreal and actuate a haptic feedback from specific data inputs. The Arduino IDE allows various libraries to be added whilst allowing it to read incoming data from its port. This method consisted of three steps; adding libraries, inputs and outputs pins to the microcontroller, begin serial use whilst setting up the Arduino pins that were used and reading the serial and performing certain actuations according what strings are read within the serial port.

Figure 9. Receiving strings from Unreal in the Arduino IDE

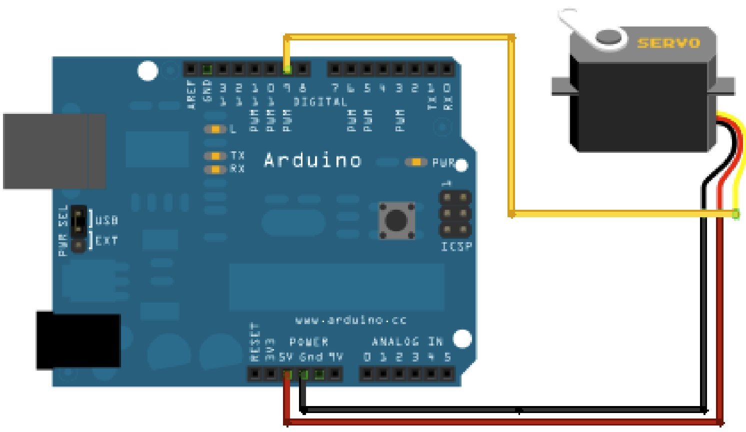


Figure 10: Arduino circuit diagram servo setup.

6.3 CONNECTION OF UNITY AND ARDUINO

Testing with Unity was carried out to determine which game engine would be best suited for the final prototype. The Unity Asset Store includes a plugin ‘Uduino’. It allows serial communication between Unity and Arduino systems. This plugin was tested in partnership with Unity’s Leap Motion plugin to see if the system was more reliable and convenient than using Unreal Engine and UE4Duino. The result was that Uduino included a simple interface to incorporate efficient pin setup but did not allow it to be integrated within the Leap Motion API as the libraries referencing systems collided producing errors in the C# scripting environment.

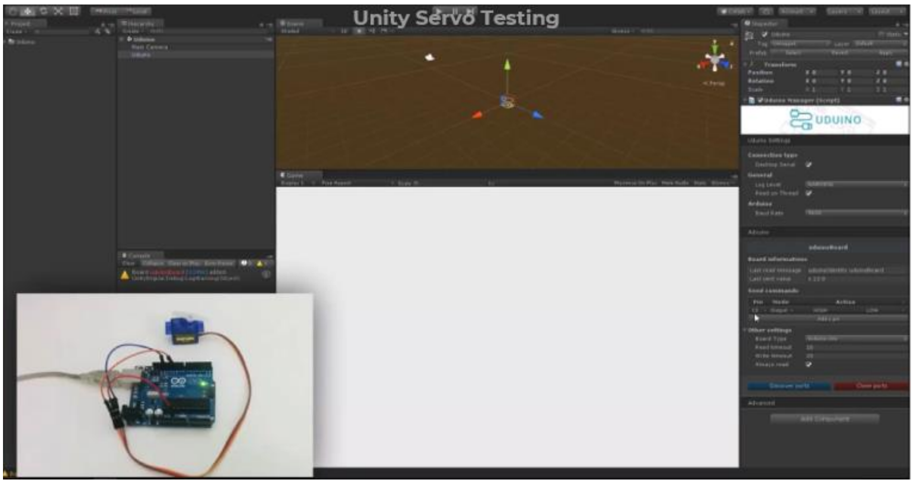


Figure 11. Uduino testing of servo rotation.

6.4 PIN ARRAYS

A pin array mechanism was designed and developed to test the application of texture simulation to a user in a VE. This incorporated the design of a pin array system using a parametric plugin Grasshopper 3D and built using a Wanhao Duplicator i3 3D printer. This enable quick and rapid prototyping with the ability to incorporate quick design changes. The mechanism was designed to be actuated through an attached servo and used a screwing mechanism to raise or lower the pins against a user’s finger tips. With development of the mechanism the issue of scale and the restricted resolution of 3D printing outlined that such development of a micro mechanism could not be produced within the time frame of the research project. This then steered the research focus on determining other ways of producing tactile feedback of simulating texture differentiation to a user in a VE.

