

People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
University M'Hamed BOUGARA – Boumerdes



Institute of Electrical and Electronic Engineering
Department of Electronics

Final Year Project Report Presented in Partial Fulfillment of
the Requirements for the Degree of

MASTER

In Electronics

Option: Computer Engineering

Title:

**Wearable Haptic Grasping Device for
Virtual Reality equipped with
Kinaesthetic, Cutaneous and Thermal**

Presented by:

Bentaleb Ahmed

Supervisors:

Dr. Benbelkacem Samir

Dr. Zitouni Abdelkader

Abstract

To grasp and manipulate objects in the real world, humans rely on fingertip contact pressure and kinaesthetic feedback of finger positions to determine proprioceptive, shape and cutaneous feedback of the object grasped. In order to elaborate haptic interfaces that can provide a realistic grasping experience we should support these modalities and produce similar forces on a user's hands. Current virtual reality (VR) and augmented reality (AR) systems can render realistic 3D objects visually, but most lack the ability to provide an environmental feedback and realistic haptic experiences.

In VR and AR context, haptic and environmental feedbacks devices must become more compact, lightweight and mobile. However, it remains challenging to group these feedbacks. A variety of wearable and portable haptic devices have been investigated with different types of exoskeletons, gloves and handheld controllers to provide users interactive and immersive environments in VR and AR. While these devices provide haptic and kinaesthetic feedback still remains a challenge. While there are conflicting constraints of mobility and performance, an effective haptic device would be a compact and lightweight mobile device with the ability to provide the necessary forces.

The aim of this project is to propose a prototype of wearable haptic feedback for grasping in VR supporting kinaesthetic, cutaneous and thermal feedback.

Dedication

In the name of Allah, the most beneficent and merciful. I devote this work to my parents who I share my joys and sorrows, my family, my teachers and friends who believed and supported me.

Acknowledgement

First and Foremost, praise is to ALLAH, the Almighty, the greatest of all, on whom ultimately, we depend for sustenance and guidance. Special appreciation and gratitude go to my supervisor, **Dr. Samir Benbelkacem** for his supervision and constant support. His invaluable help of constructive comments and suggestions throughout the experimental and thesis work have contributed to the success of this research. Furthermore, I would like to highlight my gratitude to all the teachers of the institute especially **Dr. Abdelkader Zitouni**, for their work and their support during these past five years.

Last but not least, my warmest gratitude goes to CDTA and Wameedh Scientific Club which always provided the most suitable working environment without which this thesis would not be in its present form.

Table of Content

Abstract	I
Dedication	II
Acknowledgement	III
Table of Content	IV
List of Figures	VI
List of Tables.....	IX
List of Abbreviations.....	X
Introduction	1
Chapter I: Wearable devices for Virtual Reality	2
I .1 Virtual Reality	2
I .2 Psychology of Virtual Reality	4
I .3 State of the Art	5
I .4 Our Device	5
Chapter II Design and Development of HapLeap.....	7
II .1 Hardware Design	7
II .1.1 Electrical Parts.....	7
II .1.2 Printed Circuit Board Design.....	11
II .1.3 Mechanical Structure	14
II .1.4 Virtual Reality Headset	17
II .1.5 Leap Motion Controller.....	18
II .2 Software Design & Implementation	19
II .2.1 Unity Game Engine	19
II .2.2 Leap Motion SDK.....	20
II .2.3 OpenVR SDK.....	21
II .2.4 System Diagram and Flowchart	22
II .2.4 Hand Detection and Interaction	25

II .2.5 Haptics Manager	29
II .2.6 Servo Motor Control Signal.....	30
II .2.7 Tactors Control Signal	31
II .2.8 Peltier Control Signal.....	31
II .2.9 Temperature Feedback	32
Chapter III Experiment and Evaluation.....	34
III.1 Experiments	34
Experiment 1	34
Experiment 2	35
Evaluation.....	35
Conclusion.....	37
References	38
Appendix A: Leap Motion.....	39
Appendix B: Arduino Nano.....	40
Appendix C: Evaluation Data	41
Appendix C: Project Link	42

List of Figures

Figure 1. 1: Stanley G. Weinbaum’s 1935 short story.	2
Figure 1. 2: Sensorama first VR device.	3
Figure 1. 3: Ivan Sutherland HMD.	3
Figure 2. 1: Arduino Nano.	7
Figure 2. 2: Vibro Tactile Motors.	8
Figure 2. 3: Thermoelectric Peltier TES1-03102.	8
Figure 2. 4: L293D H-Bridge.	9
Figure 2. 5: 10K NTC Thermistor.	9
Figure 2. 6: Servo Motor SG90.	10
Figure 2. 7: HC-05 Bluetooth Module.	10
Figure 2. 8: UART Module.	11
Figure 2. 9: HapLeap Circuit Diagram.	12
Figure 2. 10: UART module and HC-05 Wiring Diagram.	13
Figure 2. 11: HapLeap PCB Layout.	13
Figure 2. 12: HapLeap Test Circuit.	14
Figure 2. 13: Retractable Pulley.	14
Figure 2. 14: HapLeap Ring 3D Model.	15
Figure 2. 15: Anet A6 3D Printer.	15
Figure 2. 16: 3D printed Ring.	16
Figure 2. 17: Ring Finger. (a) Tactile Motor wires. (b) Peltier wires.	16

Figure 2. 18: Mechanical assembly of HapLeap.....	17
Figure 2. 19: Oculus Rift S HMD.	18
Figure 2. 20: Leap Motion Controller.	19
Figure 2. 21: Unity's user interface version 2019.3.1f1.	20
Figure 2. 22: Leap Motion SDK web page.	20
Figure 2. 23: Oculus Unity SDK web page.....	21
Figure 2. 24: View from the combination of Leap Motion and Oculus Rift test.	21
Figure 2. 25: General Diagram of HapLeap Application Work Flow.....	22
Figure 2. 26: Interaction Processing Flowchart.	23
Figure 2. 27: General Diagram of HapLeap.....	24
Figure 2. 28: HapLeap Side Flowchart.	24
Figure 2. 29: Leap Motion hand bone hierarchy.	25
Figure 2. 30: Hovering check from Hovering API in Interaction Behavior script.....	25
Figure 2. 31: Grasping check from Grasping API in Interaction Behavior script.....	26
Figure 2. 32: Transform Component.....	26
Figure 2. 33: Mesh Filter Component.	26
Figure 2. 34: Collider Component.....	27
Figure 2. 35: Mesh Renderer Component	27
Figure 2. 36: Rigid Body Component	28
Figure 2. 37: Interaction Behavior Component.....	28
Figure 2. 38: Material and Temp Component	29
Figure 2. 39: Serial COM Component	29
Figure 2. 40: Serial Wrapper Component	29

Figure 2. 41: Data Definition in Serial Wrapper Component	30
Figure 2. 42: Query Hand Component	30
Figure 2. 43: NTC thermistor analog signal circuit.	32
Figure 3. 1: User view inside experiment 1.	34
Figure 3. 2: Experiment 2.(a) HapLeap Setup. (b) User view.....	35
Figure 3. 3: Usability Evaluation.	36
Figure 3. 4: Subjective Evaluation.	36

List of Tables

Table 2. 1: Most used VR headsets.	18
---	----

List of Abbreviations

VR Virtual Reality.

AR Augmented Reality

HMD Head Mounted Display

3Dof 3 Degrees of freedom

6Dof 6 Degrees of freedom

UART Universal Asynchronous Receiver-Transmitter

PCB Printed Circuit Board

SD Standard Deviation

Introduction

In Virtual Reality, more effort should be made to promote natural user interactions as we can think of stimulating a part of the human body to simulate real world interaction in a natural way with the use of on hand wearables, in which they are used in our daily life, from medical applications, watches, to became a friendship symbol. This type of wearable feels natural to most of humans, beside Nukarinen et al [1], concluded that using this type of wearables through a wrist seems more promising area for human natural interaction.

In the other hand, simulations and computer based training has been extensively explored, as the positive side of such training is the safety and the margin of mistakes will not risk the human life, for example training new fireffights in simulation or even training a student in virtual lab that contains the real world work environment is much safer, as we can see that pilot simulation is used for long period before a pilot can fly with real plane, but still one issue, which is the simulation only provides 3D dimension view plus sound with no sensation which is important to drive the human instinct, like feeling the vibration of an earthquake, the smell of fire or the temperature of the room.

In this work we proposed and developed a system that will add extra 3 sensation dimensions which are Kinaesthetic, Cutaneous and Thermal sensation.

Chapter I: Wearable devices for Virtual Reality

I.1 Virtual Reality

Virtual reality (VR) is considered as a simulated experience that can be similar to or completely different from the real world [2]. Unlike traditional user interfaces, VR places the users inside the experience, instead of viewing a screen in front of them, users are immersed and able to interact with the 3D world around them in a similar manner as interacting with the real world by simulating as many senses as possible (vision, hearing, touch ...). Also, we can consider VR as the medium that with it we can simulate other mediums, we can read a book, watch television, even wear another VR headset.

The notion of 'virtual reality' came via science fiction with Stanley G. Weinbaum's 1935 short story 'Pygmalion's Spectacles' [3], this vision, goggle-based contraption offered users a holistic holographic experience.



Figure 1. 1: Stanley G. Weinbaum's 1935 short story.

In 1957 Cinematographer Morton Heilig developed 'Sensorama' as a culmination of his quest to build a more complete cinema experience. His arcade-type booth is considered the world's first virtual reality machine. In terms of navigability; the rotating chair, combined with the stereoscopic imagery, granted the user some control over the

experience, nothing comparable to VR as we know it today, yet otherworldly nonetheless [3].

