A Window to your Smartphone: Augmented Virtuality Using Depth Sensing Cameras for Near-Range Awareness

Submitted by
Ghassem Alaee Khangha
M.Sc. Candidate

Under the supervision of
Dr. Oscar Meruvia-Pastor
Dr. Edward Brown

Fall 2017

Acknowledgments

I would like to express my deep gratitude to Dr. Oscar Meruvia-Pastor and Dr. Edward Brown, my project supervisors, for their patient guidance, enthusiastic encouragement and useful critiques of this research work. I would also like to thank Dr. Lourdes Pena-Castillo, for her assistance in the data analysis. My grateful thanks are also extended to Mr. Amit Desai for his help in integrating his implementations into our experiment.

Finally, I wish to thank my family for their support and encouragement throughout my study.

Abstract

Immersive Virtual Reality (VR) Head-Mounted Displays (HMDs), such as the Oculus Rift, isolate the user visually, and sometimes aurally, from the physical environment. This partial and sometimes total visual isolation which comes as a by-product of providing a better virtual experience makes it impossible for the user to see, hear and even interact with the physical world while wearing the HMD. One way of dealing with this issue is to move towards an Augmented Virtuality (AV) design. Augmented Virtuality (AV) incorporates a part of reality into the virtual world. In this project, a unique AV environment provides near-range awareness by incorporating the users' personal mobile device within the field of view (FOV) inside the VR, i.e. while wearing an HMD. Two different methods to achieve this type of near-range awareness are evaluated. The first is a method by Desai et al., by which a Leap Motion sensor is used to track the user's hand position and orientation as the user holds a smartphone in front of the HMD as if looking at it. The second is specifically developed for this project, using an Intel Real Sense depth sensing-camera in place of the Leap Motion in Desai's approach. Results of a user study with 25 participants indicate that the implementation using the Intel Real Sense camera proposed in this project allows users to perform multiple tasks and is also preferred by the majority of the users, when compared to the approach of Desai et al. The results also suggest that some users still prefer to remove the HMD to accomplish some of the tasks.

Chapter 1: Introduction

A report by MarketsAndMarkets anticipates that the HMD market will grow up to USD 25 Billion by 2022 [1]. Over the last few years, numerous HMD products have been introduced to the market from big tech companies like Google, Microsoft, and Facebook. HMDs are being used in many fields such as gaming, medicine, and education. HMDs can be categorized into the following groups:

1. **Slide-on HMD:** has a smartphone holder where users insert a smartphone for a screen, lenses and a basic input; e.g. Google Cardboard, Samsung Gear VR [2].
2. **Discrete HMD:** has a display, lenses, rotational tracking, positional tracking, audio capability and some inputs. It should be connected to a PC or a gaming console; e.g. Oculus Rift, HTC Vive, PlayStation VR [3].
3. **Integrated HMD:** is more sophisticated and more expensive compared to the other two types. It contains everything from display to processors and camera. They do not need to be connected to a processing unit like a PC; e.g. Microsoft HoloLens, Google Glass [4].

The resolution, refresh rate, the field of view (FOV), audio, tracking capabilities and type of connectivity are some other considerations that can be used to compare the HMDs. While not all of these devices are designed and aimed to serve the same purpose, the common goal is to provide an immersive VR experience. For this goal, some of these headsets are designed with an encompassing FOV which has a high-resolution display, such as Oculus Rift, as opposed to other HMDs with a smaller FOV like Microsoft HoloLens. A higher degree of immersion is associated with an increased coverage of the FOV of the user [5]. On the other hand, having an HMD that completely covers the human FOV has an important downside: total visual isolation.

Figure 1. (Left) Oculus Rift DK2 (discrete HMD), (Right) Microsoft HoloLens (integrated HMD)

According to Milgram [6], we can consider a continuum of Reality-Virtuality (RV) as shown in Figure 2. Currently, HMDs, like Oculus Rift, are being used in VR and that means the user is being immersed in a completely Virtual Environment (VE) without any elements of the real environment. In contrast, Microsoft HoloLens is designed to be used in an Augmented Reality (AR) manner which means that the user is able to see the real environment and at the same time can see some virtual elements being projected on the panel or lenses of the HoloLens. Microsoft calls this a Mixed Reality environment. On the other hand, Augmented Virtuality (AV) refers to a primarily VE which has some real elements in it. Currently, there is not any HMDs with AV capability in the market. However, AV is usually produced with some additional camera or sensors. Mixed Reality is often used to refer to this spectrum of this RV environment.

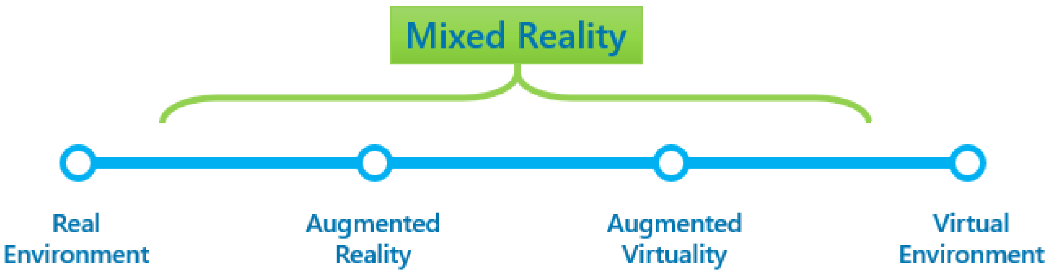

Figure 2. A simple representation of an RV continuum.

It has been forecasted that by the end of 2018, more than a third of the world population will be using smartphones [7]. With the extent of growth of social media and advances in the computational power of smartphones, both social media and smartphones are becoming major parts of our lives. The visual isolation caused by VR HMDs makes it impossible for the user to interact with the environment; hence, the user has no choice other than removing the headset for any social interactions, especially if that interaction involves a smartphone. This situation brings a certain level of displeasure to the user's VR experience. This problem could be easily solved if the user's HMD was able to deliver an AV experience in which the users were able to interact with their smartphones.

Figure 3. (left) Intel® RealSense™ Technology, (Right) Leap Motion Controller

In this project, we have implemented AV solutions to this issue using the Oculus Rift Development Kit 2 (DK2) VR HMD and equipped it with depth-sensing cameras Intel RealSense and Leap Motion Controller, shown in Figure 3. We have set out to measure the operability of these AV implementations with a user study. The user study has three conditions where each of them represents one of the possible ways of completing a set of tasks, including one condition where the user is not using any AV implementation. The key question is whether the two AV implementations can be the alternatives to removing the headset for social interactions. The results of the user study show that the participants support the idea of an AV design in VR HMDs and that they preferred one of the implementations over the other. The following sections discuss the background of the study, our implementation, the experiment, and our results in more detail.

Chapter 2: Background and Related Work

The issue of visual isolation has been addressed by other researchers in two different ways. They either tried to import a portion of reality into the VE as it is in the real world or they virtualized a part of the reality into the VE. Examples of the first approach are the works done by Steinick et al [8], Tecchia, et al [9] and Nahon et al [10]. In these examples, at least one camera was used to capture part or all of the user's body or the environment and that portion of the environment relevant to the purpose of the study was imported into the VE.

Steinick et al. used two separate cameras, an IR camera, and a USB camera, to obtain the user's body and incorporated it into a VE. They used the resulting environment to test whether the users like their augmented presence in the VE and the results showed a higher sense of presence. Tecchia et al. used an RGB-D camera to create the same effect. They also argue that by having an egocentric view of the VR and augmenting the user's body into it, the user would have a better virtual experience. As opposed to the previous two systems, Nahon et al. proposed a method to not only support a 1st person point of view of the environment but also 3rd person view. They also claimed the resulting environment would solve some of the safety issues such as hitting something or falling. Their AV environment uses a fixed Kinect to capture the scene.

An example of the second approach would be a system implemented by Desai, et al [11]. The focus of that research was to solve the visual isolation by augmenting the smartphone to the VE. In that work, they used Leap Motion and Oculus Rift DK2, as shown in Figure 4. An Android App was also used to send the screenshots and orientation data to a VR Unity application. This approach proposed two new algorithms specifically designed to detect a smartphone that is being held within a certain range and orientation in front of the Leap Motion controller. The combination of these two algorithms resulted in a real-time system capable of detecting the smartphone with 90% accuracy when a smartphone is held in front of the Leap Motion controller [11].

Figure 4. Leap Motion controller mounted on the Oculus Rift DK2.

Both approaches deliver a certain level of awareness about the user's body or the user's environment. Previous work also showed that having an egocentric AV environment increases awareness of the users and that they feel more present in the environment [8,9]. To the best of our knowledge, no study has been done in the same category as the first approach to augment, specifically, a smartphone with the operator's hands into a VE to solve the visual isolation problem. We used this as an opportunity to implement such an AV environment. Whether any of these two approaches deliver the best user experience is still an open question which needs further studies in this area. However, in this project, we examined one implementation from each of these approaches to see how operable they are and how much users would like them. More details about the implementations and the tools used in this project will be discussed in the next section.

Chapter 3: Implementation

In this chapter, we describe how to implement an AV environment that helps us to have a near-range awareness of ourselves while wearing an HMD. To implement such an environment, an RGB-D camera is required, and it should be mounted on top of the HMD, an Oculus Rift DK2. Both the camera and the HMD have a limited FOV; thus, the user has to hold the smartphone within the FOV that the camera covers, which is roughly in front of the face. The multi-touch screen is the primary input and output interface in smartphones nowadays. Because of this, hands are playing a major role in interacting with smartphones. Therefore, the operator's hands should also be augmented (included in the VR). In this implementation, using the depth perception capability of the camera and its SDK, we segment only the objects within a certain distance from the camera and augment the segmented image on the VE. The result of such process can be seen in Figure 5.

