Louvain School of Management

Opportunity study on the implementation of Air+Touch information system in organisations

Auteur·e(s) : Romain Neuville
Promoteur·rice(s) : Jean Vanderdonckt
Année académique 2021-2022
Travail de fin d'études (TFE) en vue d'obtenir le titre de
Master (60) en Sciences de Gestion
Horaire de jour / Horaire décalé

Abstract

Nowadays, Gesture recognition has become one of the main actor of Human Computer Interaction. In our previous work [17], we focused on Air+Touch gestures which suffers from a huge lack of study compared to its potential. We proved that this topic was very interesting and promising for the future of gesture recognition. We also defined a relevant Air+Touch gesture set definition related to the 3DTouchpad device [15]. This work is a continuation of our previous work as it concerns an opportunity study on the implementation of Air+Touch information system in organisations according to the 3DTouchpad. We are going to implement an application that supports 2D and 3D interactions. This prototype application is based on a relevant survey. For each type of applications founded in the literature, the respondents had to evaluate the pertinence degree of the device's use. We are going to define some targeted audience for our survey according to the different types of application. Based on the survey result, we are going to implement a prototype and perform a testing for this prototype. In this work, we are also going to provide an additional benchmarks of recognizers which are complementary to our previous one.

Acknowledgements

First of all, I would like to thank Prof. Jean Vanderdonckt who guided me, helped me a lot all along the thesis and for the trust to let me continue my work. This thesis wouldn't have been possible without him.

I would also like to thank all the researchers who have been mentioned and who have made this thesis possible through their work.

I would like to thank all the teachers who took the time to teach us the magic of computers sciences and all the administration staffs from EPL, LSM and UCLouvain.

Finally, thank you to all my friends and family. Without them I could not have achieved all this.

Contents

1 Introduction **5**
1.1 Context Of The Problem 5
1.2 Mission Statement 6
1.2.1 Statement 6
1.2.2 Research Questions 6
1.2.3 Working Hypothesis 7
1.2.4 Research Method 7
1.3 Outline of the thesis 7

2 Related Work **10**
2.1 Overview 10
2.2 Gesture Recognition Market Analysis 10
2.2.1 Market Forecast 10
2.2.2 Gesture Control Devices Hype Cycle 11
2.3 History of applications using 2D and 3D gestures together 13
2.4 Conclusion 18

3 Survey **19**
3.1 Overview 19
3.2 Survey Methodology 19
3.3 Demographic Data 20
3.3.1 Information of the respondents 20
3.3.2 Tools Frequencies 24
3.3.3 Knowledge of the Device 25
3.4 Tasks 25
3.4.1 Spontaneous ideas for application 25
3.4.2 Relevance of the types of applications chosen 26
3.4.3 New ideas 28
3.5 Dynamics Dashboard 28
3.6 Conclusion 30

4 Application Prototype **31**
4.1 Overview . . . 31
4.2 Prototype Specifications . . . 31
4.2.1 Our concept . . . 31
4.2.2 Other idea for a future work . . . 32
4.3 Gestures Definition . . . 32
4.4 Implementation . . . 34
4.5 Prototype Visualisation . . . 36
4.5.1 Prototype Utilisation . . . 36
4.5.2 Interface Components . . . 37
4.5.3 Gesture Description . . . 38
4.5.4 Zoom Effect . . . 39
4.6 Conclusion . . . 40

5 Prototype Test **41**
5.1 Overview . . . 41
5.2 Participants Information . . . 41
5.3 Conduct of the experiment . . . 42
5.4 Post-Study System Usability Questionnaire (PSSUQ) . . . 43
5.4.1 Introduction . . . 43
5.4.2 First Post-Study System Usability Questionnaire . . . 44
5.4.3 Second Post-Study System Usability Questionnaire . . . 45
5.4.4 Comparison between both Questionnaire . . . 47
5.5 Memorability Test . . . 49
5.6 Conclusion . . . 50

6 Additional Benchmarks of Recognizers **51**
6.1 Overview . . . 51
6.2 Previous Benchmarks of Recognizers . . . 51
6.3 New Benchmarks of Recognizers . . . 52
6.3.1 Goal of the Benchmarks . . . 52
6.3.2 Benchmarks Parameters . . . 52
6.3.3 Recognizers Comparison . . . 53
6.3.4 Optimal Combination of Recognizers . . . 58
6.4 Conclusion . . . 59

7 Conclusion and Future Work **60**
7.1 Final Conclusion . . . 60
7.2 Future Work . . . 61

A Appendix **70**
A.1 Prototype Test . . . 70
A.2 Kruskall-Wallis with post-how Tukey . . . 71
A.3 Recognition Rate of recognizers with gestures with One Finger . . . 73
A.4 Recognition Rate of recognizers with gestures with two Fingers . . . 76
A.5 Recognition Rate of recognizers with gestures with three Fingers . . 79
A.6 Recognition Rate of recognizers for 3D gestures . . . 82
A.7 Recognition Rate of gestures with One Finger . . . 85
A.8 Recognition Rate of gestures with two Fingers . . . 87
A.9 Recognition Rate of gestures with three Fingers . . . 90
A.10 Recognition Rate of 3D gestures . . . 92
A.11 Prototype Implementation . . . 94

Chapter 1

Introduction

1.1 Context Of The Problem

During the last decade, the interest in gesture recognition has increased. It has been proven that it can be useful in our daily lives and is no longer just a gadget. Nowadays, everyone is using 2D and 3D gestures. For instance, 2D gestures can be used with your smartphone, or the track-pad of your laptop whereas 3D gestures can be used with motion detector. There are many other relevant examples that seem to show that there is a clear and deep interest in gesture recognition.

With recent technological advances in gesture recognition, the limitations of each type of gestures became apparent. The 2D gestures suffer from a lack of expressiveness whereas 3D gestures are less accurate and can be easily confused with other gestures. In our previous work [17], we studied the combinations of 3D gestures and 2D gestures in the same working environment. The figure 1.1 is an illustration of the both kind of gestures. On the left part of the picture represents a *2D swipe left gesture with two fingers* whereas the right part is an *Air Swipe Up gesture(3D)*

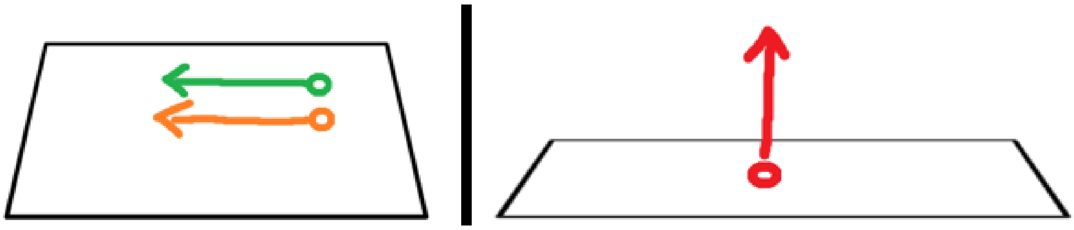


Figure 1.1: 2D gestures and 3D gestures

This combination reduces considerably the limits of each gesture. In this work, we first created an *Air+Touch gesture set definition* based on the founded literature, on two gesture elicitation studies [36] and our device, "the 3DTouchpad [15]". Then, we integrated our 3DTouchpad script inside the *QuantumLeap framework* [28] where we could test our samples from right-handed and left-handed people with different recognizers. The results were very interesting and promising. On this **link**, you can find a short video explaining how it works. Now that the technical stuff is functional, it seems relevant to implement an information system based *Air+Touch* gesture inside organizations. This kind of interaction can significantly improve the precision, the ergonomy of work and so on. For instance, "mixed" gestures could fit with medical applications [37], games [10], meetings [3]. There is an enormous potential.

1.2 Mission Statement

1.2.1 Statement

In order to prove that there is a real interest in the use of Air+Touch gestures inside information system, this work consists in an **opportunity study on the implementation of Air+Touch information system in organisations, firstly, we are going to, implement a prototype that is representative of the population's opinion and related to the resources found in the literature and then, we are going to test this prototype and to provide additional testing with different recognizers.**

1.2.2 Research Questions

The main focus on this work is about to define an *Air+Touch information system* to be implemented inside organisations. There are several steps detailed in each chapter in which their analysis and results are answered the following research questions:

- Which kind of information system is the most relevant according to our device ?
- Could Air+Touch gestures really increase the usability of an information system ?
- Which recognizer or mix of recognizers are the most accurate ?
- Is our application prototype relevant for the different participants of the experiment ?

1.2.3 Working Hypothesis

We assume the following hypothesis for this works:

Hypothesis 1. *Dataset*: used for the testing sections is diversified enough to be representative.

Hypothesis 2. *Computer Resources*: have been executed with the same laptop performance.

1.2.4 Research Method

This section is a brief overview of the research methodology of this thesis. All the different points mentioned here are going to be seen in more details in the next chapters.

- *State of Art*: consists in the maximum information gathering inside papers, books, articles which are related to the use of 2D and 3D gestures inside applications. The goal is to determine which kind of application is the most relevant for these interactions. We are going to prove that the gesture recognition market is promising and therefore, prove that our work is relevant.

- *Survey*: classifies and analyses the different data from each participant according to their personal information. The results of this survey will determine how the prototype should be designed.

- *Implementation of Prototype*: explain how the prototype was designed and which gestures are linked to which actions.

- *Prototype Test*: provide an experiment to test our prototype with several participants and then remake again the whole experiment with the same participants and compare both results. Just before the second experiment, we are going to perform a memorability test for each participant.

- *Additional testing*: provides an additional benchmarks of recognizers for our gesture set definition which results are complementary with our previous work [17].

1.3 Outline of the thesis

To achieve the central goal of this thesis, the manuscript is separated into several chapters. Each chapter is resumed in tiny explanations below to provide a better overview.

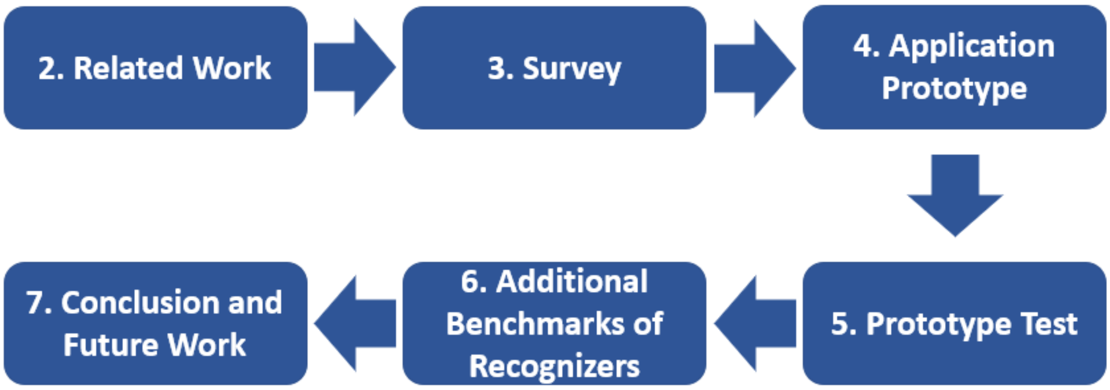


Figure 1.2: Chapters overview

- **Chapter 2: Related Work** is first going to determine that the gesture recognition market is interesting by analyzing the market forecast and the hype cycle of gestures control devices through the years. The goal is to prove that this is relevant to implement a prototype according to the 3DTouchpad device and the market. The second part of this chapter consists in retrieving articles from the literature listing applications using 2D and 3D gestures interactions. And then, we are going to classify each application to build a relevant cluster which is necessary for our survey in the next chapter.

- **Chapter 3: Survey** is going to determine which kind of applications is the most relevant according to many different people. The different respondents have to be diversified enough and related to the targeted audience that we are going to define in this chapter. We are also going to provide a strong and interesting analysis of our respondents.

- **Chapter 4: Application Prototype** explains the goals we defined for our prototype implementation. It also describes what are the other applications that are relevant that we didn't choose. We are going to explain our implementation in details and also which gesture is related to which action according to our prototype.

- **Chapter 5: Prototype Test** we are going to discuss about the participants of the experiments, the conduct of the experiment, the participants feelings and their both *PSSUQ* answers. The second *PSSUQ* took place one month later. We are going to compare them. Finally, we are going to analyze a memorability test, that took place just before the second experiment, to see which gestures are the harder to assimilate.

- **Chapter 6: Additional Benchmarks of Recognizers** are going to provide an additional testing of our gestures set definition which is complementary to our previous work. We are going to focus on different recognizers.

- **Chapter 7: Conclusion and Future Work** is a summary of all the different chapters and assumptions we made through this work. It is going to introduce the future relevant work related to our thesis.

Chapter 2

Related Work

2.1 Overview

In this chapter, we are going to start with the analysis of the gesture recognition market where we are going to observe the market forecast and the *Gartner Hype Cycle* [33] on gesture control devices through the years. Then, we are going to overview all the different resources found in the literature that are related to applications using 2D and 3D gestures together. These resources are going to determine the types of application that are relevant to include in the survey.

2.2 Gesture Recognition Market Analysis

In this section, we are going to analyze the market of gesture recognition according to two indicators. The first is the market forecast over years, whereas the second one is the "Hype Cycle" where we are going to focus on the gesture control devices. The goal of the section is to prove that gesture recognition is a promising market.

2.2.1 Market Forecast

In this subsection, we are going to focus on the **Compound annual growth rate**(CAGR) which indicates the market forecast. Unfortunately, market report from marketing company are very expensive. However, we can analyze and compare multiple CAGR from multiple different reports without paying anything.

The gesture recognition is often segmented in two main different categories: the touch-less and touch devices. As our 3DTouchpad combines both types of environment and there are few devices of this types, it looks like that the device positions itself on a niche strategy. The interest in this technology tends to grow with the market. Therefore, a high forecast increases the relevance of making a prototype application according to our device.

Table 2.1 contains the name of the company that performs the report, the estimated CAGR, the years and the reference.

DataBridge	20.5%	2021-2028	[5]
Grand View Research	19.1%	2022-2030	[24]
imarc	26.2%	2021-2026	[7]
Mordor Intelligence	27.9%	2021-2026	[8]
Total	24.43%	/	/

Table 2.1: Compound Annual growth rate per report

The market forecast from now to about 2028 is approximately equal to an increase of **24.43%**. This reinforces the relevance of our work as the market has a huge potential in the coming years.

2.2.2 Gesture Control Devices Hype Cycle

The goal of this section is to determine and analyze the hype around the gestures control devices. For instance, it can be a leap motion [23], 3DTouchpad [15] and so on. The figure 2.1 represents the evolution of the hype according to gestures devices through the years. According to all the results found [29, 30, 31, 32], we can see that the hype was in the innovation trigger phase until 2015. From 2016, the phase is now at the peak of the inflated expectations. We can see that gestures control devices hype slowly climbs through years. Unfortunately, we have not been able to access more recent resources about this domain since 2018.

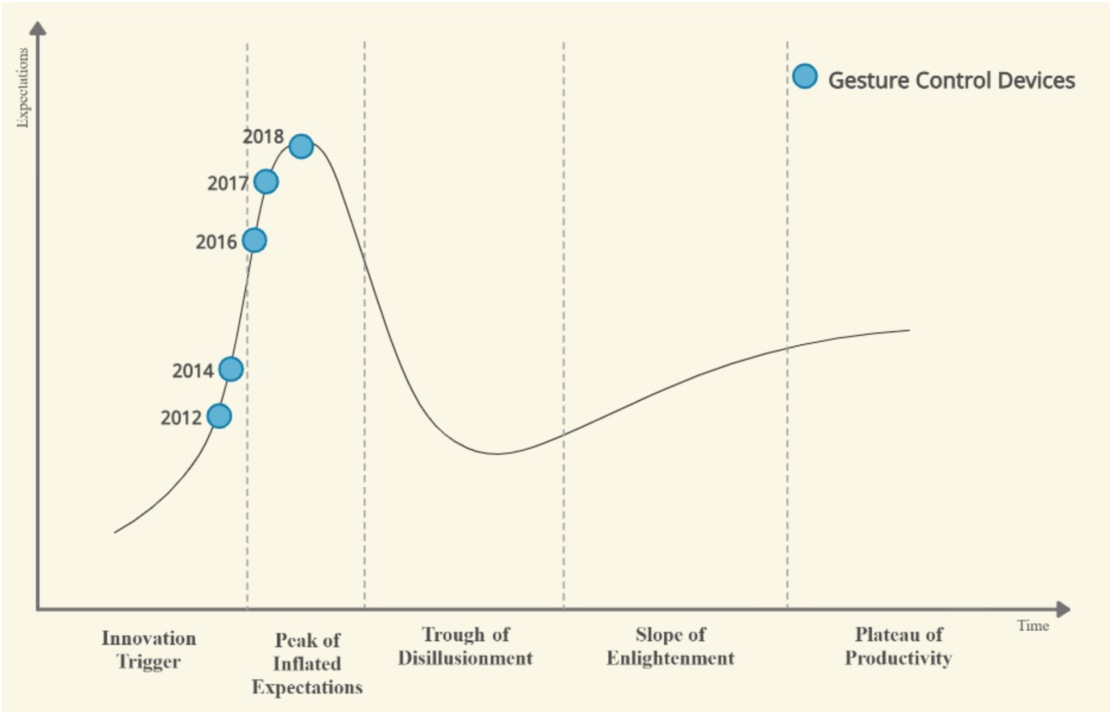


Figure 2.1: Hype Cycle of gesture control Devices through the years.

However, the domain "gesture control devices" is now included in the whole **"Digital Twin of the Person"** [1] from figure 2.2. This whole includes the way people interact with the digital world. For instance, it contains voice, vision and gesture interactions. This set is presented has one the five emerging technology trends from 2020. On figure 2.2, the digital twin seems to have a very ambitious future.

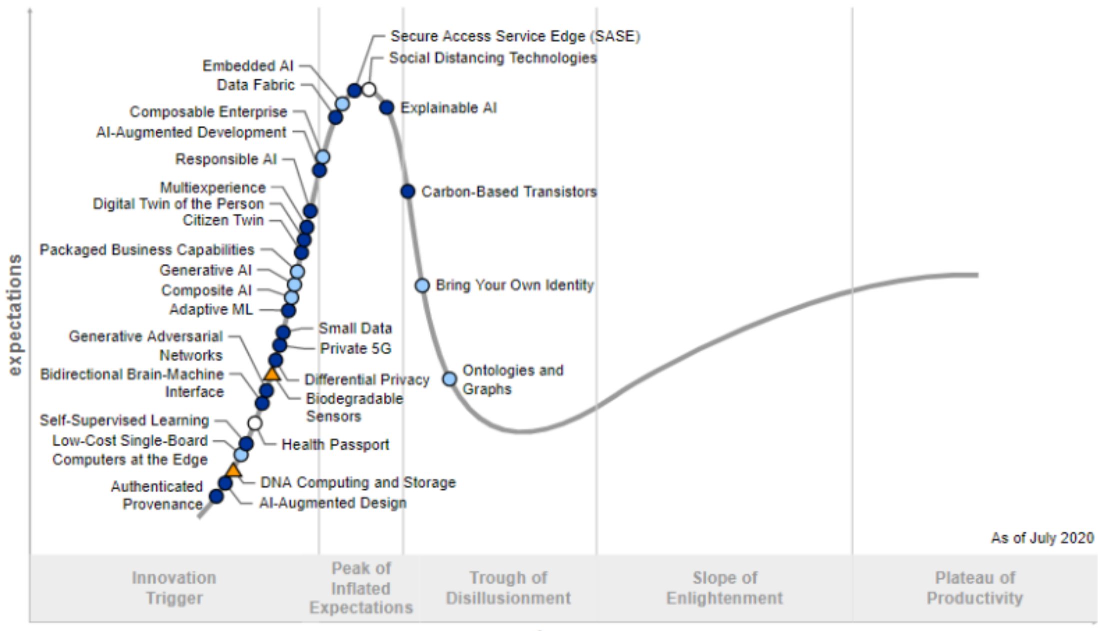


Figure 2.2: Hype Cycle for Emerging Technologies, 2020 [1]

The different resources proves that that there is a clear interest in gesture devices in our modern society and, consequently, in gesture applications related to these devices. This confirms our belief that this work is a relevant follow-up to our previous work [17].

2.3 History of applications using 2D and 3D gestures together

In this section, we are going to list all the different interesting applications which use mixed gestures in the same working environment and that we found inside the literature.

S. Malik and Joseph Laszlo published **Visual Touchpad: A Two-handed Gestural Input Device** [13] in 2004. In this paper, authors present the *Visual Touchpad* device. It combines a low-cost vision-based input device which is able to acquire two-handed gestures with two stereos hand tracking system which acquire data above the touching surface. Multi-touch gestures are allowed. To test this device, they implemented a basic application that interacts with images. They can zoom, unzoom, move the images and many other stuffs.

Alexander Schick et al. published **Extending Touch: Towards Interaction with Large-Scale Surfaces** [25] in 2009. As you can imagine, touch gestures are not suitable for large surfaces. If you have to make consequent touch gestures, it could be really annoying and not useful. Sometimes, these 2D gestures are not even possible as the starting point is to high. Therefore, in this paper, they developed an extension to the touching surface. This extension is based on 3D reconstruction with RGB camera. To test their work, they proposed an application that can be adapted to large display and where users can interact with 2D or 3D gestures.

T. Shlomer et al. published **Gesture Recognition with a Wii Controller** [26] in 2008. In this paper, authors analyzed gestures performed by one hand and they particularly choosed to work with the Wiimote controller. As you can imagine, they use a game application for their experiments.

Michel Ortega et al. published **AirMouse: Finger Gesture for 2D and 3D Interaction** [19] in 2009. This paper consists in the combinations of a classical keyboard with a dedicated area in the air where we can perform 3D gestures. The application used for this experiment can interact with desktops applications. Therefore, they proved that mixed gestures could really increase the ergonomy.

Jorg Muller et al. published **MirrorTouch: combining touch and mid-air gestures for public displays** [16] in 2014. In this paper, authors decided to focus on public terminal as we find in the station or in fast-food. They have developed these terminals so that they can interact with Air+Touch gestures. These devices combined a touch area on its whole surface with sensors which can detect mid-air gestures in the front of the terminal. According to this device, they implemented a game application on which people could see their shapes and where they have to hit falling cubes from top to win some points.

Andrew Bradgon et al. published **Code Space: Touch + Air Gesture Hybrid Interactions for Supporting Developer Meetings** [3] in 2011. In this paper, authors focused on Air+Touch gestures in a meeting environment. They developed the combination of a smartphone with a Microsoft Kinect sensor [39]. Smartphone is used for touch interaction and allow multi-touch gestures whereas the Kinect is responsible for mid-air gestures. They implement lots of different gestures that are useful to handle meeting interaction on a presentation. Participants were showing a huge interest in this new kind of interaction. It could really increase the efficiency of a meeting.

Carole Plasson et al. published **3D Tabletop AR: A Comparison of mid-air, Touch and Touch+Mid-Air Interaction** [22] in 2020. In this paper, authors developed a *Tabletop Augmented Reality(AR)* system. They used the combinations between 3D modeling and touch interactions (directly on the tabletop) to create Air+Touch gestures. For their experiment, they implemented an augmented reality application where participants could drag and move objects.

Nicolai Marquadt et al. published **The Continuous Interaction Space: Interaction Techniques Unifying Touch and Gesture on and above a**

Digital Surface [14] in 2011. They based their work on the continuous interaction space term. The devices were developed as the combination between a multi-touch surface and a dedicated area above the surface where you can perform air gestures. For this paper, authors implemented an augmented reality application where you interact with several objects. For instance, you can perform a pouring gesture to drop the content of the object.

Minseok Kim and Jae Yeol Lee published **Touch and hand gesture-based interactions for directly manipulating 3D virtual objects in mobile augmented reality** [9]. For this paper, They used first a 2D screen (a smartphone for instance) which is responsible for touch interactions. They also used the *Leap Motion sensor* [23] to track the hand of the user and sense the hand gestures performed. According to these sensors, they implemented an Augmented reality applications that is visualized on the touching screen where the users are able to interact with what is shown on the screen.

Georg Hackenberg et al. published **Lightweight Palm and Finger Tracking for Real-Time 3D Gesture Control** [6] in 2011. In this paper, they combined multi-touch interactions with a time-of-flight camera. It captures depth maps using infrared signals. According to the device, they implemented a virtual reality application which is shown on the wall with a projector. For instance, user can rotate the object shown on the screen thanks to his gestures.

G. Kouroupetoglou and A. Pino published **Using Wiimote for 2D and 3D Pointing Tasks: Gesture Performance Evaluation** [10] in 2011. The Wii controller allows touch interaction with its button and also 3D gesture thanks to the 4-LED infrared light source. For this study, they based their results on a game application.

Philip Cohen, David McGee published **Multimodal Interaction for 2D and 3D Environments** [4] in 1999. In this paper, authors used a tool that allow hand gesture and touch interaction like the Wiimote controller. According to this device, they implemented a virtual reality application. This application represent a map where we can draw "digital ink" on it or also point the map with a laser. For instance, this kind of application can be really interesting for a military briefing.

Benjamin Nuernberger et al. published **Interpreting 2D Gesture Annotations in 3D Augmented Reality** [18] in 2016. In this paper, they implemented an augmented virtual application. Thanks to the application, you draw gesture performed in 2D space or 3D space. For instance in 3D space, you can draw in front of and behind an object of the picture whereas in the 2D mode you can only draw in front of the picture.

YA Yusoff, AH Basori, F Mohamed published **Interactive hand and arm gesture control for 2D medical image and 3D volumetric medical visualization** [37] in 2013. They used a Microsoft Kinect [39] to recognize 2D and 3D gestures. According to the Kinect, users could interact with a medical application. To be precise, it is a medical image application which also allows 3D visualized model.

A Pino, E Tzemis et al. published **Using Kinect for 2D and 3D Pointing Tasks: Performance Evaluation** [21] in 2013. In this experiment, they combined a Microsoft Kinect with a mouse. In this study, they re-use the application of the study mentioned earlier: "*Using Wiimote for 2D and 3D Pointing Tasks: Gesture Performance Evaluation*[10].

Marco Barsotti, Fabio Paternò, Francesca Pulina published **A Web Framework for Cross-device Gestures Between Personal Devices and Public Displays** [2] in 2017. In this paper, they developed a framework for cross-devices interaction. For instance, this framework can be used with a kinect and a smartphone. A "car configurator" was implemented to be associated with this framework. Therefore, the gesture recognized from both devices could launch some actions as *rotate the car*, *Show an accessory*, *Select an accessory*, ...
As you can see, there are a lot of different types of relevant application handling both 2D and 3D gestures. Table 2.2 is a small comparative table about our topics.

	Game	Medical	Reality
Use	Personal	Professional	Personal/Professional
Target	Everyone	Medical Staffs	Experts
Possible Prototype	Skiing Games, Sports Games, Adventure Games	Control Device, Move dangerous, object remotely	Modelisation app, Visualisation app, VR Games
	Presentation	Ergonomics	Smart Devices
Use	Personal/Professional	Personal	Personal
Target	Everyone	Computer Workers	Everyone
Possible Prototype	handling PowerPoint, handling Interactive keyboard	Gesture = key combinations as copy, paste and so on	Handling a smart house or multiple device

Table 2.2: Comparative Table.

After analyzing all the previous papers, we can put forward the next six types of application that are representative of the literature. For each kind, we are going to mention why they are relevant.

- **Game:** It enables a fun way to use and test gestures devices. Microchip presents a game associated to the 3DTouchad in this **link**. The goal was to move a sliding penguin from left or right with Air+Touch gestures and avoid obstacles. There are also a lot of games which are using gestures technology as the Wii games, Oculus Rift games, Kinect games, Playstation games and many others. After all that, it seems relevant to include a game application as a possible choice for the prototype.

- **Medical:** This kind of application could be useful in the future. The pandemic has increased the desire to have touch-less applications to avoid the transmission of microbes. A medical application with Air+Touch gestures seems to be very interesting as it needs very precious action or scaling action. 2D gestures fit well with precious actions whereas 3D gestures fit well with scaling actions. Sometimes, it is also recommended for medical staff to avoid to stay in a contaminated room. Therefore, the use of an application that handle medical equipment from an another room with Air+Touch gestures seem to be a good alternative to the situation.