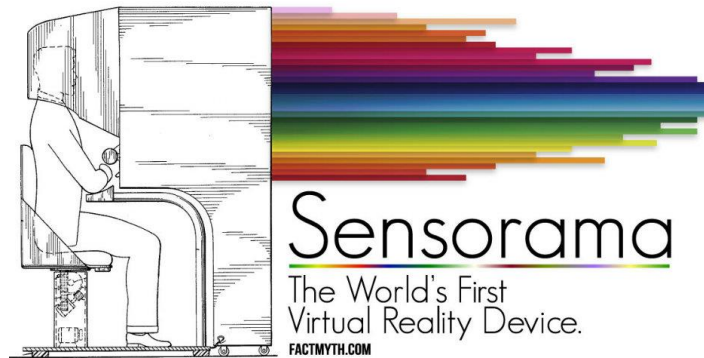


Figure 1. 2: Sensorama first VR device.

In 1960 Heilig went on to develop the Tele-Sphere Mask, considered as the first VR Head Mounted Display (HMD) [3].

Later on in 1965 American computer scientist Ivan Sutherland presented the notion of ‘the absolute display ‘. after that Sutherland created the Sword of Damocles in 1968, which was a mechanical tracking system and not the actual HMD [3].



Figure 1. 3: Ivan Sutherland HMD.

During the years many researches and institutions create different concepts for HMDs and VR to the form of VR headsets we know nowadays.

I.2 Psychology of Virtual Reality

For a virtual reality experience to be realistic and give the illusion that we are in different reality it must meet the standards of immersion which are 3 types of illusion.

Place Illusion

This illusion refers to the feeling that the user is being somewhere else. This virtual place may be a realistic place or a fantasy place. In both cases, if the user feels transported to this other location, place illusion has been successfully checked. In VR, this is achieved with the interactive rendering of the scene, if the user rotates the head, the scene will be shown accordingly with high-quality graphics. Moreover, HMDs allow the user to isolate visually from the environment, in contrast to a common screen, eliminating any distractions from the surroundings [4].

Plausibility Illusion

This illusion refers to how much the user believes that the events happening in the virtual scenario are actually occurring. For instance, a user tries to talk to a virtual character in a shop and the character doesn't respond back, in this case the plausibility Illusion is broken, but if the character responds back we say that the illusion is successfully checked [4].

Embodiment Illusion

Referred to as Body Ownership which is the feeling that user has of being in a different (virtual) body. This type of illusion is achieved by placing an avatar at the right place, for instance if user is wearing an HMD and looks down, the users expects to see a body that copy the movement of the user. In our work a virtual hand is placed in our environment and the user's hand movement is mapped to that virtual hand [4].

I.3 State of the Art

VR technology has come to be the new trend in computer interfaces but one issue makes the difference from standard computer interfaces, which is the user immersion. However, more effort should be made to promote natural user interfaces. As Prof. Robert J. Stone mentioned in his work [5] that a long history of experiments in haptics from the early 1968 in many fields such as nuclear industry, medical application and virtual reality which delivered a wide range of opportunities for researcher and commercial development that we know nowadays.

Our Focus is in haptic wearables or knows as smart wearables, as we see important role of stimulation where humans sense the warmth, cold, contact or pain by stimulating the end the skin, this is achieved using different type of stimuli systems, for example contact and pressure display and Peltier matrix, but those systems need to follow a design standard [6] such as,

Form factor: Wearable devices should alter the body size of the wearer as little as possible.

Weight: Wearable devices should tire the wearer as little as possible.

Impairment: Wearable devices should limit the motion of its wearer as little as possible.

Comfort: Wearable devices should be comfortable to wear and easy to adapt to the wearer size and shape.

I.4 Our Device

As researchers proposed different prototypes for smart wearables for virtual reality [6], they were heavyweight, bulky and wired external control system. Also, only 1 or 2 dimensions of sensations were provided to users. In our work, we proposed an optimized structure to stimulate the user's hand with 3 dimensions of sensations, named **HapLeap**, which is a wearable haptic grasping device equipped with Kinaesthetic, Cutaneous and Thermal Feedback.

HapLeap is a real-time on hand haptic device for stimulating the physical materials in a virtual environment, with the use of tactile motors to generate different vibration levels for different physical materials. Furthermore, our device allows users to grasp and interact with objects in a natural way, by blocking the fingers motion whenever the user grasps an object to simulate the force effect on the hand.

In addition, a couple of Peltier modules were used to simulate the temperature of the object being interacted with. Further details of the design and implementation of our device will be discussed on the next chapters as we will walk through the process of developing such device starting with the electronic components, device design, implementation and testing phase and finishing with evaluation of our system.

Chapter II Design and Development of HapLeap

II.1 Hardware Design

II.1.1 Electrical Parts

II.1.1.a Arduino Nano

The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328 micro-controller (Arduino Nano 3.x) operating at 16MHz frequency with 32 KB flash memory, 1KB EEPROM, 8 10-bit analog input, 22 digital input/output (6 of which are 16-bit PWM output), as well as standard and advanced communication interface: I2C, SPI, UART, mini USB and two external interrupts in which it will be used as our controller for the device.

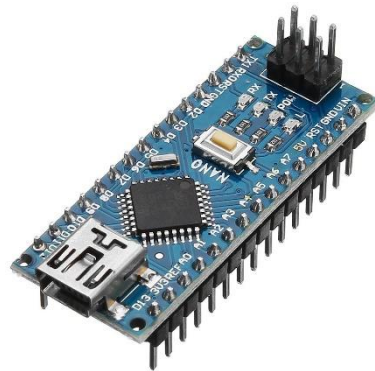


Figure 2. 1: Arduino Nano.

II.1.1.b Vibro Tactile Motors

This type of motors is used in smartphones for haptic feedback (in ringing, notification etc.) and the use of advanced vibration patterns and wave-forms to convey information to a user or operator is called haptic feedback. In our case, the vibrato tactile motors were used to indicate the type of the surface or the material that the user is currently touching with the virtual hand, the information starts when the user moves the real hand and

touches a surface or object in virtual environment then the data is transferred to the micro-controller as a value that ranges from 0 in which indicates a soft surface, up to 255 which indicates a rough surface.



Figure 2. 2: Vibro Tactile Motors.

II.1.1.c Thermoelectric Peltier TES1-03102

Thermoelectric coolers operate according to the Peltier effect. The effect creates a temperature difference by transferring heat between two electrical junctions. A voltage is applied across joined conductors to create an electric current. When the current flows through the junctions of the two conductors, heat is removed at one junction and cooling occurs. Heat is deposited at the other junction. The main application of the Peltier effect is cooling. However, the Peltier effect can also be used for heating, in our case this system is used to indicate the temperature of objects in the virtual environment the TES1-03102 operates with 3V DC and 2A with size of 15*15 mm.



Figure 2. 3: Thermoelectric Peltier TES1-03102.

II.1.1.d L293D H-Bridge

An H-bridge is an electronic circuit that switches the polarity of a voltage applied to a load. These circuits are often used in robotics and other applications to allow DC voltage to run forwards or backwards and this will be used to control the Thermoelectric Peltier in order to heat or cool the area of the finger.

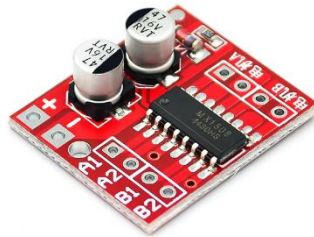


Figure 2. 4: L293D H-Bridge.

II.1.1.e Temperature Sensor

For the temperature feedback an 10K NTC thermistor is used to control the switching state of the H-Bridge.



Figure 2. 5: 10K NTC Thermistor.

II.1.1.f Micro Servo Motor SG90

A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration. It consists of a suitable motor coupled to a sensor for position feedback. It also requires a relatively sophisticated controller, often a dedicated module designed specifically for use with servomotors. The micro servo motor will be used to stop Retractable Pulley (referee to mechanical structure part) and block user's finger to simulate the grasping effect.



Figure 2. 6: Servo Motor SG90.

II.1.1.g Bluetooth Module

Bluetooth module is a basic circuit that contains a Bluetooth chip to facilitate wireless communications. In our case we will use HC-05 module that is compatible with the Arduino and uses two pins to communicate, RX pin to receive data and TX pin to transmit data. This module is used to send and receive data from the HapLeap.



Figure 2. 7: HC-05 Bluetooth Module.

II.1.1.h Universal Asynchronous Receiver-Transmitter Module

The below UART module uses CH340G integrate circuit which is a series of USB bus adapters, that provides an interface over the USB bus. The CH340G provides common signals to allow adding a UART to a computer, or converting existing UART devices to USB interface. The UART is soldered in a PCB with the Bluetooth module and plugged into the computer to send and receive data from the computer to the HapLeap.

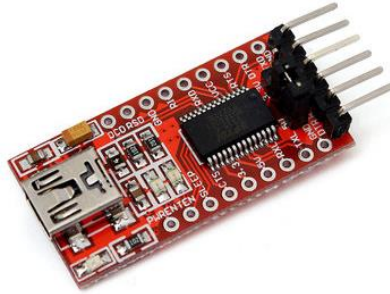


Figure 2. 8: UART Module.

II.1.2 Printed Circuit Board Design

II.1.2.a Proteus ISIS

Proteus is a circuit designing and simulation software also it is used for designing PCB Layouts. First we make our required circuit on Proteus, test it, and then make the desired PCB layout for production.

II.1.2.b Circuit Schematic Design

This section shows the main circuit and schematic diagrams for our PCB design.