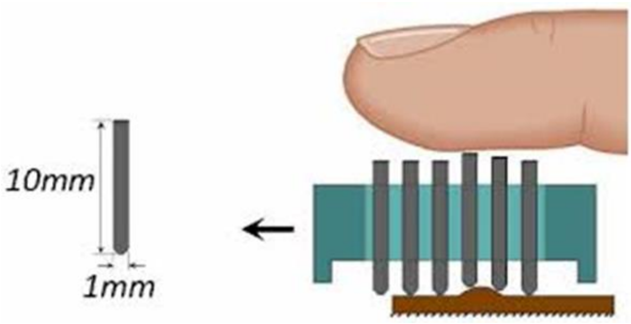


Figure 12. Pin Array Mechanism Design.

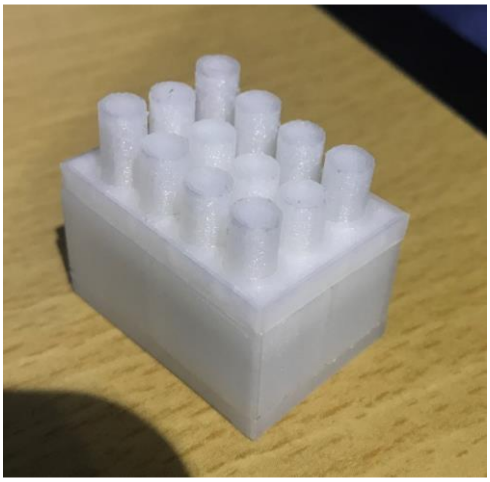


Figure 13. 3D printed pin array mechanism

6.5 VIBROTACTILE FEEDBACK

Through future background research, allowed for the concept of incorporating vibrotactile feedback in the prototype. This then led to the implementation of vibration coin motors, that same device that allows mobile phones producing vibration feedback in a compact design. Incorporating these motors in a glove system design also allowed actuation and speed controlled thorough a Pulse Width Modulation (PWM) signal as the input lead. PWM signal pins are built within the Arduino microcontroller and can be adjusted according to 2N 7000 MOSFET transistor. This component allows for PWM signal strength to be actuated based on a voltage. The design lead to one vibration motor attached to each finger tip of a nylon glove. Each motor consisted of lead wires that were all connected in a parallel connection that connected to the MOSFET transistor and Arduino microcontroller board.

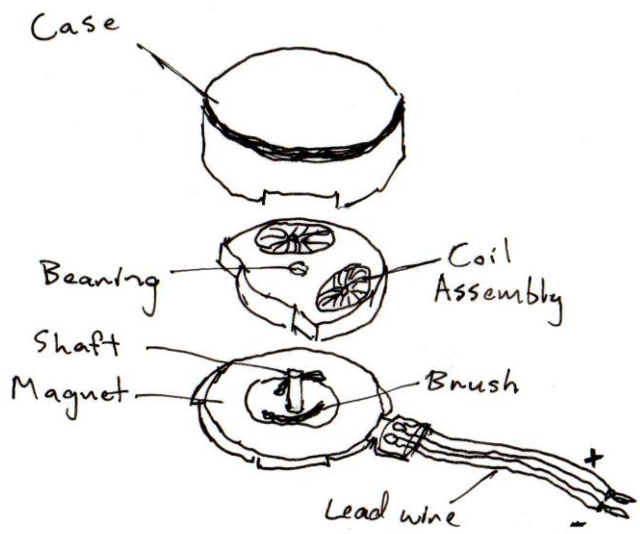


Figure 14. Coin Vibration Motor Structure.

Figure 15. Vibrotactile Glove

6.6 RESISTANCE OR FORCE FEEDBACK

Resistance or force feedback allows a user to receive information of determined actions within an environment. The approach in implementing it within a current vibrotactile glove system explored mechanisms behind applying resistance or force to a user's hand depending to their interaction with virtual objects. This stage in the prototype focused on a user being able to feel real world constrained simulated through resistance of hand and finger closure.

6.6.1 Servo Exoskeleton

An open source design called the "Servo glove" on THINGIVERSE by max09ru was tested and explored in its potential to be incorporated as a force feedback mechanism at a hand scale. The design uses a pivot and leverage mechanism to force a user's hand to open or close at a level controlled through servo and gear rotation. This design was 3D printed and tested for its capability applying force feedback to a user. The design allowed a significant force on a user's hand but performed as a clunky and jittery mechanism. This is something that the final prototype should not include as it lowers a user's ability of hand functions and natural movements. The design forwarded prototyping to investigate a custom design that would incorporate a more reliable and sleeker design and allowed natural hands movements and functions of a user.