- **Reality:** It exists three different kind of reality application: *Augmented Reality(AR), Virtual Reality(VR), Mixed Reality.* This "market" is increasing through the years. Today, we are witnessing an increase of new *"Virtual Reality center"* where you can play VR games. These applications are also very useful for modelling applications where 3D gestures are important for the ease of the use. Thus, a reality application will also be included inside the applications cluster of the future survey.

- **Presentation:** Presentation are currently more and more present in many domains. There is a clear interest for a company to increase the presentation quality. For instance, several devices have been developed as a "remote controller" which handle the presentation or a pencil that is able to write directly on the presentation. Therefore, Air+Touch gestures could really increase the presentation quality as Bragdon [3] concluded in his paper. Thus, presentation application will be put inside our possible applications cluster.

- **Increase Ergonomics:** The idea behind this application is that the device is directly connected to your desktop and so it can launch directly some action without pressing any key on your keyboard. For instance, a cross gesture will copy and another gesture will paste what was copied. Therefore, it could really increase the comfort of the user if gestures are well adapted. This is why this kind of application is taken in account.

- **Smart Devices Interaction:** In my previous work [17], I gave a lot of different examples of how the 3DTouchpad used with one or multiple smart device(s) could be promising. For instance, in a fully connected house, you can use the 2D gestures to choose the devices and the 3D gestures to launch a action. This kind of applications with Air+touch gestures are very rare.

Thus, it is relevant to include smart devices interaction inside the cluster for our survey.

Now that the six possible types of applications have been clearly identified by our state of art, we can go to the next step which is the collection of answers to our survey to determine the most relevant type for our prototype.

2.4 Conclusion

In this chapter, we first started to prove that there is a clear interest in gesture applications in our society thanks to our gesture recognition market analysis and therefore it is interesting to implement an application linked to our 3DTouchpad device. Then, we analysed multiple articles from literature to determine which types of application were the most relevant to an Air+Touch gestures application. Six types were emerging from this analysis and are going to take place in our survey from chapter 3.

Chapter 3

Survey

3.1 Overview

In this chapter, we are going to determine which type of application is the most relevant for our future prototype. First, we are going to introduce our methodology and the targeted audience to answer our survey. Once we got enough answers, we will be able to analyze the personal information of the respondents and their answers. Finally, we will choose the type of application based on this analysis.

3.2 Survey Methodology

For this survey, we are going to use *Google Form* as it is safer given the pandemic situation. This Google form is composed of 6 different sections in which we collected important data for our work.

- **Section 1:** we collected the name and surname, the year of birth, the profession, the domain work and studies of each participant.
- **Section 2:** contains questions on the frequency of technological tools use as *a computer, a smartphone, a tablet, video games console and a Kinect*. Participants answer each tool with a number between 1 and 7 according to frequency of use. 1 means that the user has never used the tool whereas 7 means that the user uses the tool almost everyday.
- **Section 3:** contains a small presentation of the *3DTouchpad* [15] device and the YouTube video which explains how the device is working with QuantumLeap Framework [28] and which are the gestures already defined. At the end of this section, respondents are asked if they are familiar with the device and if so, how often they use it.

- **Section 4:** respondents are asked what kind of application seems the most relevant to them. This question is asked before the next section in order to get their initial ideas and not to bias their answers by the different types of application chosen in chapter 2.

- **Section 5:** contains the six different types of applications defined in chapter 2. Each respondent has to mark from 1 to 7 the pertinence of the type. 1 means that the type of application with the use of the 3DTouchpad is absolutely not relevant for the respondent whereas 7 represents an absolute pertinence for the respondent.

- **Section 6:** respondents are asked if they have any additional comments to the google form and if they have any new ideas in relation to the question in section 4.

To make this survey as relevant as possible, we decided to define precise targeted respondents for each type of application.

- **Game:** The targeted audience is people who are used to playing video games.

- **Medical:** The targeted audience is people who are working in the medical sector or that are studying medicine, biology and so on.

- **Reality:** The targeted audience is people who are working in the virtual reality/augmented reality enterprise as *VR Evasion*, for instance.

- **Presentation:** The target audience is those who are used to using presentations. For instance, a teacher, a student or a manager can be relevant.

- **Increase Ergonomics:** The targeted audience is people who are spending lot of time on their computer teleworking.

- **Smart Devices Interaction:** There is no clear targeted audience here, but nowadays almost everyone uses smart devices with their smartphone, car or TV.

3.3 Demographic Data

3.3.1 Information of the respondents

52 people answered out the survey on Google Form . We tried to provide a coherent selection of respondents to be diversified enough and relevant according to what we explained previously. On the graphic 3.1, you can see the distribution over the year of birth for each respondent. As you can see, there are two different emerging "clusters", one from 1952 to 1968 and the second one from 1984 to 2000.

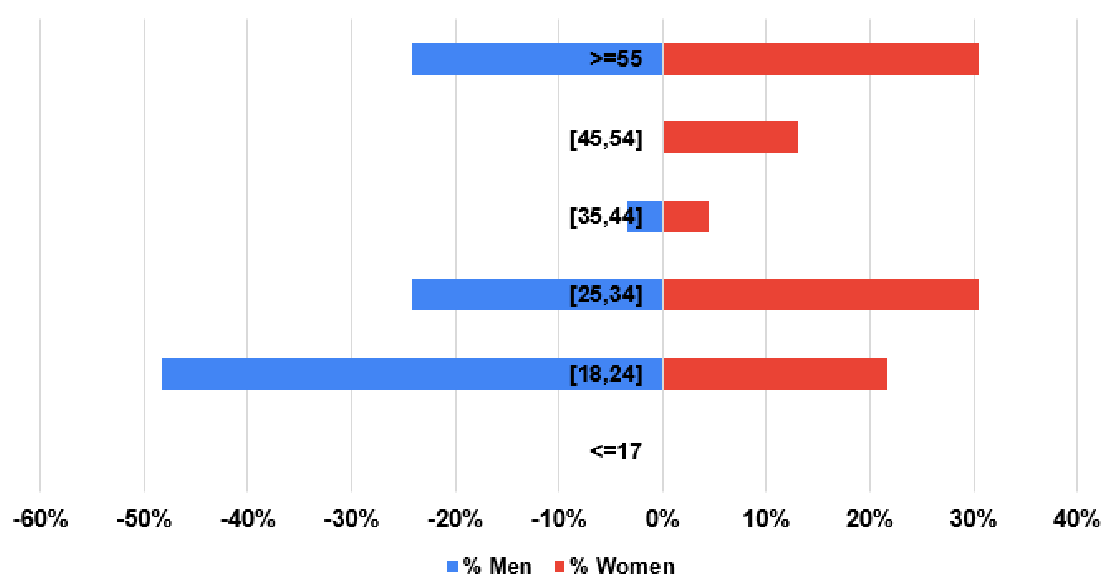


Figure 3.1: Pyramid of age of the participants.

On graphic 3.2, you can see the proportion between men and women. There are 29 men who answered the survey whereas 23 women answered. So the women are represented by 44% and men by 56%.

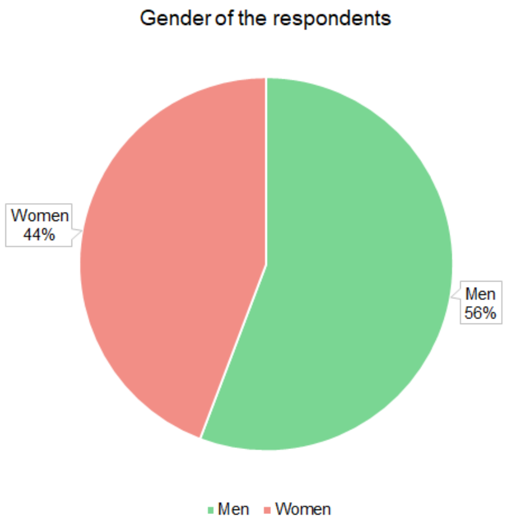


Figure 3.2: Proportion between men and women who answered the survey.

On graphic 3.3, you can analyze the distribution of the profession of our participants. There are two main professions in our survey: **Student(36%)** and **Employee(34%)**. The other professions in the survey are **Retired(6%)**, **Manager(8%)**, **Freelance(8%)**, **Civil Servant(4%)**, **Unemployed(2%)** and **Volunteering Abroad(2%)**.

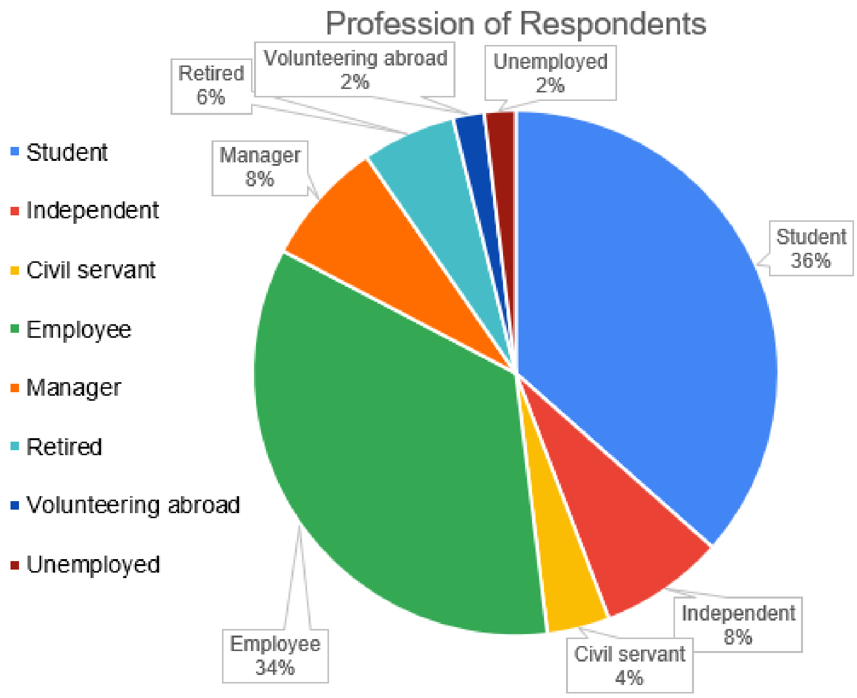


Figure 3.3: Distribution of the profession of our participants

The graphic 3.4 shows the different fields of work per respondents. As you can see, there are four fields which are more represented than the other. They are the **student(9)**, people who work in **administration(9)**, **medical staff(7)** and **teacher(6)**. These clusters mainly represent the targeted audiences that we mentioned in the beginning of this chapter. There are 17 other fields of work represented inside our surveys to be diversified enough and get a global view about our answers.

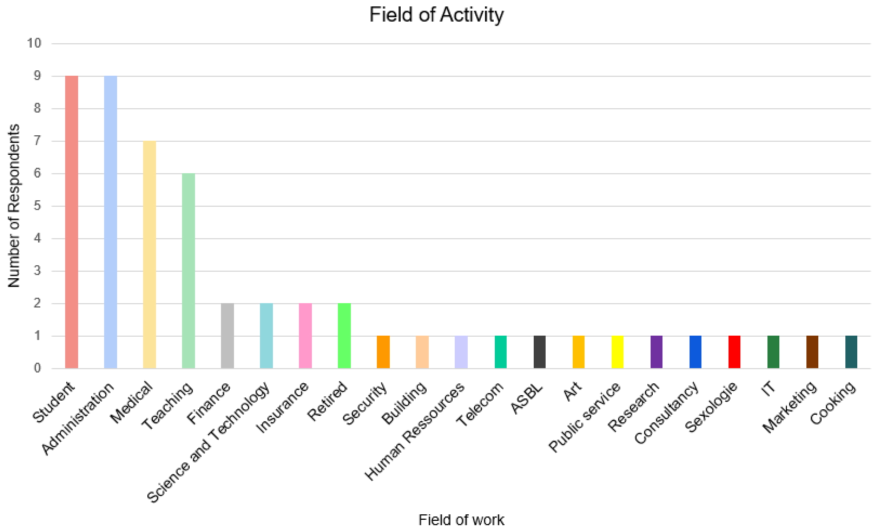


Figure 3.4: Distribution of the field of work per respondents

The graphic 3.5 shows the actual studies or the last diploma of each respondent. As you can see, the respondents are coming from many different horizons. Some are business engineers, civil engineers, nursing and so on. Apart from the targeted audiences, we tried to be as diversified as possible.

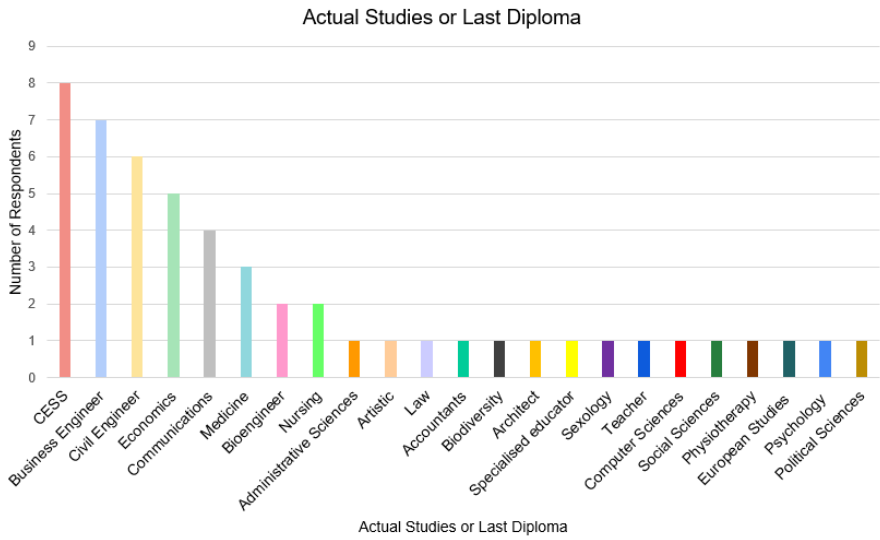


Figure 3.5: Distribution of actual studies or last diploma per respondents

To conclude this section about the personal information of our respondents, we are going to resume quickly what we have said. During this survey, We have tried to respect the constraints of the targeted respondents as much as possible to satisfy our requirements. We also tried to be the most representative of the overall to get a relevant opinion for our survey. It is important to not limit our survey to a specific group as this may introduce some subjectivity into our analysis.

3.3.2 Tools Frequencies

In this new subsection, we are going to discuss about the usage frequencies of respondents according to different tools. We asked them about computer, smartphone, tablet, video game console and Kinect. The results of the questions are resumed inside the graphic 3.6. We can see that a large proportion of respondents uses a computer and a smartphone very frequently whereas nobody uses a Kinect. This graphic is important to notice if one targeted audience is represented. It is people who often play video games. Here, one might think that it is not respected but this is because people who play video games a lot use their computers much more than game consoles. And so these people are among the one who use their computers a lot. Moreover, the use of smartphones and computer has highly increased as it is getting harder for gaming consoles to be used as much as computers and smartphones. Therefore, we are going to conclude this section by saying that "yes, the targeted audience for games application is satisfied".

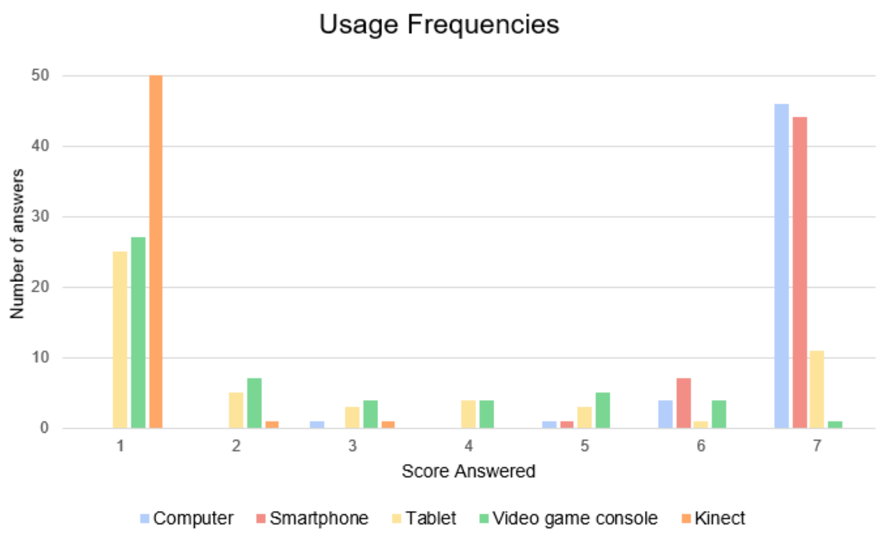


Figure 3.6: Distribution of actual studies or last diploma per respondents

3.3.3 Knowledge of the Device

In our survey, we also asked if each respondent knows the 3DTouchpad [15] device. If they know it, we also asked them to provide a score between 1 and 7 according to their frequency of use. The two graphics 3.7 show the distribution of respondents who know or don't know the device and their frequency of use if they know this device. As you can see, **11 respondents(21%)** know the 3DTouchpad and **41 respondents(79%)** have never heard about the device before. Among the 21%, 82% have almost never used it, 18% answered with a score 2 to the question. People knew the device through my work. They were involved in the sample gathering from our previous work or they were interested about my work and I showed them how it was working. Without these situations, they never heard about it.

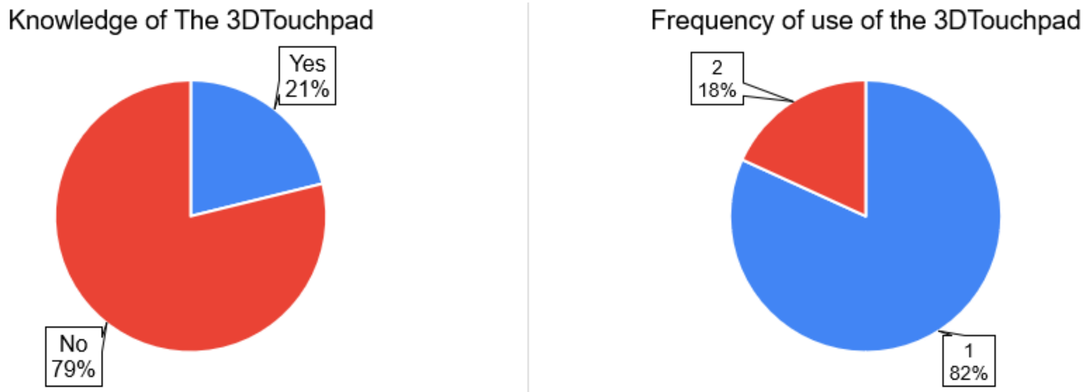


Figure 3.7: Knowledge and Frequency of use about the 3DTouchpad.

3.4 Tasks

3.4.1 Spontaneous ideas for application

For this subsection, respondents were asked to suggest ideas of application that they find relevant to the use of the 3DTouchpad. They were asked this question before they could see our proposition. This is why we called this section "the spontaneous idea". The different answers were very interesting. First, it proved that our selection in Chapter 2 was relevant as the six types of applications are represented inside the figure 3.8 and the table 3.1. It also gave us very interesting new ideas as an *application for blind or disabled people or for video editing.*

The principal propositions emerging from this section, is a **Game**, a **3D modeling**, a **drawing** and a **presentation** application.

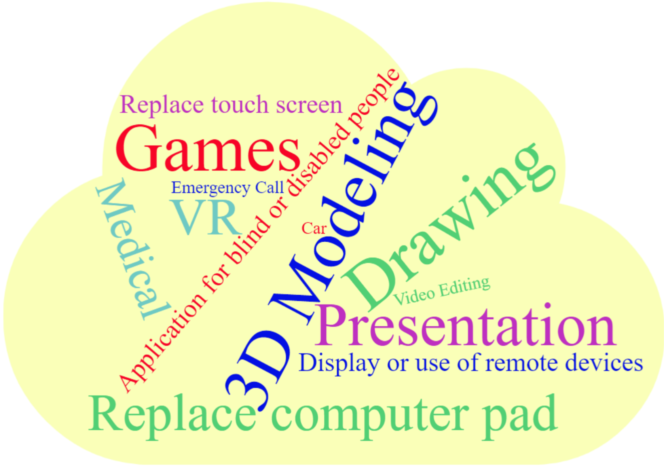


Figure 3.8: Workcloud [34] from the spontaneous ideas from respondents.

Application Proposition	#
Games	9
3D Modeling	8
Drawing	8
Presentation	7
Medical	5
Application for blind or disabled people	3
Display or use of remote devices	3
Replace computer pad	3
Replace touch screen	3
VR	3
Car	1
Emergency Call	1
Video Editing	1

Table 3.1: Respondents' proposal and the number of times it was proposed.

3.4.2 Relevance of the types of applications chosen

For this subsection, we asked respondents to answer several questions about our six types of applications defined in chapter 2. As a quick reminder, the six types are: **Game**, **Medical**,**Virtual Reality**, **Presentation**, **Increase Ergonomics**,**Smart devices Interaction**

Each respondent has to choose a value between 1 to 7 for each kind of application.

The value 1 means that this kind of application is absolutely not relevant to our device whereas the value 7 means that this kind of application fits perfectly to our device. On graphic 3.9 and in table 3.2, we can directly notice that there is one more relevant application than the other: the **presentation application**. If we compute the average relevance score for each application, we obtain the following results: *Games*(**5,28**), *Medical*(**5.56**), *VR*(**5,78**), *Presentation*(**6.02**), *Increase Ergonomics*(**5.28**), *Smart Devices Interaction*(**5.26**). So, thanks to all the different informations from this section, presentation application is the most relevant type of application according to the respondents.

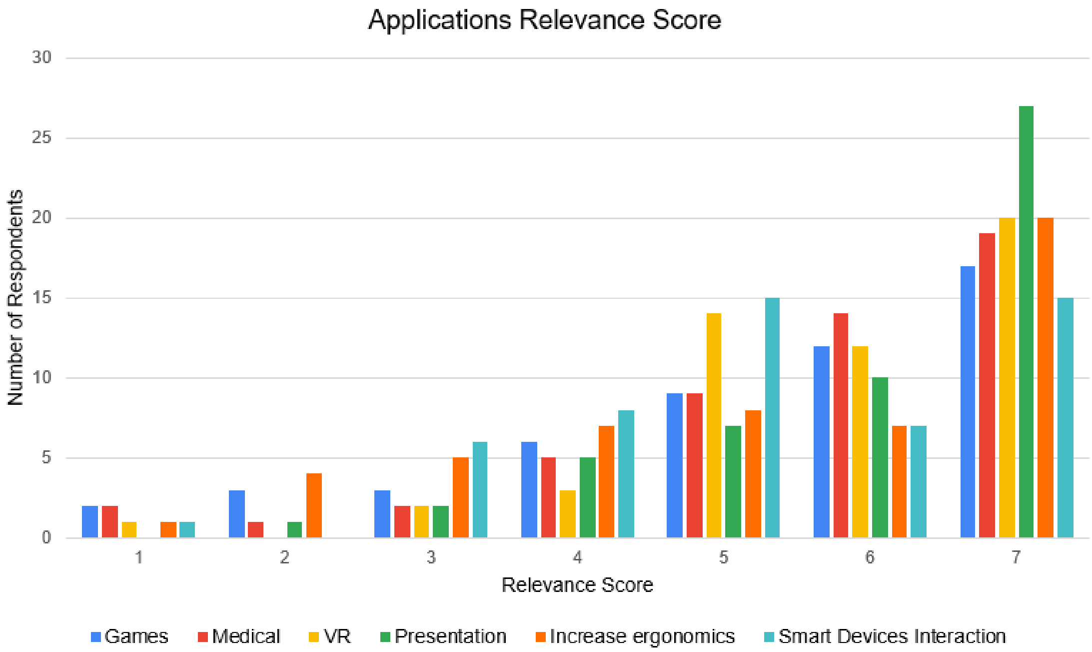


Figure 3.9: Score of relevance per respondents and per applications.

Relevance	Game	Medical	VR	Presentation	Ergonomics	Smart Devices
1	2	2	1	0	1	1
2	3	1	0	1	4	0
3	3	2	2	2	5	6
5	9	9	14	7	8	15
6	12	14	12	10	7	7
7	17	19	20	27	20	15

Table 3.2: Number of answers from the respondents per relevance score per application

3.4.3 New ideas

At the end of the survey, respondents were asked if they had some new propositions after seeing our propositions. Some people suggested some interesting new applications not covered in section 3.6.

- Application for bedridden people
- Surgery application
- Military application

Remotely controlled, teaching, meeting, game applications were also a lot mentioned.

3.5 Dynamics Dashboard

To increase the quality of this chapter, we decided to create an interactive dashboard using *Microsoft Power Bi* [35]. It is a really strong tool to help us to clearly visualize the different results according to the selected cluster. A dashboard is very powerful for showing the key performance indicator(**KPI**). On figures 3.10 and 3.11, you can discover our dashboard. We decided for our work to focus on the following KPI:

- *The mean of the different kind of application based on their targeted audience.*
- *The mean of the overall answers for each type of application.*
- *The preference by the different domains clusters.*
- *The preference according to the ages.*

And so after consulting the different results, the presentation application is the most relevant according to the the KPIs. For each cluster, it always get a good scores. Therefore, we are sure that the presentation application is the most relevant application to prototype according to our Google form results.

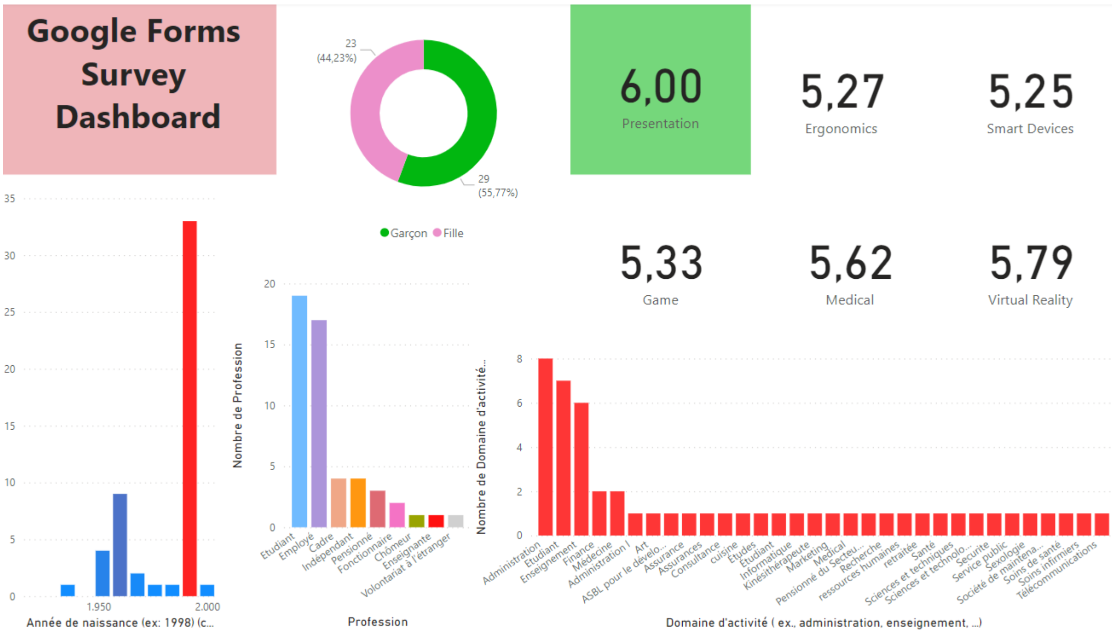


Figure 3.10: Dashboard(1/2)

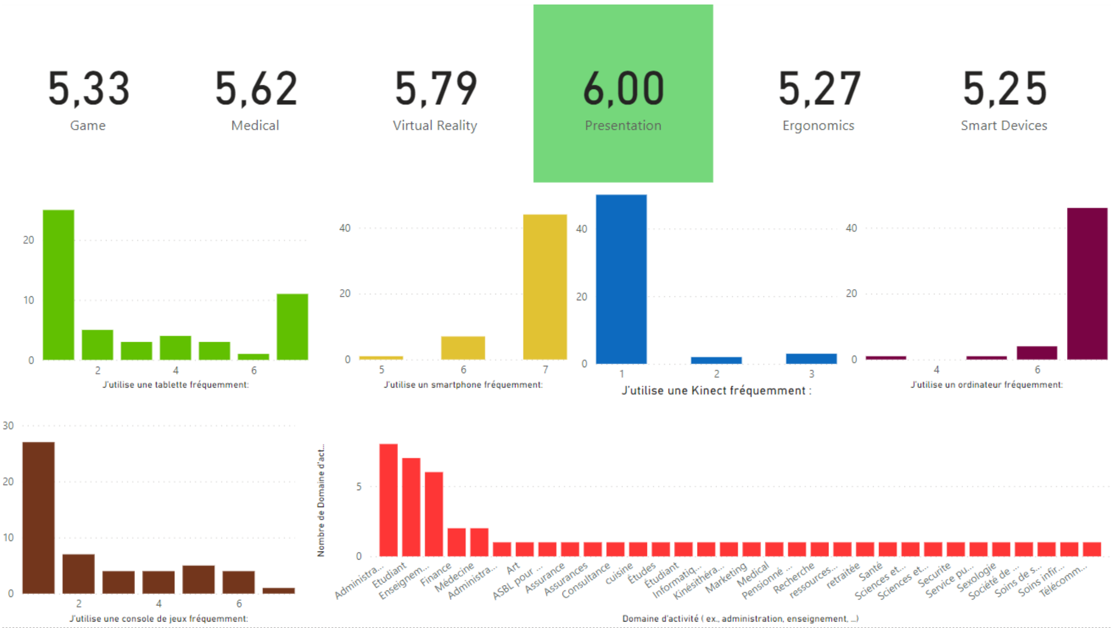


Figure 3.11: Dashboard(2/2)

3.6 Conclusion

In the first part of this chapter, we presented you our survey methodology and who was the targeted audience per application. Then, we showed the demographic data of our all participants. These data were important according to our KPIs from our dashboard. After, we analyzed the data coming from the different tasks performed by the participants. Finally, we built a *Microsoft Power BI dashboard* to visualize the prototype preference according to our KPIs. All results tend to show that an **presentation application** is the most relevant type for our prototype.