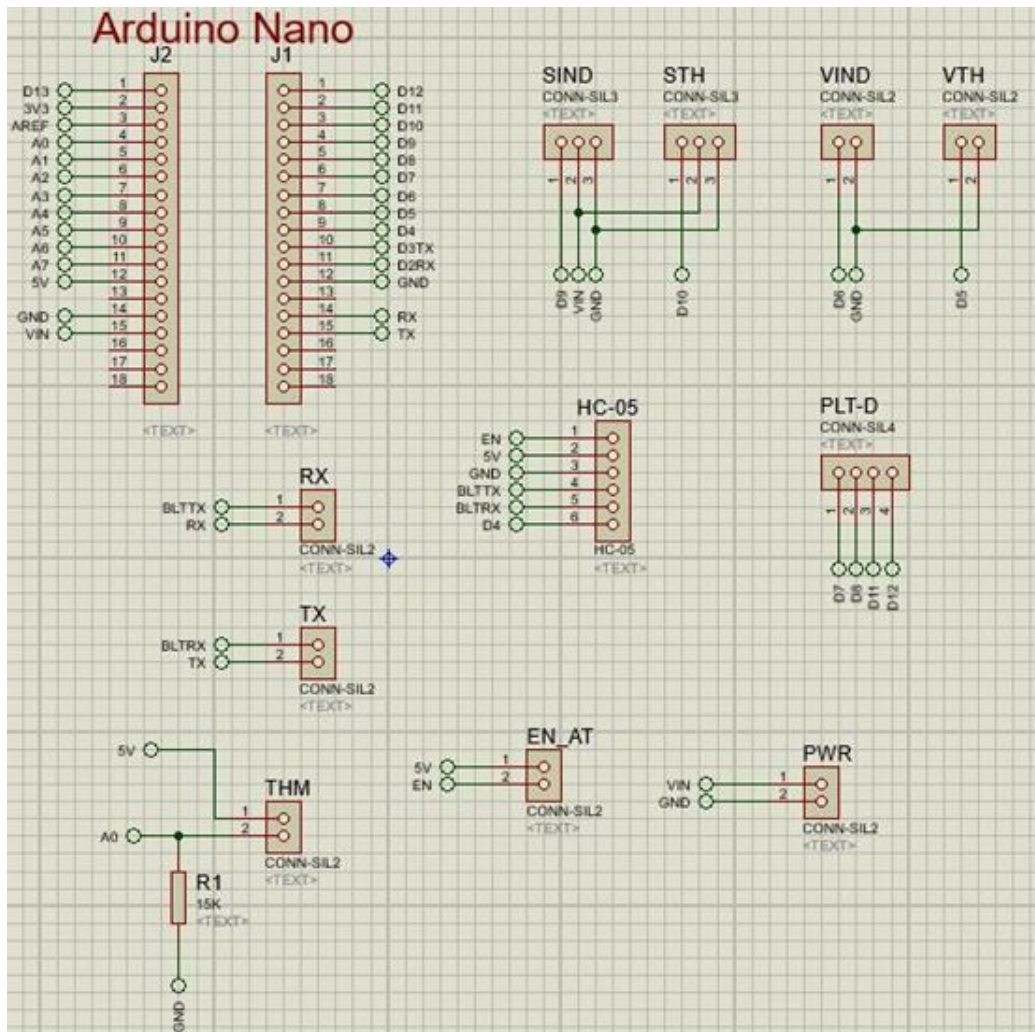


Figure 2. 9: HapLeap Circuit Diagram.

SIND: Servo motor pins for index finger.

STH: Servo motor pins for thumb finger.

VIND: Vibration motor pin for index finger.

VTH: Vibration motor pin for thumb finger.

PLT-D: H-Bridge pins to control the Peltier module.

EN_AT: pin to enable AT commands for HC-05.

RX: pin to allow RX connection from HC-05 to Arduino nano (remove it when uploading code).

TX: pin to allow TX connection from HC-05 to Arduino nano (remove it when uploading code).

The wiring diagram of the UART and HC-05 modules for the computer side to send and receive data from HapLeap.

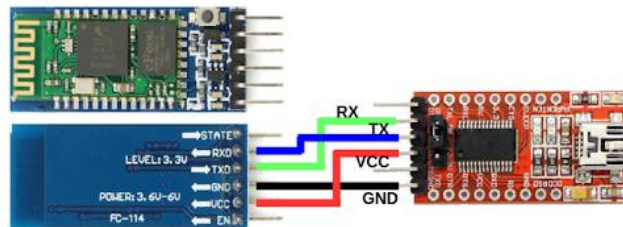


Figure 2. 10: UART module and HC-05 Wiring Diagram.

II.1.2.c PCB Layout

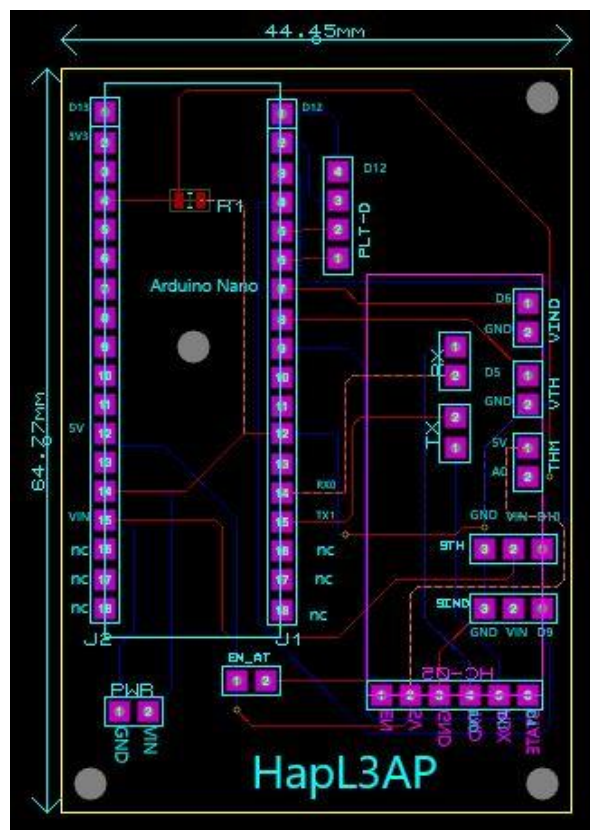


Figure 2. 11: HapLeap PCB Layout.

Due to the current world covid-19 pandemic the printed circuit board could not be made, we used a testing board for experiments test.

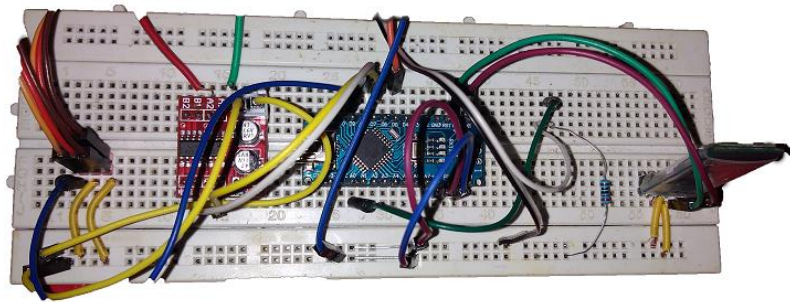


Figure 2. 12: HapLeap Test Circuit.

II.1.3 Mechanical Structure

II.1.3.a Retractable Pulley

In order to control the users hand and create the feeling of grasping we used retractable pulley, which contain a thin rope you can pull and when released it will return back to the ideal state. With the help if the micro servo motors we can control the movement of the rope by blocking or releasing the pulley.



Figure 2. 13: Retractable Pulley.

II.1.3.b 3D Design

For the 3D design of HapLeap we used Blender, which is a free and open-source 3D computer graphics software tool used for creating animated films, visual effects, art, 3D printed models, motion graphics, interactive 3D applications, and computer games. First

we designed a set of rings that will hold different actuators and sensors tested them with virtual hand on Blender and we chose the best design before 3D printing.



Figure 2. 14: HapLeap Ring 3D Model.

II.1.3.c 3D Printing

We used Anet A6 3D printer with black PLA material (polylactic acid which is a vegetable-based plastic material) which is the most used material for 3D printing.



Figure 2. 15: Anet A6 3D Printer.

After 3D printing the ring, here is the result:



Figure 2. 16: 3D printed Ring.

II.1.3.d Assembly

For faster prototyping we have consider the left hand only with the index and thumb fingers, after printing the finger ring, cleaning and placing the Peltier and tactile motor is shown in the below figure.

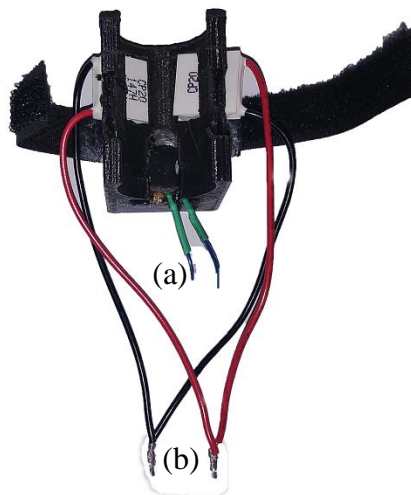


Figure 2. 17: Ring Finger. (a) Tactile Motor wires. (b) Peltier wires.

Then we assembled different parts into a wrist bracelet as below,



Figure 2. 18: Mechanical assembly of HapLeap.

II.1.4 Virtual Reality Headset

A virtual reality headset is a head-mounted device that provides stereo view for the wearer. Virtual reality headsets are widely used with video games but they are also used in other applications, including simulators. They comprise a stereoscopic head-mounted display (providing separate images for each eye), stereo sound, and head motion tracking sensors (which may include gyroscopes, accelerometer, magnetometers, structured light systems etc.). Some VR headsets also have eye tracking sensors and controllers.

There are many types for VR headset, we divided them as follows with some of the most used headsets in the virtual reality industry:

3Dof refers to rotation tracking only, where user can only rotate the head to see inside the VR environment.

6Dof refers to rotation and position tracking where user can rotate the head to see inside the VR environment also walk and use hand to navigate.

Table 2. 1: Most used VR headsets.

	Cardboard/ GearVR	Oculus Go	Oculus Rift S	Oculus Quest	HTC Vive
HMD	3Dof	3Dof	6Dof	6Dof	6Dof
Controller	3Dof	3Dof	6Dof	6Dof	6Dof
Tracking	-	-	inside-out	inside-out	Lighthouse
Equipment	Smart Phone	Stand Alone	PC Powered	Stand Alone	PC Powered

In our case we will be using Oculus Rift S without the controller, Figure below shows a setup of Oculus Rift S.



Figure 2. 19: Oculus Rift S HMD.

