Comparison between Developing a Mobile Virtual Reality Game and a Desktop Application

Yueying Liu
College of Information and Computer Sciences
yueyingliu@umass.edu
11/26/2017

Advisor: Timothy Richards
Second Committee Member: Shane Mecklenburger
Research Type: Project

1. Introduction

The goal of this project has three components:

On the technical level, the goal is to develop a 3D mobile game on the Android platform using a game engine, Unity. Google Cardboard will be used to create VR experience with a mobile phone. Research should be completed on using Unity, understanding the mobile platform, enhancing real-time rendering, and game design.

On the artistic level, this project aims to develop 3D modeling and design skills so as to give the players the best gaming experience. The art of camera angle, 3D model design, lighting, texturing, icon design and component arrangements will be studied and reviewed during playtest.

Last but not least, the research component will focus on comparing and contrasting the writing of a non-game application and the writing of a game application as a programmer, and the experience difference for players.

The hypotheses for the differences are: 1. for programmers, their design focus would for a game would largely depend on the user experience, whereas for desktop applications the design focus would be on achieving desired functionalities. 2. for the users, a VR game player need to use more parts of the brain than a desktop application user. This may make their user-experience easier or harder depending on each individual. 3. as VR is more similar to the reality, the user may find it easier navigating the VR application than navigating a desktop application which abstracts many things.

The project would be developed using Unity as the game engine and ideally on both Android and Apple operating systems. Building the game and user testing are as important to ensure user experience.

At first a runnable prototype of the game shall be developed for preliminary playtest. The subsequent refinement of the game shall be done considering the user feedback from the playtesting. Throughout this process, the similarities and differences between developing traditional desktop programs and developing mobile VR game shall be noted for future analysis.

Developing VR technology, including hardware, software and tricks in user experience, through games is common. Actually gaming industry is expected to drive the development of VR as how it drove the development of graphic displays in the past.

A variety of hardwares and softwares are developed to support VR these days. Some may become the future, some may never be shown to the consumers. Of all these, mobile VR is kind of “known” to survive this technological revolution. However, due to the versatility of mobile phones, its user experience is innately not as good as specialized devices. Many technological and design tricks must be deployed to reduce the complexity of the program so as to ensure user experience, especially avoiding cybersickness which is caused by any inconsistency of the human balance system with the VR visuals.

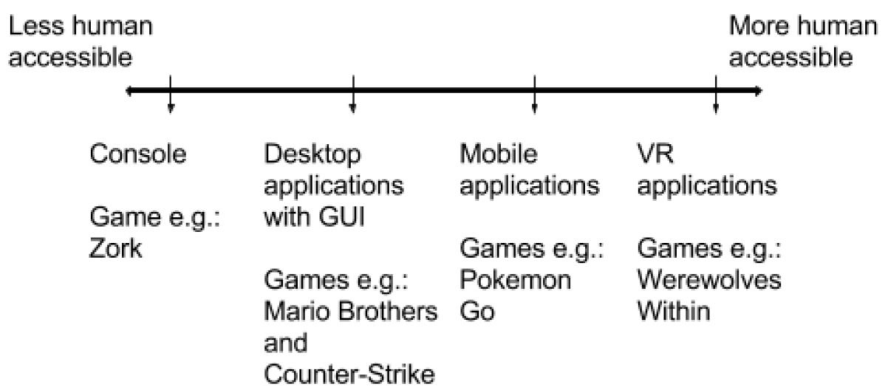
In my project, I will explore some techniques and designs to improve user experience. Due to the novelty of the VR field, there has yet been any systematic writing on user experience for mobile VR. By studying the similarities and differences with developing traditional desktop applications and my project, I can not only prove or disprove my hypotheses, but also provide some guidance on how to make the mobile VR experience better. The experience in developing

VR can be applied to many different fields, like room decoration prototypes, pilot simulation, etc.