Chapter 4

Application Prototype

4.1 Overview

In the previous chapter, we determined the most relevant types of applications related to Air+Touch gestures from 3DTouchpad according to answers of 52 respondents. A presentation application results from the analysis of the survey. That is what this chapter is about. First, we are going to discuss about our prototype specifications. Then, we are going to explain our implementation and which gestures are chosen according to which actions. Finally, we are going to show you some pictures of the prototype.

4.2 Prototype Specifications

In this section, we are going to mention the purpose of our prototype. Then, we are going to speak about the other prototypes that are also relevant but that we let for a future work.

4.2.1 Our concept

The general idea is a **slide show** application that we can manage with Air+Touch gestures. Nowadays, almost everyone is making slide show presentation for their works, for their meetings, for their sports, for lectures and so on. One major thing about slide show is that we often interact behind the computer. For instance, when you have to move to the previous or the next slide, you have to press arrows from your computer, whereas if you can interact with your slide discreetly with gestures and a small device, this can improve the exchange and make it more natural. This is why we think that it is an interesting and promising prototype according to Air+Touch gestures. During this work, we were contacted by a friend who is currently a teacher and he is using the slides shows for his classes a lot. He expressed his interest in this type of device because he believes that it could really improve the quality of his presentations.

4.2.2 Other idea for a future work

- **A cross-device meeting application:** It could be interesting to implement this kind application.In their paper [3], Andrew Bradgon et al. already realized this kind of application. This could really increase meetings quality and efficiency.

- **Interaction with an interactive board:** As with an interactive board, we can make many different things. It could be really interesting to link the 3DTouchpad to the board and therefore, manage everything with it as a slide show, a video, and so on.

4.3 Gestures Definition

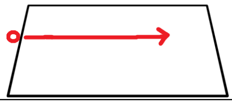
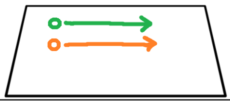
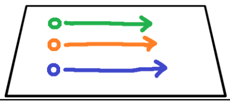
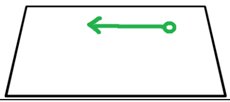
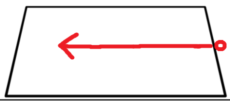
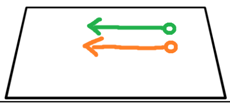
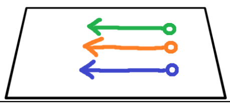
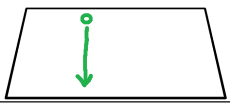
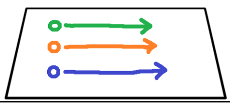
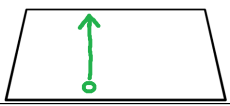
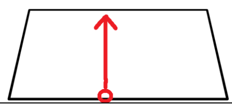
In this section, we are going to mention each chosen gestures from the gesture set definition of our previous work [17] according to the prototype functionalities. We based our choices of gestures and actions according to the common gestures of our today life as using a smartphone and also to the two elicitation studies [36] analysis performed on our previous thesis. For instance, left swipe is associated to the "previous slide" action. If the user wants to go back two slides, he can use his two fingers to swipe. We tried our best to choice gestures that are relevant to their related actions. Table 4.1 contains gestures names, the related action to its gesture and a small drawing to help you visualise the gestures correctly. On the pictures, colors have different significations:

- **Red:** Gestures performed above the device. (*3D gesture*)

- **Green:** Gestures performed by the first finger which has touched the device. (*2DTouch1*)

- **Orange:** Gestures performed by the second finger which has touched the device. (*2DTouch2*)

- **Blue:** Gestures performed by the third finger which has touched the device. (*2DTouch3*)

Gesture Name	Dimension	Related Action	Draw
SwipeRight1Touch	2D	Next slide	
AirSwipeRight	3D	Next Slide	
SwipeRight2Touch	2D	Advances Two Slides	
SwipeRight3Touch	2D	Advances Three Slides	
SwipeLeft1Touch	2D	Previous Slide	
AirSwipeLeft	3D	Previous Slide	
SwipeLeft2Touch	2D	Back Two Slides	
SwipeLeft3Touch	2D	Back Three Slides	
SwipeDown1Touch	2D	Go to Gesture Description	
AirSwipeBackward	3D	Go to Gesture Description	
SwipeUp1Touch	2D	Quit Gesture Description	
AirSwipeForward	3D	Quit Gesture Description	

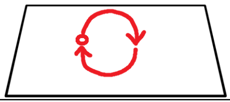
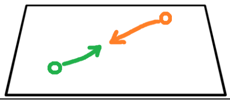
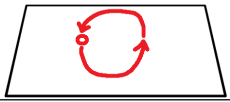
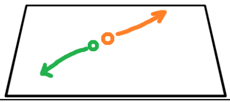
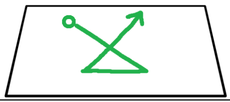
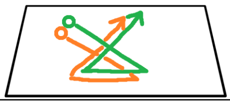
AirCircleRight	3D	Zoom	
ZoomPinchIn2Touch	2D	Zoom	
AirCircleLeft	3D	Unzoom	
ZoomPinchOut2Touch	2D	Unzoom	
Cross1Touch	2D	End the presentation	
Cross2Touch	2D	Restart the presentation	

Table 4.1: Gestures names, related actions and drawing of each gesture.

The gesture description is a slide which mentions each chosen gesture and it related action. The goal is to provide a reminder when the user needs it.

4.4 Implementation

Before going through this section, we are going to make a brief reminder on how the whole software is working. The figure 4.1 is a representation of what we are going to explain. First, a gesture is performed on the 3DTouchpad. The C script receives the raw data from the gesture performed and sends it to the QuantumLeap framework [28]. The framework analyzes the data and transmits it to the recognizer. If the recognizer recognizes a gesture, it will send this gesture to the connected application. In our work, the connected application is our prototype. Depending on the gesture received, the prototype runs the related action. This is how the whole process is working.

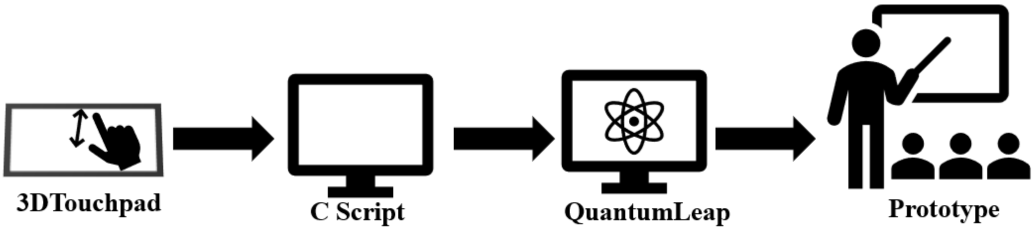


Figure 4.1: The process from the device to our prototype.

For this work, we are not going to talk about the C script and the QuantumLeap framework as they don't change from our previous work [17]. As a quick reminder, figure 4.2 is the overall architecture from QuantumLeap Framework [28].

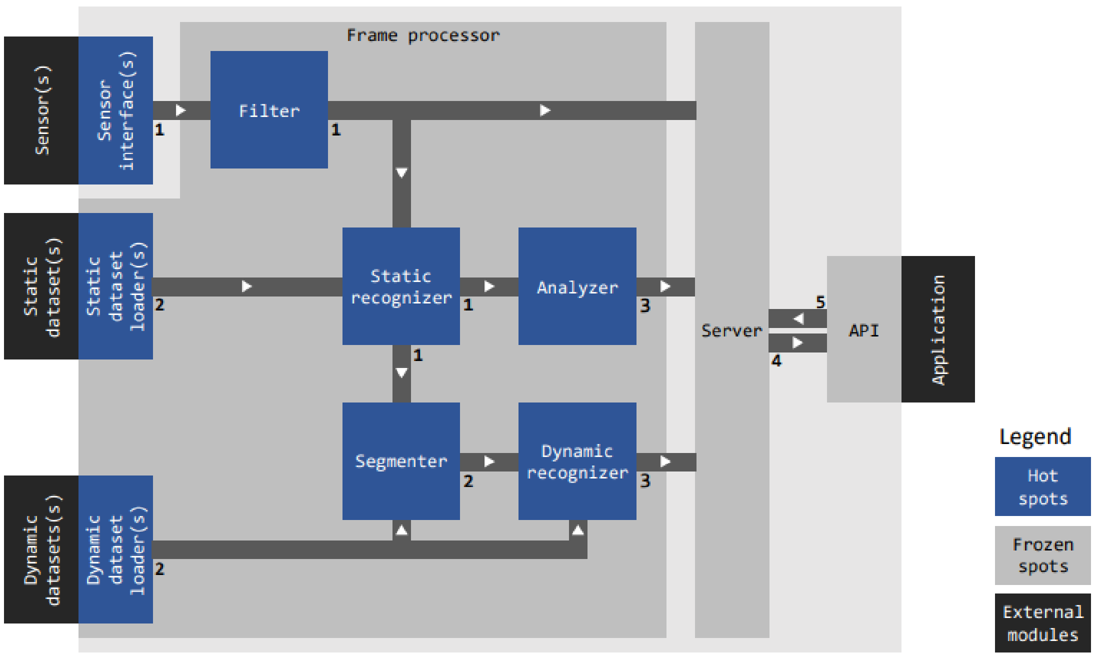


Figure 4.2: QuantumLeap Framework Architecture. [28]

In this work, We are just going to discuss about the implementation of our prototype. The whole implementation of our prototype can be found in appendix A.11.

- **Import quantumleapjs module:** thanks to Arthur Sluÿters work, there is a module to import which is called *quantumleapjs*. This module is responsible for the management of gesture. So, it connects the prototype to the framework.

When a gesture is received, it allows the prototype to know which gestures have been received. We are not going to detail his work but only what needs to be changed in order to handle our prototype.

- **Import the slides:** we need to import the slides. For instance:

```
import Slide1 from './Presentation_Folder/Slide1.JPG';
import Slide2 from './Presentation_Folder/Slide2.JPG';
import Slide3 from './Presentation_Folder/Slide3.JPG';
```

- **Bind gestures and variables definitions:** we need to bind the gestures that we want to use, the table containing the different slides and the table that is responsible for the zoom action.

```
// This is how we bind the different gestures
this.onGesture = this.onGesture.bind(this);
this.onAirSwipeRight = this.onAirSwipeRight.bind(this);
this.onAirSwipeLeft = this.onAirSwipeLeft.bind(this);
this.onAirCircleLeft = this.onAirCircleLeft.bind(this);
this.onAirCircleRight=this.onAirCircleRight.bind(this);
```

- **Received Gestures:** Once all the desired gestures are well mentioned to the gesture handler, you need to set up what the prototype will run when a gesture is received from QuantumLeap. For instance when a AirSwipeRight is received, the presentation goes to the next slide. In Javascript, this can be seen as:

```
onAirSwipeRight(){
if(this.counterSlide + 1 < this.SlideTable.length-1){
this.counterSlide = this.counterSlide+1
this.setState({
  image: this.SlideTable[this.counterSlide],
  displayTime: 999,
});
}
```

This is how we decided to implement our prototype which satisfy the different requirements that we mentioned earlier in this chapter.

4.5 Prototype Visualisation

4.5.1 Prototype Utilisation

Figure 4.3 is a representation of someone using our prototype. On the pink square, you can see the interface prototype on the laptop whereas in the red zone, you can see that the user is performing some gestures above the 3DTouchpad.

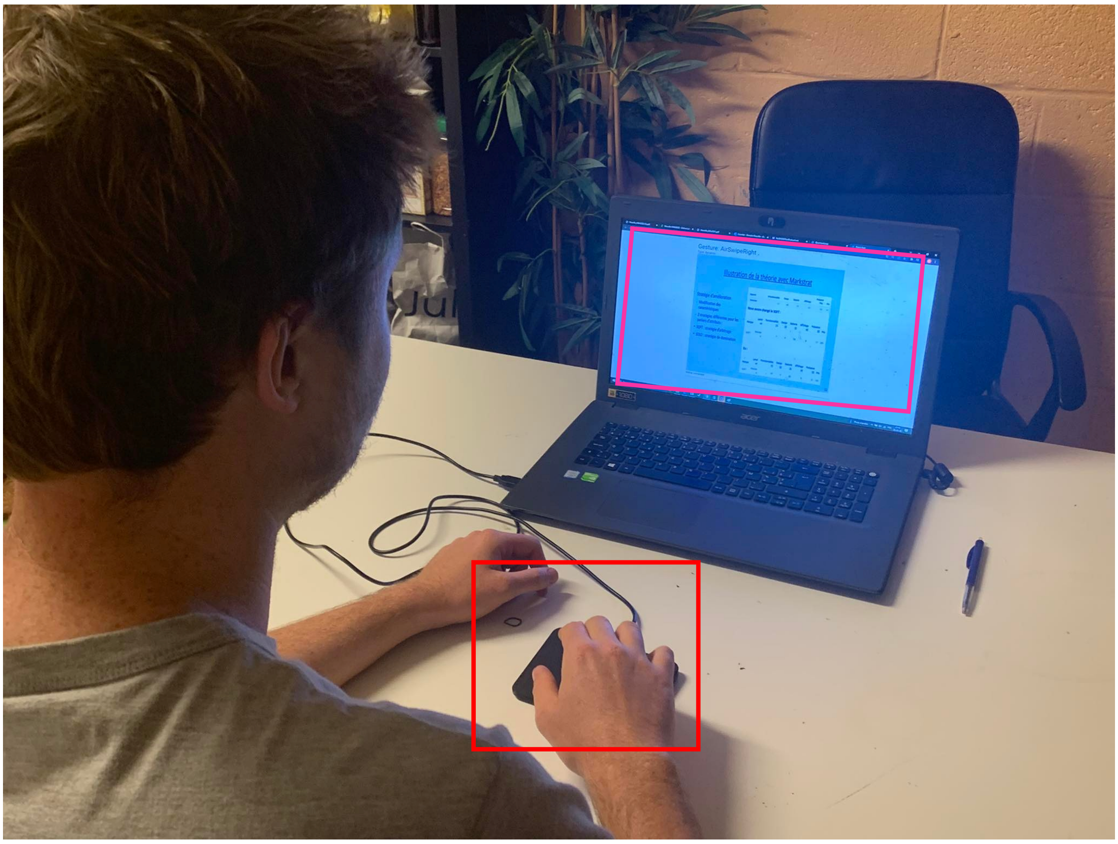


Figure 4.3: Picture of a participant using the prototype with the 3DTouchpad

4.5.2 Interface Components

The interface of our prototype is divided into three components. The first part is the one surrounded by the red line on figure 4.4, it contains the information relative to the last gesture send by *QuantumLeap Framework* and received by our prototype. Here, you can see that the last gesture received is an AirSwipeRight, its related action is to move to the next slide. Therefore, we moved from the slide 1 to the slide 2. The second part is the green zone, it shows the current slide of the presentation. It will change according to the gestures received. The last part is the blue one, it simply mentions to us that we are well connected to *QuantumLeap Framework* [28]

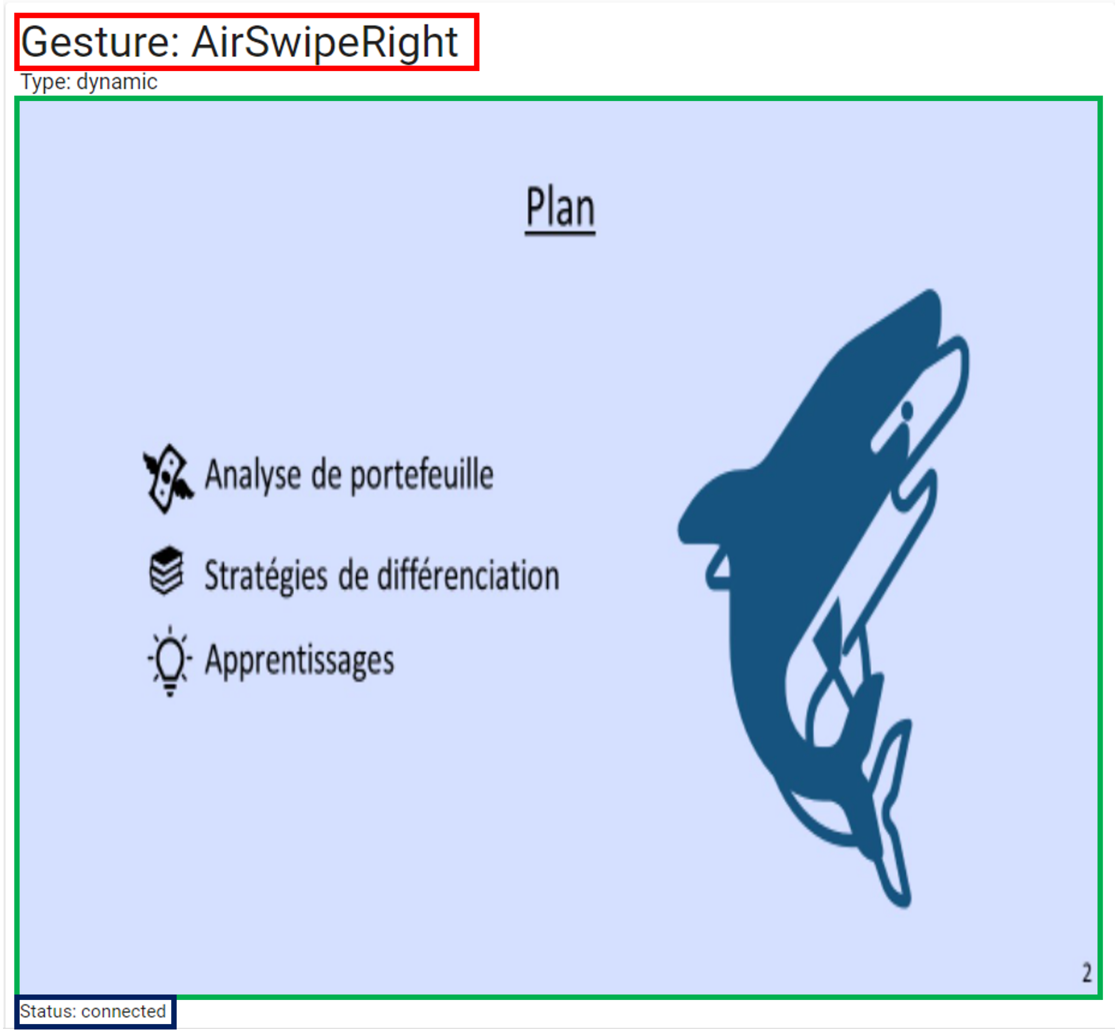


Figure 4.4: Prototype Interface.

4.5.3 Gesture Description

Figure 4.5 represent the gesture description of our prototype. Its goal is to help users if they don't remember a gesture, they can reach it by performing an *AirSwipeBackward* or a *SwipeDown1Touch*. It is an important tool for our prototype as it is hard to assimilate some gestures or actions. This allows the participant not to feel lost in case of doubt.

Gesture: AirSwipeBackward

Type: dynamic

Gesture Name	Action description
SwipeRight1Touch	Next Slide
AirSwipeRight	Next Slide
SwipeRight2Touch	Advance two Slides
SwipeRight3Touch	Advance three Slides
SwipeLeft1Touch	Previous Slide
AirSwipeLeft	Previous Slide
SwipeLeft2Touch	Back two Slides
SwipeLeft3Touch	Back three Slides
SwipeDown1Touch	Gesture Description
AirSwipeBackward	Gesture Description
SwipeUp1Touch	Quit Gesture Description
AirSwipeForward	Quit Gesture Description
AirCircleRight	Zoom on the slide
AirCircleLeft	Dezoom on the slide
ZoomPinchIn2Touch	Zoom on the slide
ZoomPinchOut2Touch	Dezoom on the slide
Cross1Touch	End the presentation
Cross2Touch	Go to the first Slide

Status: connected

Figure 4.5: Gesture Description

4.5.4 Zoom Effect

An other functionalities of our prototype is that we can zoom on a slide. For instance, figure 4.6 contains the slide "before zooming" on the left part of the figure and "after zooming" on the right part. To zoom, user needs to perform an *AirCircleLeft* or a pinch-in gesture. Once he has finish with zoom, he can perform a gesture to dezoom or go to next slide and therefore and then, he can continue his presentation.

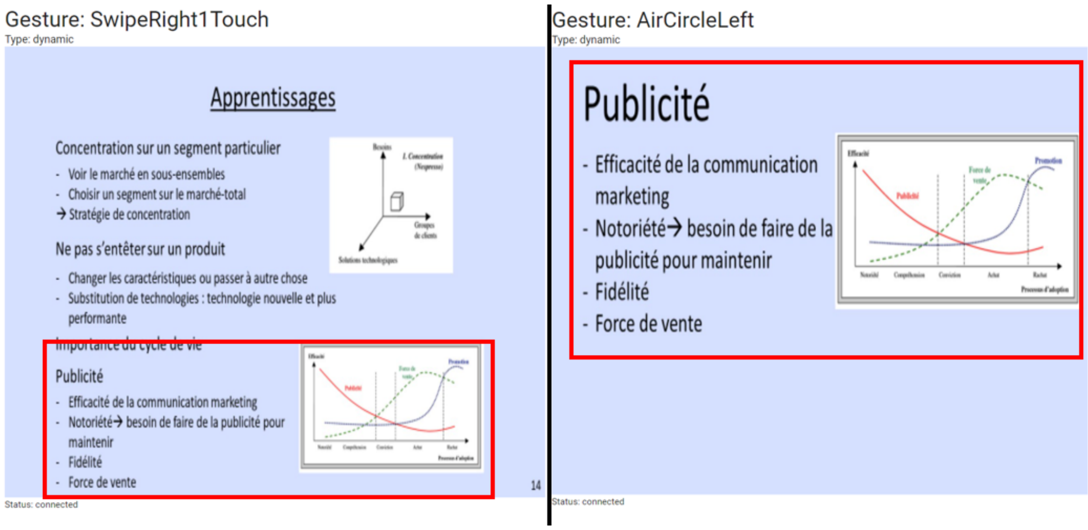


Figure 4.6: The zoom effect.

4.6 Conclusion

During this chapter, we have explained on what the prototype is based and why this is interesting. We also introduced some other possibles prototypes that are promising and interesting. We are going to let these prototypes for our next work. After that, we explained which gestures were chosen from the 34 different gestures defined in our previous work [17]. We explained the related action for each gesture. We also provided a small draw to help readers to visualise correctly each gesture. And to finish the chapter, we have discussed the implementation of our prototype and we also included the overall implementation in appendix A.11 for people who are interested.

Chapter 5

Prototype Test

5.1 Overview

The objective of this chapter is to test the prototype that we implemented in chapter 4. First of all, we are going to overview the personal information of our participants. Then, we are going to explain how the experiments were conducted.The same experiment was performed twice for each participant with one month of interval. After, we are going through the answers of the two **Post-Study System Usability Questionnaire (PSSUQ)** from the both experiments. Finally, we are going to analyse the memorability of participants according to the chosen gestures of our prototype after one month without use.

5.2 Participants Information

All the participants who performed the testing of our prototype were asked some questions. We collected their **age**, **gender**, **status** and **field of work**. Table 5.1 resumes all these participants information.

ID	Age	Gender	Status	Field Of Work
1.	26	Woman	Worker	Psychology
2.	24	Woman	Worker	Physiotherapist
3.	54	Woman	Worker	Financial
4.	24	Man	Worker	Teaching
5.	23	Man	Student	Bio-engineer
6.	25	Man	Student	Economy
7.	55	Man	Worker	Manager
8.	23	Woman	Student	Psychology
9.	23	Man	Student	Teaching
10.	23	Woman	Student	Communication

Table 5.1: Personal information of our participants

The ten participants were already involved in this work through chapter 3. They answered our *Google Forms*. It was nice as they already knew the device and the goal of this work. They feel very interested and curious on what was the type of application which emerged from their answers.

We also tried to be as diversified as possible. We chose **5** women and **5** men, **5** students and **5** workers. We got people from psychology, teaching, communication and so on.

5.3 Conduct of the experiment

First of all, the presentation starts with a slide containing the different gestures and their related action. People were free to take the time they wanted to try to master the different gestures. After that, they were free to move from slide to slide with our device. Sometimes, we asked them to move back or advance three slides with the adequate gesture. Sometimes, we also asked them to perform a zoom then an unzoom action on the slide. They naturally came back to the gesture description slide when they didn't remember what was the related gesture. We also randomly asked them to end the presentation and then restart the presentation with the "cross" gestures. As the gestures used are very easy, everyone easily understood how to use the device with our prototype. They didn't take too much time.

This experiment were performed twice for each participant. After, they had to answer the *PSSUQ*.

Before the second experiment, we decided to perform a memorability test. For instance, we asked each participant to perform an action as move to the next slide with a 2D gesture. Each action were asked in a random order according to each participant. According to the results, we are going to be able to analyze which gestures were easy to assimilate and which are not.

5.4 Post-Study System Usability Questionnaire (PSSUQ)

5.4.1 Introduction

This questionnaire expresses the satisfaction of an user with the usability of our prototype. This questions came directly when a participant finished the experiment. The responses are going to help us to understand what are the points which are good and what are the points to be improved. Each question has to be answered with a value between 1 to 7. The value 1 means that the user is strongly disagreeing with the statement. Whereas value 7 means that the user is strongly agreeing with the question. Table 5.2 resumes the sixteen questions of the questionnaire. After all these questions, we are going to ask what are the positive and negative feelings about our prototype for each participant.

ID	Question
1	Overall, I am satisfied with the ease of use of this system
2	The system was simple to use
3	I was able to complete the tasks and scenarios quickly with this system
4	I felt comfortable with the system
5	The system was easy to learn
6	I expect to become productive quickly using this system
7	The system provided me with clear error messages to solve the problems
8	I was able to correct every mistake simply and quickly
9	Information provided by the system, such as online help, messages, documentation, was clearly provided by the system
10	I easily found the information I needed
11	The information was useful to me in completing the tasks and scenarios
12	Information was clearly presented on the screen
13	The system interface was nice
14	I liked using the system interface
15	The system had all the functions I wanted
16	Overall, I am satisfied with the system

Table 5.2: The 16 Questions of the PSSUQ.

5.4.2 First Post-Study System Usability Questionnaire

Questionnaire Results

The overall results from figure 5.1, 5.2 is very promising. Our prototype obtains a mean of **5.49** for the whole questionnaire. As you can see, they strongly agree with question (**2, 3, 11 ,12**). That means that our implementation is simple to use, can be easily mastered and the information and the interface are clear. In contrast, question 6, has a low average, we believe that this poor result is due to the fact that the participants were not in an atmosphere of presentation. And so they may have had difficulty projecting themselves and visualizing how it would improve a presentation. The other response averages are in line with what we expected and prove the relevance of our work.