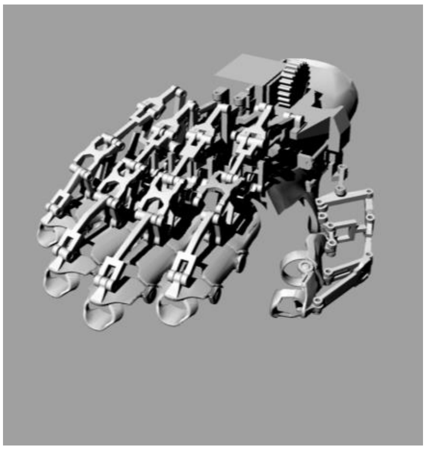


Figure 16. Servo Glove by max09ru.

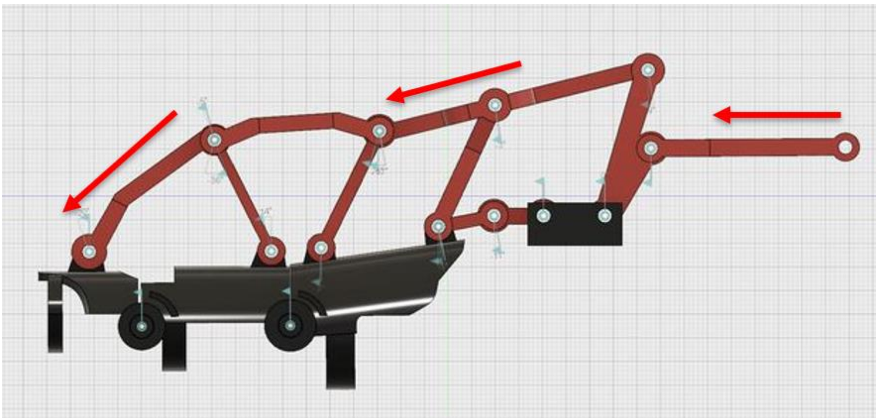


Figure 17. Leaverage Mechanism Design by max09ru.

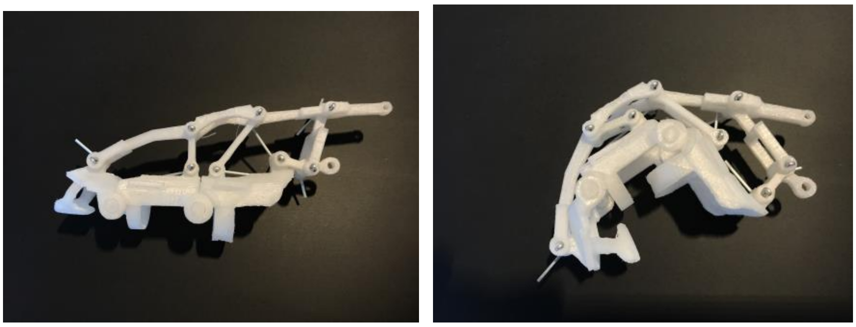


Figure 18. 3D printed mechanism for testing.

6.6.2 Channel pulley mechanism

The custom development design lead to the prototype of a channel pulley mechanism that used the hands natural knuckle joints to be used as channel guides for a fixed string attach to the glove fingertip ends and to the leverage arm of a servo. The design is to apply resistance throughout the channels and onto the user’s hands to limit their hand closure. This purpose was to simulate shape and volume of virtual objects. The mechanism allows thin and strong string to be pulled through each channel with ease, so the servo can adjust and be actuated smoothly. The servo was secured using Velcro fastening system on a user’s wrist. Testing this designed mechanism determined a better solution on applying resistance to a user’s hand with a confined and lower profiled design than the servo glove.