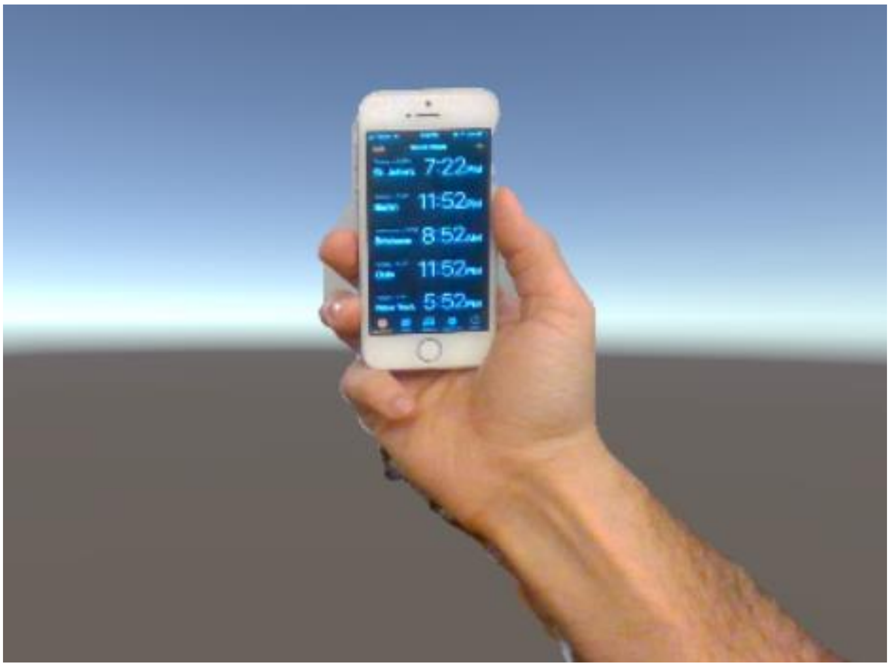

Figure 5. A segmented image of a hand holding a phone integrated with the virtual background.

Hardware requirements:

The following components are used in this implementation:

1. Oculus Rift

   Oculus Rift Developer Kit 2 (DK2) together with the Leap Motion sensor was used in this project (Figure 1). Oculus DK2 provides a 1920x1080 (960x1080 per eye) resolution and a maximum refresh rate of 75Hz [12]. Oculus Legacy Runtime for Windows 0.8.0.0-beta [13] is the only required software package needed to be installed prior the integration of Oculus Rift with Unity. This means that the developer does not need to write any code for this integration as long as the mentioned package is installed on the machine.
2. Intel RealSense Technology

   Intel's RealSense Developer Kit (SR300), as shown in Figure 3, was used in the implementation of this project. It is a depth-sensing camera that can be used for close-range depth perception. This version of the camera required an Intel 6^{th} generation (or above) processor and would only work on a machine with a USB 3.0 port and Microsoft Windows 10 as the operating system. Its highest color resolution is 1920x1080 at 30 frames per second (fps) and its highest depth resolution is 640x480 with an optimal distance between 20 cm to 1.5 meters for best depth perception. For dynamic background segmentation, which comes as an algorithm module in its SDK, this resolution is decreased to 1280x720 [14]. The SDK used at the time of preparation of this project was version 2016 R3 [15].

Algorithm

As mentioned earlier, Unity is the development environment for this project. The RealSense SDK provides a C# interface for Unity. To set up the Unity environment, the developer should follow the instructions provided in the documentation webpage [16].

There are three main steps to build an AV environment with the components mentioned above. Figure 5 shows the overall process.

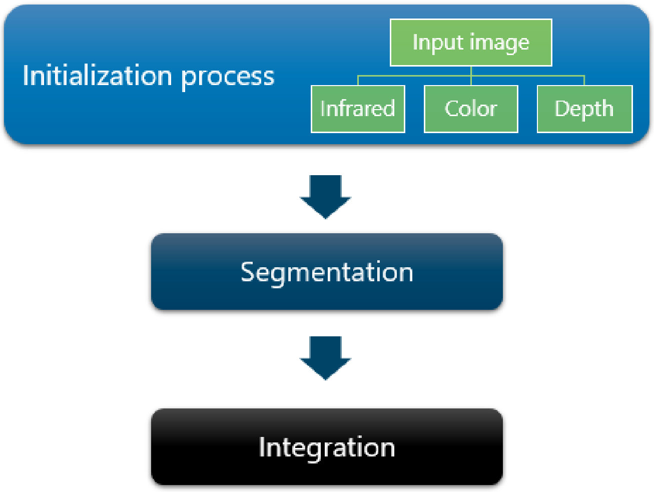

Figure 6. The overall process of a dynamic background segmentation process.

- Initialization:

  The two required namespaces in the RealSense SDK are 'Intel.RealSense' and 'Intel.RealSense.Segmentation'. In this step, we initialize the interfaces and the *GameObject* that shows the segmented image. Since the input image has a lower resolution than the maximum available resolution, reconfiguring the *StreamProfile* object to a higher resolution is necessary. The same process is repeated for the depth image.

- Segmentation:

  In segmentation, we select the objects that are close to the camera from the captured frames. This process is repeated for every single frame that is being sent from the camera to the previously initialized object in the start() function. The segmentation module comes with an event subscription mechanism that a function/method subscribes to, an event called 'FrameProcessed'. The subscriber/observer gets an instance of the image for each frame and then within the function, we segment the image in a way that only the foreground object being selected is shown.

- Integration:

  Displaying the segmented image along other *GameObjects* in the scene takes several lines of code in different functions. We define an 'IntPtr' pointer object in the global scope and attach it to the texture property of a *Material* Object. The *GameObject* which shows the segmented image is also connected to the same *Material* object. At the end of the segmentation process, the segmented image gets copied into the previously defined pointer. When the pointer gets updated, the *Material* object gets updated and finally, the *GameObject* shows the segmented image.

A simple version of this implementation is provided in a GitHub repository for educational purposes [17]. With this implementation, the user is able to see the augmented smartphone and operator's hands. The smartphone should be held within the FOV of the camera, and with approximately 30 cm distance in front of the camera. The result is an egocentric view of the smartphone within the VE, as shown in Figure 5.

Chapter 4: User Study

Experiment Setup

Based on the method explained in the previous section, we implemented an AV environment to provide smartphone access to a user wearing the HMD. Even though this implementation works most of the time, it has some limitations. For instance, the user must hold the phone at a certain distance from the RealSense camera. It cannot be too close (less than 10 cm) nor too far. After several tests, it has been found that the optimal distance for depth perception is more than what Intel has advertised. We found that optimal background segmentation is possible if the smartphone is held at approximately 30 cm from the camera.

State 1: Unable to segment and show the smartphone.

State 2: Segmented image would fade when the smartphone is not in the visible range. [closer than 30 cm]

State 3: Incorrect segmentation with flickering effect and not detecting the borders of the smartphones or part of the user's hand correctly.

State 4: Relatively a successful segmentation with some small errors on the edges.

Figure 7. Different states of segmentation.

Another limitation of the segmentation module in the RealSense SDK is that it finds only one continuous object in the foreground based on the depth and distance of that object with respect to the camera. This characteristic of the segmentation, on some occasions, causes the segmented image to flicker and in the worst case, it cannot segment anything as an object in the foreground. Figure 7 shows four different states which this segmentation may produce.

To determine whether our proposed implementation is operable, a user study is designed. As mentioned earlier, the work which was done by Desai, et al. was also designed to interact with one's smartphone while wearing an HMD. In this user study, these two implementations are evaluated to see which is more operable and more acceptable to the users. A baseline condition was also needed for this experiment. The baseline condition is the current state of technology, where users remove the headset whenever they want to have any social interactions, especially when a smartphone is involved. These three conditions will be identified with the following names in the rest of this report.

- Condition 1 (C1): Removing the headset [baseline]
- Condition 2 (C2): Using Intel RealSense [proposed here]
- Condition 3 (C3): Using Leap Motion [Desai, et al.]

To test the operability of these two implementations, 5 tasks are selected. These tasks are representative of the most likely tasks that a user may perform during a VR experience. Six fundamental interaction tasks which are independent of application and hardware are identified by Foley et al: Select (choosing between alternatives), Position (indicating a position on the display), Orient (rotating a symbol on the screen), Path (combination of Position and Orient), Quantify (entering a numeric value), Text (entering a text string) [18]. The 5 selected tasks comprise some of these interactions tasks.

Here are the descriptions of those tasks:

- Task 1: Answering a phone call

  Answering a phone call on an Android phone includes two main steps: swiping up the screen when it is ringing, answering, and putting the call-in-progress (or on speaker mode) by tapping on the speaker icon. The latter step is added to this task to make it easier for the participant to hear the other side of the conversation while wearing an HMD. The participant starts the task by hearing the ringtone and ends the task when the call is on speaker. This task includes 2 interaction tasks: Position and Select.

- Task 2: Doing some calculations

  The participants start the task by unlocking the phone (swiping up the screen) and ultimately launching the calculator app. Before starting the task, the participants know what calculation they are going to do, and during each attempt, they can request the calculation to be read to them whenever they want. The task ends with the participant reading out the result. This task includes 3 interaction tasks: Position, Select, and Text (or Quantify).