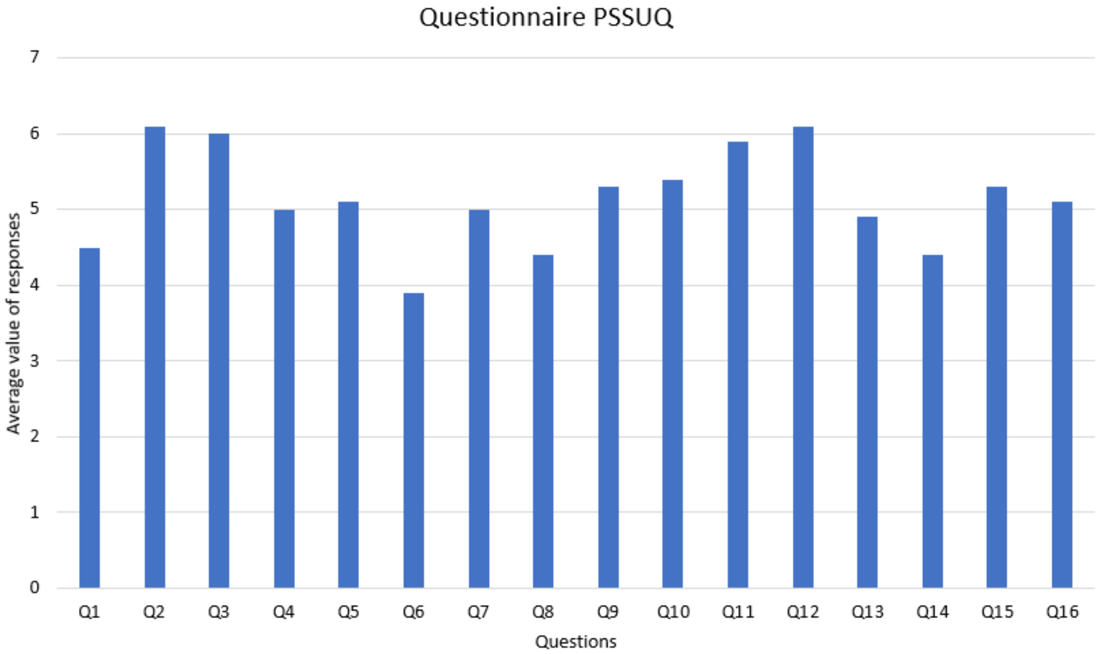


Figure 5.1: Average response from the first PSSUQ

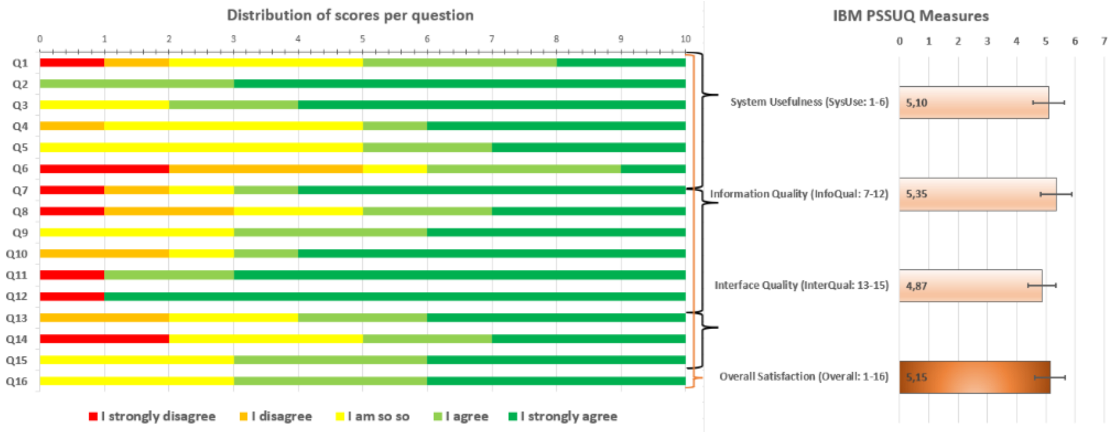


Figure 5.2: Distribution of scores per question and IBM PSSUQ Measures

Feelings

We also collected some feelings about our device and our prototype. Through this experience, participants were often intrigued by the 3DTouchpad. They often did not know that devices which are able to handle Air+Touch gestures exist. The participants were therefore very captivated and interested as it was new technology for them. They think it can be very promising for the future. In general, they were very positive about our work. However, there are also some negatives points to mention. First, as mentioned earlier, they were not in an optimal presentation environment. It was more of a hands-on situation. Therefore, in the future, it could be interesting to create an adequate atmosphere for the prototype test. An other point is that the prototype has not a full-screen functionalities. For a future work, it may be interesting to increase the number of functionalities.

5.4.3 Second Post-Study System Usability Questionnaire

Questionnaire Results

Figure 5.3, 5.4 result confirm once again that our prototype is very promising according to the results. Our prototype obtains a mean of **5.31** for the whole questionnaire. However, there are 3 questions that get a lower average than the other questions. There are questions **6**, **7** and **13**. Therefore, it could be very interesting to base a future work to embellish the interface. Thanks to this improvement, the expectation to become productive quickly using the system (question 6) will also increase. Other questions have satisfying averages that meet our expectations.

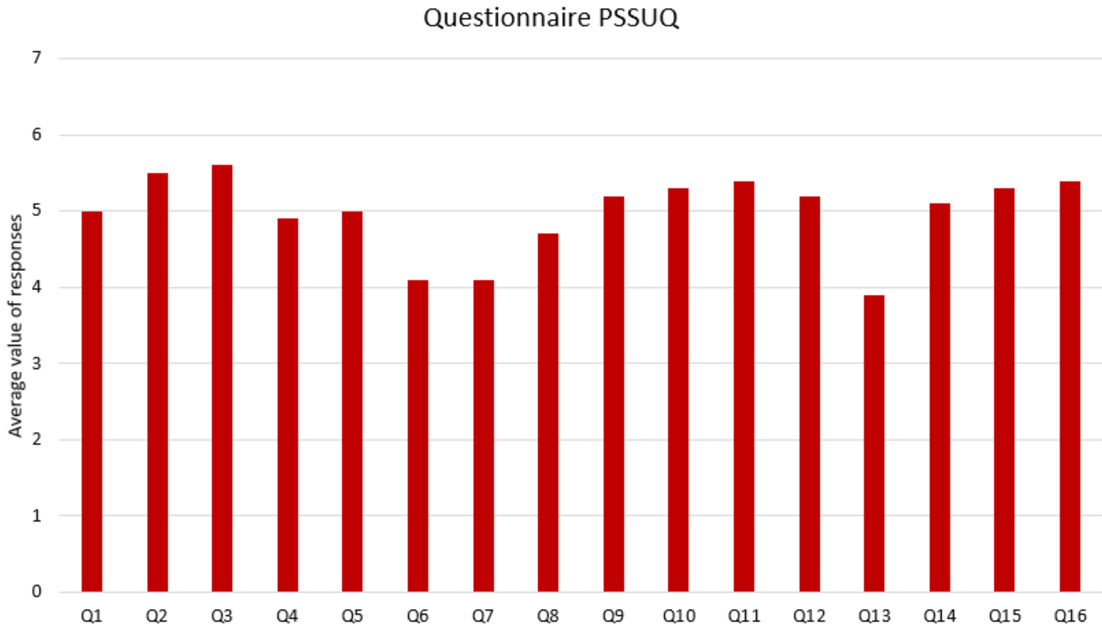


Figure 5.3: Average response from the second PSSUQ

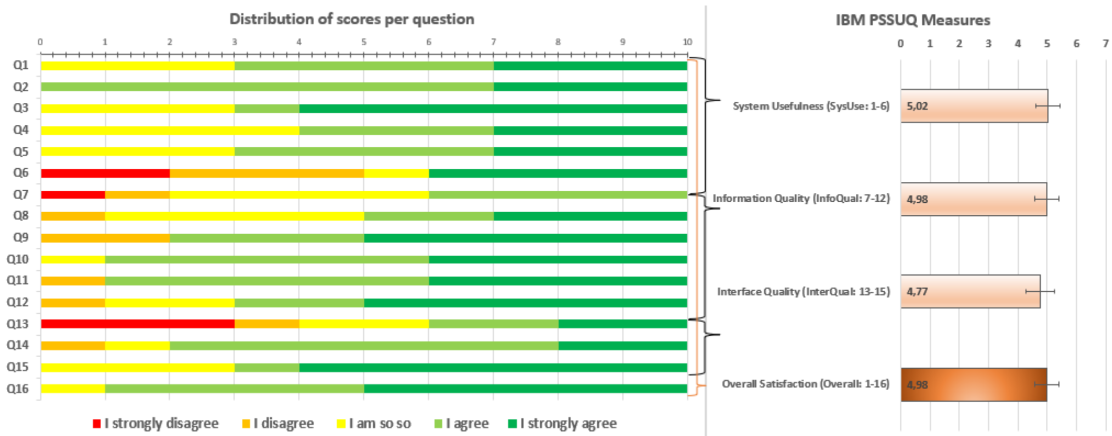


Figure 5.4: Distribution of scores per question and IBM PSSUQ Measures

Feelings

During this second test, people looked less enthusiastic in general as they already knew the prototype and the device. Therefore, the results show more of their true feelings about our prototype. They also gave us some very relevant advice. First, two participants said that it could be very helpful to have the *"Go to Gesture Description"* gesture information directly on the screen. Sometimes, they feel as "trapped" when they didn't remembering this gesture. Secondly, after performing the memorability test with participants, they were efficient directly. This is probably also related to the fact that it was the second time that they performed the experiment. Therefore, it could be interesting to include a kind of tutorial where user can see the action, a small video of the gesture wanted and a small exercise to perform this gesture. A negative point that emerges from this second study is that the interface is not so beautiful as it contains some practicals informations. For instance, it prints the gesture recognized by QuantumLeap. In my case, I think that these kinds of components are essentials as sometimes recognizers can miss-matches and it saves us from being lost. However, it could very interesting to embellish the interface in a future work.

5.4.4 Comparison between both Questionnaire

In this subsection, we are going to focus on the comparison of the both PSSUQ. The figures 5.5,5.6 contain the average score for each question. The red columns are the averages for the second *PSSUQ* whereas the blue ones are related to the first one.

First of all, we can see that the participants satisfactions with the ease of use has increase as they were more familiar with the environment and gestures. The agreement averages over the four questions which people were slightly agreeing on the first *PSSUQ*, obtain a lower average in the second. This can be explained by the fact that people has "mastered" the halve of the gesture set and their related action(see next section). Therefore, they underestimated their use and the helping information. Secondly, we can observe that participants mainly disliked more the interface in the second questionnaire. According to their feelings, this is because they attached more importance to the interface and less to discovering the device and the application. The last variation between both *PSSUQ* is according to question 7. This variation occurs because participants were more familiar with the device and therefore, they enjoyed more the prototype. For the remaining questions, averages are nearly the same between both questionnaire.

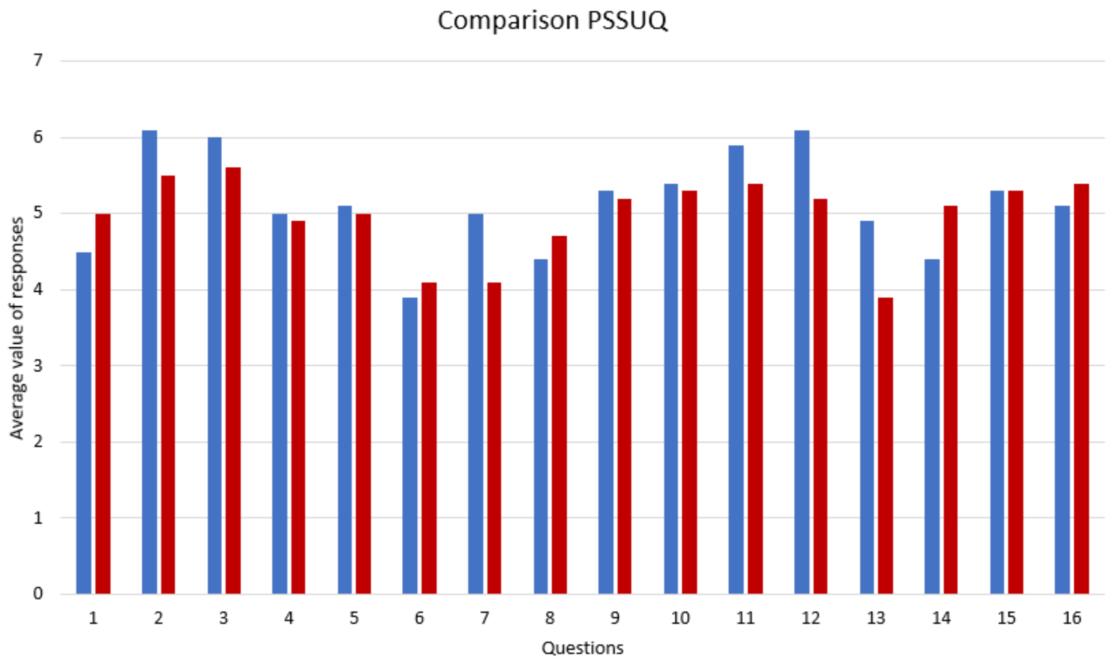


Figure 5.5: Average comparison from both PSSUQ

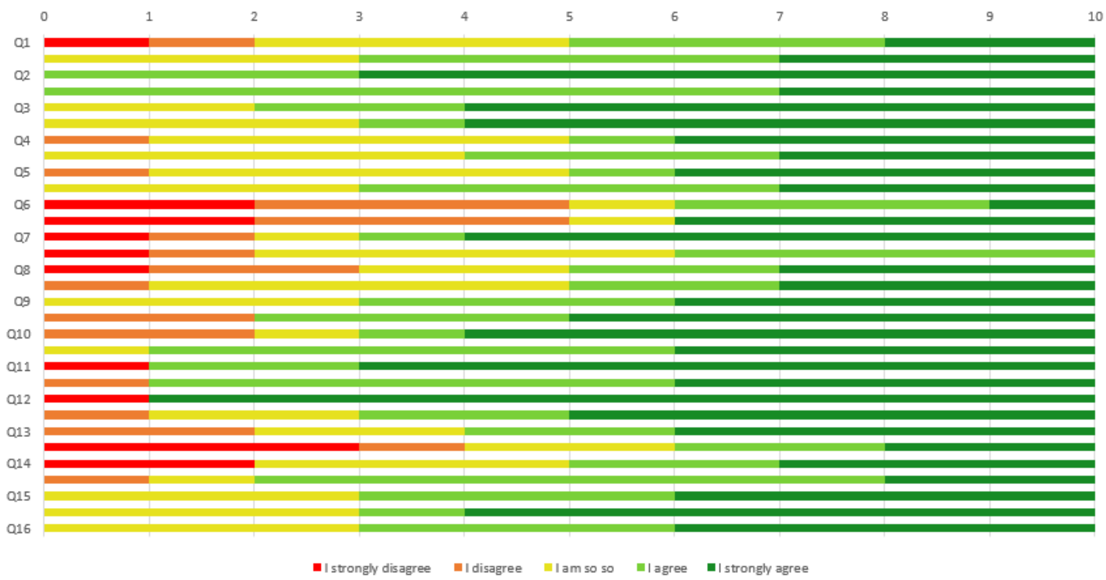


Figure 5.6: Distribution of scores per question

5.5 Memorability Test

In this new section, we are going to analyze the memorability of our participants to our different gestures. During this test, we asked to perform the gesture related to a dedicated action. For instance, we asked them to perform gesture related to the *"Advances two slides action"* with a 2D gesture. If the participant remembered the right gesture, he was given a score of 1, otherwise 0. This test was performed with an interval of 30 to 39 days without using the prototype. The goal was to analyze the participants remembering about 18 chosen gestures. Each action was asked randomly to avoid any influence on the order of the questions.

In appendix A.1, you can find the a table resuming the results from each participants according to each gesture. On the figure 5.7, we can see that gestures which obtained an high rate of remembering, are the swiping gestures(2D & 3D) and the pinch(2D) gestures. These results are due to the fact that these gestures and their related action are almost the same as when using a smartphone. The swiping gestures with multiple fingers were assimilated well. The other gestures as *Swipe Up, Swipe Down, Cross, Circle* obtained a lower score. This is probably because some gestures are less common and actions are less usual. These result are very important as we know which gestures are more problematic and harder to assimilate. Therefore, if we want to implement a tutorial, we will really focus on these gestures.

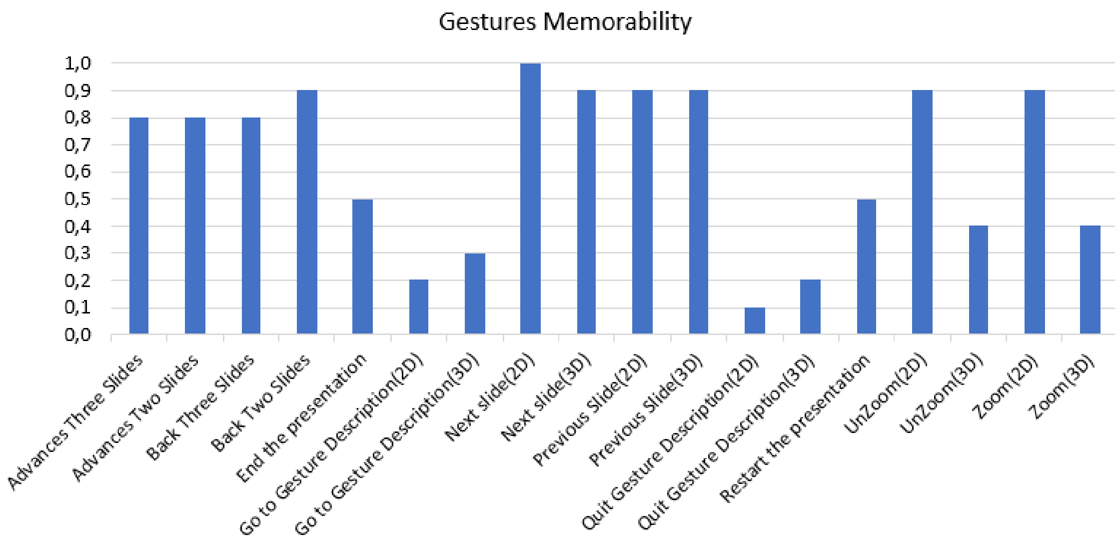


Figure 5.7: Gestures Memorability test

5.6 Conclusion

In this chapter, we first mentioned the personal information of each participant. Then, we presented how experiments were conducted. After, we analyzed both PSSUQ answers of participants and we also discussed about their general feelings on our prototype. Then, we compared the average of each question of both questionnaire. Finally, we analyzed a memorability test that was conduct before the second experiment that happens nearly one month after the previous one. All of these different analysis gave us some relevant new ideas or point to improve as, for instance, create a tutorial, embellish the interface or add useful informations on the interface. This analysis proved the relevance of our work, we got the results we wanted.

Chapter 6

Additional Benchmarks of Recognizers

6.1 Overview

In this chapter, we are going to provide an additional testing to complete our previous work [17]. This analysis is going to be based on different points of interest. The two analyses together are going to be complementary and therefore, they are going to cover a larger set. Firstly, we are going to do a quick reminder on what our previous benchmarks of recognizers were based on. Then, we are going to dress the points on what we want to focus. Finally, we are going to provide a deep, coherent and interesting analysis.

6.2 Previous Benchmarks of Recognizers

Before explaining, it is important to remind you of these definitions.

- **User-dependent Scenario:** Recognizers are trained with samples only performed by the same user.
- **User-Independent Scenario:** Recognizers are trained with samples. There is no constraint on who performed the samples.
- **Recognition Rate:** It is the total number of gestures correctly identified by the recognizer divided by the overall number of tested gestures. Each recognizer must recognize correctly almost all of the gestures [17].

In our previous thesis, we decided to focus our analysis on parameters that can vary. All the criteria are resumed in table 6.1.

Parameters	Variations
Testing Scenario	User-Dependent or User-Independent
Dataset	Right-handed samples, Left-handed samples or Both hand samples
Training Templates	Recognizers are trained with 1 to 8 samples
Recognizers	μF or Jacknife

Table 6.1: Table containing the different parameters and their different variations from our previous testing.

First, we decided to restrict our analysis to two recognizers as the other recognizers were struggling to handle the 34 gestures together. And then, we analyzed the whole gesture set at the same time. We analyzed both recognizers according to the both scenarios.

The benchmark results showed that μF and *Jacknife* could achieve a sufficiently high recognition rate to allow an application using these 34 gestures, despite the scenario chosen. We are not going deeper for this chapter. If you are interest in the result, don't hesitate to read the chapter 5 from our previous work [17].

6.3 New Benchmarks of Recognizers

6.3.1 Goal of the Benchmarks

As in our previous study we tested all our gestures according to two scenarios and three types of dataset, we are going to focus on the recognizers and how they react according to the type of gesture(*2DTouch1*, *2DTouch2*, *2DTouch3*, *3D*). The goal is to compare and to determine some interesting facts between recognizers and our gestures. Some recognizers could not handle the whole gesture set together. For instance, they can confuse 3D gestures with 2D gesture or they can't understand with how many finger(s) the gesture was performed. After analyzing our entire results, it may be interesting to use recognizers associated to one gesture type only. Imagine 3 Cent recognizers acquire an high recognition rate with 2DTouch1 gesture, jacknife with 3D, ... It may possibly increase our recognition rate. This is why it could be really interesting.

6.3.2 Benchmarks Parameters

For our testing, there are two kinds of parameter, the fixed and the non-fixed parameter. Table 6.2 sums up the different parameters and their variations. The light blue color means that the parameter is fixed to the mentioned value whereas the green means that the parameter has some possible variation according to the testing performed. So here, we are always going to use our right-handed dataset to train our recognizers according to the user-independent scenario.

Parameters	Variations
Testing Scenario	User-Independent
Dataset	Right-Handed samples
Training Templates	Recognizers are trained with 1 to 8 samples
Recognizers	μF, μV, 3 Cent, GSPDa, $P3+, Jacknife
Gesture Type	2DTouch1, 2DTouch2, 2DTouch3, 3D

Table 6.2: Parameters of our benchmarks.

6.3.3 Recognizers Comparison

The goal of this subsection is to distinguish which recognizer is better than the others according to the number training templates used and according to the type of gesture. We are going to analyze recognition rate for each recognizer trained with 1, 4 and 8 templates. Based on the *Kruskall-Wallis with post-how Tukey test results* from appendix A.2, we are going to determine a recognizers classification according to their recognition rate to each type of gesture when they are trained with eight templates.

2D Gestures with 1 finger

On figure 6.1, we can directly see that recognizers *μV, GSPDa, $P3+* don't acquire an high recognition rate to be relevant enough for our gesture set definition from our previous work. When the recognizer is trained with 1 template, μF looks to be the best recognizer for these gestures. However, with more templates, jacknife is the best recognizer overall and it converges quickly to the 100% recognition rate. We can also mention that 3 Cent recognizer can obtain good result with 4/8 training templates.

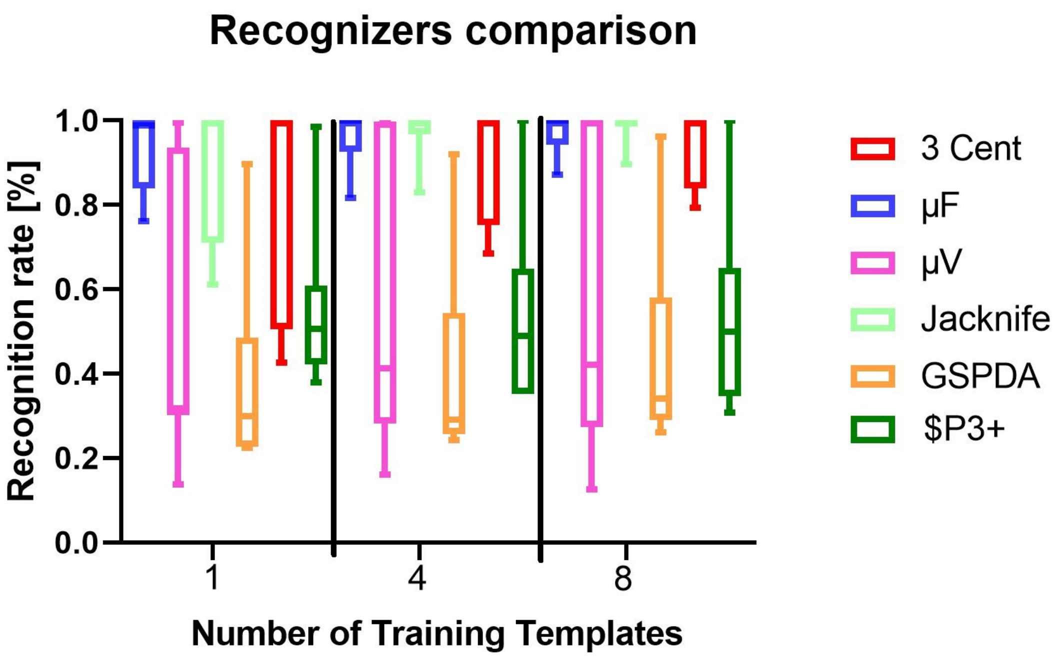


Figure 6.1: Recognizers comparison for 2D gestures with one finger.

1	2	3	4	5	6
Jacknife	μF	3Cent	μV	GSPDA	$P3+

Table 6.3: Classification based on Kruskall-Wallis test results A.2.

2D Gestures with 2 fingers

On figure 6.2, GSPDa and μV obtain very bad recognition rate for these kind of gestures. $P3+ is better than the two mentioned recognizers but unfortunately, it is not as good as the three other recognizers. Therefore, it seems that $P3+ is still not relevant to consider for these two fingers gestures. For the three remaining recognizers, it seems that μF acquires the best results. It has the higher median for each number of training templates. It also has a greater box compared to 3 Cent and Jacknife. So, they could also be a good alternatives to μF.

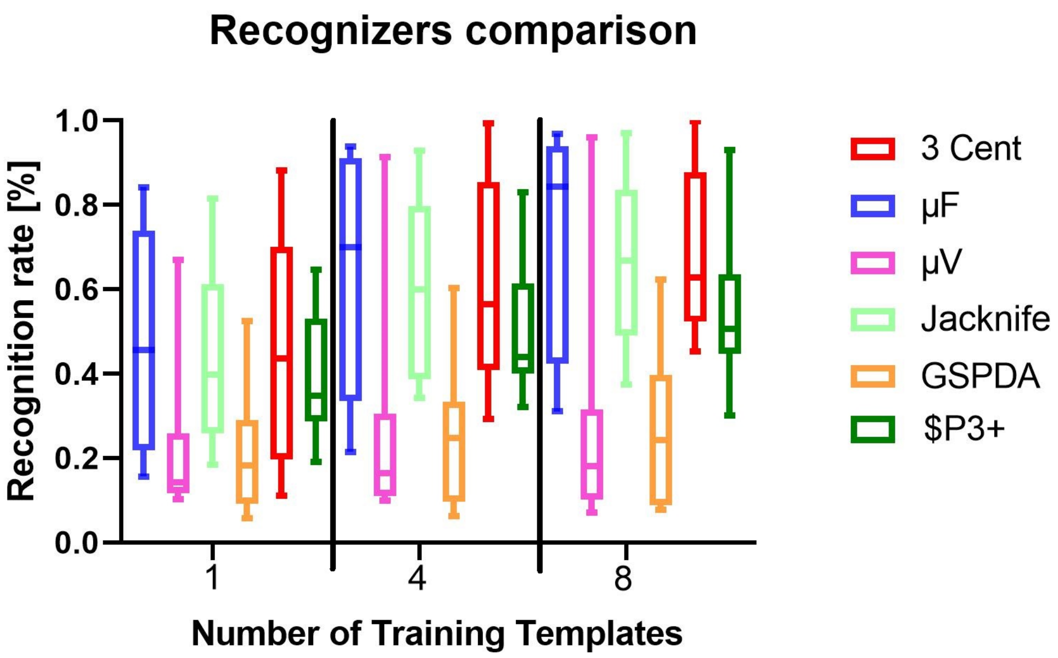


Figure 6.2: Recognizers comparison for 2D gestures with two fingers.