II.1.5 Leap Motion Controller

A computer (USB linked) hardware sensor device that supports hand and finger motions as input that requires no hand contact or touching. It uses stereoscopic cameras and infrared light with a wavelength of 850 nanometers, which is outside the visible light spectrum to detect hand motions in 3D space, providing high-speed, marker-less,

intuitive 3D hand input [7]. Leap Motion Inc. provides an SDK for different development platforms, to be used later on in the software implementation.



Figure 2. 20: Leap Motion Controller.

II.2 Software Design & Implementation

After finishing the hardware part, we need a way to control our device in an efficient way, for this the following pieces of software were used:

II.2.1 Unity Game Engine

Unity is a cross-platform game engine developed by Unity Technologies, first announced and released in June 2005 at Apple Inc.'s Worldwide Developers Conference as a Mac OS X-exclusive game engine. As of 2018, the engine had been extended to support more than 25 platforms. The engine can be used to create three-dimensional, two-dimensional, virtual reality, and augmented reality games, as well as simulations and other experiences. The engine has been adopted by industries outside video gaming, such as film, automotive, architecture, engineering and construction [8].



Figure 2. 21: Unity's user interface version 2019.3.1f1.

II.2.2 Leap Motion SDK

As mentioned in the hardware part that we are using Leap Motion for hand tracking, in order to use this sensor with unity we will be using a software development kit (SDK) version 4.0.0+52173 which is a collection of software development tools in one installable package made for Unity game engine. With this SDK we will be able to read the raw data of the hand and fingers joints in the 3d space and map this data into our simulation in real-time.

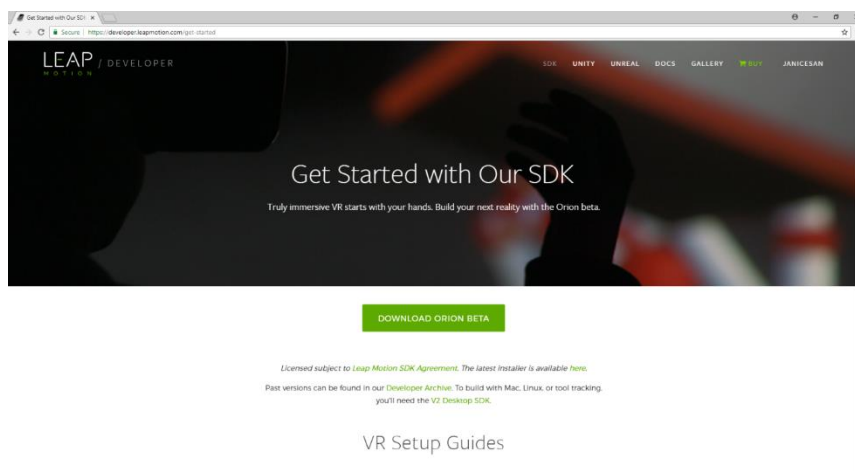


Figure 2. 22: Leap Motion SDK web page.

II.2.3 OpenVR SDK

OpenVR is a software development kit and application programming interface (API) developed by Valve for supporting the SteamVR and other virtual reality devices. In our case it is used with Unity engine to integrate Oculus VR headset. The integration SDK for Oculus Rift is downloaded from Oculus website.

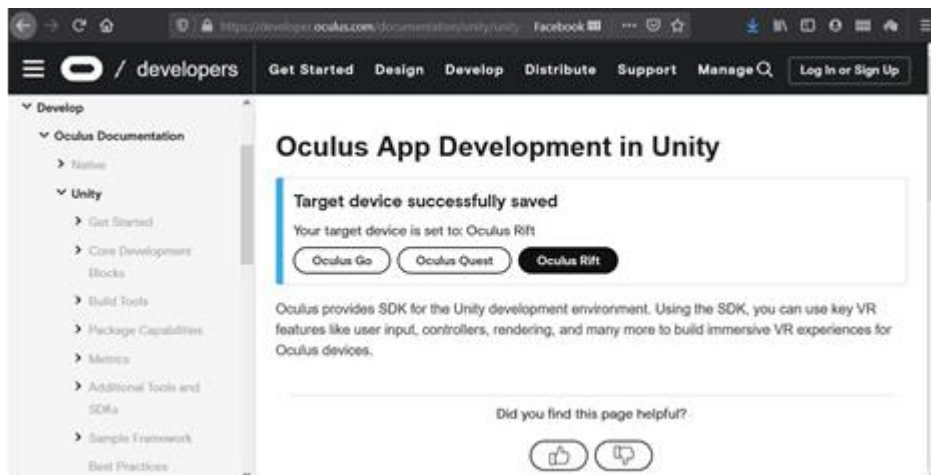


Figure 2. 23: Oculus Unity SDK web page.

Combining the Leap Motion and Oculus SDKs into Unity and running a simple test gave us the following result,

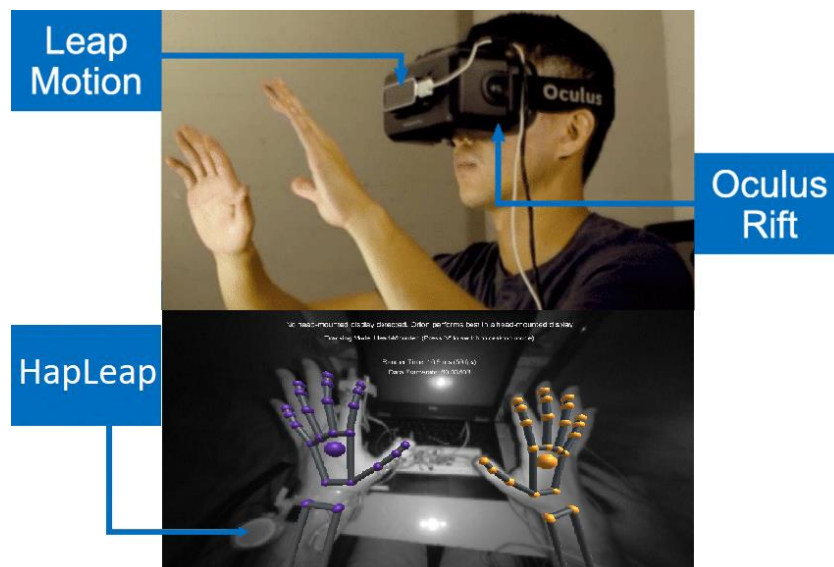


Figure 2. 24: View from the combination of Leap Motion and Oculus Rift test.

II.2.4 System Diagram and Flowchart

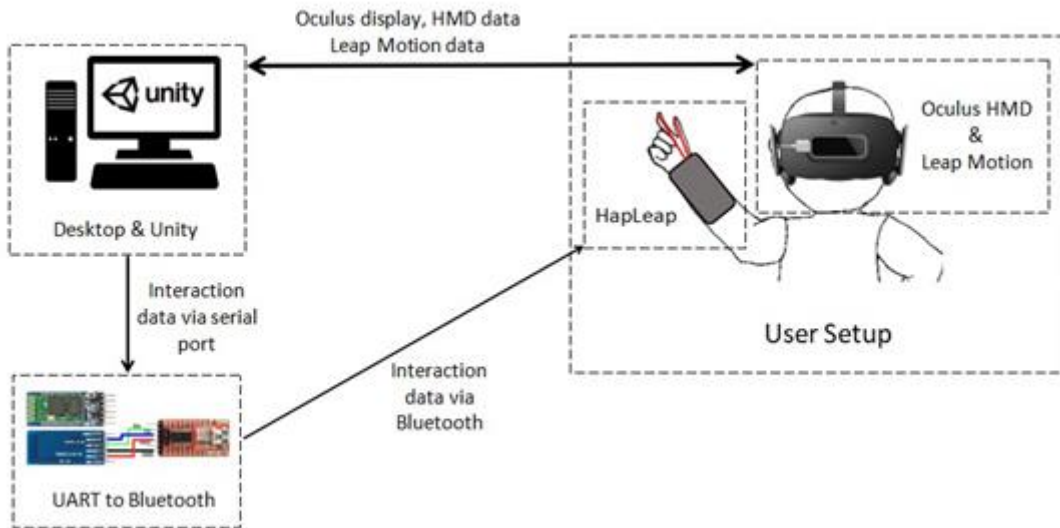


Figure 2. 25: General Diagram of HapLeap Application Work Flow.

Application Side

Application side is responsible for rendering and displaying the simulation inside the Oculus Head Mounted Display (HMD), in addition reading the Leap Motion data to visualize the user's hand and make necessary computations to get the interaction data and detect hand actions to be transmitted to HapLeap.

Hand actions can divide into 3, grasp (enables grasping stimulation, surface material and surface temperature stimulation), touch (enables surface material and surface temperature stimulation) and hover (enables surface temperature stimulation).

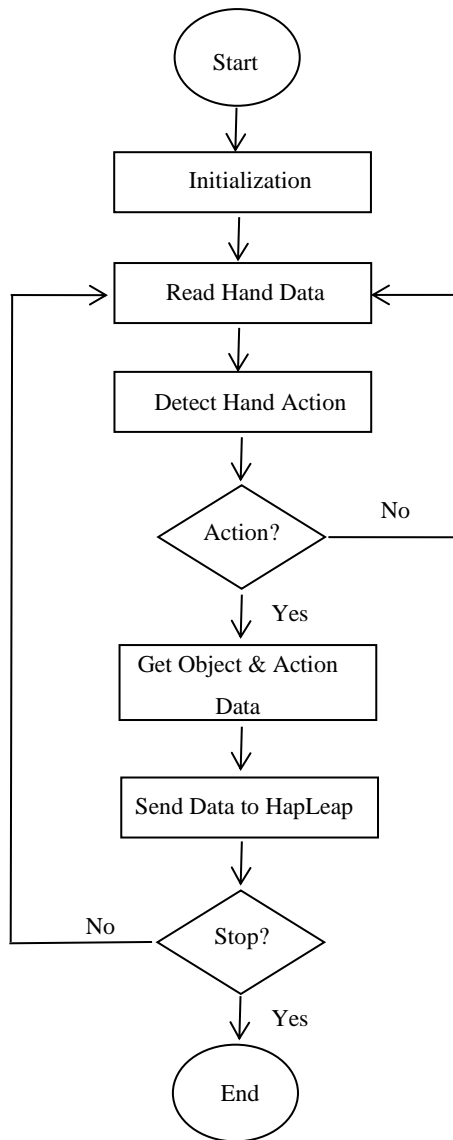


Figure 2. 26: Interaction Processing Flowchart.