- Task 3: Reading a text message

  There are different ways to read an unread text message on an Android phone. The participants were asked to read the text message only by launching the messages app through the list of applications and not by using the notification bar. The task starts with the text tone, then the participant unlocks the phone and launches the messages app and ends with the participant reading out the text message. This task includes 2 interaction tasks: Position and Select.

- Task 4: Initiating a phone call

  In this task, for every attempt, the participant gets a name and should call that person by launching the contact app. The task starts with participant unlocking the phone and ends with the ringtone of the recipient's phone. This task includes 2 interaction tasks: Position and Select.

- Task 5: Navigating through the Viking Village

  In this task, the participants start the task from a starting point in a 3D environment and end the task by reaching another point. In condition 1, the participants use a keyboard to navigate, but in conditions 2 and 3, they use an Android app. At the beginning of the task, the android app is launched and ready to be used by the participant. This task includes 2 interaction tasks: Position and Select.

To simulate a gaming condition, we used the Viking Village, which is a free sample Unity project designed by Unity technologies [19]. Viking Village is a 3D environment that requires Unity 5.0 and above. Figure 8 shows a screenshot of its environment.

Figure 8. A screenshot of the Viking Village.

Implementation

To integrate these two implementations into the Viking Village, some changes have been made in the initial implementations. In condition 1 (C1), nothing has changed in the environment of Viking Village. Keyboard and mouse are the default controllers for moving or changing where the camera is pointing, and the user has to remove the HMD to operate these devices. In the other two conditions, the user is not going to remove the HMD and instead will interact with the app using the smartphone, therefore, an android app is designed to send directions to the Unity VR application. Figure 9 shows the UI of the Android app with four arrows for navigation purposes.

Figure 9. UI of the Screen Capture App.

In condition 2 (C2), other than adding the segmented image as a foreground layer in front of the camera object and allowing for on-air gesture interaction, an additional feature for using the smartphone as a gaming controller is incorporated. In other words, the Unity project as the client side of this communication waits for the smartphone, the server, to send the navigation directions. The Unity project needs the IP address and the port number from which the smartphone is sending its messages. Both the smartphone and the machine running the Unity project should be using the same network, preferably connecting to a router as shown in Figure 10.

Figure 10. A one-way communication between the smartphone and the Unity project.

The android app is also a major component in Desai's implementation, since condition 3 (C3) has already used this communication schema in its initial implementation, the only changes for implementing this in our condition 2 would be the integrating of the necessary visual layers into the default Viking Village project and adding the navigation capability. Figure 11 shows these two conditions after the integration into the Viking Village project.

Figure 11. (Left) Screenshot from condition 2, (Right) Screenshot from condition 3.

Data Collection

We collected the following data in this experiment:

1. Questionnaires:

   Two questionnaires were involved: a pre-questionnaire and a post-questionnaire. The pre-questionnaire collected personal information about the participants including the amount of time they spend on social media and different types of games on a weekly basis. The post-questionnaire was answered at the end of the experiment before the interview. The post-questionnaire collected the participant's preferences for the conditions in different tasks. The pre-questionnaire and post-questionnaire are available in the Appendix section of this report.

2. Timing measurements and pass/fail rates:

   All of the sessions were videotaped for measuring how much time the participant spends on each of the task attempts. At the same time, successful and unsuccessful attempts of the participants were recorded on result forms. The data from the pass and fail rates are shown in the Appendix.

3. Interview:

   As a part of debriefing, we asked the participants about their performance and how they felt about each of the tasks and the conditions, the reasons behind their questionnaire answers and about the possible ways to improve the conditions.

Chapter 5: Discussion of Results

Demographic:

Out of 25 participants who had accepted to take part in our experiment, only one person could not complete the user study due to the motion sickness that was caused by wearing the Oculus Rift for the first time. Participants were all undergraduate and graduate level students of MUN and they were mainly students from the Computer Science and Electrical Engineering departments. The participants can be split into two age groups for comparison, Newman, and Newman [20]: The Later Adolescence (18-24 years old) (40%) and the Early Adulthood (25-34 years old) (60% of the participants). Figure 12 show how much time they report they spend on different types of games and immersive devices on a weekly basis.

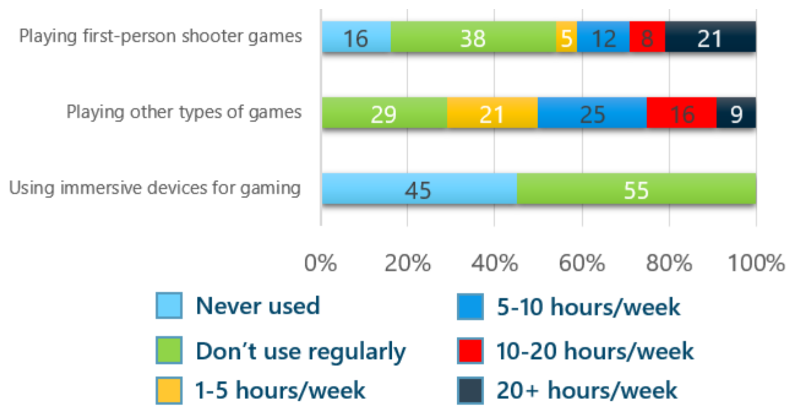

Figure 12. How much time the participants spend on a weekly basis.

Time measurements:

Each participant was asked to repeat each Task 3 times for each of the conditions. The timing data is available in the Appendix. The following Figures show the pairwise comparison of the mean time with respect to each condition. In each figure, three different comparisons can be seen, from top to bottom:

- C2-C1: time difference between condition 2 and condition 1
- C3-C1: time difference between condition 3 and condition 1
- C3-C2: time difference between condition 3 and condition 2