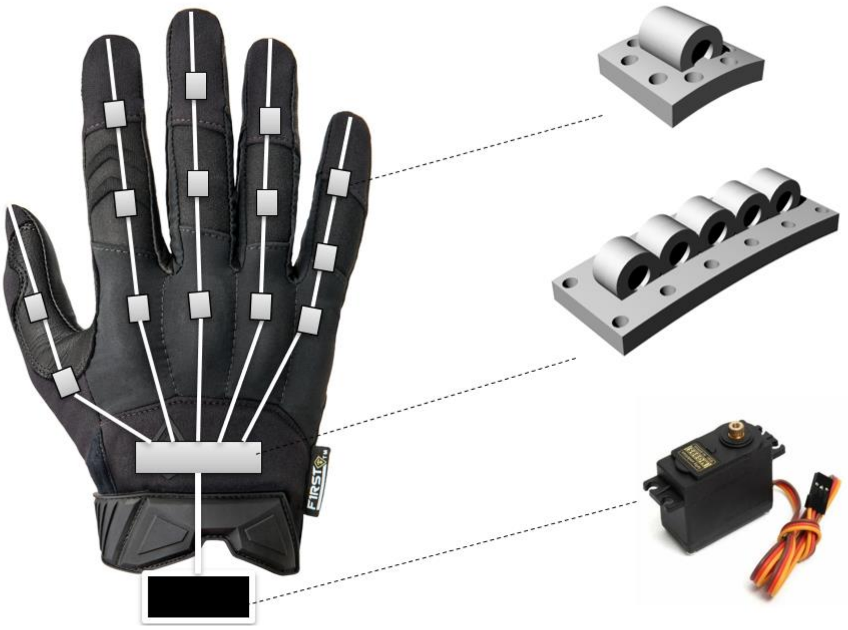


Figure 19. Force Feedback Channel Mechanism Design

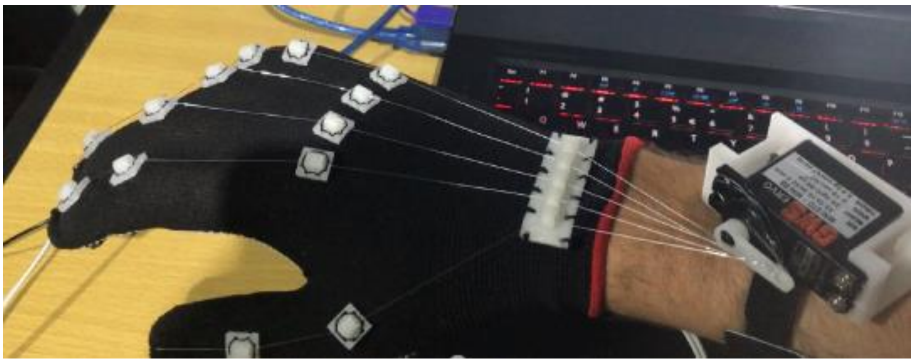


Figure 20. Channel Mechnism Protoype Testing

6.7 FINAL PROTOTYPING

6.7.1 Haptic glove prototype

The final prototype included the method of collaborating the vibrotactile glove design with the force feedback channel mechanism into one system. Extension cables were made for compatibility between the Arduino microcontroller and the servo and vibration motors with also having a significant amount if length allowing as users hand not be constricted when wearing the prototype. The Leap Motion uses infrared sensors to determine depth or height of hands and position of finger joints. This makes it difficult to detect dark fabrics such as gloves on hands, so to work around this limitation introduced the application of inserting that haptic glove prototype inside of a white nylon glove. This allowed similar tracking precision than compared with having skin visible to the Leap Motion sensor.

Figure 21. Final Haptic Glove Prototype.

6.7.2 Arduino and Unreal setup and communication

Setting up the Unreal project included the incorporation of importing various objects with different volumes, so a user could test different actuations of the resistance feedback. This involved importing a FBX file inside the Unreal project and allowing physics and collision generation as it needed to be recognised as a static mesh actor to have an interactive interface with the Leap Motion controllers in the environment. Next stage was to link grabbing events with a designated grabbing strength to allow the user to pickup virtual objects in the Unreal Engine environment. These grabbing events were assigned to an individual object and linked to a UE4Duino component that would communicate which object is grabbed within Unreal to the Arduino com port. After this process occurs the

Arduino IDE reads the incoming object grasp ID and outputs a signal to the desired pins connected servo and vibration motors. According to the objects ID determined the amount closure a user could have caused through servo actuation. The user picking up a virtual object with a large volume or shape restricted their hand to minimal closure, whereas picking up a small volume object allowed maximize closure of a user's hand. This actuation simulates what shape their hand would be grasping the object physically in real life. With the object ID's being read by the Arduino IDE consisted of variable speed actuation of the vibration motors that simulated feedback of differentiated surface interaction. Objects with assigned smooth surfaces included a consistent and even rumble of the vibration motors where as objects with assigned rough surfaces feedback a jittering and uneven rumble. The collaboration of a vibrotactile and force feedback system determines a real-time feedback to a user's hand being tracked constantly through the Leap Motion sensor.