HapLeap Side

After receiving interaction data via Bluetooth, the HapLeap control unit will generate the output signals for the vibrato tactile motors (tactors), servo motors and the thermoelectric peltier, which we will go through in details later on.

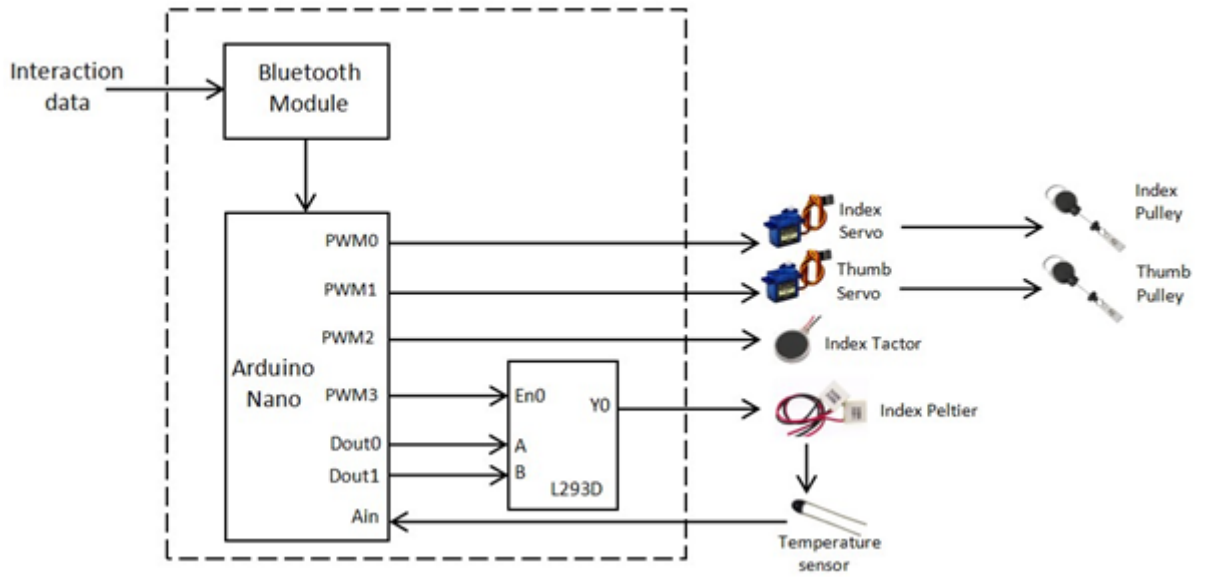


Figure 2. 27: General Diagram of HapLeap.

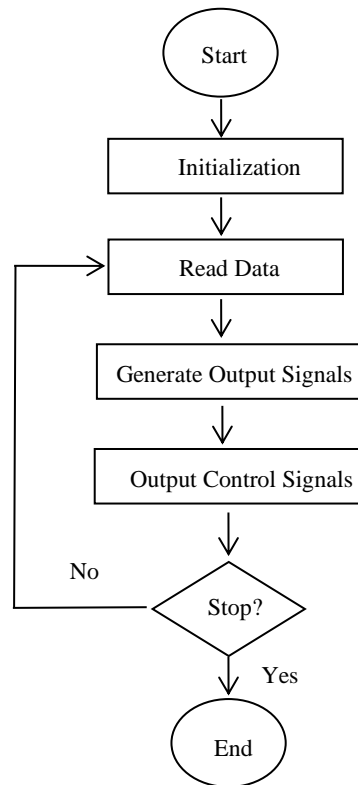


Figure 2. 28: HapLeap Side Flowchart.

II.2.4 Hand Detection and Interaction

At the basic level, the Leap Motion SDK provides the tracking data in the form of frames, each frame object contains a set of tracked entities, such as hands, fingers and other tools that will help us represent the hand in our simulation [9].



Figure 2. 29: Leap Motion hand bone hierarchy.

In addition, the Leap Motion SDK allows us to access the raw data sensor, first we check if we are detecting the hand (left hand in our case), then go through all the individual fingers, in our case the index and thumb, then check for 2 main interaction actions, touching or grasping. The touching action is defined as checking the hovering distance with the interaction object such as table, cup and the surrounding environment and setting a minimum hovering distance and when that distance is reached we call it a touch, this is done using the Hovering API from the leap motion SDK.

```
#region Hovering API
/// <summary> Gets whether any interaction controller is nearby. </summary>
3 references
public bool isHovered { get { return _hoveringControllers.Count > 0; } }
```

Figure 2. 30: Hovering check from Hovering API in Interaction Behavior script.

For the other interaction action, which is the grasping actions, is defined as the collision of the hand or the fingers with the interaction object and the grasp force of the hand to reach certain value, this is also done using Grasping API provided by the leap motion SDK

```
#region Grasping API

/// <summary> Gets whether this object is grasped by any interaction controller. </summary>
25 references
public bool isGrasped { get { return _graspingControllers.Count > 0; } }
```

Figure 2. 31: Grasping check from Grasping API in Interaction Behavior script.

In order for the interaction behaviors to work, we need 3d objects to interact with, for this we setup **Interaction Objects** which are the essentially blocks for our system, they contain the interaction behavior needed to trigger the touch and grasping actions, in addition to a set of components contained in a **Game Object** which is the fundamental object in Unity that act as container for our scripts.

The components are listed as follow:

Transform: Defines the position, rotation and scale of the 3D object in the 3D space of our simulation inside unity game engine, any Game Object will contain a transform even if we are not using it as Game Objects lay in the 3D space.

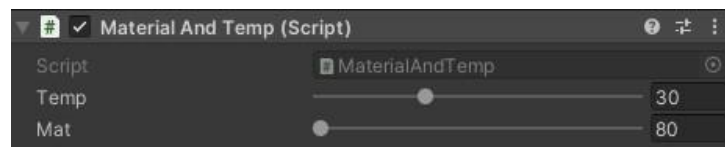


Figure 2. 32: Transform Component.

Mesh Filter: Holds the 3D mesh of our 3d model like a cup.



Figure 2. 33: Mesh Filter Component.

Collider: Defines the shape of a game object for the purposes of physical collisions.

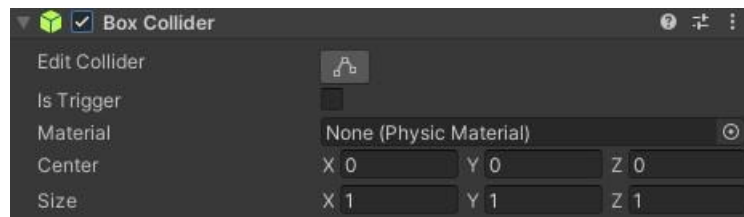


Figure 2. 34: Collider Component

Mesh Renderer: Contains parameters that makes an object appear in the screen and Unity can render its mesh.

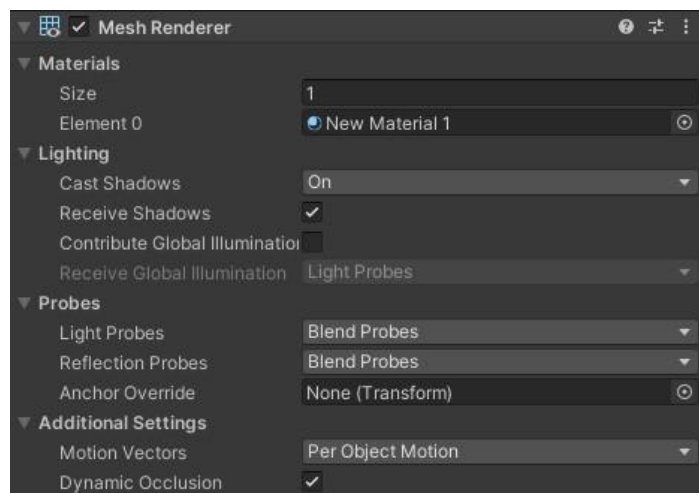


Figure 2. 35: Mesh Renderer Component

Rigid Body: Enables the game objects to act under the control of physics. The rigid body can receive forces and torque to make the object move in a realistic way, influenced by gravity and act under added forces via scripting, or interact with other objects.

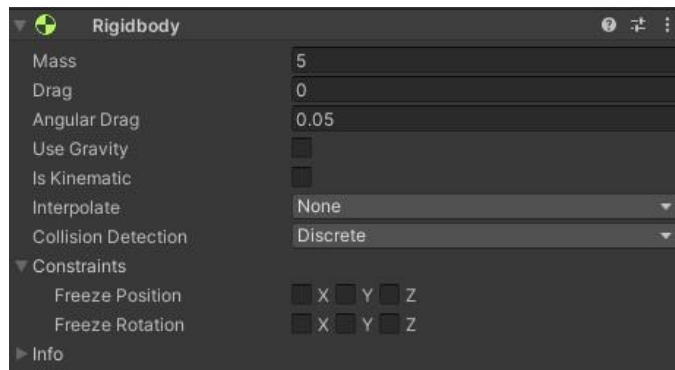


Figure 2. 36: Rigid Body Component

Interaction Behavior: Interaction behavior is a component that enable game objects to interact with interaction controllers (user’s hand) in a physically intuitive way. By default, they represent objects that can be poked, prodded, smacked, grasped, and thrown around by interaction controllers, including Leap hands. They also provide a public API for hovering, contact, and grasping callbacks for creating physical interfaces or overriding the default physical behavior of the object.