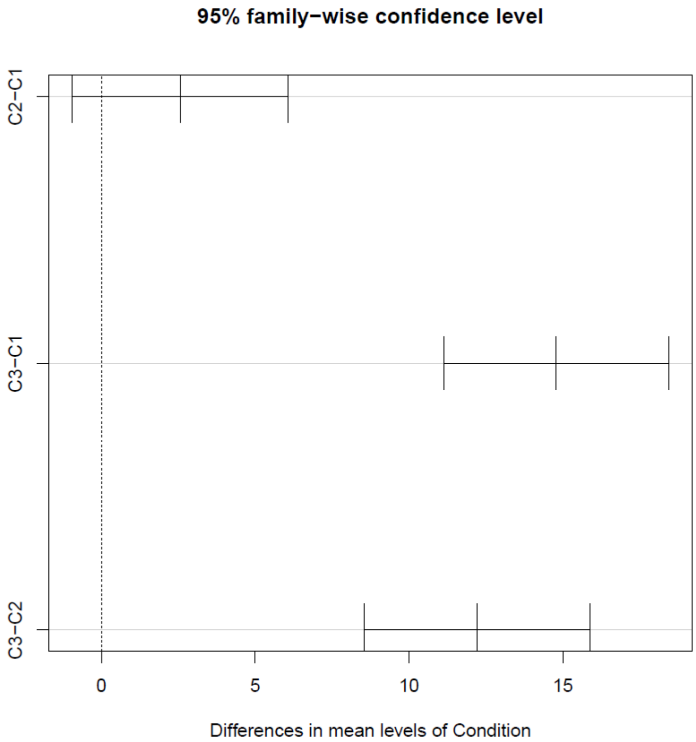

Figure 13. Time differences in seconds for Task 1

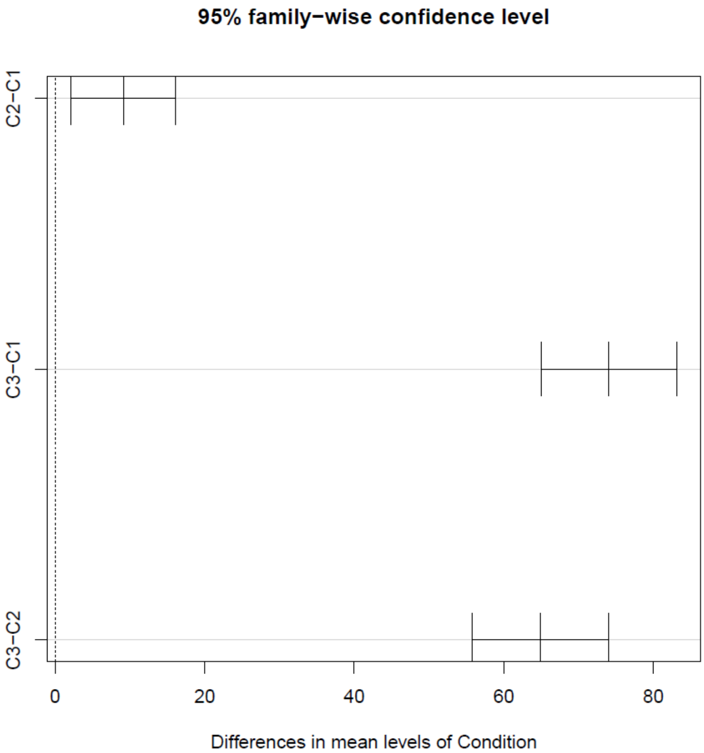

Figure 14. Time differences in seconds for Task 2

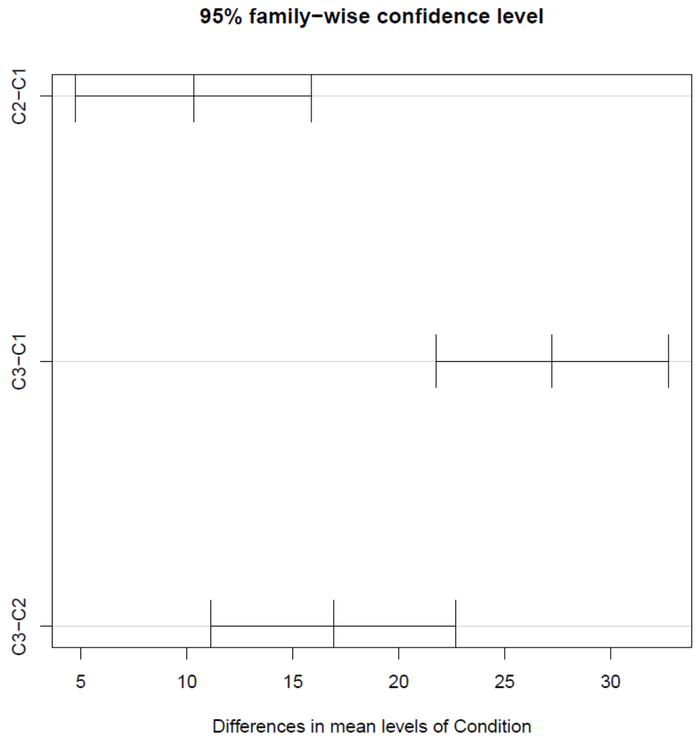

Figure 15. Time differences in seconds for Task 3

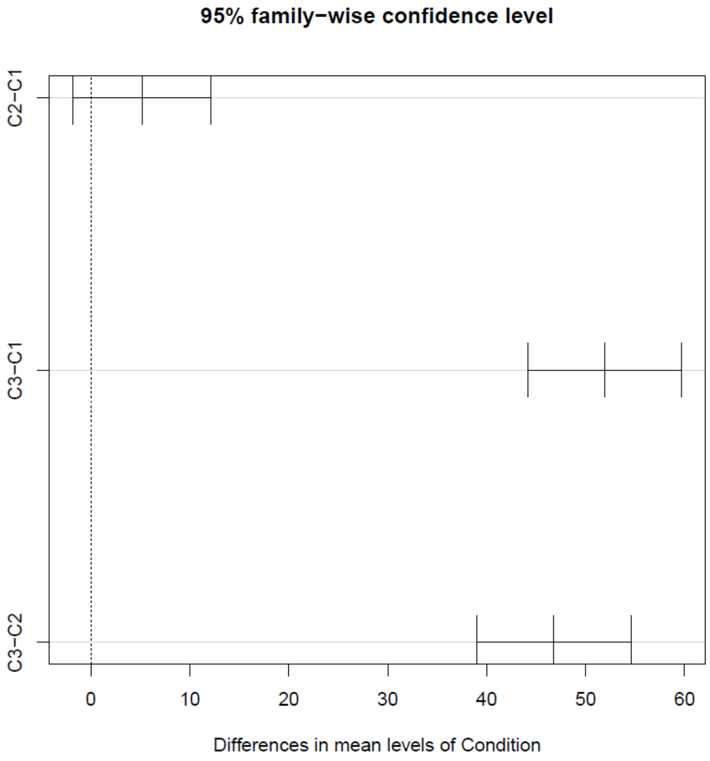

Figure 16. Time differences in seconds for Task 4

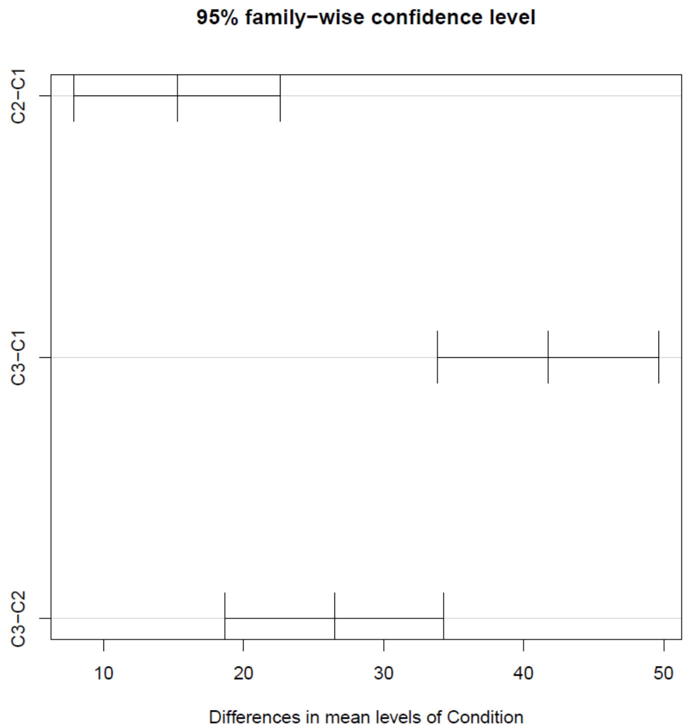

Figure 17. Time differences in seconds for Task 5

The complete analysis on timing is available in the Appendix. The followings observations are made based on that analysis:

- For Tasks 1 and 4, the mean time of condition 3 is statistically significantly different than the mean time of condition 1 and condition 2. Condition 2 can be executed in a shorter amount of time than condition 3. For both of these tasks, there were only a few participants who used condition 2 and were able to finish the tasks sooner than condition 1.
- For Tasks 2, 3 and 5: The mean time of condition 3 is statistically significantly different than the mean time of condition 1 and condition 2, and the mean time of condition 2 is statistically significantly different than the mean time of condition 1. The participants were able to finish these tasks faster in condition 2 than condition 3, however, they were faster in the condition 1 than condition 2.
- For Task 5: The mean time of attempt 3 is statistically different than the mean time of attempt 1. Users were faster in attempt 3 than in attempt 1, suggesting

that, as expected, there is a learning effect that comes from doing multiple attempts.

Participants' preferences:

As a part of the post-questionnaire, participants were asked to choose the best and the worst conditions based on five different factors: ease of use, ease of learning, no frustration, fun, and speed of use. The only statistical difference between condition 1 and 2 was the fun factor. 75% of the participants chose condition 2 as the most fun condition out the three conditions. In addition, condition 3 was the worst condition for all of the 5 factors. Figure 18 shows the results of these questions.

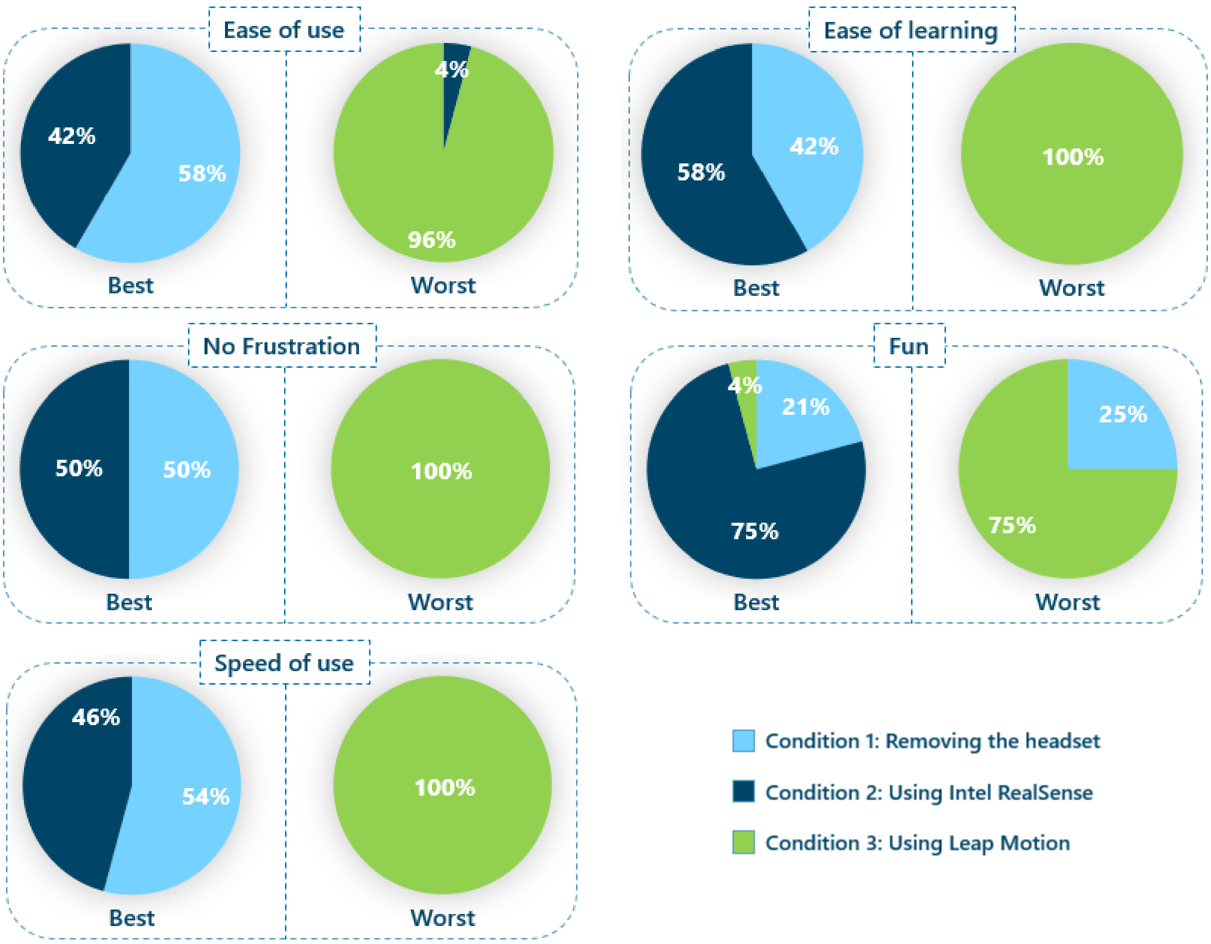

Figure 18. Users' preferences on the best and the worst conditions based on 5 different factors.