2. Background

It is a trend that computer programs are becoming more accessible to humans. At the very beginning, computer programs were executed using the console - people type in a command which is usually hard to understand and prone to errors, hit the enter key, and get the results. Consoles are still frequently used among programmers nowadays. This kind of programs are the most inaccessible for humans as the users need to memorize the computer commands in order to communicate with the computer. Gradually, with the introduction of Graphical User Interface(GUI) since 1963, desktop programs, whether running on a web browser or on its own as an executable program, transformed from text-only to using graphics and icons, which make visualization easier and can be understood by people speaking different languages. As the measurements of hardware running computer programs become smaller, mobile applications become popular, and their market is still expanding rapidly. Mobile applications increase the human accessibility by its increased portability and the ability to sense the direction and orientation of the device with accelerometers. Nevertheless, the above-mentioned programs all require the user to view a two-dimensional(2D) interface from the three-dimensional(3D) world. Although with years of using the existing GUI, we are used to this form of abstraction, the most accessible programs for humans should be viewing from 3D to 3D. Virtual Reality(VR), whose foundation already laid down in 1838[3] but developed slowly through the years due to hardware constraints, is one of the hottest research area in computer science[4] and aims to allow the users to view the contents of the program in 3D directly. This is achieved by simulating how the 3D world is projected onto human retinas and interpreted into 3D setting by the brain: The user puts on a VR headset that displays two slightly different images, which belong to the same scene but viewed from the angles that simulate what the two eyes see, in front of the left and right eyes separately. The two images are called stereoscopic images. Human brains interpret the two images into a 3D setting.

Computer game, a subcategory of computer programs, follows the trend, too. Early console games like *Zork* have the same human-computer interactions as any other console programs. With GUI and a mouse which can locate any point on the screen, desktop games have a large variety, ranging from 2D games like *Mario Brothers* which abstract the world onto a 2D plane, to 3D model games like *Counter-Strike* which project a 3D world onto a 2D screen. The multitouch screen and accelerometers of mobile devices give the players more natural and direct control of the games. *Pokemon Go*, for example, utilizes the geographical location and its change of the player on the earth to simulate the player moving in the world similar to the earth but has additional gaming components. Again, all these games are limited by the hardware that even if the game creates a 3D world, the player could only see the 2D projection of the world. VR games, on the other hand, allows the players to immerse into a virtual 3D world just like they are in the real world. For example the players of the game *Werewolves Within* would find themselves gathered around a campfire in the medieval town of Gallowston. Standing-up, turn their heads around, lying down would directly change the view in the game. They just need to move as per normal to trigger the changes in the virtual world.



Spectrum of Computer Programs and Example of Games

The most avant-garde types of computer program today are VR and mobile, due to their novelty. Actually the term VR was coined only in 1987[5], and the first mobile application was released only in early 90's[6]. Being such new and blossom research areas, there are many things to learn and explore.

Currently the most prominent challenge for VR is ensuring fast real-time rendering and high resolution display, which contradict each other. In other words, when a person enters a virtual world, the person expects to see the world as clear and smooth as the real world. To see things clearly requires all objects in the virtual world to have high resolutions. To see things smoothly, the hardware needs to finish calculating the next frame to be displayed for both eyes within 1/60 seconds which is the frame rate for most of the games. The higher resolution of the objects, the slower the calculation. Every VR project face this challenge, and need to find the balancing point based on the content and purpose of the project.

Coincidentally, mobile development research is leaning towards VR and something slightly different: Augmented Reality(AR) which is displaying on the screen the reality with computer-generated data superimposed. For example, having a pair of rabbit ears during video chat. AR has the same calculation-resolution dilemma as VR. The dilemma is intensified by the dilemma for any mobile devices: having better hardware or having smaller sizes.

Technologies are developed to solve this problem. Various game engines are developed optimising the displays on respective target devices. Unity developed by Google is targeted at mobile devices which paired with their head-mounted devices could be used to have VR experiences. Maya developed by Autodesk has many similar functionalities on modeling and rendering as that on Unity, with just more sophisticated features. The models made in Maya is compatible in Unity. It is the software I choose for modeling.