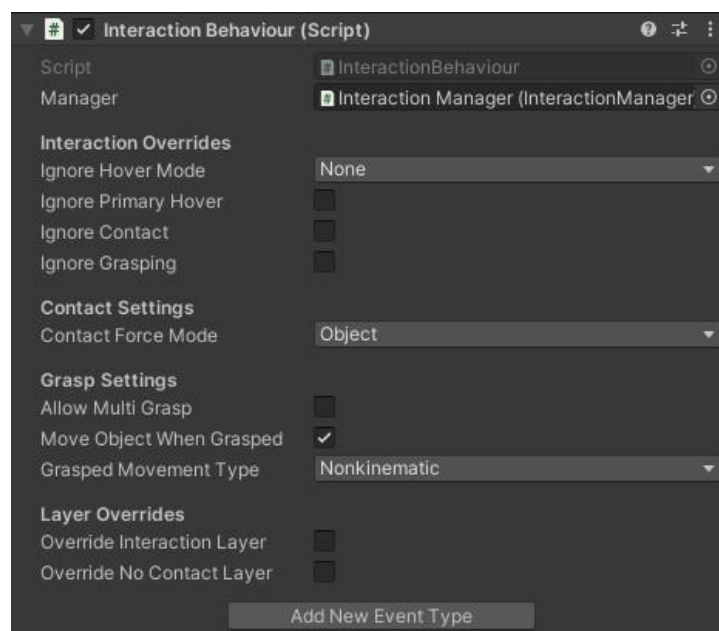


Figure 2. 37: Interaction Behavior Component

Material and Temp: Component that holds the physical material data of the object's surface and the object's temperature.

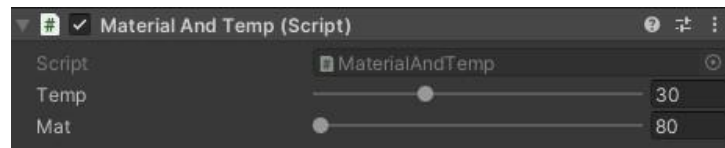


Figure 2. 38: Material and Temp Component

II.2.5 Haptics Manager

Haptics manager is the game object responsible for setting the communication port, reading interaction action, receive object physical material, temperature and then transmit that data to HapLeap. The haptics manager also is a container of components, the base component is the transform component, the other components are as follow:

Serial COM: Serial COM is the base communication component, that uses C sharp Input/Output port directive (System.IO.Ports) in which it defines the serial port communicative between our simulation and the HapLeap device. It is used to open port, write and read from that port with pre-defined transfer rate of data (Baud Rate), in our case 38400 baud rate is used.

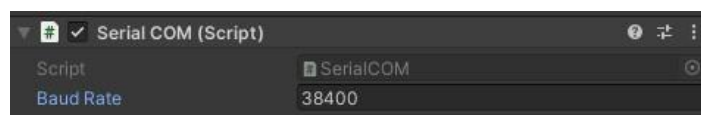


Figure 2. 39: Serial COM Component

Serial Wrapper: it is a Serial COM wrapper component that contains simplified functionality from the Serial COM component to send data over the serial port also contains the definition of the hand active state, grasping state, touch state, surface type and object temperature.

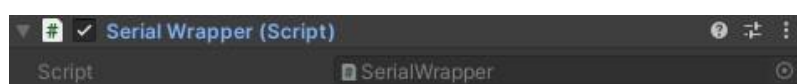


Figure 2. 40: Serial Wrapper Component

```

#region SettersAndGetters
3 references
public int IsHandGrasp { get => isHandGrasp; set => isHandGrasp = value; }
3 references
public int IsHandActive { get => isHandActive; set => isHandActive = value; }
4 references
public int IndexContact { get => indexContact; set => indexContact = value; }
3 references
public int SurfaceType { get => surfaceType; set => surfaceType = value; }
3 references
public int ObjectTemp { get => objectTemp; set => objectTemp = value; }
#endregion

```

Figure 2. 41: Data Definition in Serial Wrapper Component

Query Hand: Holds a copy of the Interaction Behavior of all interaction objects in our simulation, checks the active state of the hand, the touching state and the grasping state, then sets the surface type and temperature in the Serial Wrapper component.

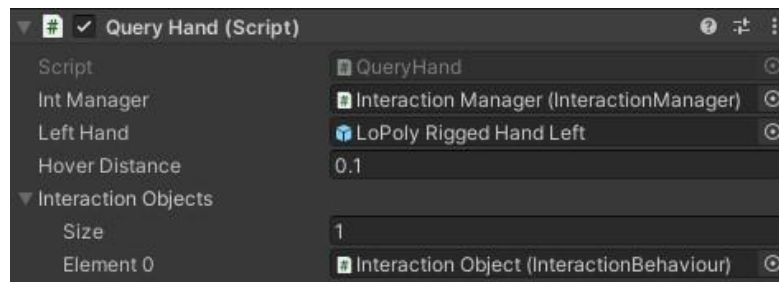


Figure 2. 42: Query Hand Component

II.2.6 Servo Motor Control Signal

The servo motors are used for grasping sensation, when user grasp an object the grasping state will be true and the HapLeap control unit will generate a signal to rotate the servo motors to stop the fingers at certain position to give the sensation of grasping, in our device we set the grasping for the index and thumb fingers only.

II.2.7 Tactors Control Signal

We used Pulse Width Modulated (PWM) Waveform to control the tactors. With PWM signal the power loss in the switching transistors is small, since the transistors are either full ON which allows the flow of current or fully OFF which blocks the flow of current. This allow the proposed device to run under less heat dissipation and power consumed with low cost, efficient and long lasting equipment. In case we have an action between the user's hand and an interaction object the surface type value is transmitted to HapLeap and processed with the control unit. The vibration intensity is, then, controlled by driving the tactors with a series of ON/OFF pulses and varying the duty cycle which is the ratio of the time that the output is ON compared to when the output is OFF, while keeping the frequency constant. More the pulse ON is longer the faster the tactors will rotate. This leads to rough vibrations. Likewise, the shorter the pulse of ON the slower the motor will rotate which leads to softer vibrations. In our device we set the haptic feedback for the index finger only.

II.2.8 Peltier Control Signal

Using the same concept of PWM signal, we control the peltier with another generated signal from the micro-controller, in addition, the peltier control signal is affected by a closed loop control system from the micro-controller to maintain a reasonable temperature value that will not exceed some maximum value for safety. This control system in accomplished using a temperature sensor placed between of the Peltier and the skin, then the current temperature is feed to the control unit in addition to the interaction data received via Bluetooth. The generated signal is connected to L293D integrated circuit. The latter is a typical motor driver that allows 2 DC motors to drive on 2 directions. The 1st driver channel is used for the Peltier with 2 directions, as the 1st direction is used to heat up the area of the finger and the 2nd direction is used cool down the area if the finger for faster and reliable sensation of the object is begin interacted with in simulation. In our device we set the thermal feedback for the index finger only.

II.2.9 Temperature Feedback

Analog Voltage Reading

To measure the temperature, we need to measure the resistance of the 10K NTC thermistor. However, a micro-controller does not have a resistance-meter built in, instead, it only has a voltage reader known as an analog-digital-converter. So what we have to do is convert the resistance into a voltage, and we'll do that by adding another resistor and connecting them in series. Now we just measure the voltage in the middle, as the resistance changes, the voltage changes too, according to the simple voltage-divider equation. We just need to keep one resistor fixed.

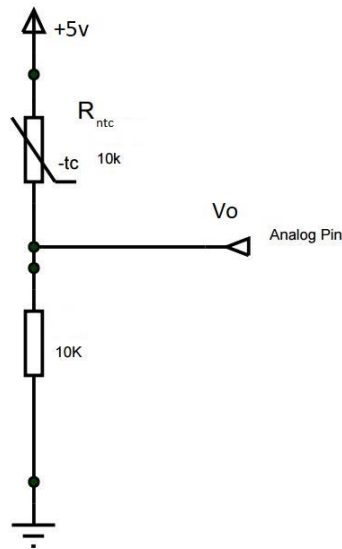


Figure 2. 43: NTC thermistor analog signal circuit.

Using a fixed 10K ohm resistor with the NTC resistance called R_{ntc} and output voltage V_o , we get:

$$V_o = \frac{R_{ntc}}{(R_{ntc} + 10K)} \times V_{cc} \quad (3.1)$$

with $V_{cc} = 5v$

The output voltage is then connected to the analog input of the micro-controller with ADC resolution of 10 bits, the output voltage from the voltage divider will go from 0(0V)

to $2^{10} - 1 = 1023(5V)$, this value gives a step of $\frac{1023}{V_{ref}}$, as V_{ref} is the analog reference voltage and it is set to 5V, we get

$$ADC = \frac{R_{ntc}}{(R_{ntc} + 10K)} \times 1023 \quad (3.2)$$

Concluding that:

$$R_{ntc} = \frac{ADC \times 10k}{1023 - ADC} \quad (3.3)$$

Temperature Conversion

In order to have the temperature value, we will use the Steinhart–Hart equation as follow

$$\frac{1}{T} = A + B \times \ln(R_{ntc}) + C \times (\ln(R_{ntc}))^3 \quad (3.4)$$

However, this equation is fairly complex, instead we will use the simplified B equation as follow

$$\frac{1}{T} = \frac{1}{T_0} + \frac{1}{B} \times \ln\left(\frac{R_{ntc}}{R_{ntc_0}}\right) \quad (3.5)$$

$T_0 = 25^\circ C = 298.15 K$, $B = 3950$ (for 10K NTC)

$R_{ntc_0} = 10K\Omega$ (at T_0)

With the feedback of the temperature and the interaction data the control unit will generate the appropriate signal for the Peltier.

Chapter III Experiment and Evaluation

III.1 Experiments

To demonstrate the immersive experience of the proposed device, we investigated 2 VR experiments. First of all, a user wears a VR HMD device to view a VR environment to be familiar with using the Leap Motion and how this concept works. For the second experiment, user is asked to perform several task using our device such as grasp and touch.