In another question in the post-questionnaire, participants were asked to choose their desired condition for each of the tasks. They were asked specifically that if they were playing a game and they had to choose one of the 3 conditions to do a certain task, what would they choose? Figure 19 shows the results of these questions.

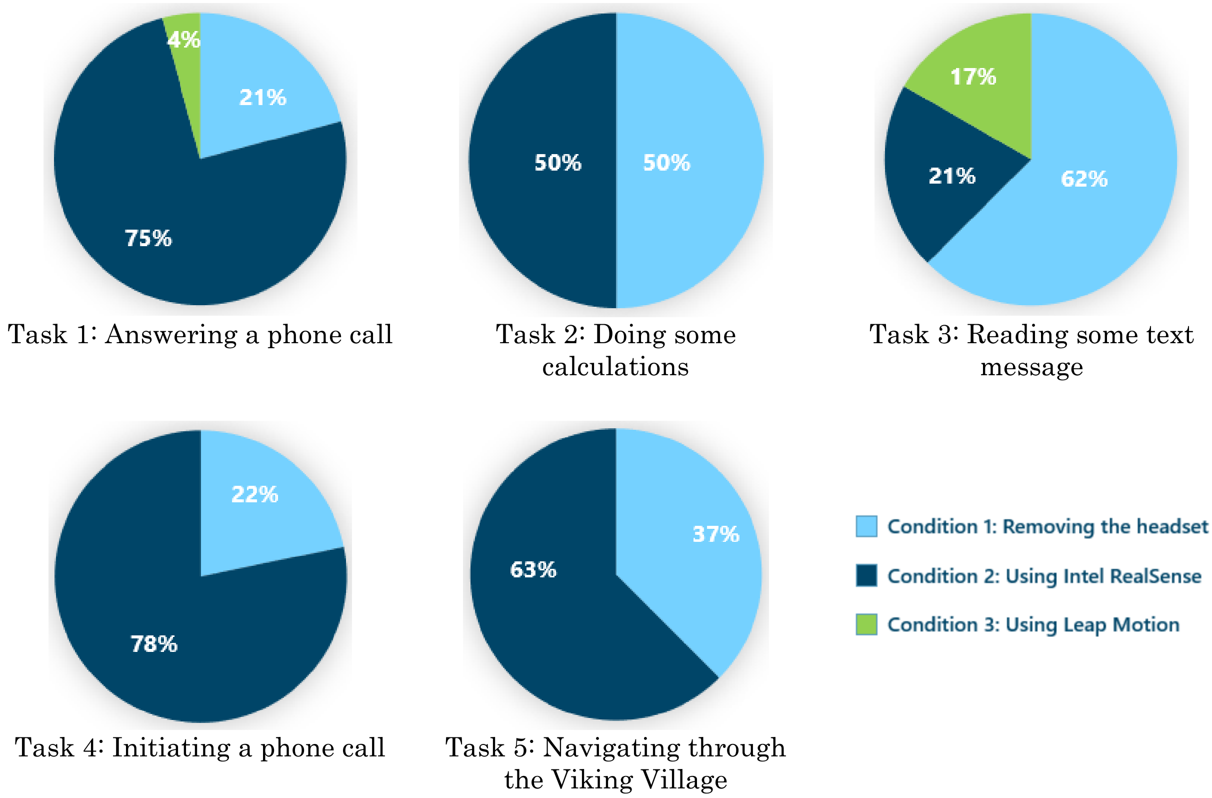

Figure 19. Users' preferences on their desired conditions for each of the tasks.

The condition 2 was preferred over the other two conditions in 3 out of 5 tasks. In Task 2, condition 1 and 2 each with 50% were equally preferred. Condition 1, with the majority of the vote, was preferred only for Task 3. Task 3 is the only task that the condition 2 was voted with less than 50% preference among the 5 tasks. For this same task, condition 3 with 17% preference reached its peak of preference among all tasks, indicating that the users found some advantage of condition 3 for this particular task.

Interview:

One of the questions that were asked of the participants during the interview was to choose the easiest and the hardest tasks considering only conditions 2 and 3. Figure 20 shows that Task 1 was considered as the easiest task and Task 2 as the hardest task. Interestingly, Task 5, navigating through the Viking Village was found the second easiest task.

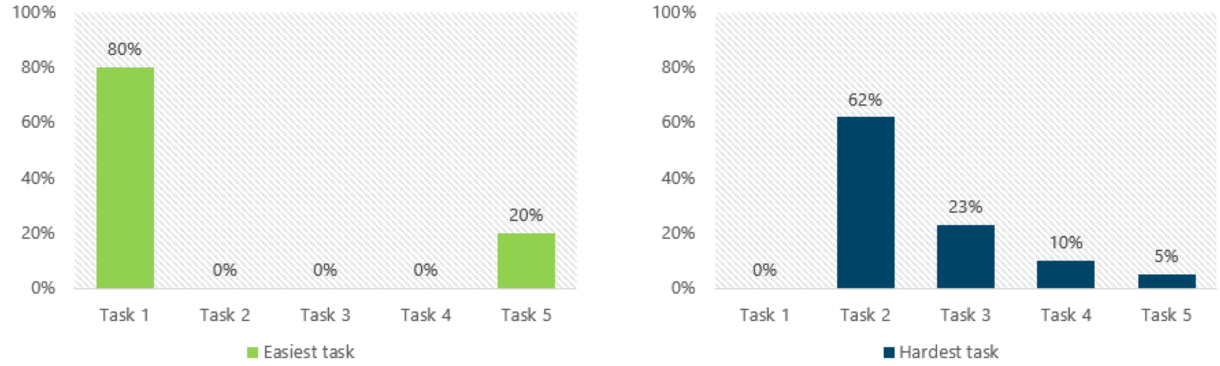

Figure 20. Easiest and hardest tasks from participants' perspective.

Task 1 was about answering a phone call and involved only one swipe up and one tap on the speaker icon on the screen. Task 5 had only four buttons or arrows on the phone's screen that the participants had to choose from to navigate through Viking Village. On the other hand, Task 2 was about doing some calculations, and it involved different digits and operators to be selected to complete the task. According to the instructions given to the participants, in only tasks 2, 3 and 4, the participants had to pick up the smartphone and launch the appropriate app from the list of applications, however, in tasks 1 and 5, there was no need to launch any apps. This can be one of the reasons that caused more work for the participants, especially in condition 3 where the participants spent more time than in the other two conditions.

During the interview, participants were asked about the reasons why they were unhappy about the augmented phone in the resulting environments. Based on the conversations and our **informal evaluation**, the following observations emerge:

- There are some major reasons behind the fact that they did not like condition 3. First, the participants could not register their fingers with what they see as the phone in the VR. Second, the delay on what participants could see as the virtualized smartphone on the screen was one of the reasons that caused more human errors and eventually spending more time on that condition. On the other hand, the legibility of the screen was the only upside of condition 3 and that is why they preferred to choose it on Task 3 with 17%.

- For two reasons, the participants did not like the condition 2. Some of them could not read the texts clearly, especially on Task 3, and some of them did not like to see small parts of background environment on the edges of the augmented smartphone and the hands due to imperfect segmentation, as shown in Figure 7.
- The blurriness of the texts in condition 2 and the difficulty of selecting and launching the apps in condition 3 could be the main reasons that some of the participants would prefer to remove the headset on some of the tasks.
- One of the things that participants had difficulty with was the navigation controller on the phone's screen. They did not consider it as intuitive as the keyboard or other gaming controllers because they had to hold it in a sweet spot, and that was uncomfortable for them if they had to continue holding it there for a long period of time like it was required in Task 5. This is natural because people do not usually hold their phone in front of their face when looking towards the front and instead look down or lower their gaze temporarily.
- The context of the tasks is one of the important factors on how the participants would react to a condition. It may not be realistic to say that the users are willing to perform all kinds of tasks in the middle of the game. It depends on the game and the importance of the tasks, and if a task is going to take twice as much time and energy as it normally should, then removing the headset, would not be considered a problem by the users.

Chapter 6: Conclusion

In this project, we have implemented an AV environment in which the HMD (an Oculus Rift DK2) was equipped with a depth-sensing camera (Intel RealSense) to address the problem of visual isolation. In this implementation, users experience near-range awareness, which helps them interact with their smartphone without removing the headset. To test whether our implementation is operable enough to replace the current state of the technology, a user study was designed with three conditions. In the user study, condition 1 was the current state of the technology, where a user would have to remove the headset to perform certain tasks, condition 2 represented our implementation, and condition 3 represented an implementation done by Desai, et al [13]. Overall, we found out that our implementation is more promising than the other implementation. The participants experienced some difficulties in both conditions and that indicates that these implementations need more improvements. However, participants preferred our implementation in 3 out of 5 tasks.