1	2	3	4	5	6
μF	3Cent	Jacknife	μV	GSPDA	$P3+

Table 6.4: Classification based on Kruskall-Wallis test results A.3.

2D Gestures with 3 fingers

On figure 6.3, GSPDa, μV and $P3+ didn't acquire an enough high recognition rate to be interesting. Here, we can also see that 3 Cent recognizer has an high median but also has a greater box than the others. Jacknife and μF has an high recognition rate. The one which is clearly the best for this kind of gestures, is μF.

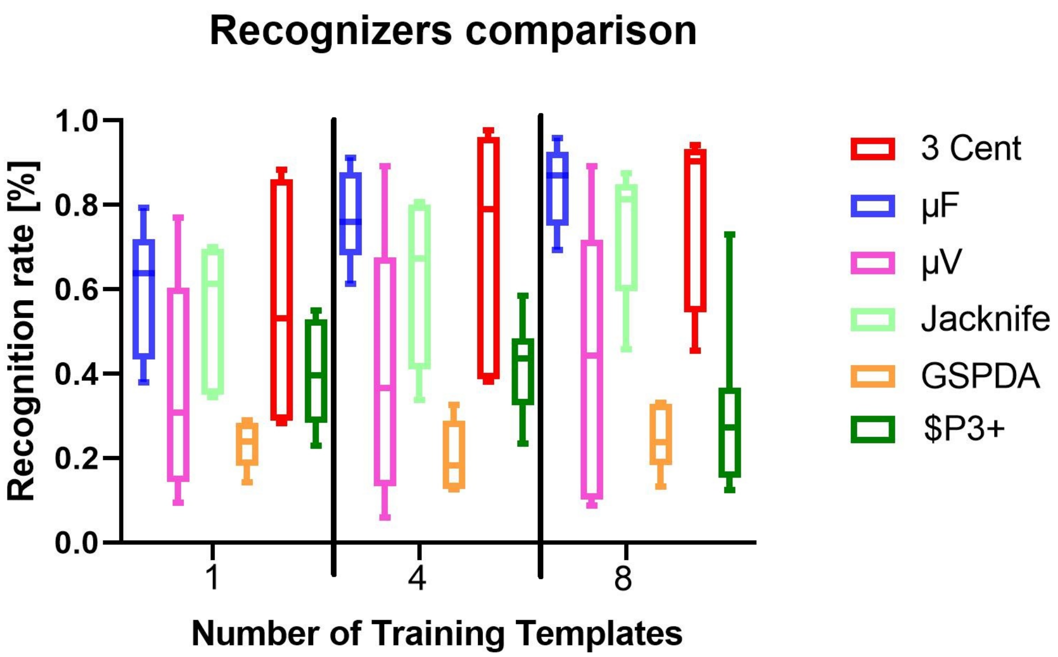


Figure 6.3: Recognizers comparison for 2D gestures with three fingers.

1	2	3	4	5	6
μF	3Cent	Jacknife	μV	GSPDA	$P3+

Table 6.5: Classification based on Kruskall-Wallis test results A.4.

3D Gestures

On figure6.4, GSPDa, μV and $P3+ are not relevant for 3D gestures. 3 Cent recognizer has bad result when it is trained with few training templates. But, as the number of training templates increase, the recognition rate slightly converges to 100%. However, Jacknife and μF are converging much faster. μF is the one to choose, if you are using one or two training template. However, if you are using three or more training templates, jacknife is the best one.

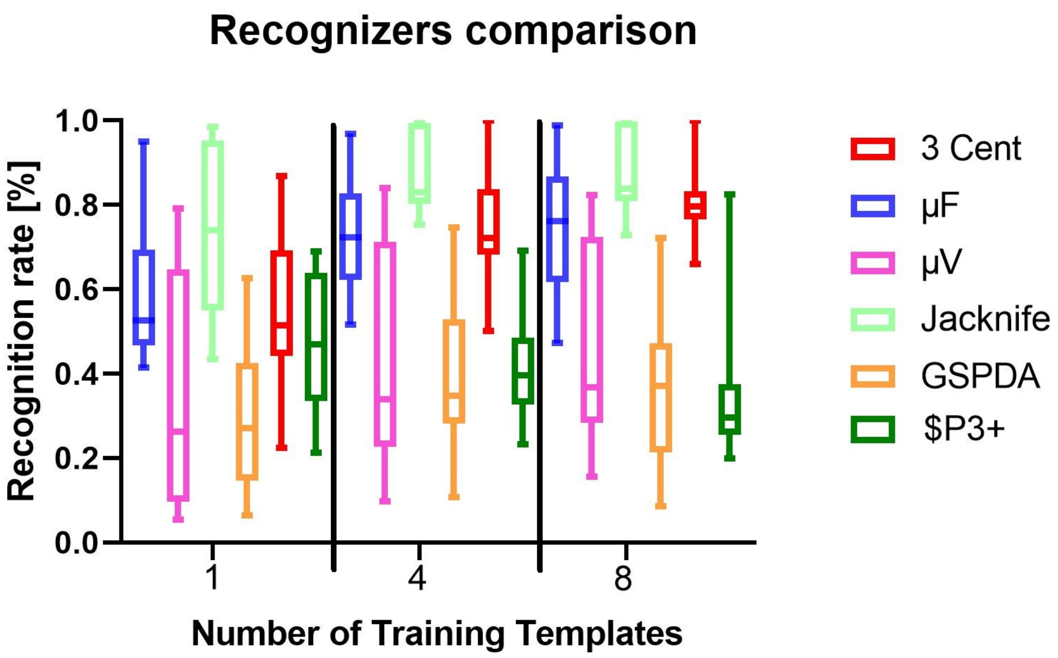


Figure 6.4: Recognizers comparison for 3D gestures.

1	2	3	4	5	6
Jacknife	3Cent	μF	μV	GSPDA	$P3+

Table 6.6: Classification based on Kruskall-Wallis test results A.5.

General Comparison Conclusion

In this section, we first proved that our *Air+Touch* gesture set definition defined in our previous work [17] is relevant and provide an high recognition rate with some recognizers. On this work, we mainly focused on the recognizers. Through our analysis, we can easily determine that **GSPDa, $P3+, μV** are absolutely not adequate to the gestures that we defined. **3 Cent** recognizer obtains often a good recognition rate. Unfortunately, you can only use one type of gesture with this recognizer. For instance, if you use it with 2D gestures with 1 finger and 3D gestures, it will completely mess up and will provide a low recognition rate. However, **Jacknife** and **μF** are satisfying our requirements as our previous benchmark analysis [17]. In appendix A, you can find the other figures as the recognition rate for each gesture for each recognizer and the recognition rate of each recognizer per number of training templates used.

6.3.4 Optimal Combination of Recognizers

Through this analysis, we somehow proved that it could be relevant to use multiple recognizers where each one is responsible for one type of gesture. Based on previous results, we can imagine the following optimal combination of recognizers.

Gesture Type	Recognizer
2DTouch1	Jacknife
2DTouch2	μF
2DTouch3	μF
3D	Jacknife

Table 6.7: Combination of recognizers according to the gesture type

As each recognizer is responsible for a smaller group of gestures, this increase probably the overall recognition rate of our gesture definition **despite the execution time**.

The execution time is one of the major issue related to this combination. As you can see on figure 6.5, there is only a small difference between each recognizer for each gesture. It does not seem relevant to "double" the execution time for such a small improvement. After the analysis of the results, this mixture of recognizers does not perform better than Jacknife. So Jacknife is once again the best recognizer.

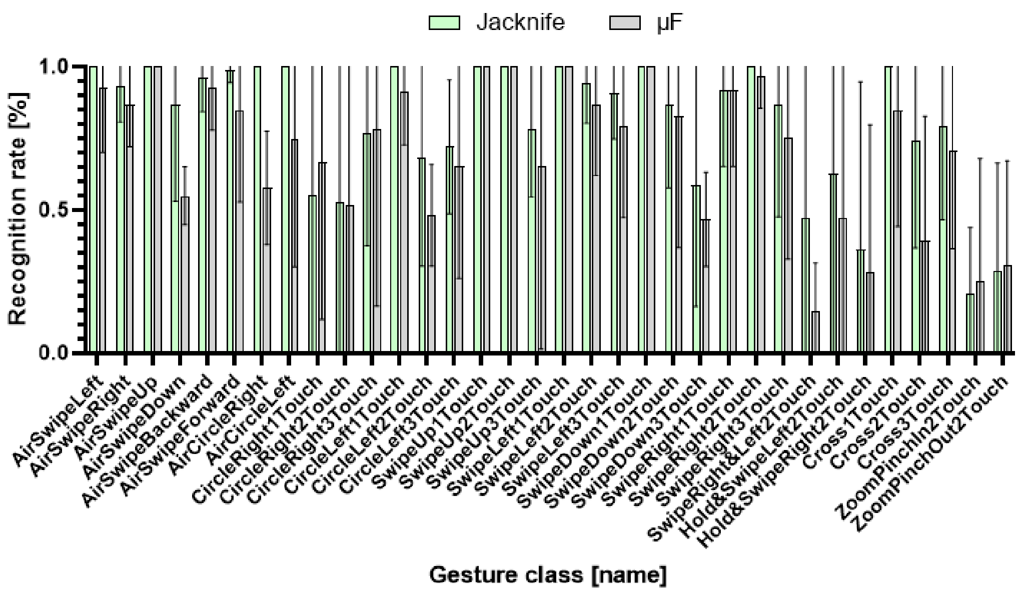


Figure 6.5: Recognition rate of both recognizers for each gesture.

6.4 Conclusion

In this chapter, we firstly recalled how our first analysis was carried out and what the results were. Then, we explained the goal of this new benchmarks. We analyzed the different results. Based on them, we tried to propose a combination to increases the recognition rate of our gesture set definition. Unfortunately, it was not relevant as it also increases the overall execution time. As our previous analysis, Jacknife is overall the best tested recognizers.

Chapter 7

Conclusion and Future Work

7.1 Final Conclusion

In this chapter, we are going to mention all the different contributions which were discussed in each chapter from this work.

In chapter 1, we introduced our context problem and all the different assumptions, research methods and questions about our subject.

In chapter 2, we first proved that the gesture recognition market is interesting by analyzing its forecast and the hype cycle of gesture control devices through the years to prove that there is a real interest for this work. Then, we gathered all the resources which contain information about application that are able to handle both 2D gestures and 3D gestures in the same environment. From this analysis, we defined a relevant cluster of the different founded types of application for our prototype.

In chapter 3, we performed a survey to determine which kind of application is the most relevant. The result of the survey is that the presentation application was preferred by respondents.

In chapter 4, we defined our prototype goals. We also determined which gestures were relevant to which action. And then, we presented the overall implementation structure of our prototype.

In chapter 5, we tested our prototype with ten participants two times with nearly one month of interval. We also explained our methodology for the experiment. Then, we analyzed both PSSUQ answers and feelings from each participant . We also performed a memorability test on each gesture for each participant. The multiple analysis proved that our prototype is relevant to our initials goals. The participants feelings assume that Air+Touch devices could be very interesting in the future. So, this has further enhanced the relevance and interest of our work on Air+Touch gestures. However, the participants feelings have emerged some negative points as we have to embellish the interface and maybe create a kind of tutorial in a future work.

In chapter 6, we performed a new benchmarks of recognizers. The goal was to complete our previous analysis by focusing on the different recognizers. We analyzed each recognizer according to each type of gesture. We also tried to propose an optimal mix to increase the overall recognition rate of our gesture set. A major issue related to the mix is that it really increases the execution time. According to this issue, we didn't assume this mix anymore. Jacknife is again the best tested recognizer. However, according to the parameters, μF could be a better solutions.

7.2 Future Work

During the whole work, we travelled through a lot of different subjects. Some have left us with prospects for future work or improvements. we are going to resume all of them:

- **Add functionality for our prototype:** As mentioned in chapter 5, it could be very interesting to continue to add new functionality and to expand the application.
- **Increase our gesture set definition:** Increase the number of gesture related to the 3DTouchpad by adding some 3D and 2D new gestures.
- **Provide a deeper prototype result:** Perform a prototype test with much more participants and also perform a gesture elicitation study [36] to determine which are the relevant gestures related to which action.
- **Add new recognizers to QuantumLeap [28]:** Try to find new recognizers that fit well with our gesture set definition and if yes, process some benchmarks to determine which one is the best.
- **Improve the prototype:** Create a tutorial, embellish the interface and put permanently information about the gesture that show all the gestures and their related action.
- **A work focusing on combinations:** Air+Touch offers an incredible potential for combinations as Air+Touch, Touch+Air+Touch, Air+Touch+Air and so on. As Marco Barsotti et al. [2], it may be interesting to propose a similar work. For instance, this application could allow the selection of a light with a 2D gesture and the reduction of the light with a 3D gesture. These actions could be relevant and adequate with the use of our device.

List of Figures

1.1 2D gestures and 3D gestures 5
1.2 Chapters overview 8

2.1 Hype Cycle of gesture control Devices through the years. 12
2.2 Hype Cycle for Emerging Technologies, 2020 [1] 13

3.1 Pyramid of age of the participants. 21
3.2 Proportion between men and women who answered the survey. . . . 21
3.3 Distribution of the profession of our participants 22
3.4 Distribution of the field of work per respondents 23
3.5 Distribution of actual studies or last diploma per respondents . . . 23
3.6 Distribution of actual studies or last diploma per respondents . . . 24
3.7 Knowledge and Frequency of use about the 3DTouchpad. 25
3.8 Workcloud [34] from the spontaneous ideas from respondents. . . . 26
3.9 Score of relevance per respondents and per applications. 27
3.10 Dashboard(1/2) 29
3.11 Dashboard(2/2) 29

4.1 The process from the device to our prototype. 35
4.2 QuantumLeap Framework Architecture. [28] 35
4.3 Picture of a participant using the prototype with the 3DTouchpad . 37
4.4 Prototype Interface. 38
4.5 Gesture Description 39
4.6 The zoom effect. 40

5.1 Average response from the first PSSUQ 44
5.2 Distribution of scores per question and IBM PSSUQ Measures . . . 45
5.3 Average response from the second PSSUQ 46
5.4 Distribution of scores per question and IBM PSSUQ Measures . . . 46
5.5 Average comparison from both PSSUQ 48
5.6 Distribution of scores per question 48
5.7 Gestures Memorability test 49

6.1 Recognizers comparison for 2D gestures with one finger. 54
6.2 Recognizers comparison for 2D gestures with two fingers. 55
6.3 Recognizers comparison for 2D gestures with three fingers. 56
6.4 Recognizers comparison for 3D gestures. 57
6.5 Recognition rate of both recognizers for each gesture. 58

A.1 Table representing the memorability for each participant to each gesture 70
A.2 Kruskall-Wallis with post-how Tukey for recognizers with 2D gestures with one finger. 71
A.3 Kruskall-Wallis with post-how Tukey for recognizers with 2D gestures with two fingers. 71
A.4 Kruskall-Wallis with post-how Tukey for recognizers with 2D gestures with three fingers. 72
A.5 Kruskall-Wallis with post-how Tukey for recognizers with 3D gestures. 72
A.6 Recognition Rate for $P3+ recognizer for gesture with 1 finger . . . 73
A.7 Recognition Rate for 3 Cent recognizer for gesture with 1 finger . . 73
A.8 Recognition Rate for GSPDA recognizer for gesture with 1 finger . 74
A.9 Recognition Rate for μF recognizer for gesture with 1 finger 74
A.10 Recognition Rate for μV recognizer for gesture with 1 finger 75
A.11 Recognition Rate for Jacknife recognizer for gesture with 1 finger . 75
A.12 Recognition Rate for $P3+ recognizer for gesture with 2 fingers . . 76
A.13 Recognition Rate for 3 Cent recognizer for gesture with 2 fingers . . 76
A.14 Recognition Rate for GSPDA recognizer for gesture with 2 fingers . 77
A.15 Recognition Rate for μF recognizer for gesture with 2 fingers 77
A.16 Recognition Rate for μV recognizer for gesture with 2 fingers 78
A.17 Recognition Rate for Jacknife recognizer for gesture with 2 fingers . 78
A.18 Recognition Rate for $P3+ recognizer for gesture with 3 fingers . . 79
A.19 Recognition Rate for 3 Cent recognizer for gesture with 3 fingers . . 79
A.20 Recognition Rate for GSPDA recognizer for gesture with 3 fingers . 80
A.21 Recognition Rate for μF recognizer for gesture with 3 fingers 80
A.22 Recognition Rate for μV recognizer for gesture with 3 fingers 81
A.23 Recognition Rate for Jacknife recognizer for gesture with 3 fingers . 81
A.24 Recognition Rate for $P3+ recognizer for 3D gestures 82
A.25 Recognition Rate for 3 Cent recognizer for 3D gestures 82
A.26 Recognition Rate for GSPDA recognizer for 3D gestures 83
A.27 Recognition Rate for μF recognizer for 3D gestures 83
A.28 Recognition Rate for μV recognizer for 3D gestures 84
A.29 Recognition Rate for Jacknife recognizer for 3D gestures 84
A.30 Recognition Rate for 3 Cent recognizer per gesture with 1 finger . . 85
A.31 Recognition Rate for GSPDA recognizer per gesture with 1 finger . 85

A.32 Recognition Rate for μF recognizer per gesture with 1 finger 86
A.33 Recognition Rate for μV recognizer per gesture with 1 finger 86
A.34 Recognition Rate for Jacknife recognizer per gesture with 1 finger . 87
A.35 Recognition Rate for 3 Cent recognizer per gesture with 2 fingers . 87
A.36 Recognition Rate for GSPDA recognizer per gesture with 2 fingers . 88
A.37 Recognition Rate for μF recognizer per gesture with 2 fingers 88
A.38 Recognition Rate for μV recognizer per gesture with 2 fingers . . . 89
A.39 Recognition Rate for Jacknife recognizer per gesture with 2 fingers . 89
A.40 Recognition Rate for 3 Cent recognizer per gestures with 3 fingers . 90
A.41 Recognition Rate for GSPDA recognizer per gestures with 3 fingers 90
A.42 Recognition Rate for μF recognizer per gestures with 3 fingers . . . 91
A.43 Recognition Rate for μV recognizer per gestures with 3 fingers . . . 91
A.44 Recognition Rate for Jacknife recognizer per gestures with 3 fingers 92
A.45 Recognition Rate for 3 Cent recognizer per 3D gestures 92
A.46 Recognition Rate for GSPDA recognizer per 3D gestures 93
A.47 Recognition Rate for μF recognizer per 3D gestures 93
A.48 Recognition Rate for μV recognizer per 3D gestures 94
A.49 Recognition Rate for Jacknife recognizer per 3D gestures 94

List of Tables

2.1 Compound Annual growth rate per report 11
2.2 Comparative Table. 16

3.1 Respondents' proposal and the number of times it was proposed. . . 26
3.2 Number of answers from the respondents per relevance score per application 27

4.1 Gestures names, related actions and drawing of each gesture. 34

5.1 Personal information of our participants 42
5.2 The 16 Questions of the PSSUQ. 43

6.1 Table containing the different parameters and their different variations from our previous testing. 52
6.2 Parameters of our benchmarks. 53
6.3 Classification based on Kruskall-Wallis test results A.2. 54
6.4 Classification based on Kruskall-Wallis test results A.3. 55
6.5 Classification based on Kruskall-Wallis test results A.4. 56
6.6 Classification based on Kruskall-Wallis test results A.5. 57
6.7 Combination of recognizers according to the gesture type 58

Bibliography

[1] Gartner identifies five emerging trends that will drive technology innovation for the next decade.

[2] Marco Barsotti, Fabio Paternò, and Francesca Pulina. A web framework for cross-device gestures between personal devices and public displays. In *Proceedings of the 16th International Conference on Mobile and Ubiquitous Multimedia*, pages 69–78, 2017.

[3] Andrew Bragdon, Rob DeLine, Ken Hinckley, and Meredith Ringel Morris. Code space: touch+ air gesture hybrid interactions for supporting developer meetings. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, pages 212–221, 2011.

[4] Philip Cohen, David McGee, Sharon Oviatt, Lizhong Wu, Josh Clow, Rob King, Simon Julier, and Lawrence Rosenblum. Multimodal interaction for 2d and 3d environments [virtual reality]. *IEEE Computer Graphics and Applications*, 19(4):10–13, 1999.

[5] DataBridge. Global gesture recognition and touchless sensing market – industry trends and forecast to 2028.

[6] Georg Hackenberg, Rod McCall, and Wolfgang Broll. Lightweight palm and finger tracking for real-time 3d gesture control. In *2011 IEEE Virtual Reality Conference*, pages 19–26. IEEE, 2011.

[7] imarc. Gesture recognition market: Global industry trends, share, size, growth, opportunity and forecast 2021-2026.

[8] Mordor Intelligence. Gesture recognition market - growth, trends, covid-19 impact, and forecasts (2022 - 2027).

[9] Minseok Kim and Jae Yeol Lee. Touch and hand gesture-based interactions for directly manipulating 3d virtual objects in mobile augmented reality. *Multimedia Tools and Applications*, 75(23):16529–16550, 2016.

[10] Georgios Kouroupetroglou, Alexandros Pino, Athanasios Balmpakakis, Dimitrios Chalastanis, Vasileios Golematis, Nikolaos Ioannou, and Ioannis Koutsoumpas. Using wiimote for 2d and 3d pointing tasks: gesture performance evaluation. In *International Gesture Workshop*, pages 13–23. Springer, 2011.

[11] Joseph J LaViola and Robert C Zeleznik. A practical approach for writer-dependent symbol recognition using a writer-independent symbol recognizer. *IEEE Transactions on pattern analysis and machine intelligence*, 29(11):1917–1926, 2007.

[12] Nathan Magrofuoco, Paolo Roselli, and Jean Vanderdonckt. Two-dimensional stroke gesture recognition: A survey. *ACM Computing Surveys (CSUR)*, 54(7):1–36, 2021.

[13] Shahzad Malik and Joe Laszlo. Visual touchpad: a two-handed gestural input device. In *Proceedings of the 6th international conference on Multimodal interfaces*, pages 289–296, 2004.

[14] Nicolai Marquardt, Ricardo Jota, Saul Greenberg, and Joaquim A Jorge. The continuous interaction space: interaction techniques unifying touch and gesture on and above a digital surface. In *IFIP Conference on Human-Computer Interaction*, pages 461–476. Springer, 2011.

[15] Microchip. 3DTouchPad Sell Sheet. `https://ww1.microchip.com/downloads/en/DeviceDoc/00001835A.pdf`.

[16] Jörg Müller, Gilles Bailly, Thor Bossuyt, and Niklas Hillgren. Mirrortouch: combining touch and mid-air gestures for public displays. In *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services*, pages 319–328, 2014.

[17] Romain Neuville. *Air+Touch gesture recognition : algorithms, software, and experiment*. PhD thesis, UCL - Ecole polytechnique de Louvain, 2021.

[18] Benjamin Nuernberger, Kuo-Chin Lien, Tobias Höllerer, and Matthew Turk. Interpreting 2d gesture annotations in 3d augmented reality. In *2016 IEEE symposium on 3D user interfaces (3DUI)*, pages 149–158. IEEE, 2016.

[19] Michael Ortega and Laurence Nigay. Airmouse: Finger gesture for 2d and 3d interaction. In *IFIP Conference on Human-Computer Interaction*, pages 214–227. Springer, 2009.

[20] Mehdi Ousmer, Arthur Sluÿters, Nathan Magrofuoco, Paolo Roselli, and Jean Vanderdonckt. Recognizing 3d trajectories as 2d multi-stroke gestures. *Proceedings of the ACM on Human-Computer Interaction*, 4(ISS):1–21, 2020.

[21] Alexandros Pino, Evangelos Tzemis, Nikolaos Ioannou, and Georgios Kouroupetroglou. Using kinect for 2d and 3d pointing tasks: performance evaluation. In *International Conference on Human-Computer Interaction*, pages 358–367. Springer, 2013.

[22] Carole Plasson, Dominique Cunin, Yann Laurillau, and Laurence Nigay. 3d tabletop ar: a comparison of mid-air, touch and touch+ mid-air interaction. In *Proceedings of the International Conference on Advanced Visual Interfaces*, pages 1–5, 2020.

[23] Leigh Ellen Potter, Jake Araullo, and Lewis Carter. The leap motion controller: a view on sign language. In *Proceedings of the 25th Australian computer-human interaction conference: augmentation, application, innovation, collaboration*, pages 175–178, 2013.

[24] Grand View Research. Gesture recognition market size, share trends analysis report by technology (touch-based, touchless), by industry (automotive, consumer electronics, healthcare), by region, and segment forecasts, 2022 - 2030.

[25] Alexander Schick, Florian van de Camp, Joris Ijsselmuiden, and Rainer Stiefelhagen. Extending touch: towards interaction with large-scale surfaces. In *Proceedings of the ACM international conference on interactive tabletops and surfaces*, pages 117–124, 2009.

[26] Thomas Schlömer, Benjamin Poppinga, Niels Henze, and Susanne Boll. Gesture recognition with a wii controller. In *Proceedings of the 2nd international conference on Tangible and embedded interaction*, pages 11–14, 2008.

[27] Arthur Sluÿters, Sébastien Lambot, and Jean Vanderdonckt. Hand gesture recognition for an off-the-shelf radar by electromagnetic modeling and inversion. In *27th International Conference on Intelligent User Interfaces*, pages 506–522, 2022.

[28] Arthur Sluÿters, Mehdi Ousmer, Paolo Roselli, and Jean Vanderdonckt. Quantumleap, a framework for engineering gestural user interfaces based on the leap motion controller. *Proc. ACM Hum. Comput. Interact.*, 6(EICS):1–47, 2022.

[29] `https://www.gartner.com/en/documents/2822817/hype-cycle-for-human-computer-interaction-2014`. Gartner hype cycle - human machine interface 2014.

[30] https://www.gartner.com/en/documents/3368017/hype-cycle-for-human-machine-interface-2016. Gartner hype cycle - human machine interface 2016.

[31] https://www.gartner.com/en/documents/3764163/hype-cycle-for-human-machine-interface. Gartner hype cycle - human machine interface 2017.

[32] https://www.gartner.com/en/documents/3882879/hype-cycle-for-human-machine-interface-2018. Gartner hype cycle - human machine interface 2018.

[33] https://www.gartner.com/en/information-technology/research/hype-cycle. Gartner hype cycle.

[34] https://www.wordclouds.com/. Workcloud online generator.

[35] powerbi.microsoft.com/fr-fr/. Microsoft power bi.

[36] Santiago Villarreal-Narvaez, Jean Vanderdonckt, Radu-Daniel Vatavu, and Jacob O Wobbrock. A systematic review of gesture elicitation studies: What can we learn from 216 studies? In *Proceedings of the 2020 ACM Designing Interactive Systems Conference*, pages 855–872, 2020.

[37] Yusman Azimi Yusoff, Ahmad Hoirul Basori, and Farhan Mohamed. Interactive hand and arm gesture control for 2d medical image and 3d volumetric medical visualization. *Procedia-Social and Behavioral Sciences*, 97:723–729, 2013.

[38] Shumin Zhai, Per Kristensson, Caroline Appert, Tue Andersen, and Xiang Cao. Foundational issues in touch-surface stroke gesture design: An integrative review. foundations of trends in human-computer interaction 5, 2 (feb 2012), 97–205, 2012.

[39] Zhengyou Zhang. Microsoft kinect sensor and its effect. *IEEE multimedia*, 19(2):4–10, 2012.