Experiment 1

We used an application from Leap Motion gallery called Blocks which is an interactive game that allows users to Create and interact with shapes, manipulate gravity and more in this VR experience. The actions that are allowed are as follow: spawn cubes and prisms, grab and stack them, or knock them over, select between different shapes using a simple menu that appears alongside your outstretched hand, swing your hands up to turn off gravity, spin objects into space and bring everything down to earth. All of this is instructed by an avatar to help users through the test.

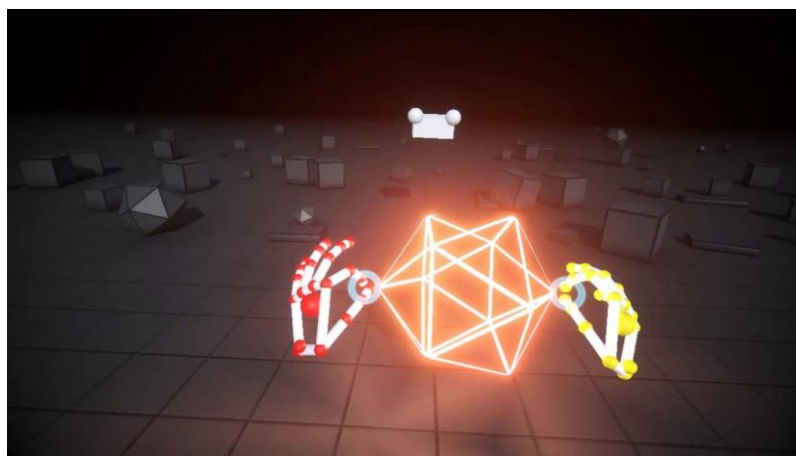


Figure 3. 1: User view inside experiment 1.

Experiment 2

We introduced HapLeap to the participants, then participants were asked to interact with the environment as they interact in real life with real objects (grasp, touch, hover).

In the experiment, the 4 cubes have different temperate and surface material in which the red cube is the warmest and has rough surface material, and the blue cube is the coldest and has the smoothest surface material among the other cubes.

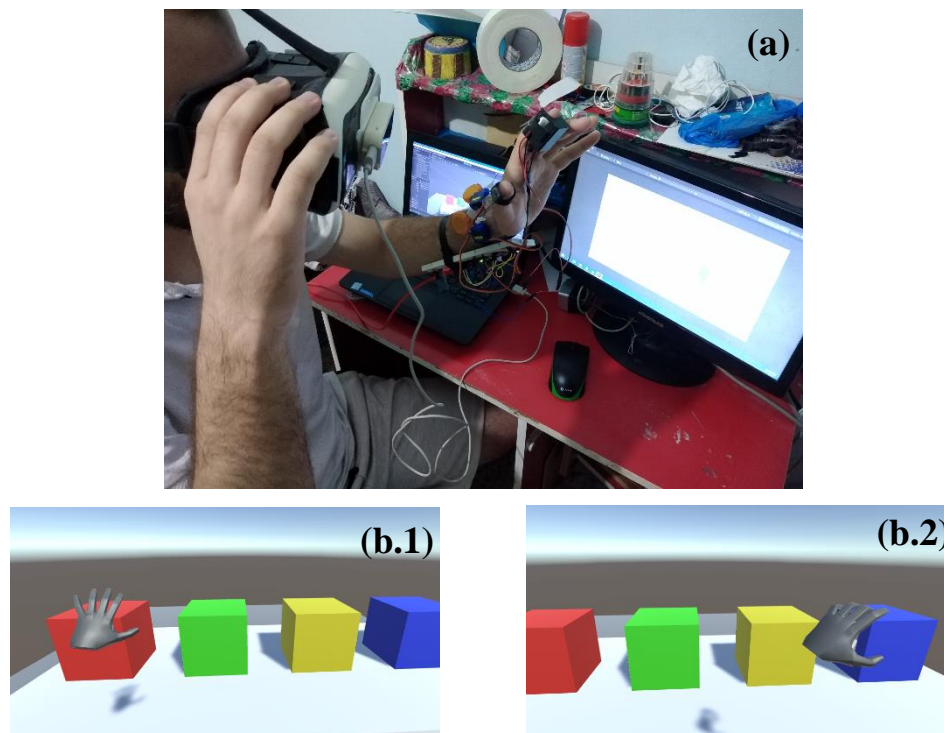


Figure 3. 2: Experiment 2.(a) HapLeap Setup. (b) User view.

Evaluation

The USE [Davis 1989] (Usefulness, Satisfaction and Ease of Use) questionnaire was used to evaluate subjectively HapLeap interaction. A total of 10 subjects (5 females, 5 males) aged from 15 to 50 (mean of age 30.1 and (SD=13.13)), participated and asked to rate the experiments in which our device was use (Experiment 2).

The results are shown in the following figure (refer to appendix C for the evaluation data).

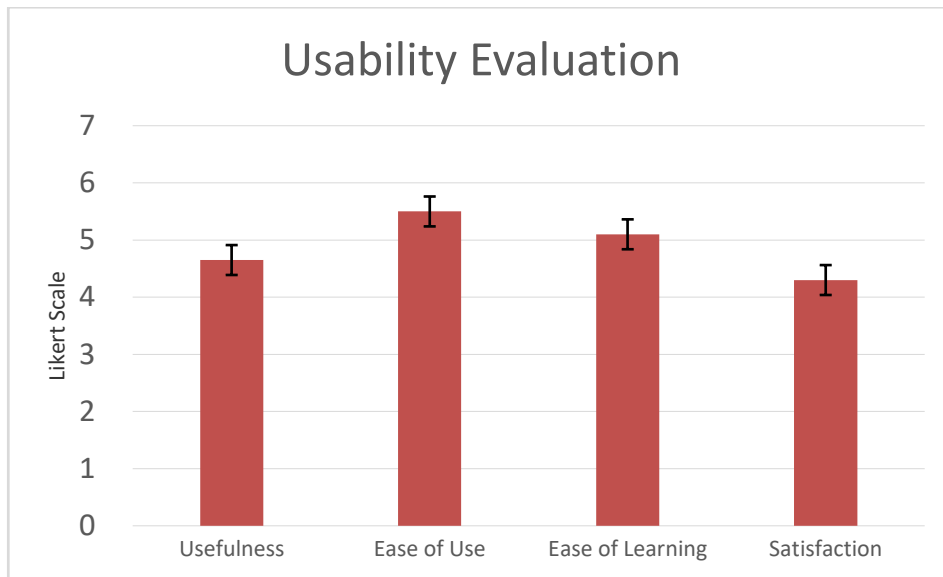


Figure 3. 3: Usability Evaluation.

In the second step of the evaluation, we asked the participants to answer a questionnaire related to smoothness of interaction, realism, synchronization and perception of stimulation.

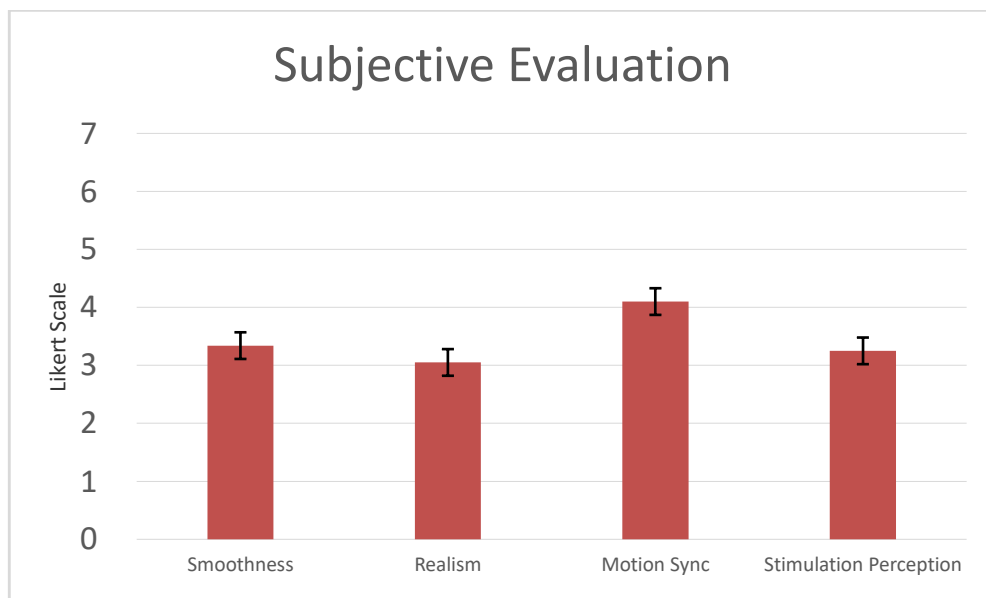


Figure 3. 4: Subjective Evaluation.

Conclusion

Virtual reality is the new trend in information technology, which knows a remarkable growth in many domains especially with the new world situation where social distancing is a need, VR can be used in trainings, education and be present in places where we are in the comfort of our homes, but the lack of feedback and sensation will break the illusion and sometimes give motion sickness or just break the enjoyment of the experience.