Despite the fact that the participants in condition 2 could not be as fast as they were in condition 1, condition 2 was adequately responsive, especially in a gaming environment like the Viking Village and it was considered the most fun condition among the 3 conditions. The results also suggest that for users to support such an AV environment and for an implementation to be accepted by the users, they need to be as fast and efficient as the baseline condition. Condition 2 had the least mean time difference with our baseline condition 1 in all the 5 tasks and the users had less difficulty in their interactions when compared to condition 3.

Chapter 7: Future work

Further refinements are necessary on the proposed prototypes. Based on the results, increasing the image quality and consistency in augmentation (which depends on the segmentation algorithm) for condition 2 would increase support for this solution, whereas decreasing the response time and increasing the accuracy of condition 3 would do so as well. Designing prototypes that work in different circumstances might need a better understanding of the context where the tasks are taking place. Further research should be done on what kind of tasks should be targeted for an AV environment. Finally, we plan to submit the results of this work to a journal or conference in the area of VR and HCI.

References

[1] MarketsAndMarkets's webpage, https://www.marketsandmarkets.com/Market-Reports/head-mounted-display-hmd-market-729.html, last accessed 2017/12/04

[2] XinReality's webpage, https://xinreality.com/wiki/Slide-on_HMD, last accessed 2017/12/04

[3] XinReality's webpage, https://xinreality.com/wiki/Discrete_HMD, last accessed 2017/12/04

[4] XinReality's webpage, https://xinreality.com/wiki/Integrated_HMD, last accessed 2017/12/04

[5] Basting, Oliver, Arnulph Fuhrmann, and Stefan M. Grünvogel. "The effectiveness of changing the field of view in a HMD on the perceived self-motion." *3D User Interfaces (3DUI), 2017 IEEE Symposium on*. IEEE, 2017.

[6] Milgram, Paul, et al. "Augmented reality: A class of displays on the reality-virtuality continuum." *Telemanipulator and telepresence technologies*. Vol. 2351. No. 11. 1994.

[7] Mobile Marketing Magazine's webpage, http://mobilemarketingmagazine.com/24bn-smartphone-users-in-2017-says-emarketer, last accessed 2017/12/04

[8] Bruder, Gerd, et al. "Enhancing presence in head-mounted display environments by visual body feedback using head-mounted cameras." CyberWorlds, 2009. CW'09. International Conference on. IEEE, 2009.

[9] Tecchia, Franco, et al. "I'm in VR!: using your own hands in a fully immersive MR system." Proceedings of the 20th ACM Symposium on Virtual Reality Software and Technology. ACM, 2014.

[10] Nahon, David, Geoffrey Subileau, and Benjamin Capel. "“Never Blind VR” enhancing the virtual reality headset experience with augmented virtuality." Virtual Reality (VR), 2015 IEEE. IEEE, 2015.

[11] Desai, Amit P., Pena-Castillo, Lourdes, Meruvia-Pastor, Oscar. “A Window to your Smartphone: Exploring Interaction and Communication in Immersive VR with Augmented Virtuality”

[12] Oculus Rift's Guides' webpage, https://support.oculus.com/guides/dk2/latest/concepts/ug-hardware-setup/#gsg_oneworld_use, last accessed 2017/12/04

[13] Oculus Rift's Developers' webpage, https://developer.oculus.com/downloads/package/oculus-runtime-for-windows/, last accessed 2017/12/04

[14] Intel® RealSense™ SDK 2016 R3 Documentation's web page, https://software.intel.com/sites/landingpage/realsense/camera-sdk/v2016r3/documentation/html/index.html?doc_advanced_working_with_multiple_modaliti.html, last accessed 2017/12/04

[15] Intel® RealSense™ SDK 2016 R3 Documentation's web page, https://software.intel.com/sites/landingpage/realsense/camera-sdk/v2016r3/documentation/html/index.html, last accessed 2017/12/04

[16] Intel® RealSense™ SDK 2016 R3 Documentation's web page, https://software.intel.com/sites/landingpage/realsense/camera-sdk/v2016r3/documentation/html/doc_devguide_unity__application.html, last accessed 2017/12/04

[17] Ghassem Alaee's GitHub repository's page, https://github.com/salaee/Seg3D, last accessed 2017/12/04

[18] Foley, James D., Victor L. Wallace, and Peggy Chan. "The human factors of computer graphics interaction techniques." IEEE computer Graphics and Applications 4.11 (1984): 13-48.

[19] Viking Village at Unity asset store, http://u3d.as/bqF, last accessed 2017/12/04

[20] Newman, Barbara M., and Philip R. Newman. *Development through life: A psychosocial approach*. Cengage Learning, 2017.

Appendix A – Questionnaires

Experiment Pre-Questionnaire

no. __

Please answer the following questions:

What is you age? _________

Are you Male(M) or Female(F)? _______

What is your occupation? ____________________________

Are you left handed or right handed? ___________

What brand and model of smartphone do you have? ________________

What OS does your phone have?

◯ Android ◯ iOS ◯ Windows Mobile ◯ Other (please indicate): _____________

How much time do you spend on each of the following technologies on a weekly basis:

	Never Used	Do not use regularly	1-5 hours	5-10 hours	10-20 hours	20+ hours
Using smartphones, tablets and wearables						
Online searching or shopping (e.g. google, amazon)						
Social networking (e.g. Facebook, Instagram, twitter)						
First-person style games (e.g. GTA, World of Warcraft)						
Other types of games (on smartphone or desktop pc)						
Using immersive devices for gaming (e.g. Head-mounted display)						
Using immersive devices for other purposes						

Will you be wearing any of the following during the study?

◯ Prescription Glasses

◯ Contact lenses

◯ None of the above

Experiment Post-Questionnaire

no. __

1. If you wore correction glasses during the experiment, rate the comfort wearing the VR goggles
   with correction glasses:

1 2 3 4
5

(No problem at all) *(Neutral)*
(Lots of problems)

2. Which of the technologies would you prefer to use to accomplish the following tasks during a game? (mark with an "X" each of your choices)

#	Tasks description	No VR technology (remove the headset)	1st VR technology	2nd VR technology
1	Picking up the phone and answering a call			
2	Doing some calculation on the calculator app			
3	Reading a text message			
4	Initiate a phone call by using the contact list			
5	Completing a navigation challenge			

3. From your experience during the experiment, rank the 3 technologies as best-worst based on the following factors:
   (circle **one best** and **one worst** in each table row)

Factor	No VR technology (remove the headset)	1st VR technology	2nd VR technology
Ease of use	Best Worst	Best Worst	Best Worst
Ease of learning	Best Worst	Best Worst	Best Worst
No Frustration	Best Worst	Best Worst	Best Worst
Fun	Best Worst	Best Worst	Best Worst
Speed of use	Best Worst	Best Worst	Best Worst

4. Do you have any other comments that could help us understand your answers to the previous question or any comment regarding the technologies you used in the experiment?

Appendix B – Timing and Pass/Fail Data

TASK 1

	Condition 1 - Removing the headset						Condition 2 - RealSense						Condition 3 - Leap Motion					
Participants	T1-C1-A1	Duration	T1-C1-A2	Duration	T1-C1-A3	Duration	T2-C2-A1	Duration	T1-C2-A2	Duration	T1-C2-A3	Duration	T2-C3-A1	Duration	T1-C3-A2	Duration	T1-C3-A3	Duration
P1	PASS	8	PASS	7	PASS	10	-	-	PASS	14	PASS	14	PASS	30	PASS	20	PASS	25
P2	PASS	9	PASS	14	PASS	9	PASS	27	PASS	19	PASS	8	PASS	35	PASS	13	PASS	24
P3	PASS	12	PASS	9	PASS	9	PASS	11	PASS	7	PASS	7	PASS	8	PASS	9	PASS	41
P4	PASS	11	PASS	9	PASS	7	PASS	13	PASS	8	PASS	10	PASS	14	PASS	20	PASS	59
P5	PASS	14	PASS	6	PASS	8	PASS	17	PASSS	10	PASS	7	FAIL	0	FAIL	0	FAIL	0
P6	PASS	12	PASS	11	PASS	9	PASS	18	PASS	9	PASS	12	PASS	11	PASS	8	PASS	12
P7	PASS	7	PASS	8	PASS	9	PASS	10	PASS	9	PASS	7	PASS	27	PASS	27	PASS	21
P8	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
P9	PASS	10	PASS	8	PASS	8	PASS	7	PASS	18	PASS	12	PASS	47	PASS	21	PASS	14
P10	PASS	9	PASS	6	PASS	6	PASS	7	PASS	8	PASS	20	PASS	16	PASS	9	PASS	38
P11	PASS	6	PASS	6	PASS	6	PASS	7	PASS	6	PASS	9	PASS	20	FAIL	0	FAIL	0
P12	PASS	8	PASS	7	PASS	6	PASS	8	PASS	7	PASS	7	PASS	20	PASS	18	PASS	11
P13	PASS	10	PASS	8	PASS	5	PASS	10	PASS	19	PASS	10	FAIL	0	PASS	19	PASS	43
P14	PASS	13	PASS	7	PASS	6	PASS	10	PASS	12	PASS	10	FAIL	0	PASS	40	PASS	23
P15	PASS	10	PASS	5	PASS	6	PASS	15	PASS	30	PASS	9	PASS	22	PASS	21	PASS	17
P16	PASS	7	PASS	5	PASS	4	PASS	15	PASS	7	PASS	9	PASS	33	PASS	10	PASS	10
P17	PASS	13	PASS	6	PASS	5	PASS	12	PASS	10	PASS	9	PASS	18	PASS	17	PASS	17
P18	PASS	8	PASS	7	PASS	5	PASS	8	PASS	8	PASS	7	PASS	19	PASS	15	PASS	18
P19	PASS	11	PASS	8	PASS	9	PASS	10	PASS	8	PASS	13	FAIL	0	FAIL	0	PASS	80
P20	PASS	15	PASS	9	PASS	8	PASS	11	PASS	9	PASS	9	PASS	24	PASS	21	PASS	20
P21	PASS	10	PASS	9	PASS	8	PASS	23	PASS	19	PASS	10	PASS	34	PASS	59	PASS	71
P22	PASS	8	PASS	7	PASS	8	PASS	19	PASS	10	PASS	8	PASS	15	PASS	10	PASS	9
P23	PASS	8	PASS	9	PASS	8	PASS	8	PASS	6	PASS	9	PASS	12	PASS	24	PASS	16
P24	PASS	6	PASS	8	PASS	11	PASS	4	PASS	5	PASS	5	FAIL	0	PASS	23	PASS	10
P25	PASS	10	PASS	8	PASS	16	PASS	13	PASS	11	PASS	9	PASS	12	FAIL	0	PASS	17