Appendix A

Appendix

A.1 Prototype Test

Gesture	Participant 1	Participant 2	Participant 3	Participant 4	Participant 5	Participant 6	Participant 7	Participant 8	Participant 9	Participant 10	Moyene
Advances Three Slides	1	1	0	1	1	1	1	0	1	1	0,8
Advances Two Slides	1	1	0	1	0	1	1	1	1	1	0,8
Back Three Slides	1	1	0	1	1	1	0	1	1	1	0,8
Back Two Slides	1	1	0	1	1	1	1	1	1	1	0,9
End the presentation	1	1	0	1	1	1	0	0	0	0	0,5
Go to Gesture Description(2D)	0	0	0	0	0	1	1	0	0	0	0,2
Go to Gesture Description(3D)	1	0	0	0	0	0	1	0	1	0	0,3
Next slide(2D)	1	1	1	1	1	1	1	1	1	1	1,0
Next slide(3D)	1	1	1	1	1	1	1	0	1	1	0,9
Previous Slide(2D)	1	1	1	1	1	1	1	0	1	1	0,9
Previous Slide(3D)	1	1	1	1	1	1	1	0	1	1	0,9
Quit Gesture Description(2D)	0	0	0	0	0	0	1	0	0	0	0,1
Quit Gesture Description(3D)	0	0	0	0	0	0	1	0	1	0	0,2
Restart the presentation	0	0	0	1	1	1	0	0	1	1	0,5
UnZoom(2D)	1	1	1	1	0	1	1	1	1	1	0,9
UnZoom(3D)	0	0	0	1	1	0	1	0	0	1	0,4
Zoom(2D)	1	1	1	1	1	1	1	0	1	1	0,9
Zoom(3D)	0	0	0	1	1	0	0	0	1	1	0,4

Figure A.1: Table representing the memorability for each participant to each gesture

A.2 Kruskall-Wallis with post-how Tukey

Tukey's multiple comparisons test	Mean Diff.	95.00% CI of diff.	Below threshold?	Summary	Adjusted P Value
µF vs. µV	0.3845	0.03439 to 0.7345	Yes	*	0.0243
µF vs. Jacknife	-0.01176	-0.3618 to 0.3383	No	ns	>0.9999
µF vs. GSPDA	0.5281	0.1781 to 0.8782	Yes	***	0.0008
µF vs. 3 Cent	0.02972	-0.3204 to 0.3798	No	ns	0.9998
µF vs. $P3+	0.4294	0.07935 to 0.7795	Yes	**	0.0089
µV vs. Jacknife	-0.3962	-0.7463 to -0.04615	Yes	*	0.0188
µV vs. GSPDA	0.1437	-0.2064 to 0.4937	No	ns	0.8170
µV vs. 3 Cent	-0.3547	-0.7048 to -0.004668	Yes	*	0.0455
µV vs. $P3+	0.04496	-0.3051 to 0.3950	No	ns	0.9988
Jacknife vs. GSPDA	0.5399	0.1898 to 0.8900	Yes	***	0.0006
Jacknife vs. 3 Cent	0.04148	-0.3086 to 0.3916	No	ns	0.9992
Jacknife vs. $P3+	0.4412	0.09111 to 0.7913	Yes	**	0.0067
GSPDA vs. 3 Cent	-0.4984	-0.8485 to -0.1483	Yes	**	0.0017
GSPDA vs. $P3+	-0.09870	-0.4488 to 0.2514	No	ns	0.9561
3 Cent vs. $P3+	0.3997	0.04963 to 0.7498	Yes	*	0.0174

Figure A.2: Kruskall-Wallis with post-how Tukey for recognizers with 2D gestures with one finger.

Tukey's multiple comparisons test	Mean Diff.	95.00% CI of diff.	Below threshold?	Summary	Adjusted P Value
µF vs. µV	0.3999	0.1270 to 0.6729	Yes	***	0.0008
µF vs. Jacknife	0.04056	-0.2324 to 0.3135	No	ns	0.9979
µF vs. GSPDA	0.4267	0.1538 to 0.6997	Yes	***	0.0003
µF vs. 3 Cent	0.007194	-0.2658 to 0.2802	No	ns	>0.9999
µF vs. $P3+	0.1503	-0.1226 to 0.4233	No	ns	0.5908
µV vs. Jacknife	-0.3594	-0.6323 to -0.08640	Yes	**	0.0034
µV vs. GSPDA	0.02682	-0.2461 to 0.2998	No	ns	0.9997
µV vs. 3 Cent	-0.3927	-0.6657 to -0.1198	Yes	**	0.0010
µV vs. $P3+	-0.2496	-0.5225 to 0.02338	No	ns	0.0923
Jacknife vs. GSPDA	0.3862	0.1132 to 0.6591	Yes	**	0.0013
Jacknife vs. 3 Cent	-0.03336	-0.3063 to 0.2396	No	ns	0.9992
Jacknife vs. $P3+	0.1098	-0.1632 to 0.3827	No	ns	0.8446
GSPDA vs. 3 Cent	-0.4195	-0.6925 to -0.1466	Yes	***	0.0004
GSPDA vs. $P3+	-0.2764	-0.5494 to -0.003436	Yes	*	0.0455
3 Cent vs. $P3+	0.1431	-0.1298 to 0.4161	No	ns	0.6406

Figure A.3: Kruskall-Wallis with post-how Tukey for recognizers with 2D gestures with two fingers.

Tukey's multiple comparisons test	Mean Diff.	95.00% CI of diff.	Below threshold?	Summary	Adjusted P Value
µF vs. µV	0.4212	0.1132 to 0.7291	Yes	**	0.0027
µF vs. Jacknife	0.1038	-0.2042 to 0.4117	No	ns	0.9102
µF vs. GSPDA	0.5933	0.2854 to 0.9013	Yes	****	<0.0001
µF vs. 3 Cent	0.04528	-0.2627 to 0.3532	No	ns	0.9977
µF vs. $P3+	0.5248	0.2169 to 0.8328	Yes	***	0.0001
µV vs. Jacknife	-0.3174	-0.6253 to -0.009426	Yes	*	0.0402
µV vs. GSPDA	0.1722	-0.1358 to 0.4801	No	ns	0.5520
µV vs. 3 Cent	-0.3759	-0.6838 to -0.06793	Yes	**	0.0093
µV vs. $P3+	0.1037	-0.2043 to 0.4116	No	ns	0.9107
Jacknife vs. GSPDA	0.4895	0.1816 to 0.7975	Yes	***	0.0004
Jacknife vs. 3 Cent	-0.05850	-0.3664 to 0.2494	No	ns	0.9923
Jacknife vs. $P3+	0.4210	0.1131 to 0.7290	Yes	**	0.0028
GSPDA vs. 3 Cent	-0.5480	-0.8560 to -0.2401	Yes	****	<0.0001
GSPDA vs. $P3+	-0.06851	-0.3765 to 0.2394	No	ns	0.9842
3 Cent vs. $P3+	0.4795	0.1716 to 0.7875	Yes	***	0.0005

Figure A.4: Kruskall-Wallis with post-how Tukey for recognizers with 2D gestures with three fingers.

Tukey's multiple comparisons test	Mean Diff.	95.00% CI of diff.	Below threshold?	Summary	Adjusted P Value
µF vs. µV	0.3054	0.04249 to 0.5684	Yes	*	0.0145
µF vs. Jacknife	-0.1292	-0.3922 to 0.1337	No	ns	0.6863
µF vs. GSPDA	0.3825	0.1195 to 0.6454	Yes	**	0.0012
µF vs. 3 Cent	-0.05927	-0.3222 to 0.2037	No	ns	0.9840
µF vs. $P3+	0.3912	0.1282 to 0.6541	Yes	***	0.0009
µV vs. Jacknife	-0.4347	-0.6976 to -0.1717	Yes	***	0.0002
µV vs. GSPDA	0.07701	-0.1859 to 0.3400	No	ns	0.9506
µV vs. 3 Cent	-0.3647	-0.6276 to -0.1018	Yes	**	0.0021
µV vs. $P3+	0.08572	-0.1772 to 0.3487	No	ns	0.9239
Jacknife vs. GSPDA	0.5117	0.2487 to 0.7746	Yes	****	<0.0001
Jacknife vs. 3 Cent	0.06997	-0.1930 to 0.3329	No	ns	0.9670
Jacknife vs. $P3+	0.5204	0.2574 to 0.7833	Yes	****	<0.0001
GSPDA vs. 3 Cent	-0.4417	-0.7047 to -0.1788	Yes	***	0.0001
GSPDA vs. $P3+	0.008706	-0.2542 to 0.2716	No	ns	>0.9999
3 Cent vs. $P3+	0.4504	0.1875 to 0.7134	Yes	***	0.0001

Figure A.5: Kruskall-Wallis with post-how Tukey for recognizers with 3D gestures.