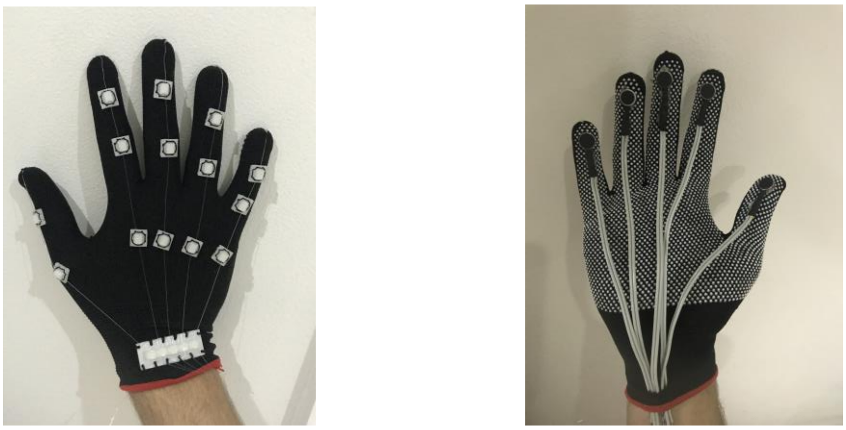


Figure 22. Collaboration Design of Force Feedback and Vibrotactile Designs.

Figure 23. Real-time Testing with Haptic Glove Prototype and Unreal Project.

7. Significance of Research

The main objective of this research is to situate a methodology, for using haptic technologies VR. New users to VR are heavily confronted by hardware and usability issues, to this research includes user interaction with virtual objects and the ability to design using haptic feedback systems. Progressively VR is being more used in architecture at the end of the design process. With this architects and designers design through simply designing through ‘clicks’, having a feedback enables the user to obtain information on what is happening in the VE around them. This research provides the significant opportunity of extending the application of haptic technologies into VR systems allowing more compelling and immersive experiences. This research determines the avenue of incorporating feedback design in the design process using VR systems. This could potentially allow architects to manipulate designs and determine predefined constraints through haptic feedback. There is also another use where clients can have the potential to experience VR designs through interacting with virtual objects.

8. Evaluation of Research Project

In evaluating the research project, the prototype produces has extensively investigated and explored the integration of haptic technologies in VE's. The prototype explores a methodology of mediating the interaction between virtual objects and the user in a VE. With more time made available to certain areas, a lot more detail could have been emphasized in the development of mechanism design and functionality of VR system setups. A lack of interoperability between software or plugins was a major obstacle as it obstructed at times. The lack of resources that outline function and use of plugins used was also an obstacle. Future work includes the developments of a full collaborated and integrated haptic product partnered with an application for commercial VR systems. This would consist of the collaboration with PTW and Aron Sheldon's research study (figure reference) that compliments both sides of research with the objectives of improving usability and reducing complexity of VR and AR systems, allowing user and VE to communicate effectively in design practice. In the scope of this research project there is a sound correlation in the understanding of VR and haptic technology that contributes to its acceptance in mainstream industries. With further development and user testing of this research, we will see an increasing relationship between VR and haptic system integration within the design practice.

9. Conclusion

Conclusively to the research, the outcomes have reflected the overarching goals in defining, relevant research methods in VR and its relation to haptic technology systems. The objective of this research has set out to explore the use of haptic technologies in a VE providing a user to interact with the surrounding context. The prototyped haptic glove system allows a user to determine a level of sensory feedback with touch to determine the communication between the user and VE. Through the iterative steps taken within the methodology in achieving user interaction and feedback in VR expresses how future developments of sensory interaction and feedback can support design in the built environment. This research determines the level of integration needed to allow significant feedback from the VE through the mediation enabled by haptic technology systems.

Acknowledgements

Giving my full appreciation and gratitude to UNSW Course Coordinator, Professor Mathias Hank Haeulser for the ongoing support, ss well as supportive staff: Nicole Gardner and Alessandra Fabbri,. I would like to give thanks to PTW Architects' supportive mentor; Tiara Dobbs for her ongoing support in the development of this research. Lastly, I would like to thank Aron Sheldon, and his participation within the research project. \

References

Biocca, F., Levy, R. M. (1995) Communication in the Age of Virtual Reality. Lawrence Erlbaum Associates, Publishers. Hillsdale, New Jersey.

El Saddik, A., Orozco, M., Eid, M., Cha, J. (2011) Haptics Technologies: Bringing Touch to Multimedia. Springer Heidelberg Dordrecht. London, New York.

Mihelj, M., Novak, D., Beguš. (2014) Virtual Reality Technology and Applications. Springer Dordrecht Heidelberg. London, New York.

Kumar, S., Bhavani, S. (2017) A New Dimension of Impressiveness into Virtual Reality through Haptic Technology. IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI), India.

Srinivasan, A. M. (2005) What is Haptics? Laboratory for Human and Machine Haptics: The Touch Lab. Massachusetts Institute of Technology, Massachusetts.

O'Malley, K. M., Gupta, A (2008) Haptic Interfaces. Morgan Kaufmann Publishers Inc. San Francisco.