TASK 2

		Condition 1 - Removing the headset							Condition 2 - RealSense						Condition 3 - Leap Motion			
Participants	T2-C1-A1	Duration	T2-C1-A2	Duration	T2-C1-A3	Duration	T2-C2-A1	Duration	T2-C2-A2	Duration	T2-C2-A3	Duration	T2-C3-A1	Duration	T2-C3-A2	Duration	T2-C3-A3	Duration
P1	PASS	7	PASS	7	PASS	7	PASS	35	PASS	36	PASS	35	FAIL	0	FAIL	0	FAIL	0
P2	PASS	16	PASS	15	PASS	15	PASS	57	PASS	29	PASS	21	FAIL	0	FAIL	0	FAIL	0
P3	PASS	16	PASS	18	PASS	20	PASS	31	PASS	27	PASS	19	PASS	92	PASS	30	FAIL	0
P4	PASS	14	PASS	15	PASS	14	PASS	18	PASS	18	FAIL	0	PASS	68	PASS	44	PASS	101
P5	PASS	24	PASS	11	PASS	9	PASS	18	PASSS	21	PASS	27	FAIL	0	FAIL	0	FAIL	0
P6	PASS	13	PASS	13	PASS	11	PASS	15	PASS	14	PASS	13	FAIL	0	PASS	67	FAIL	
P7	PASS	14	PASS	14	PASS	16	PASS	24	PASS	18	PASS	24	PASS	44	PASS	40	PASS	34
P8	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
P9	PASS	26	PASS	14	PASS	18	PASS	21	PASS	16	PASS	15	PASS	65	FAIL	0	FAIL	0
P10	PASS	5	PASS	11	PASS	11	PASS	28	PASS	16	PASS	15	FAIL	0	FAIL	0	PASS	113
P11	PASS	20	PASS	12	PASS	16	PASS	63	PASS	20	PASS	62	PASS	76	FAIL	0	FAIL	0
P12	PASS	12	PASS	15	PASS	13	PASS	13	PASS	17	PASS	11	PASS	90	FAIL	0	PASS	103
P13	PASS	15	PASS	17	PASS	27	PASS	36	PASS	49	PASS	38	FAIL	0	FAIL	0	FAIL	0
P14	PASS	15	PASS	13	PASS	13	PASS	14	PASS	17	PASS	23	FAIL	0	PASS	143	PASS	75
P15	PASS	11	PASS	14	PASS	28	PASS	29	PASS	20	FAIL	0	PASS	167	FAIL	0	PASS	138
P16	PASS	16	PASS	13	PASS	14	PASS	20	PASS	24	PASS	20	FAIL	0	FAIL	0	FAIL	0
P17	PASS	11	PASS	13	PASS	11	PASS	18	PASS	17	PASS	16	FAIL	0	PASS	80	FAIL	0
P18	PASS	14	PASS	14	PASS	11	PASS	18	PASS	17	PASS	31	FAIL	0	FAIL	0	FAIL	0
P19	PASS	21	PASS	34	PASS	20	FAIL	0	PASS	40	PASS	39	PASS	90	PASS	121	PASS	87
P20	PASS	25	PASS	15	PASS	24	FAIL	0	PASS	12	PASS	32	FAIL	0	FAIL	0	FAIL	0
P21	PASS	14	PASS	17	PASS	20	FAIL	0	PASS	56	PASS	25	PASS	73	FAIL	0	PASS	158
P22	PASS	16	PASS	13	PASS	13	PASS	20	PASS	21	PASS	21	PASS	127	PASS	124	FAIL	0
P23	PASS	29	PASS	19	PASS	17	PASS	27	PASS	26	PASS	20	PASS	51	PASS	101	PASS	93
P24	PASS	13	PASS	10	PASS	13	PASS	8	PASS	15	PASS	18	-	-	-	-	-	-
P25	PASS	13	PASS	20	PASS	11	PASS	14	PASS	26	PASS	17	FAIL	0	FAIL	0	FAIL	0

TASK 3

		Condition 1 - Removing the headset							Condition 2 - RealSense						Condition 3 - Leap Motion			
Participants	T3-C1-A1	Duration	T3-C1-A2	Duration	T3-C1-A3	Duration	T2-C2-A1	Duration	T3-C2-A2	Duration	T3-C2-A3	Duration	T2-C3-A1	Duration	T3-C3-A2	Duration	T3-C3-A3	Duration
P1	PASS	6	PASS	5	PASS	8	PASS	15	PASS	61	PASS	6	PASS	35	PASS	27	PASS	40
P2	PASS	20	PASS	31	PASS	10	PASS	40	PASS	21	PASS	19	FAIL	0	FAIL	0	FAIL	0
P3	PASS	15	PASS	9	PASS	10	PASS	19	PASS	16	PASS	8	PASS	27	PASS	27	FAIL	0
P4	PASS	14	PASS	8	PASS	7	PASS	21	PASS	13	PASS	41	PASS	23	PASS	44	FAIL	0
P5	PASS	13	PASS	5	PASS	8	FAIL	0	PASSS	17	PASS	37	FAIL	0	FAIL	0	FAIL	0
P6	PASS	11	PASS	9	PASS	10	PASS	10	PASS	9	PASS	26	PASS	42	PASS	29	PASS	14
P7	PASS	10	PASS	8	PASS	6	PASS	10	PASS	13	PASS	29	PASS	29	PASS	36	PASS	20
P8	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
P9	PASS	9	PASS	9	PASS	8	PASS	41	PASS	14	FAIL	0	PASS	34	PASS	21	PASS	15
P10	PASS	11	PASS	6	PASS	8	PASS	9	PASS	12	FAIL	0	PASS	19	PASS	23	PASS	22
P11	PASS	8	PASS	8	PASS	8	FAIL	0	FAIL	0	FAIL	0	PASS	69	PASS	16	PASS	11
P12	PASS	12	PASS	7	PASS	7	PASS	16	PASS	9	PASS	18	PASS	14	PASS	16	PASS	26
P13	PASS	13	PASS	11	PASS	11	PASS	27	FAIL	0	FAIL	0	FAIL	0	PASS	69	PASS	46
P14	PASS	10	PASS	10	PASS	11	PASS	20	PASS	13	PASS	23	PASS	22	PASS	14	PASS	31
P15	PASS	13	PASS	9	PASS	9	PASS	20	PASS	21	FAIL	0	PASS	44	PASS	65	PASS	21
P16	PASS	9	PASS	7	PASS	6	PASS	27	PASS	15	FAIL	0	PASS	25	PASS	61	FAIL	0
P17	PASS	11	PASS	8	PASS	8	PASS	10	PASS	28	PASS	11	PASS	27	PASS	17	PASS	25
P18	PASS	13	PASS	9	PASS	8	PASS	14	PASS	13	PASS	9	PASS	34	PASS	19	PASS	16
P19	PASS	14	PASS	12	PASS	11	FAIL	0	FAIL	0	PASS	20	PASS	80	PASS	50	PASS	56
P20	PASS	14	PASS	12	PASS	11	PASS	28	PASS	27	PASS	30	PASS	64	PASS	18	PASS	40
P21	PASS	16	PASS	9	PASS	9	PASS	26	FAIL	0	PASS	31	PASS	88	PASS	27	PASS	65
P22	PASS	10	PASS	8	PASS	7	PASS	9	PASS	20	PASS	24	PASS	62	PASS	79	PASS	20
P23	PASS	9	PASS	12	PASS	9	PASS	24	PASS	22	PASS	29	PASS	45	PASS	23	PASS	77
P24	PASS	9	PASS	7	PASS	7	PASS	11	PASS	18	PASS	14	PASS	32	PASS	71	FAIL	0
P25	PASS	12	PASS	7	PASS	9	PASS	20	PASS	17	FAIL	0	PASS	52	PASS	79	PASS	24