A.3 Recognition Rate of recognizers with gestures with One Finger

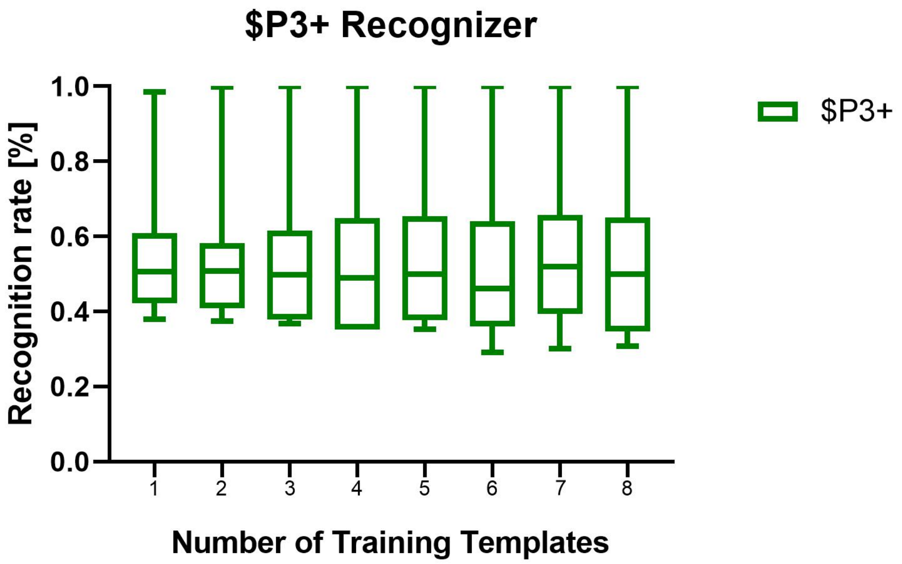


Figure A.6: Recognition Rate for $P3+ recognizer for gesture with 1 finger

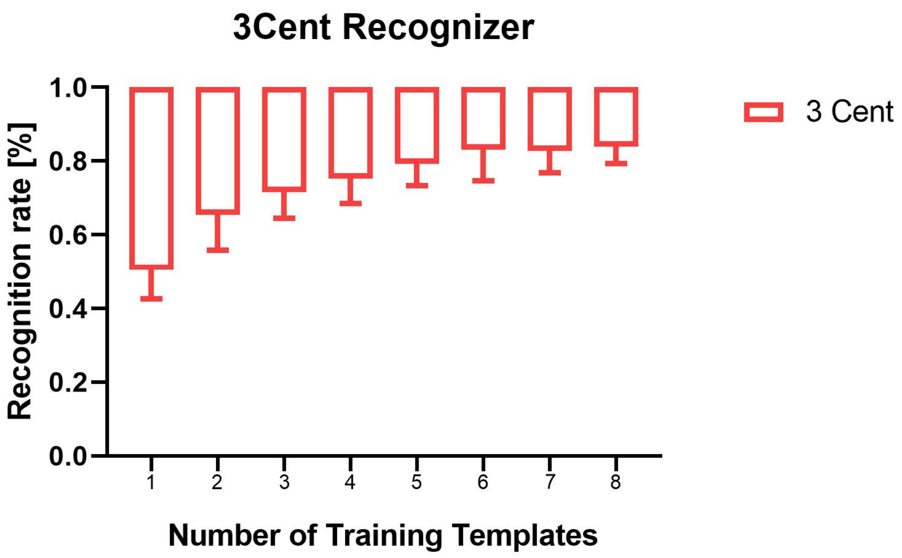


Figure A.7: Recognition Rate for 3 Cent recognizer for gesture with 1 finger

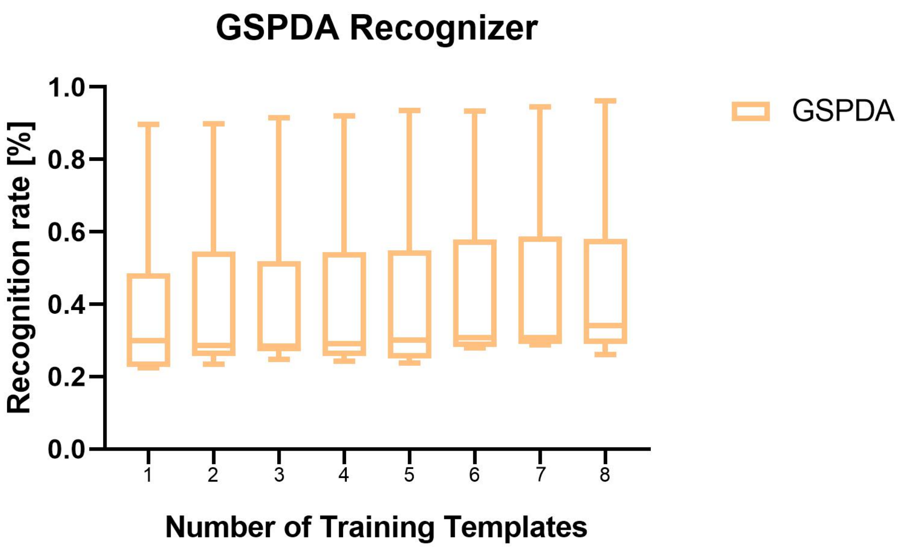


Figure A.8: Recognition Rate for GSPDA recognizer for gesture with 1 finger

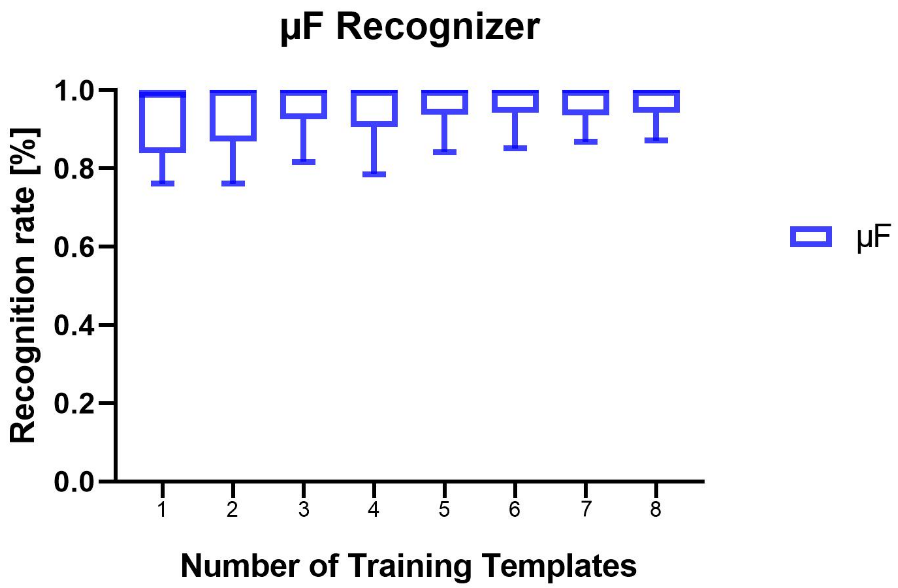


Figure A.9: Recognition Rate for μF recognizer for gesture with 1 finger

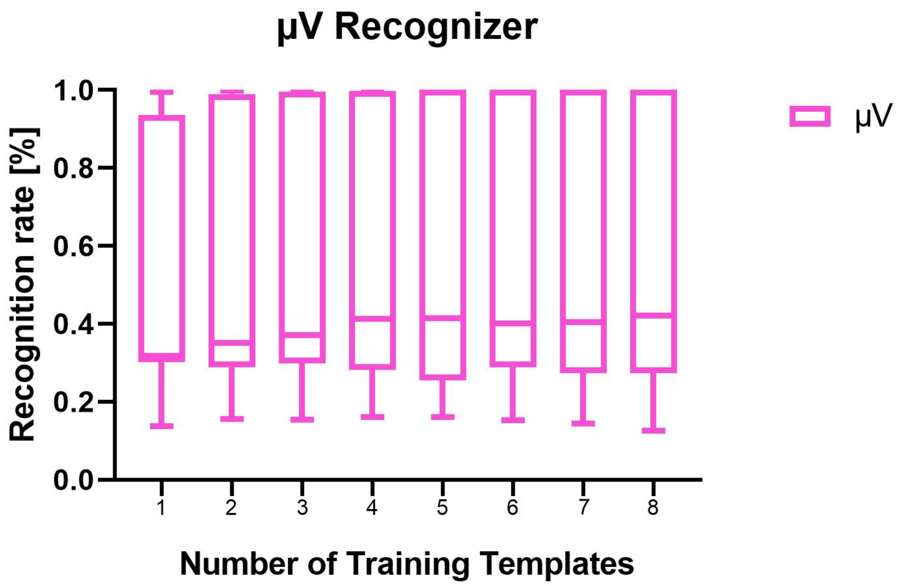


Figure A.10: Recognition Rate for μV recognizer for gesture with 1 finger

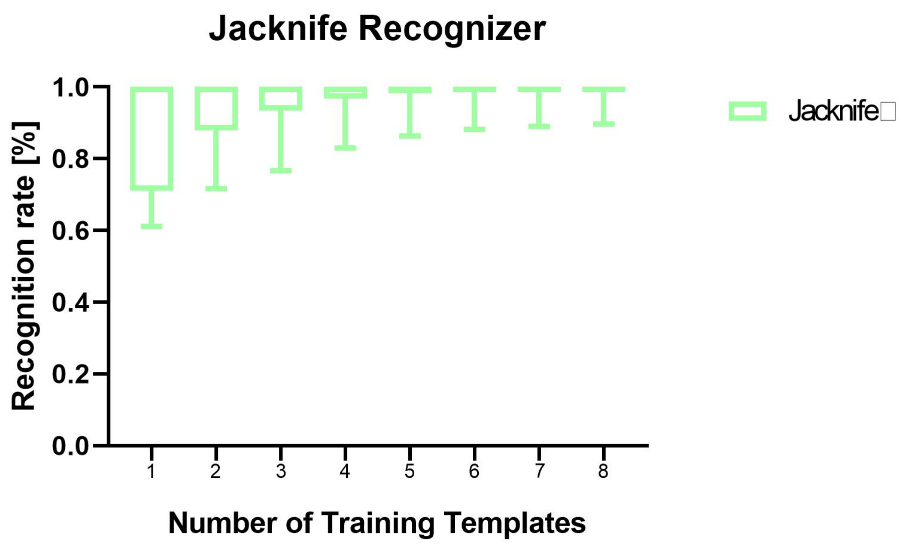


Figure A.11: Recognition Rate for Jacknife recognizer for gesture with 1 finger

A.4 Recognition Rate of recognizers with gestures with two Fingers

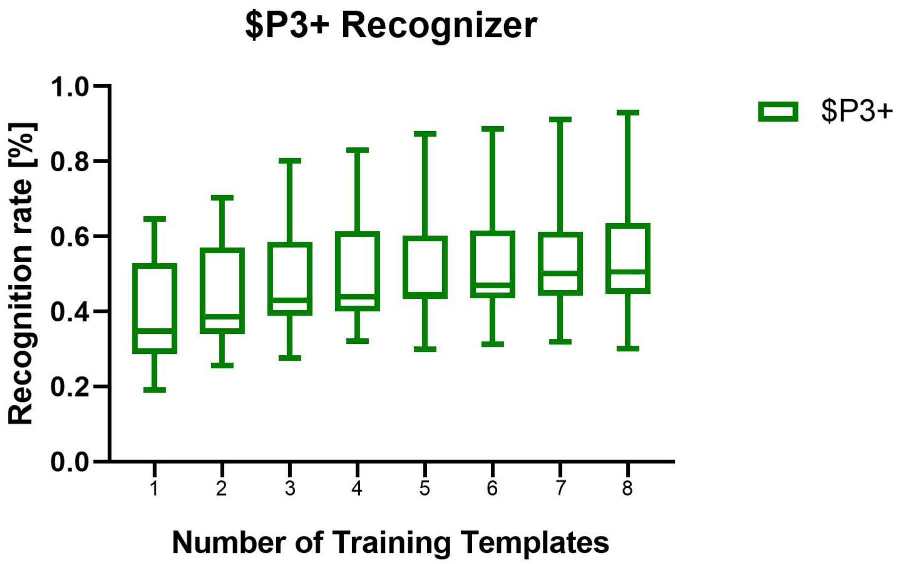


Figure A.12: Recognition Rate for $P3+ recognizer for gesture with 2 fingers

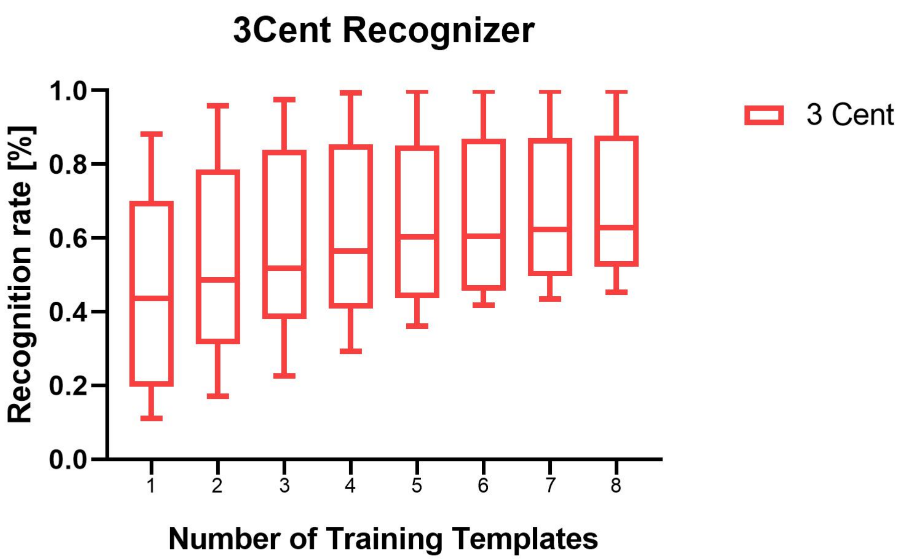


Figure A.13: Recognition Rate for 3 Cent recognizer for gesture with 2 fingers

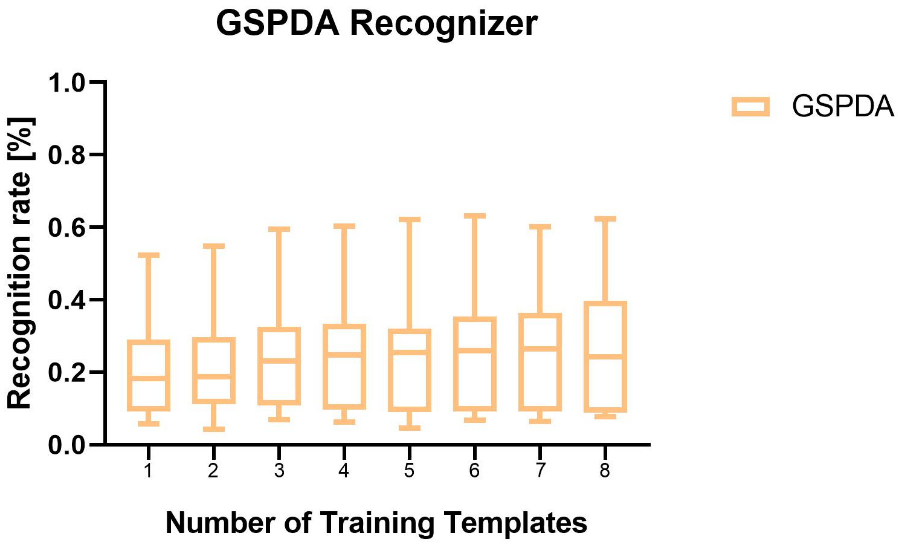


Figure A.14: Recognition Rate for GSPDA recognizer for gesture with 2 fingers

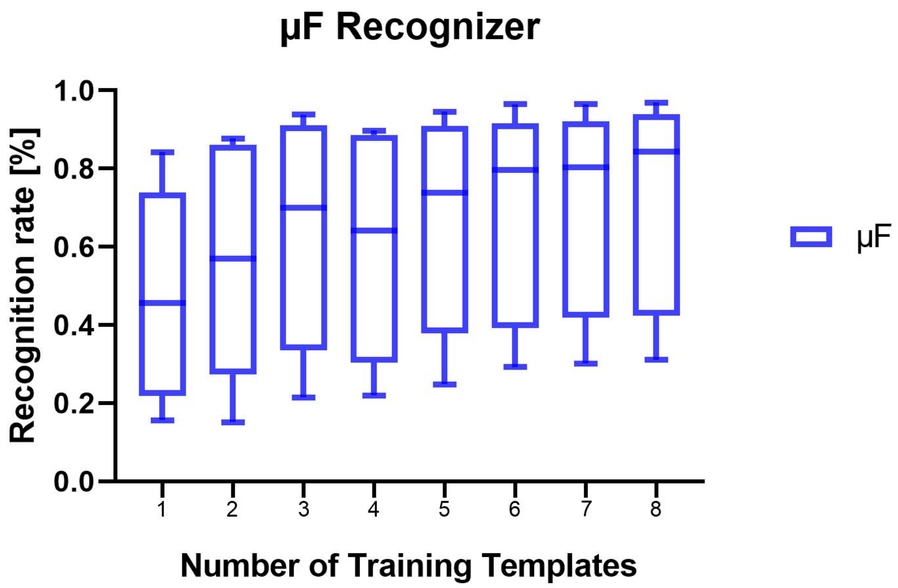


Figure A.15: Recognition Rate for μF recognizer for gesture with 2 fingers

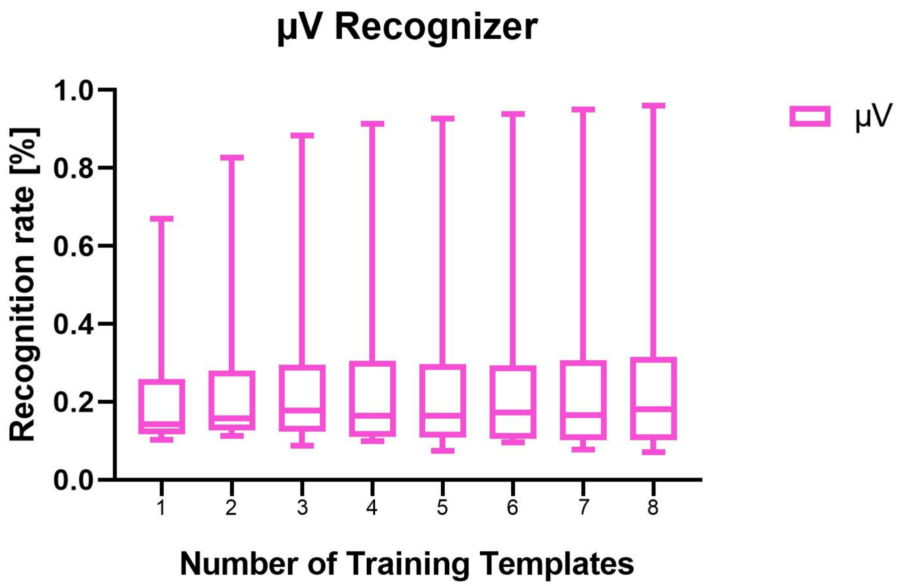


Figure A.16: Recognition Rate for μV recognizer for gesture with 2 fingers

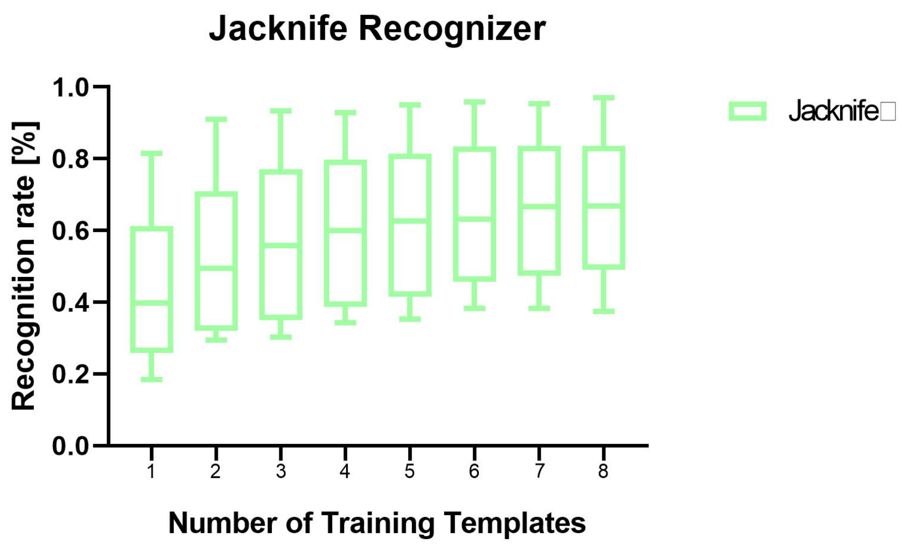


Figure A.17: Recognition Rate for Jacknife recognizer for gesture with 2 fingers

A.5 Recognition Rate of recognizers with gestures with three Fingers

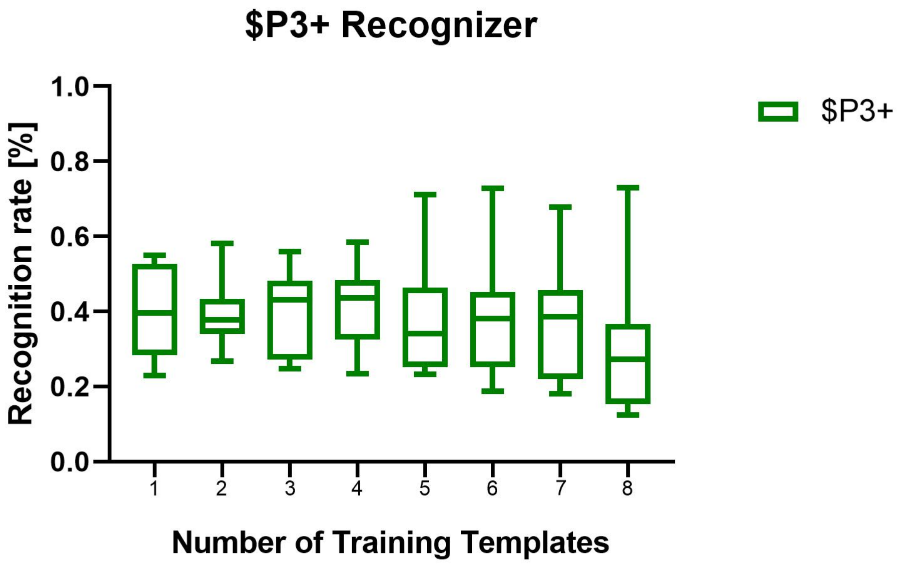


Figure A.18: Recognition Rate for $P3+ recognizer for gesture with 3 fingers

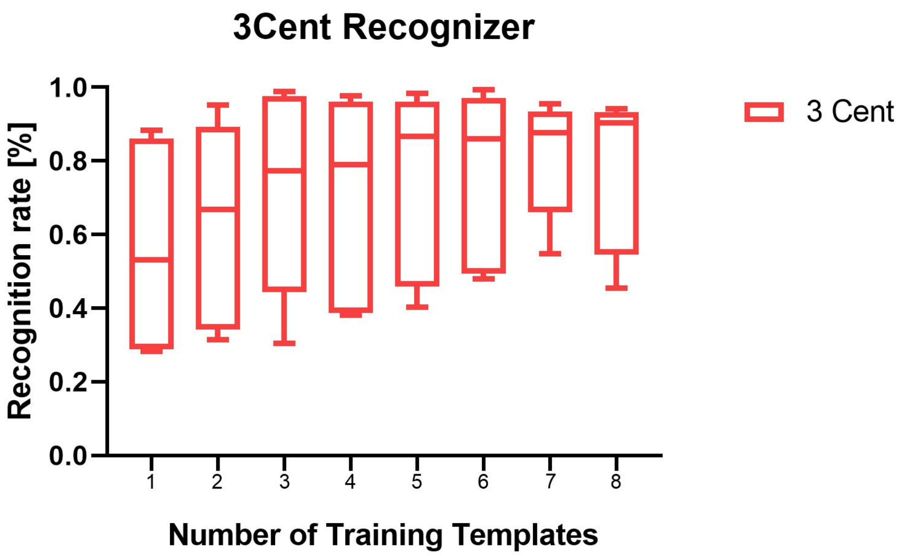


Figure A.19: Recognition Rate for 3 Cent recognizer for gesture with 3 fingers

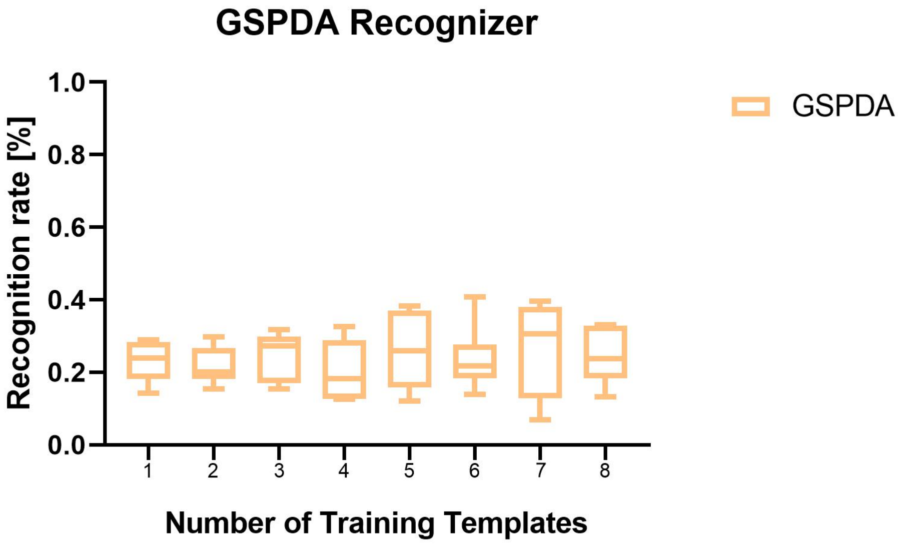


Figure A.20: Recognition Rate for GSPDA recognizer for gesture with 3 fingers

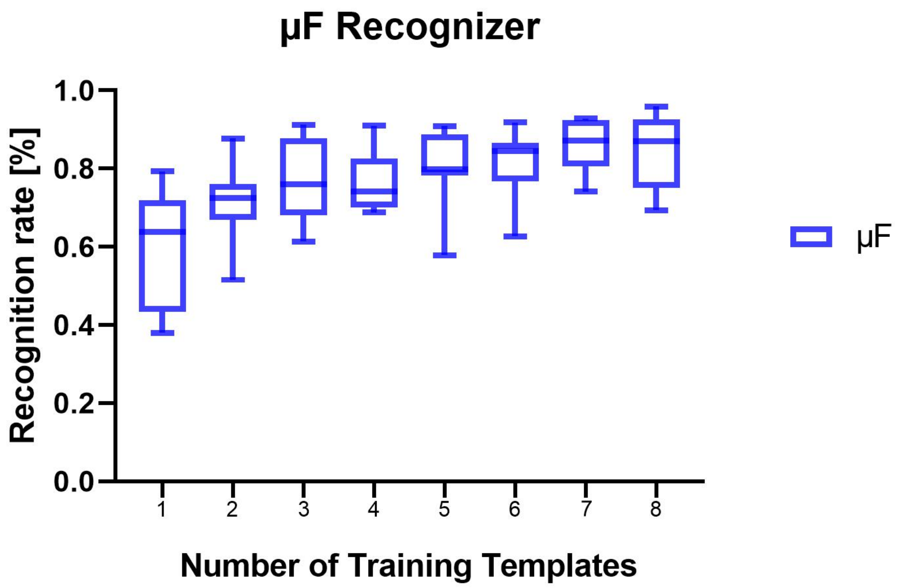


Figure A.21: Recognition Rate for μF recognizer for gesture with 3 fingers

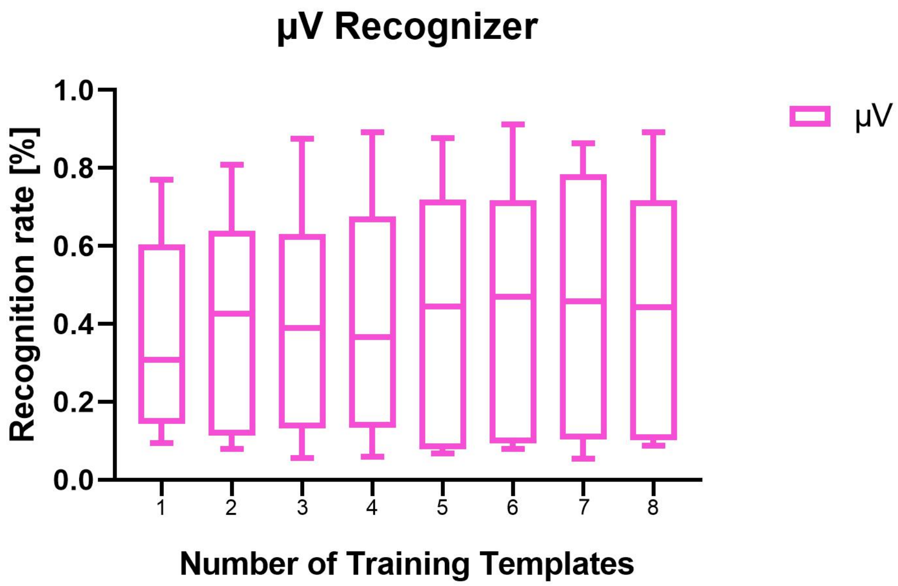


Figure A.22: Recognition Rate for μV recognizer for gesture with 3 fingers

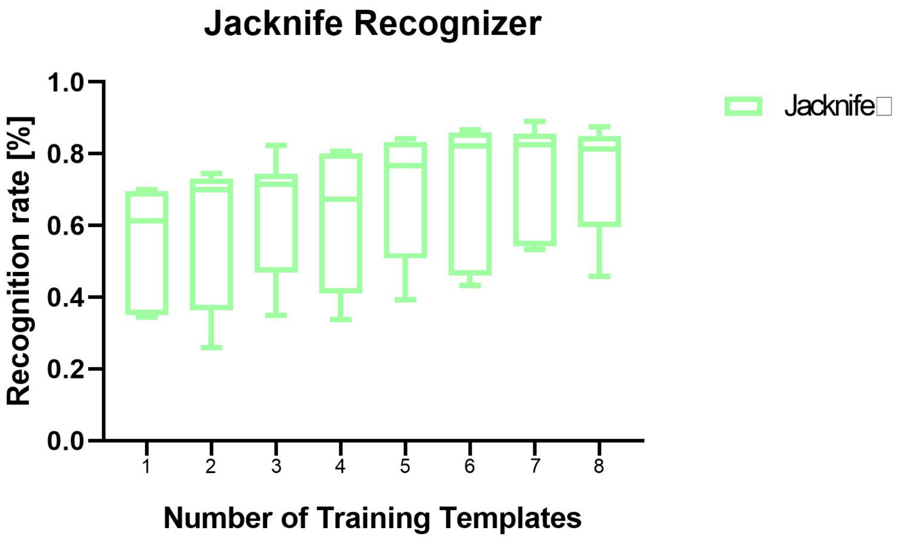


Figure A.23: Recognition Rate for Jacknife recognizer for gesture with 3 fingers

A.6 Recognition Rate of recognizers for 3D gestures

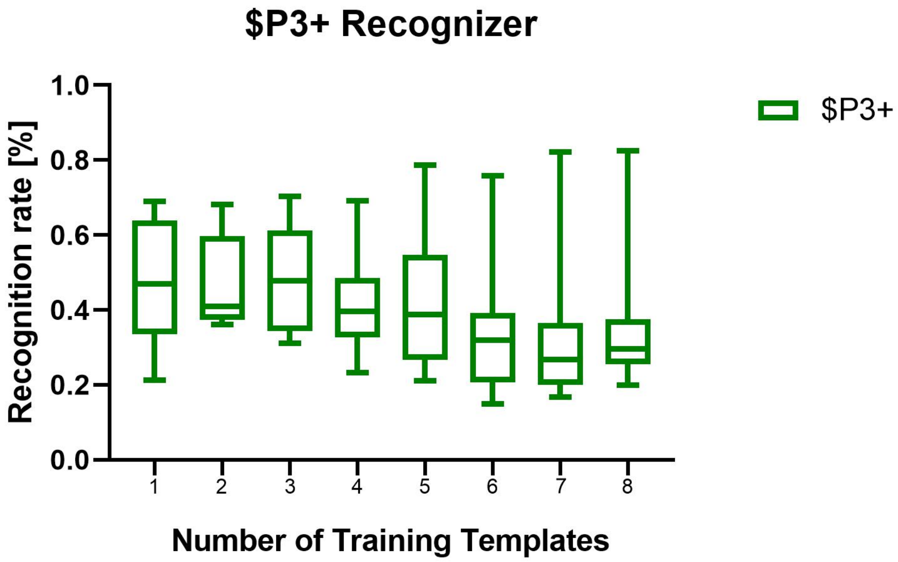


Figure A.24: Recognition Rate for $P3+ recognizer for 3D gestures

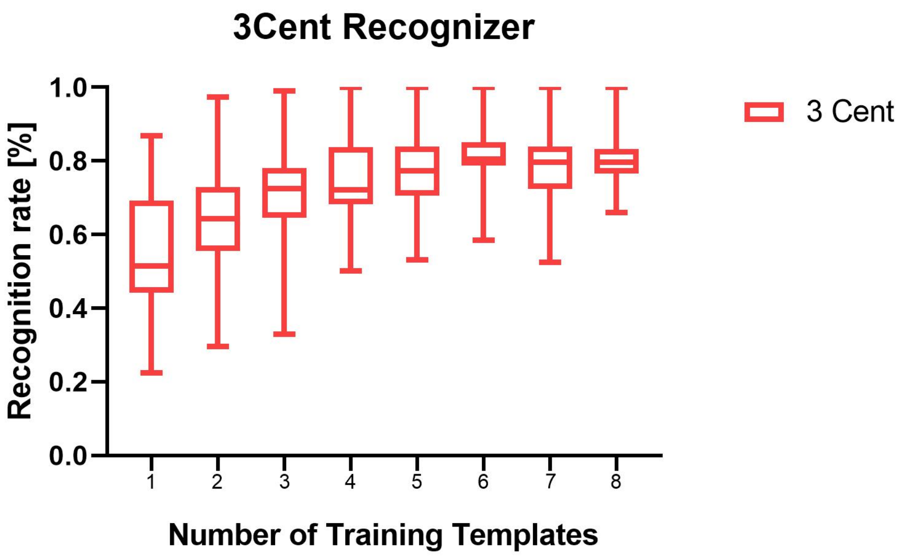


Figure A.25: Recognition Rate for 3 Cent recognizer for 3D gestures

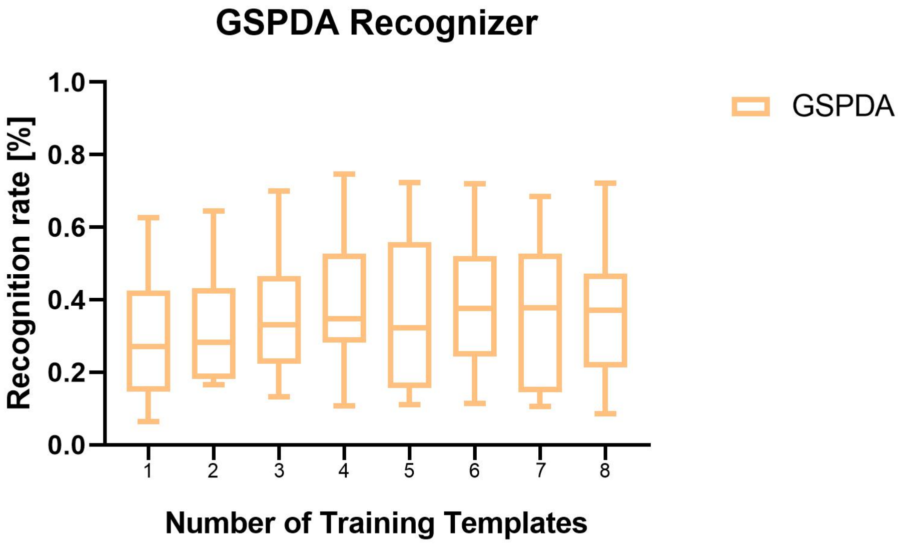


Figure A.26: Recognition Rate for GSPDA recognizer for 3D gestures

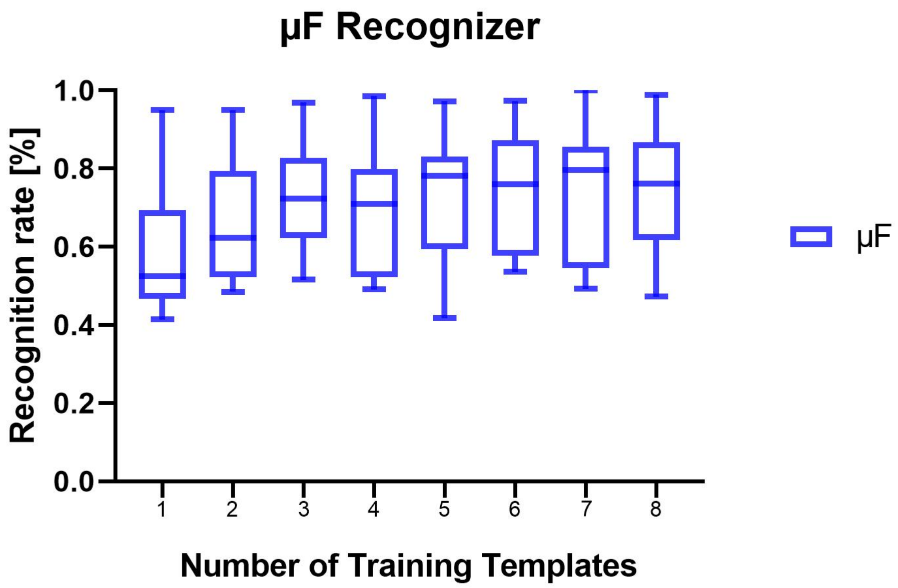


Figure A.27: Recognition Rate for μF recognizer for 3D gestures

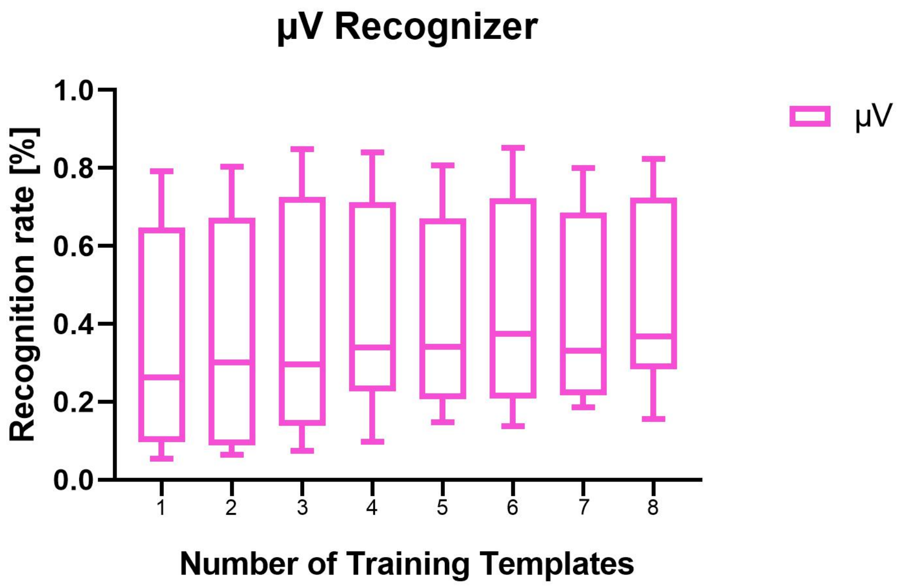


Figure A.28: Recognition Rate for μV recognizer for 3D gestures

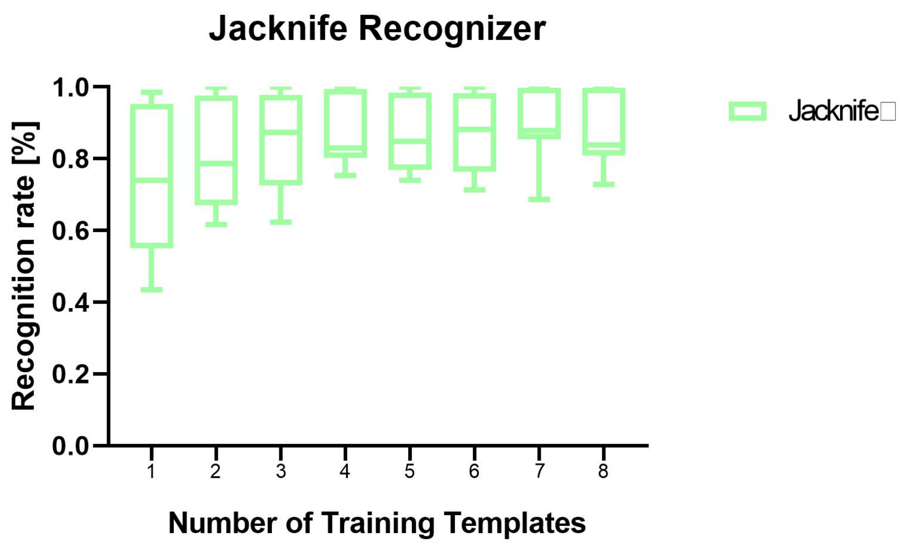


Figure A.29: Recognition Rate for Jacknife recognizer for 3D gestures

A.7 Recognition Rate of gestures with One Finger

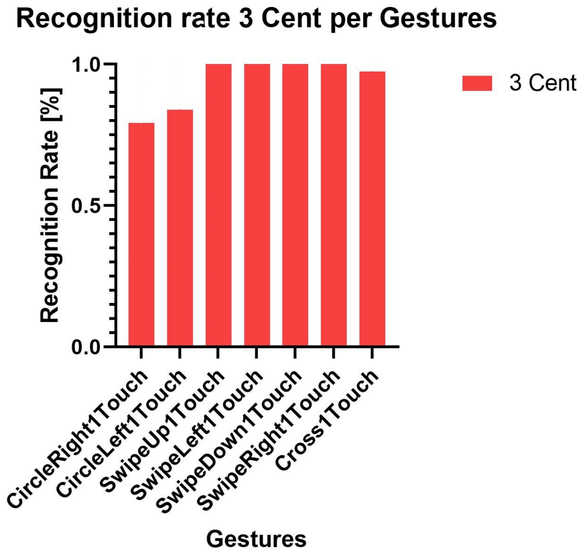


Figure A.30: Recognition Rate for 3 Cent recognizer per gesture with 1 finger

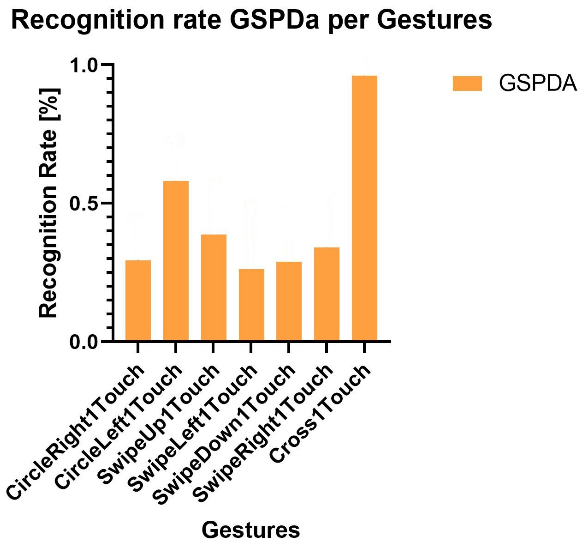


Figure A.31: Recognition Rate for GSPDA recognizer per gesture with 1 finger

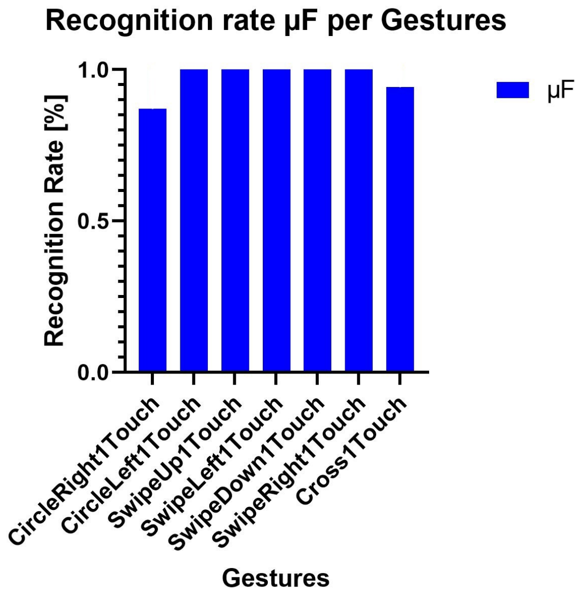


Figure A.32: Recognition Rate for μF recognizer per gesture with 1 finger

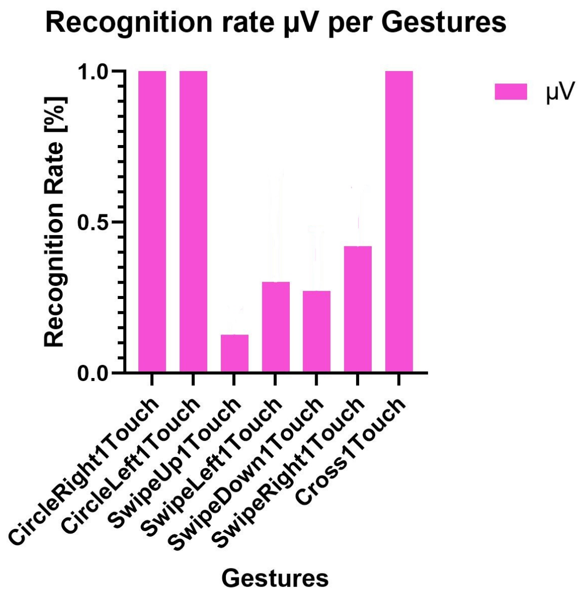


Figure A.33: Recognition Rate for μV recognizer per gesture with 1 finger

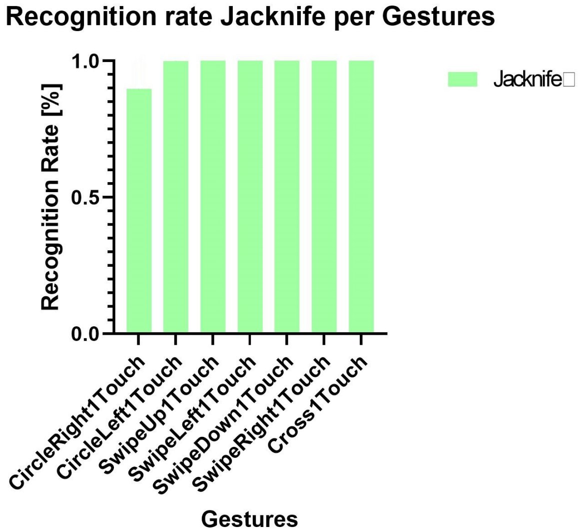


Figure A.34: Recognition Rate for Jacknife recognizer per gesture with 1 finger

A.8 Recognition Rate of gestures with two Fingers

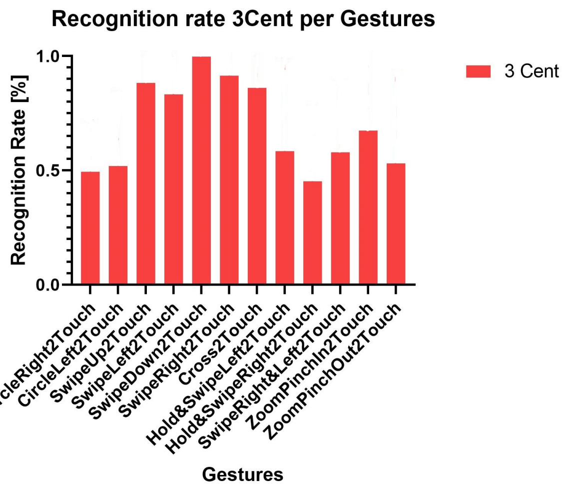


Figure A.35: Recognition Rate for 3 Cent recognizer per gesture with 2 fingers

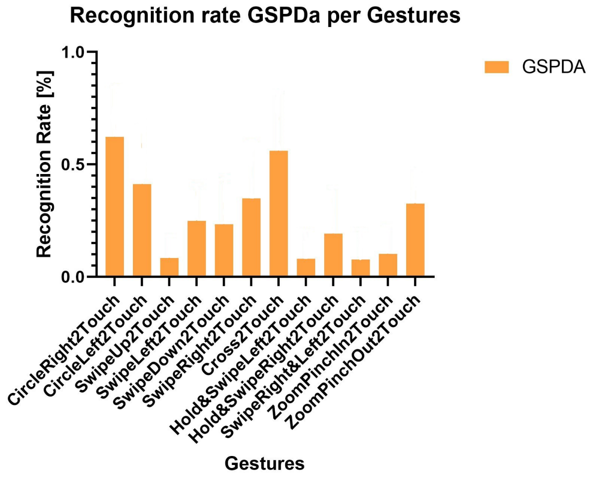


Figure A.36: Recognition Rate for GSPDA recognizer per gesture with 2 fingers

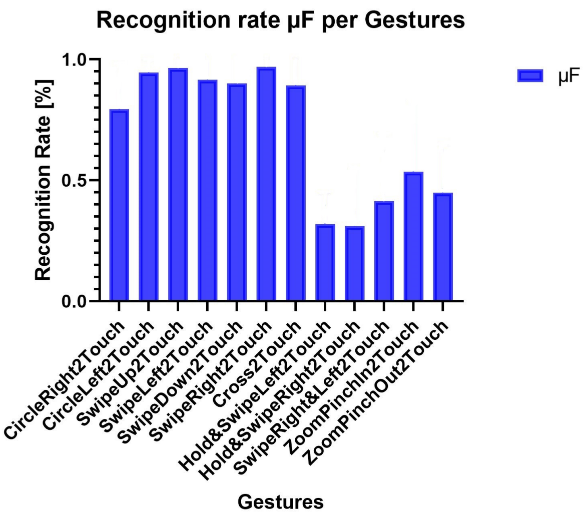


Figure A.37: Recognition Rate for μF recognizer per gesture with 2 fingers

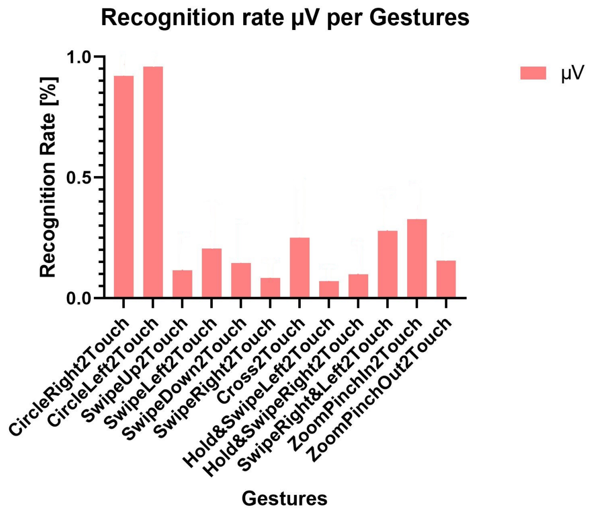


Figure A.38: Recognition Rate for μV recognizer per gesture with 2 fingers

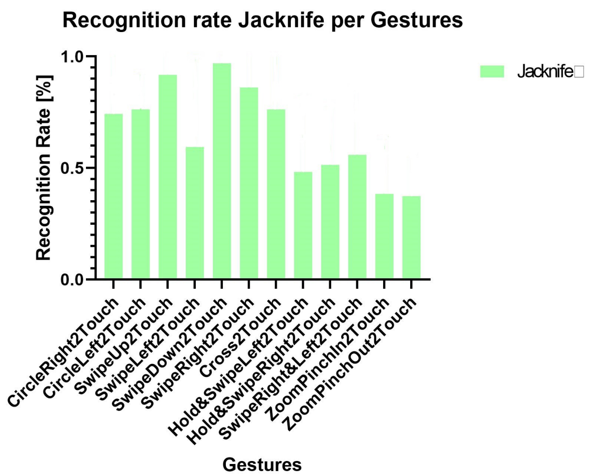


Figure A.39: Recognition Rate for Jacknife recognizer per gesture with 2 fingers

A.9 Recognition Rate of gestures with three Fingers

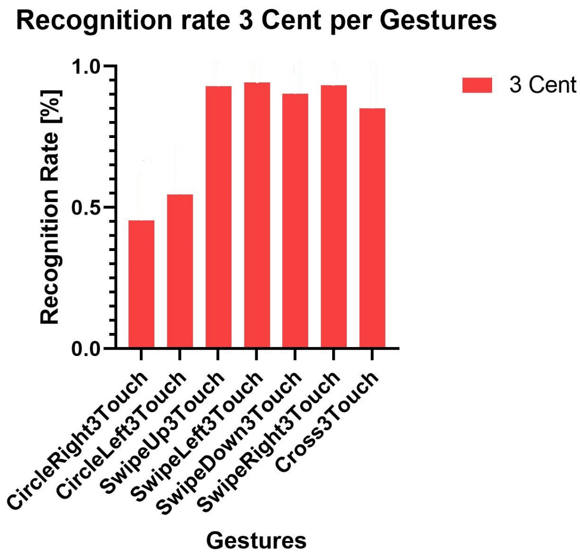


Figure A.40: Recognition Rate for 3 Cent recognizer per gestures with 3 fingers

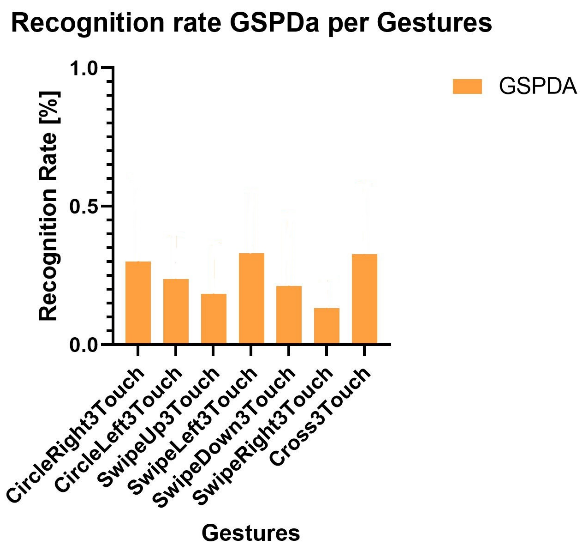


Figure A.41: Recognition Rate for GSPDA recognizer per gestures with 3 fingers

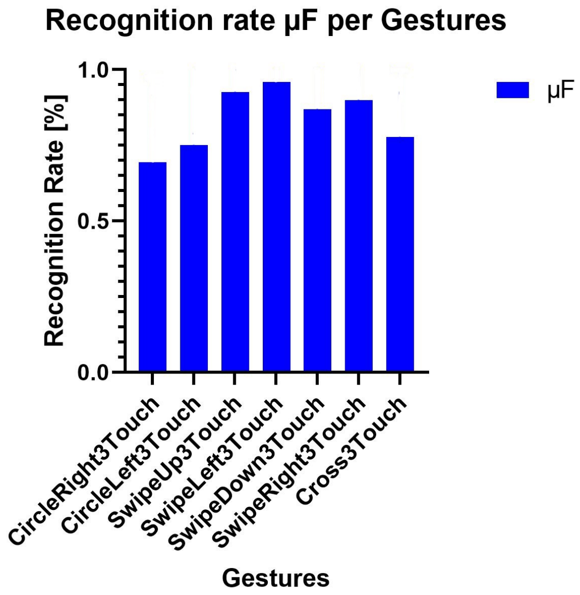


Figure A.42: Recognition Rate for μF recognizer per gestures with 3 fingers

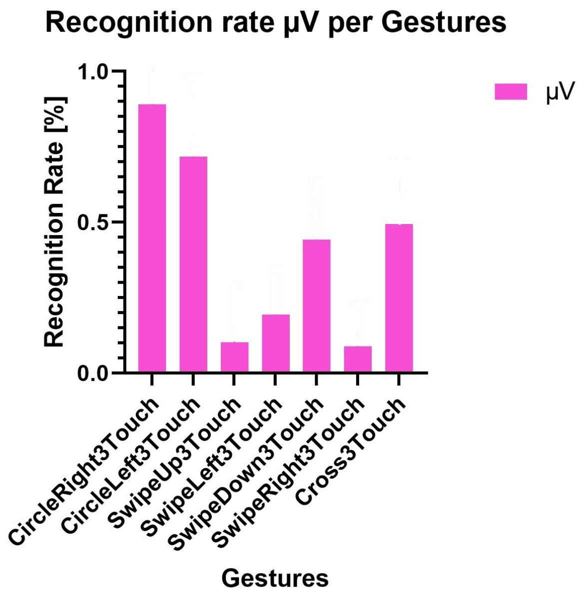


Figure A.43: Recognition Rate for μV recognizer per gestures with 3 fingers

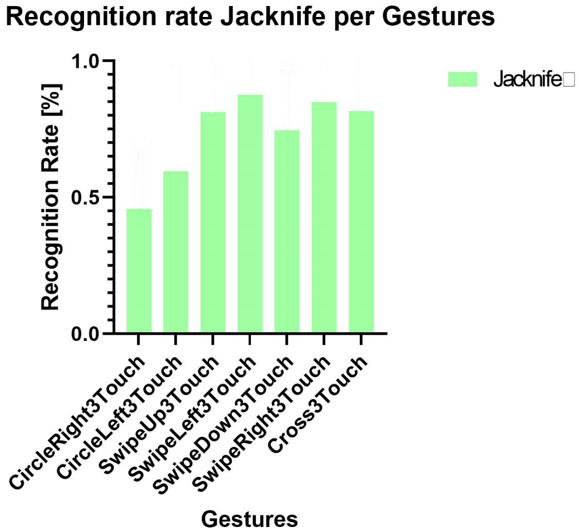


Figure A.44: Recognition Rate for Jacknife recognizer per gestures with 3 fingers

A.10 Recognition Rate of 3D gestures

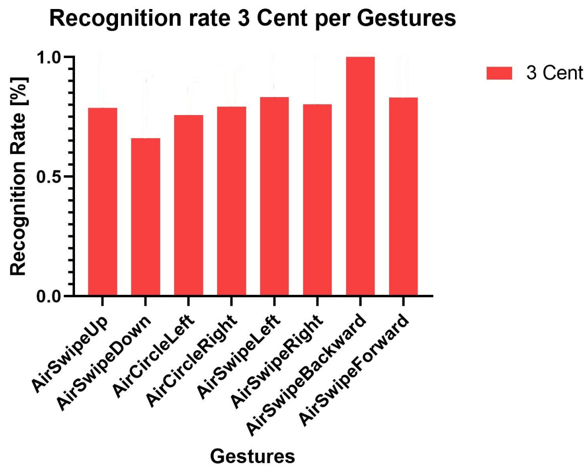


Figure A.45: Recognition Rate for 3 Cent recognizer per 3D gestures

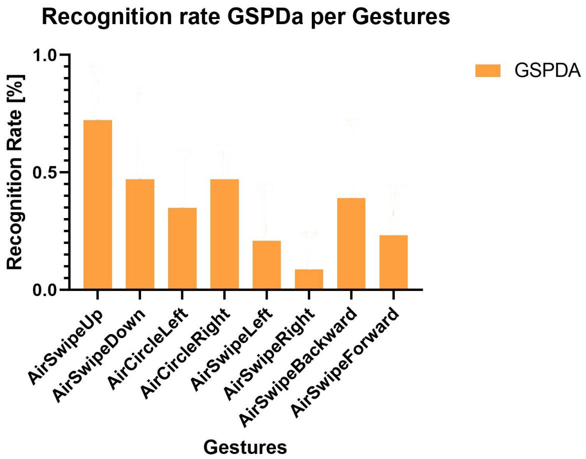


Figure A.46: Recognition Rate for GSPDA recognizer per 3D gestures

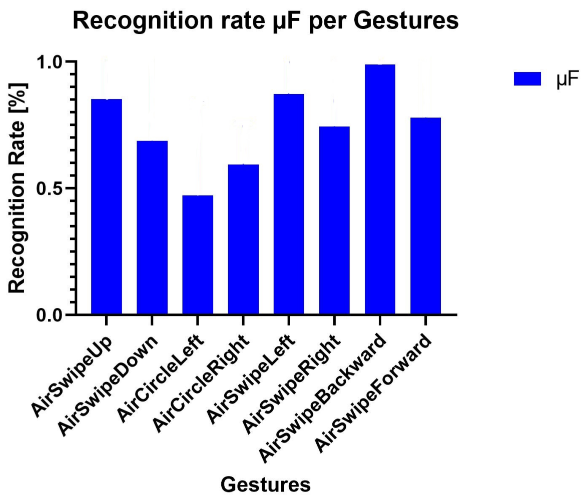


Figure A.47: Recognition Rate for μF recognizer per 3D gestures

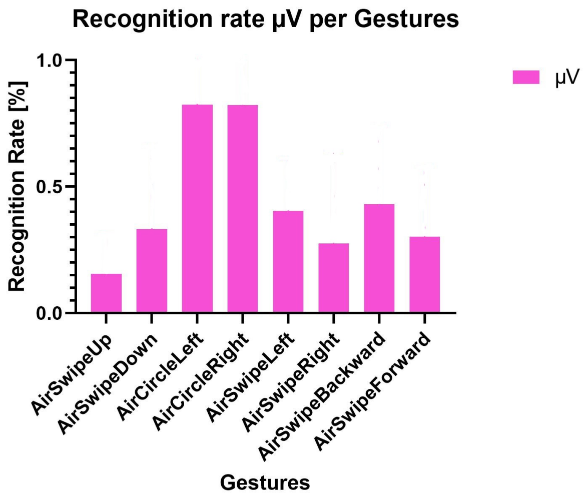


Figure A.48: Recognition Rate for μV recognizer per 3D gestures

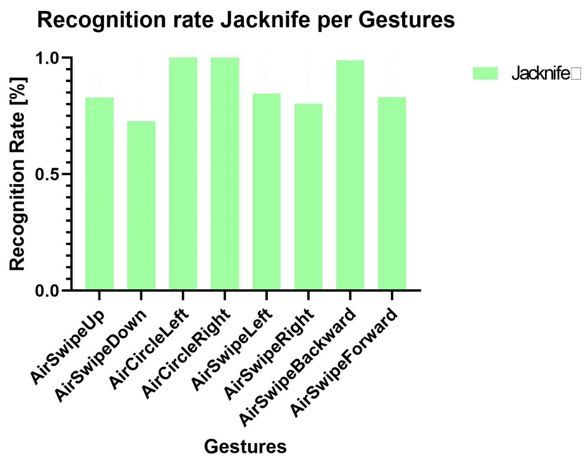


Figure A.49: Recognition Rate for Jacknife recognizer per 3D gestures

A.11 Prototype Implementation

```
  import React from 'react'
import { withStyles } from '@material-ui/core/styles';
// Images

//Slide
import HelpGesture from './Picture_pptx/HelpGesture.jpg';
import Slide1 from './Picture_pptx/Slide1.JPG';
```

```
import Slide2 from './Picture_pptx/Slide2.JPG';
import Slide3 from './Picture_pptx/Slide3.JPG';
import Slide4 from './Picture_pptx/Slide4.JPG';
import Slide5 from './Picture_pptx/Slide5.JPG';
import Slide6 from './Picture_pptx/Slide6.JPG';
import Slide7 from './Picture_pptx/Slide7.JPG';
import Slide8 from './Picture_pptx/Slide8.JPG';
import Slide9 from './Picture_pptx/Slide9.JPG';
import Slide10 from './Picture_pptx/Slide10.JPG';
import Slide11 from './Picture_pptx/Slide11.JPG';
import Slide12 from './Picture_pptx/Slide12.JPG';
import Slide13 from './Picture_pptx/Slide13.JPG';
import Slide14 from './Picture_pptx/Slide14.JPG';
import Slide15 from './Picture_pptx/Slide15.JPG';

//Zoom Slides
import Slide1Zoom from './Picture_pptx/Slide1Zoom.JPG';
import Slide2Zoom from './Picture_pptx/Slide2Zoom.JPG';
import Slide3Zoom from './Picture_pptx/Slide3Zoom.JPG';
import Slide4Zoom from './Picture_pptx/Slide4Zoom.JPG';
import Slide5Zoom from './Picture_pptx/Slide5Zoom.JPG';
import Slide6Zoom from './Picture_pptx/Slide6Zoom.JPG';
import Slide7Zoom from './Picture_pptx/Slide7Zoom.JPG';
import Slide8Zoom from './Picture_pptx/Slide8Zoom.JPG';
import Slide9Zoom from './Picture_pptx/Slide9Zoom.JPG';
import Slide10Zoom from './Picture_pptx/Slide10Zoom.JPG';
import Slide11Zoom from './Picture_pptx/Slide11Zoom.JPG';
import Slide12Zoom from './Picture_pptx/Slide12Zoom.JPG';
import Slide13Zoom from './Picture_pptx/Slide13Zoom.JPG';
import Slide14Zoom from './Picture_pptx/Slide14Zoom.JPG';

import { Button, Container, Paper, Typography }
from '@material-ui/core';
import { createMuiTheme, ThemeProvider } from '@material-ui/core';

import GestureHandler from "quantumleapjs";


const theme = createMuiTheme();

const styles = (theme) => ({
  root: {
    height: '100vh',
    width: '100vw',
    boxSizing: 'border-box',
    paddingTop: theme.spacing(2),
    paddingBottom: theme.spacing(2),
    backgroundColor: theme.palette.background.default,
  },
```

```
  container: {
    padding: theme.spacing(2),
    backgroundColor: theme.palette.background.paper,
  },
  imageContainer: {
    height: '1000px',
    width: '100%',
    backgroundColor: theme.palette.background.default,
  },
  buttonsContainer: {
    display: 'flex',
    alignItems: 'center',
    '& > *': {
      margin: theme.spacing(1),
    },
  },
  image: {
    display: 'block',
    height: '1000px',
    width: '100%',
    margin: 'auto',
  },
  dot: {
    height: '40px',
    width: '40px',
    backgroundColor: '#bbb',
    borderRadius: '50%',
    display: 'inline-block',
    position: 'fixed',
  },
});

class App extends React.Component {
  constructor(props) {
    super(props);
    this.state = {
      name: '/',
      type: '/',
      image: HelpGesture,
      displayTime: 999,
      connected: false,
    };
    this.counterSlide = 0
    this.SlideTable = [Slide1,Slide2,Slide3,Slide4,Slide5,Slide6,
    Slide7,Slide8,Slide9,Slide10,Slide11,Slide12,Slide13,
    Slide14,Slide15]
    this.SlideZoomTable = [Slide1Zoom,Slide2Zoom,Slide3Zoom,
    Slide4Zoom,Slide5Zoom,Slide6Zoom,Slide7Zoom,Slide8Zoom,
    Slide9Zoom,Slide10Zoom,Slide11Zoom,Slide12Zoom,