Bermejo, C., Hui, P. (2017) A Survey on Haptic Technologies for Augmented Reality. ArXiv, Cornell University Library, Ithaca, New York.

Otadury, A. M., Okamura, A., Subramanian, S. (2016) Haptic Technologies for Direct Touch in Virtual Reality. URJC Madrid.

Jang, S., Kim, H. L., Tanner, K., Ishii, H., Follmer, S. (2016) Haptic Edge Display for Mobile Tactile Interaction. CHI Conference on Human Factors in Computing Systems. San Jose, California.

Fitzgerald, D., Ishii, H. (2018) Mediate: A Spatial Tangible Interface for Mixed Reality. CHI Conference on Human Factors in Computing Systems. Montreal, Quebec.

Gi-Hun Yang, Ki-Uk Kyung, M. A. Srinivasan and Dong-Soo Kwon, "Quantitative tactile display device with pin-array type tactile feedback and thermal feedback," *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, Orlando, FL, 2006, pp. 3917-3922.

M. Konyo K. Akazawa S. Tadokoro and T. Takamori "Tactile feel display for virtual active touch" IEEE/RSJ Int. Conf. Intelligent Robotics and Systems pp 3744-3750 2003

Kyung K.U. Son S.W. Kwon D.S. Kim M. "Design of an integrated tactile display system" Proceedings of IEEE International Conference on Robotics and Automation. New Orleans USA pp. 776-781 2004

Alvina, J. et al. OmniVibe: Towards Cross-body Spatiotemporal Vibrotactile Notifications for Mobile Phones. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI '15, ACM, 2015, 2487–2496.

CyberGlove Systems Inc. CyberGrasp Glove. http://www.cyberglovesystems.com/cybergrasp. Last accessed: 12.09.2018.

Massie, T. H., and J. K. Salisbury. The PHANToM Haptic Interface: A Device for Probing Virtual Objects. In Proceedings of the ASME Winter Annual Metting ’94, Dynamics and Control 1994, 1994, 295–301.

Hayward, V., Astley, O., Cruz-Hernandez, M., Grant, D., and Robles-De-La-Torre, G. 2004. Haptic interfaces and devices. Sensor Review, 24, 1. 16–29.

Image References

Figure 5: Haptic system: interaction between a human and the haptic interface represents a bidirectional exchange of information (Mihelj, 2014)

Figure 12: Feeling through Tactile Displays: A Study on the Effect of the Array Density and Size on the Discrimination of Tactile Patterns - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/Passive-tactile-arrays-with-pin-array-sizes-of-1-cm-2-pins-diameter_fig1_224205653 [accessed 12 Nov, 2018]