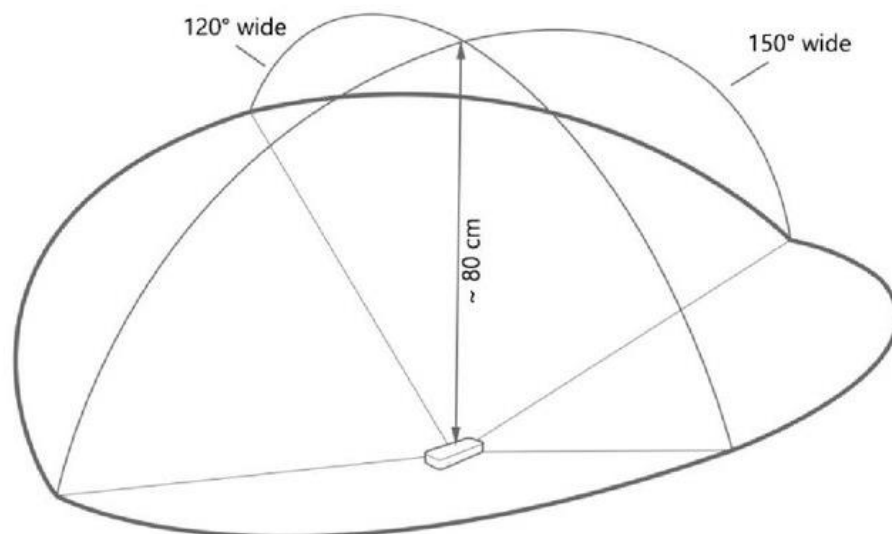
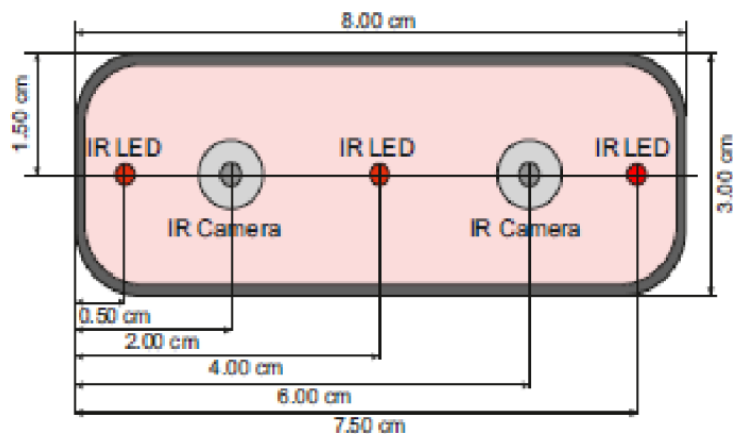
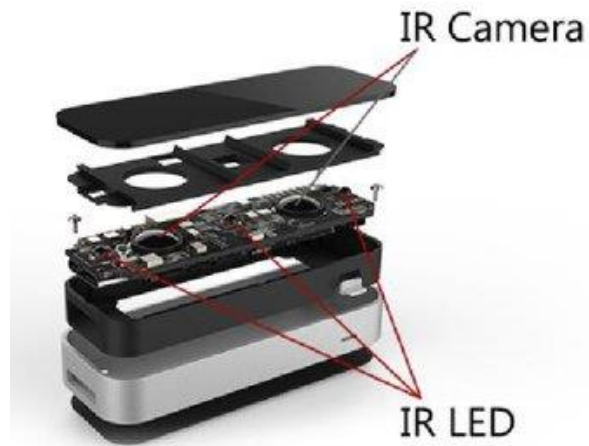
In this work we presented a Wearable Haptic Grasping Device for Virtual Reality equipped with Kinaesthetic, Cutaneous and Thermal Feedback named HapLeap, which is a real-time on hand haptic device that allows a dimensions of immersion to VR application, where the three illusion of VR were applied, as the user can see the virtual hands and feel the interaction with the surrounding environment in which the environment will respond to the user, all of this with extra feedback were the user can engage deeply with the three sensations that our system provides.

The results show a promising aspect of the system and its use in the enhancement of the VR technology. The proposed device will be improved further more to handle extra sensation and feedback such as VR notification, alerts and to guide the user to fully immersed experience.

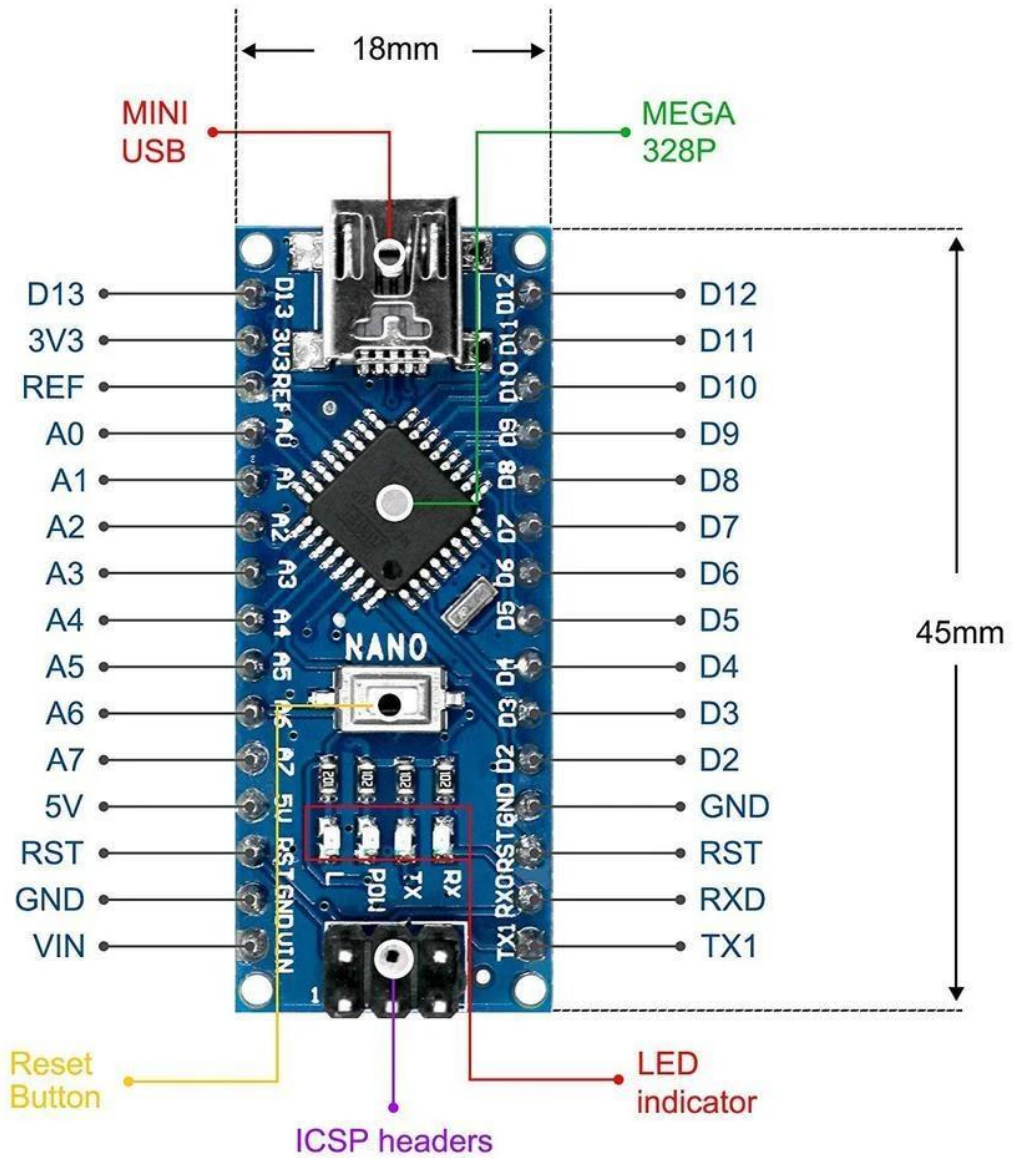
References

- [1] Nukarinen et Al, "Hands-free Vibrotactile Feedback for Object Selection Tasks in Virtual Reality," *VRST 2018: 24th ACM Symposium on Virtual Reality Software and Technology (VRST '18)*, 2018.
- [2] wikipedia, "wikipedia," [Online]. Available:
https://en.wikipedia.org/wiki/Virtual_reality.
- [3] D. Cooke, "Pebble Studios," 17th August 2017. [Online]. Available:
<https://www.pebblestudios.co.uk/2017/08/17/when-was-virtual-reality-invented/>.
- [4] dbudhrani, "steemit," [Online]. Available:
<https://steemit.com/virtualreality/@dbudhrani/virtual-reality-2-the-three-illusions>.
- [5] P. R. J. Stone, "Haptic Feedback: A Potted History, From Telepresence to Virtual Reality," MUSE Virtual Presence, Chester House, 79 Dane Road, Sale, M33 7BP, UK.
- [6] Pacchierotti et al, "WEARABLE HAPTIC SYSTEMS FOR THE FINGERTIP AND THE HAND: TAXONOMY, REVIEW, AND PERSPECTIVES," *IEEE TRANSACTIONS ON HAPTICS*, vol. 10, 2017.
- [7] Leap Motion Inc., Leap Motion Inc., [Online]. Available:
<http://blog.leapmotion.com/hardware-to-software-how-does-the-leap-motion-controller-work/>.
- [8] Wikipedia, "Unity (game engine)," [Online]. Available:
[https://en.wikipedia.org/wiki/Unity_\(game_engine\)](https://en.wikipedia.org/wiki/Unity_(game_engine)).
- [9] Leap Motion Inc., [Online]. Available: <http://blog.leapmotion.com/getting-started-leap-motion-sdk/>.

Appendix A: Leap Motion



Appendix B: Arduino Nano



Appendix C: Evaluation Data

USE: Usefulness **ES:** Ease of use **EL:** Ease of learning **ST:** Satisfaction

SM: Smoothness **RL:** Realism **MS:** Motion sync **SP:** Stimulation perception

		Usability Evaluation (0 to 7)				Subjective Evaluation (0 to 7)			
User	Age	USE	ES	EL	ST	SM	RL	MS	SP
1	21	3	6	5	4	4	4	5	4
2	22	4	5	4	3	3	4	4	3
3	48	6	5	5	4	2.5	3	5	3
4	43	4	5	6	5	4	3	5	4
5	18	3.5	6.5	5	5	3	3	4	5
6	16	5	5	6	4	3.5	2	3	2.5
7	40	4.5	4.5	4	5	4	3	4.5	3
8	28	6	6	6	5	2.4	3	4.5	2
9	15	5	6	5	4	3	2.5	3	2
10	50	5.5	6	5	4	4	3	3	4
Average		4.65	5.5	5.1	4.3	3.34	3.05	4.1	3.25

Mean Age = 30.1

Standard deviation of the age = 13.13

Appendix C: Project Link

You can find the full project at my GitHub account:

<https://github.com/bentalebahmed/HapL3ap>