TASK 4

	Condition 1 - Removing the headset						Condition 2 - RealSense						Condition 3 - Leap Motion					
Participants	T4-C1-A1	Duration	T4-C1-A2	Duration	T4-C1-A3	Duration	T2-C2-A1	Duration	T4-C2-A2	Duration	T4-C2-A3	Duration	T2-C3-A1	Duration	T4-C3-A2	Duration	T4-C3-A3	Duration
P1	PASS	20	PASS	18	PASS	19	PASS	18	PASS	18	PASS	15	FAIL	0	FAIL	0	PASS	48
P2	PASS	20	PASS	16	PASS	20	PASS	15	-	-	-	-	FAIL	0	FAIL	0	FAIL	0
P3	PASS	25	PASS	21	PASS	18	PASS	11	PASS	22	PASS	23	FAIL	0	FAIL	0	FAIL	0
P4	PASS	26	PASS	18	PASS	17	FAIL	0	PASS	23	PASS	28	PASS	86	PASS	117	PASS	50
P5	PASS	22	PASS	16	PASS	17	PASS	48	PASS	21	PASS	17	FAIL	0	FAIL	0	FAIL	0
P6	PASS	20	PASS	25	PASS	20	PASS	24	PASS	21	PASS	25	PASS	54	PASS	38	PASS	35
P7	PASS	25	PASS	18	PASS	19	PASS	30	PASS	31	PASS	26	PASS	60	PASS	80	PASS	25
P8	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
P9	PASS	23	PASS	23	PASS	20	PASS	24	PASS	25	PASS	22	PASS	25	PASS	101	PASS	134
P10	PASS	20	PASS	15	PASS	18	PASS	35	PASS	25	PASS	21	PASS	82	PASS	46	PASS	33
P11	PASS	21	PASS	20	PASS	19	PASS	34	PASS	84	PASS	29	PASS	40	PASS	87	PASS	56
P12	PASS	21	PASS	19	PASS	18	PASS	31	PASS	27	PASS	16	PASS	59	PASS	136	FAIL	0
P13	PASS	21	PASS	18	PASS	22	PASS	29	FAIL	0	PASS	32	FAIL	0	PASS	97	PASS	142
P14	PASS	25	PASS	24	PASS	21	PASS	22	PASS	29	PASS	24	PASS	45	PASS	34	PASS	57
P15	PASS	22	PASS	19	PASS	21	PASS	22	PASS	23	PASS	21	FAIL	0	FAIL	0	PASS	52
P16	PASS	27	PASS	25	PASS	21	PASS	34	PASS	20	PASS	19	FAIL	0	FAIL	0	FAIL	0
P17	PASS	20	PASS	17	PASS	17	PASS	20	PASS	21	PASS	20	PASS	71	PASS	44	PASS	42
P18	PASS	20	PASS	17	PASS	18	PASS	20	PASS	19	PASS	26	PASS	43	PASS	74	PASS	58
P19	PASS	24	PASS	24	PASS	26	PASS	28	PASS	28	PASS	25	PASS	96	FAIL	0	PASS	125
P20	PASS	22	PASS	18	PASS	20	PASS	28	PASS	49	PASS	24	PASS	129	FAIL	0	PASS	31
P21	PASS	24	PASS	22	PASS	23	PASS	26	PASS	22	PASS	31	FAIL	0	PASS	88	PASS	75
P22	PASS	18	PASS	21	PASS	20	PASS	25	PASS	25	PASS	27	PASS	92	PASS	45	FAIL	0
P23	PASS	23	PASS	20	PASS	20	PASS	26	PASS	28	PASS	31	PASS	105	PASS	89	PASS	77
P24	PASS	34	PASS	17	PASS	20	PASS	23	PASS	20	PASS	20	-	-	-	-	-	-
P25	PASS	19	PASS	19	PASS	19	PASS	22	PASS	29	PASS	27	PASS	117	PASS	87	PASS	103

TASK 5

	Condition 1 - Removing the headset						Condition 2 - RealSense						Condition 3 - Leap Motion					
Participants	**T5-C1-A1**	**Duration**	**T5-C1-A2**	**Duration**	**T5-C1-A3**	**Duration**	**T2-C2-A1**	**Duration**	**T5-C2-A2**	**Duration**	**T5-C2-A3**	**Duration**	**T2-C3-A1**	**Duration**	**T5-C3-A2**	**Duration**	**T5-C3-A3**	**Duration**
P1	-	-	-	-	-	-	PASS	60	PASS	65	PASS	58	FAIL	0	FAIL	0	PASS	82
P2	-	-	-	-	-	-	PASS	102	PASS	83	PASS	90	FAIL	0	FAIL	0	PASS	140
P3	PASS	60	PASS	55	PASS	52	PASS	75	PASS	58	PASS	75	PASS	80	PASS	105	PASS	87
P4	PASS	58	PASS	48	PASS	48	PASS	63	PASS	56	PASS	63	PASS	77	PASS	65	PASS	57
P5	PASS	64	PASS	55	PASS	60	PASS	75	PASS	67	PASS	65	FAIL	0	FAIL	0	FAIL	0
P6	PASS	71	PASS	52	PASS	50	PASS	70	PASS	67	PASS	69	PASS	80	PASS	127	PASS	127
P7	PASS	49	PASS	48	PASS	47	PASS	60	PASS	65	PASS	65	PASS	85	PASS	60	PASS	60
P8	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
P9	PASS	44	PASS	60	PASS	50	PASS	62	PASS	54	PASS	58	PASS	86	PASS	65	PASS	76
P10	PASS	52	PASS	46	PASS	47	PASS	71	PASS	60	PASS	60	PASS	90	PASS	83	PASS	60
P11	PASS	48	PASS	50	PASS	46	PASS	64	PASS	61	PASS	62	PASS	108	PASS	88	PASS	70
P12	PASS	50	PASS	48	PASS	47	PASS	66	PASS	58	PASS	60	PASS	81	PASS	75	PASS	62
P13	PASS	62	PASS	54	PASS	48	PASS	79	PASS	69	PASS	70	PASS	106	PASS	181	PASS	85
P14	PASS	69	PASS	51	PASS	57	PASS	72	PASS	69	PASS	82	PASS	104	PASS	83	PASS	69
P15	PASS	51	PASS	52	PASS	57	PASS	81	PASS	71	PASS	71	PASS	90	PASS	108	PASS	98
P16	PASS	55	PASS	48	PASS	53	PASS	77	PASS	62	PASS	64	FAIL	0	FAIL	0	FAIL	0
P17	PASS	55	PASS	58	PASS	54	PASS	78	PASS	65	PASS	65	PASS	75	PASS	86	PASS	65
P18	PASS	51	PASS	49	PASS	49	PASS	69	PASS	60	PASS	59	PASS	141	PASS	139	PASS	102
P19	PASS	57	PASS	53	PASS	50	PASS	84	PASS	74	PASS	69	FAIL	0	FAIL	0	FAIL	0
P20	PASS	84	PASS	65	PASS	67	PASS	89	PASS	65	PASS	61	PASS	209	FAIL	0	FAIL	0
P21	PASS	65	PASS	55	PASS	52	PASS	77	PASS	69	PASS	68	PASS	141	PASS	142	PASS	66
P22	PASS	44	PASS	47	PASS	44	-	-	-	-	-	-	PASS	74	PASS	73	FAIL	0
P23	PASS	56	PASS	55	PASS	55	PASS	77	PASS	97	PASS	75	PASS	100	PASS	137	PASS	132
P24	PASS	56	PASS	48	PASS	49	PASS	67	PASS	65	PASS	65	-	-	-	-	-	-
P25	PASS	60	PASS	52	PASS	59	PASS	73	PASS	66	PASS	74	PASS	98	PASS	117	PASS	67

Appendix C – Analysis

Tukey Honest Significant Differences: Pairwise comparison of the mean times of the conditions

$Task1
Tukey multiple comparisons of means
95% family-wise confidence level

Fit: aov(formula = Time ~ Condition + Attempt, data = tmp)

condition	diff	lwr	upr	p adj
C2-C1	2.56964	-0.9357271	6.075007	0.1961957
C3-C1	14.78506	11.1378974	18.432230	0.0000000
C3-C2	12.21542	8.5564963	15.874351	0.0000000

$Task2
Tukey multiple comparisons of means
95% family-wise confidence level

Fit: aov(formula = Time ~ Condition + Attempt, data = tmp)

condition	diff	lwr	upr	p adj
C2-C1	9.142677	2.15384	16.13151	0.0065372
C3-C1	74.079981	65.06014	83.09982	0.0000000
C3-C2	64.937304	55.80050	74.07411	0.0000000

$Task3
Tukey multiple comparisons of means
95% family-wise confidence level

Fit: aov(formula = Time ~ Condition + Attempt, data = tmp)

condition	diff	lwr	upr	p adj
C2-C1	10.32895	4.761729	15.89617	5.79e-05
C3-C1	27.24727	21.782877	32.71166	0.00e+00
C3-C2	16.91832	11.133555	22.70309	0.00e+00

$Task4
Tukey multiple comparisons of means
95% family-wise confidence level

Fit: aov(formula = Time ~ Condition + Attempt, data = tmp)

condition	diff	lwr	upr	p adj
C2-C1	5.169118	-1.786102	12.12434	0.1874221
C3-C1	51.928191	44.215147	59.64124	0.0000000
C3-C2	46.759074	38.956946	54.56120	0.0000000

$Task5
Tukey multiple comparisons of means

95% family-wise confidence level

Fit: aov(formula = Time ~ Condition + Attempt, data = tmp)

condition	diff	lwr	upr	p adj
C2-C1	15.25494	7.899912	22.60997	6.2e-06
C3-C1	41.74414	33.865022	49.62326	0.0e+00
C3-C2	26.48920	18.686741	34.29166	0.0e+00

Attempt comparison in Task 5:

Attempt	diff	lwr	upr	p adj
A2-A1	-4.321397	-11.96332	3.320530	0.3769955
A3-A1	-9.253968	-16.86527	-1.642670	0.0125810
A3-A2	-4.932571	-12.57450	2.709356	0.2815123