Figure 2 and 6: By Tiara Dobbs, 2018

Figure 1,3,4,7,8,9,10,11,13,14,15,16,17,18,19,20,21,22,23:
The author,

Appendix

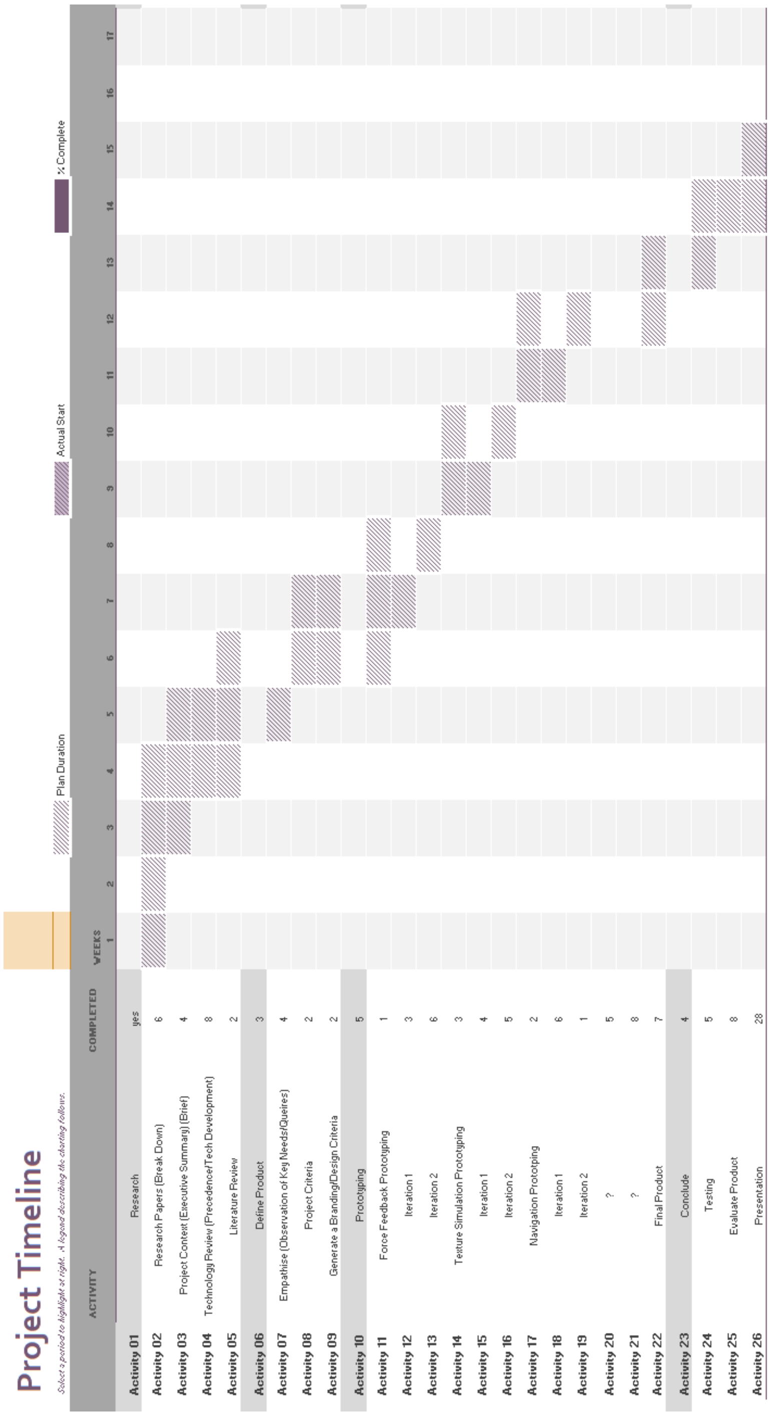
Appendix A. Technology Review

Appendix A

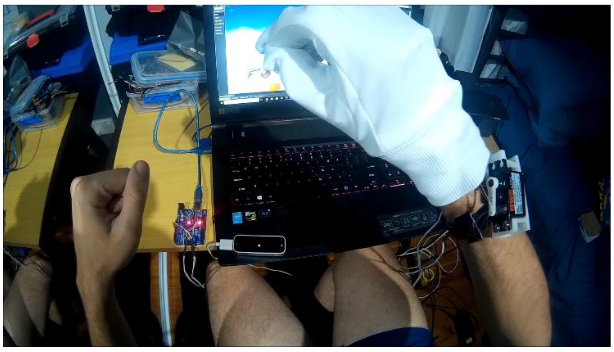
Technology/software	*Features*	*Positives*	*Negatives*
HTC Vive Pro	• *Dual lens camera on headset* • *Room scale tracking* • *6 DoF controllers*	• *Higher resolution than other virtual reality headsets*	• *Complicated and heavy setup with 3 outputs* • *Needs a separate power source other than pc being used* • *Large and heavy controllers* • *Heavy headset* • *Expensive relative to Oculus Rift*
Oculus Rift	• *Dynamic and smaller grasping controllers* • *Removable headphones*	• *Simple setup with HDMI and USB 3.0 as outputs* • *Quick and easy setup* • *Compatible in a small working space*	• *No safety features only warnings* • *Beneficial to workspaces that are particularly small*
Unreal Engine 4	• *Visual editors* • *Visual blueprint scripting using C++* • *Visual debugging*	• *High quality rendering and lighting capabilities* • *Higher graphics performance* • *Minimal knowledge of programming needed* • *Suitable for prototyping* • *Price is reasonable with quality*	• *Not suitable for simple projects* • *More work needed for projects due to vast amount of options*
Unity	• *Written in C# for productive development*	• *Cross-platform integration* • *Convenient for VR/AR and mobile devices* • *Simple interface and structure* • *Large community and forums for resources*	• *Slow rendering speeds* • *Low graphics performance and capabilities*
Leap Motion	• *Uses infrared technology*	• *Low price for a hand tracking device* • *Sleek, small design*	• *Too sensitive at times* • *Doesn't pick up particular glove materials or colours* • *Sensor region is smaller than kinect*