```

```
    Slide13Zoom , Slide14Zoom ]
    // Bind
    this.onGesture = this.onGesture.bind(this);
    this.onAirSwipeRight = this.onAirSwipeRight.bind(this);
    this.onAirSwipeLeft = this.onAirSwipeLeft.bind(this);
    this.onAirCircleLeft = this.onAirCircleLeft.bind(this);
    this.onAirCircleRight=this.onAirCircleRight.bind(this);
        this.onAirSwipeForward = this.onAirSwipeForward.bind(this);
        this.onAirSwipeBackward=this.onAirSwipeBackward.bind(this);

    this.onSwipeRight1Finger = this.onSwipeRight1Finger.bind(this);
    this.onSwipeRight2Fingers = this.onSwipeRight2Fingers.bind(this);
    this.onSwipeRight3Fingers = this.onSwipeRight3Fingers.bind(this);

    this.onSwipeLeft1Finger = this.onSwipeLeft1Finger.bind(this);
    this.onSwipeLeft2Fingers = this.onSwipeLeft2Fingers.bind(this);
    this.onSwipeLeft3Fingers = this.onSwipeLeft3Fingers.bind(this);

    this.onSwipeDown1Finger = this.onSwipeDown1Finger.bind(this);

    this.onSwipeUp1Finger = this.onSwipeUp1Finger.bind(this);

        this.onCross1Touch = this.onCross1Touch.bind(this);
        this.onCross2Touch = this.onCross2Touch.bind(this);

        this.onPinchIn = this.onPinchIn.bind(this);
        this.onPinchOut = this.onPinchOut.bind(this);



    this.timer = null;
    this.gestureHandler = new GestureHandler();

  }

  componentDidMount() {
    this.gestureHandler.registerGestures("dynamic", ["AirSwipeLeft",
    "AirSwipeRight","AirCircleLeft","AirCircleRight","AirCircleRight",
    "AirSwipeForward","AirSwipeBackward","SwipeLeft1Touch",
    "SwipeRight1Touch","SwipeUp1Touch","SwipeDown1Touch",
    "SwipeLeft2Touch","SwipeRight2Touch","SwipeDown2Touch",
    "SwipeLeft3Touch","SwipeRight3Touch","Cross1Touch",
    "Cross2Touch","ZoomPinchIn2Touch","ZoomPinchOut2Touch"])

    this.gestureHandler.addListener('gesture', (event) => {
              console.log(event.gesture.name);
      switch (event.gesture.name) {
        case "AirSwipeRight":
          this.onAirSwipeRight();
```

```
          break;
        case "AirSwipeLeft":
          this.onAirSwipeLeft();
          break;
        case "AirCircleLeft":
          this.onAirCircleLeft();
          break;
        case "AirCircleRight":
          this.onAirCircleRight();
          break;

                case "AirSwipeForward":
         this.onAirSwipeForward();
         break;

                case "AirSwipeBackward":
         this.onAirSwipeBackward();
         break;


              case "SwipeRight1Touch":
                this.onSwipeRight1Finger();
                break;
              case "SwipeLeft1Touch":
                this.onSwipeLeft1Finger();
                break;
              case "SwipeUp1Touch":
                this.onSwipeUp1Finger();
              break;
        case "SwipeDown1Touch":
          this.onSwipeDown1Finger();
          break;
                case "SwipeRight2Touch":
                this.onSwipeRight2Fingers();
                break;
              case "SwipeLeft2Touch":
                this.onSwipeLeft2Fingers();
                break;

                case "SwipeRight3Touch":
                this.onSwipeRight3Fingers();
                break;
              case "SwipeLeft3Touch":
                this.onSwipeLeft3Fingers();
                break;


                  case "Cross1Touch":
                  this.onCross1Touch();
```

```
                  break;
                  case "Cross2Touch":
                  this.onCross2Touch();
                  break;


                  case"ZoomPinchIn2Touch":
                  this.onPinchIn();
                  break;

                  case"ZoomPinchOut2Touch":
                  this.onPinchOut();
                  break






        default:
          console.log("Unsupported gesture");
      }

      this.onGesture(event.gesture.type, event.gesture.name);
    });

    this.gestureHandler.addListener('connect', (event) => {
      this.setConnected(true);
    });

    this.gestureHandler.addListener('disconnect', (event) => {
      this.setConnected(false);
    });

    this.gestureHandler.connect();


    // Timer
    this.timer = setInterval(() => {
      if (this.state.displayTime > 0) {
        this.setState((prevState) => {
          return { displayTime: prevState.displayTime - 1 };
        });
      } else if (this.state.image) {
        this.setState({ name: '/', type: '/', image: '' });
      }
    }, 100);
  }
```

```
componentWillUnmount() {
  clearInterval(this.timer);
  this.gestureHandler.disconnect();

}

setConnected(connected) {
  this.setState({
    connected: connected,
  });
}

onGesture(type, name) {
  this.setState({
    name: name,
    type: type,
  });
}


onSwipeRight1Finger(){
  if(this.counterSlide + 1 < this.SlideTable.length-1){
    this.counterSlide = this.counterSlide+1
  this.setState({
    image: this.SlideTable[this.counterSlide],
    displayTime: 999,
  });
  }
}

onSwipeRight2Fingers(){
  if(this.counterSlide + 2 < this.SlideTable.length-1){
    this.counterSlide = this.counterSlide+2
  this.setState({
    image: this.SlideTable[this.counterSlide],
    displayTime: 999,
  });
  }
}

onSwipeRight3Fingers(){
  if(this.counterSlide + 3 < this.SlideTable.length-1){
    this.counterSlide = this.counterSlide+3
  this.setState({
    image: this.SlideTable[this.counterSlide],
    displayTime: 999,
  });
  }
}
```

```
onSwipeLeft1Finger(){
  if(this.counterSlide - 1 > -1){
    this.counterSlide = this.counterSlide-1
  this.setState({
    image: this.SlideTable[this.counterSlide],
    displayTime: 999,
  });
  }
}

onSwipeLeft2Fingers(){
  if(this.counterSlide - 2>-1){
    this.counterSlide = this.counterSlide-2
  this.setState({
    image: this.SlideTable[this.counterSlide],
    displayTime: 999,
  });
  }
}

onSwipeLeft3Fingers(){
  if(this.counterSlide -3 > -1){
    this.counterSlide = this.counterSlide-3
  this.setState({
    image: this.SlideTable[this.counterSlide],
    displayTime: 999,
  });
  }
}

onSwipeUp1Finger(){
  this.setState({
    image: this.SlideTable[this.counterSlide],
    displayTime: 999,
  });
}

onSwipeDown1Finger(){
  this.setState({
    image: HelpGesture,
    displayTime: 999,
  });
}

onCross1Touch(){
  this.counterSlide = this.SlideTable.length -1
  this.setState({
    image: this.SlideTable[this.counterSlide],
```

```
      displayTime: 999,
    });
  }
  onCross2Touch(){
    this.counterSlide = 0
    this.setState({
      image: this.SlideTable[this.counterSlide],
      displayTime: 999,
    });
  }



  onAirSwipeRight(){
    if(this.counterSlide + 1 < this.SlideTable.length-1){
    this.counterSlide = this.counterSlide+1
    this.setState({
      image: this.SlideTable[this.counterSlide],
      displayTime: 999,
    });
  }
  }

  onAirSwipeLeft(){
    if(this.counterSlide -1 > -1){
    this.counterSlide = this.counterSlide-1
    this.setState({
      image: this.SlideTable[this.counterSlide],
      displayTime: 999,
    });
  }
  }

  onAirCircleLeft(){
    this.setState({
      image: this.SlideZoomTable[this.counterSlide],
      displayTime: 999,
    });
  }

  onAirCircleRight(){
    this.setState({
      image: this.SlideTable[this.counterSlide],
      displayTime: 999,
    });
  }

        onAirSwipeBackward(){
    this.setState({
```

```
    image: HelpGesture,
    displayTime: 999,
  });
      }
      onAirSwipeForward(){
  this.setState({
    image: this.SlideTable[this.counterSlide],
    displayTime: 999,
  });
      }

      onPinchIn(){
              this.setState({
    image: this.SlideZoomTable[this.counterSlide],
    displayTime: 999,
  });
      }

      onPinchOut(){
  this.setState({
    image: this.SlideTable[this.counterSlide],
    displayTime: 999,
  });
      }



render() {
  const { classes } = this.props;
  const { image, name, type, connected } = this.state;
  return (
    <div className={classes.root}>
      <ThemeProvider theme={theme}>
        <Container>
          <Paper className={classes.container}>
            <Typography variant='h3'>
              {`Gesture: ${name}`}
            </Typography>
            <Typography variant='h5'>
              {`Type: ${type}`}
            </Typography>
            <div className={classes.imageContainer}>
              {image && <img className={classes.image}
              alt={name} src={image} />}
            </div>
            <Typography variant='h6'>
              {`Status: ${connected ? 'connected'
              : 'disconnected'}`}
            </Typography>
```

```
                <div className={classes.buttonsContainer}>
                  {}
                </div>
              </Paper>
            </Container>
          </ThemeProvider>
        </div>
      );
  }
}

export default withStyles(styles)(App);
```

Abstract :

Résumé :

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
Louvain School of Management
Place des Doyens, 1 bte L2.01.01, 1348 Louvain-la-Neuve
Boulevard Emile Devreux 6, 6000 Charleroi, Belgique
Chaussée de Binche 151, 7000 Mons, Belgique
www.uclouvain.be/lsm