		• *Wide range of compatibility for apps and software's and game engines* • *Sensitivity*	
Microsoft Kinect	• *RGB colour VGA camera* • *Depth sensor* • *Multi-array microphone*	• *Wide sensor*	• *Sensitivity is weaker than Leap Motion*
Arduino	• *Designed as basic controller* • *Wide range of libraries*	• *Cheaper than Raspberry Pi* • *Easy integration of sensors* • *Minimal setup* • *Can be battery powered*	• *Isn't a complete computer* • *Suitable for hardware-based tasks*
Raspberry Pi 3 B	• *Full programable computer*	• *Suited to software and software developers* • *Can run Linux, Android and Windows 10*	• *Relatively expensive* • *SD card needed*
UE4Duino	• *Plugin for Unreal to Arduino communication* • *Allows current nodes to be connected in visual scripting editor*	• *Provides communication through visual scripting C++* • *Free plugin* • *Minimal knowledge required*	• *Minimal resources and information on its implementation in Unreal*
Uduino	• *Plugin for Unity to Arduino communication* • *Uses C# scripting*	• *Wide range of resources and tutorials online* • *Ability to be used with other libraries easily*	• *Costs $17 US* • *Extensive C# knowledge required*

Appendix B. Project Timeline



Appendix C. Screen Captures of Final Prototype

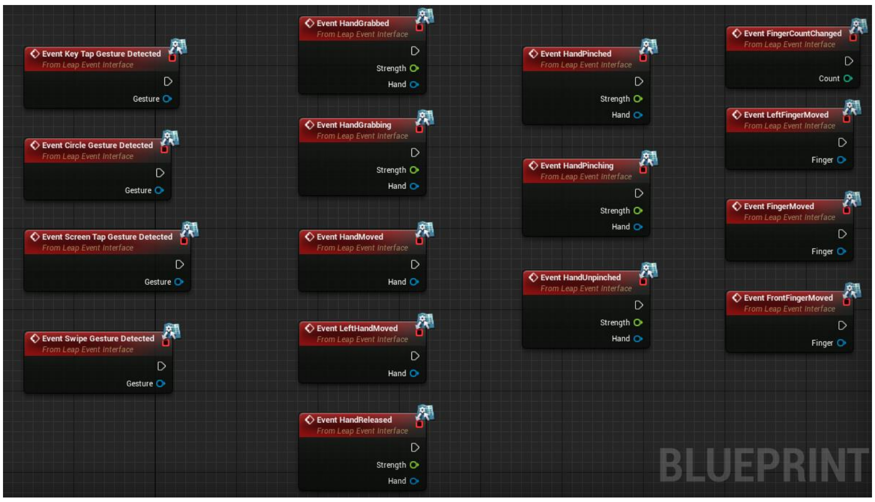


Appendix D. Command Arduino Outputs

Appendix A

Event Action Collision	Command Sent (Input)	Vibration Motors (Output) PWM	Servo Angle (Output) Degree
Rough Surface	Intensity1	255	
Even Surface	Intensity2	180	
Smooth Surface	Intensity3	100	
Large Object	Highresistance		180
Medium Object	Mediumresistance		120
Small Object	Lowresistance		45

Appendix E. Leap Motion Event Components



Appendix F. Expenses Table

Appendix A

Product/s	Qty	Price (AUD)
Leap motion controller	1	$169
Fabric gloves	4	$15
Wiring	1	$7
Vibration motors	19	$15
Servo	2	$30
Velcro straps	5	$5
Uduino	1	$17
Arduino board	1	$30
3D printing filament	1	$29
Total		$317

Appendix G. Haptic Interface Types

Haptic Interface Types	Pros	Cons
Ground-referenced	• *Can produced high levels of force if needed* • *Don't have to wear them* • *Accurate trackers*	• *Limited movement when using them* • *Safety concerns*
Body-referenced	• *More freedom of motion than ground-referenced* • *Provide more control with direct manipulation of virtual objects*	• *User has to bear weight of device* • *Can be burdensome to put on*
Tactile	• *Smaller and more lightweight when compared to force displays* • *Useful for simulating touch* • *Can provide vestibular sensations*	• *Difficult to get sensations correct* • *Often stimulate only a small area of skin*
In-Air	• *Do not have to wear a device* • *Sensations can be located anywhere*	• *Does not provide precise force or tactile information* • *Limited range*
Combination	• *Useful for combining tactile and force feedback*	• *More complex* • *Can be burdensome to wear*
Passive	• *Useful when haptics is required for a specific object or physical proxy* • *Easy to design and use* • *Increase sense of realism*	• *Constant force and tactile sensation* • *Limited by specificity*

Appendix H (Glossary of terms)

VR	Virtual Reality
AR	Augmented Reality
MR	Mixed Reality
VE/VE's	Virtual Environments
UI	User Interface
UX	User Experience
AEC	Architecture, Engineering, Construction
BIM	Building Information Modelling
CGI	Computer Generated Imagery
API	Application Program Interface
SDK	Software Developer Kit
IDE	Integrated Development Environment

Appendix H. Glossary of terms used throughout the paper.