3. Methodology

In principle, the artistic designing and modeling will precede the programming, so it may provide data and testing later on in the project. Small and simple models will be built and imported into the game project for making a mini version of the game with the basic functionalities realized. After the codes successfully run on the desktop in the game engine, the game will be packaged and downloaded into a phone so as to test the game using the Google Cardboard. Google Cardboard (which is a cheap headset that holds a phone in front of the eyes and physically block the sight from the left eye to the right half of the screen and vice versa), paired with VR Cinema for Cardboard (which is a VR mode converter that splits the screen in landscape position into two smaller screens on the left and right which displays the stereoscopic images) will be used to test the VR experience of the game. Adjustments shall be made until the VR experience is optimized for the smaller version of the game. More detailed designs and other functionalities will be gradually added to the game until the technology required is too much for an honors project, or the hardware cannot handle the file sizes.

Throughout the process of building this project, similarities and differences between writing a non-game application and a game application will be noted. Attention shall be paid to the comparison between game engines and non-game development environments, unique adaptations to mobile hardware, and attention to user experience.

The Art Department at UMass Amherst provides computers with great performance for graphics intensive applications. 3D modeling tools like Maya are already installed in these computers, and free student version can be downloaded from Autodesk website. There is no need to purchase software for modeling. Art 374, Introduction to Computer Animation is the course that teaches 3D modeling using Maya, in preparation for the 3D animation production in the second half of the two-semester course sequence. I am taking this course this semester. I will be proficient in building 3D models mid-semester.

The learning of Unity, VR Cinema for Cardboard (the VR mode converter) and Android will depend on the Internet. There are plenty of video and word tutorials online, made by both the tool developers and experts in using certain tools. The VR in Cardboard on Android devices tutorial page on Unity website (https://unity3d.com/learn/tutorials/s/virtual-reality) is a great place to get started. NurFACE GAMES (https://www.patreon.com/nurfacegames) provides more sophisticated tutorials for Unity VR development. Online forums are also great places to ask questions. The Unity AR/VR Forum (https://forum.unity.com/forums/ar-vr-xr-discussion.80/), for example, is the official forum for Unity AR/VR development. Engineers who developed the tool would answer constructive questions in the forum. By searching others' posts, I can find answers to many of my questions.

Once a viable prototype has been developed, the playtest shall be conducted to ensure the game is user-friendly. As college students are the target audience of the game, UMass students will be invited to test the game. They will be given a Google Cardboard and a phone with the test version of the game installed, and play the game following the instructions in the game, with no help from the game developer. After the playtest for the VR game is done, testers will be invited to play the desktop console game *Zork*. They will play on a website which simulates the console environment. They will be surveyed on the differences between a mobile VR game and desktop console game. The survey results will be used to study the user experience differences between mobile VR games and desktop console games.

Testers' feedback on the VR game shall be analyzed during each round of testing and will be used as reference for any changes and improvements. Two major rounds of testing are expected and would happen in the Spring semester.

A journal will be kept for writing the final thesis and also to assist in the creative process. Every step will be recorded in the journal and will include the following: the tools and methods used, what they will be used for, the reasoning behind the choices or designs, the intended results, the actual results, and a following analysis. This will ensure that every decision will be justified and the journal shall act as a formal record for future decision making regarding this project.

4. Evaluation

1. Finish use cases

   a. Write a report on all the use cases to be developed. Discuss the use cases during a meeting

2. Finish maze design
   a. Draw out the designs and discuss during a meeting
3. Finish a preliminary game prototype according to the use cases
   a. Bring the game prototype to a meeting to get feedback
4. Finish game map decoration
   a. Present the 3D model of the map during a meeting to get feedback
5. Finish a runnable game prototype for user testing
   a. The prototype shall fulfill all the use cases, does not crash, and with all VR functionalities enabled
   b. Present the prototype in a meeting
6. Draft user testing questionnaire and feedback forms. Justify the knowledge elicitation method
   a. Write a report for the thoughts behind the questionnaire and feedback forms
   b. Discuss in a meeting
7. Conduct the first round of user testing
   a. Finish a user testing analysis
   b. Present the user testing analysis in a meeting
8. Modify the game based on the user feedback
   a. Present the modified game in a meeting
9. Finish a draft on analyzing the similarities and differences between developing a traditional desktop applications and developing a mobile VR game
   a. Present the draft in a meeting before oral defense
10. Oral defense
11. Final submission of the codes and Project manuscript

The milestones will be evaluated based on the presentation in meeting or the deliverable submitted. Each milestone is preferably weighed the same but subject to changes. Committee members provide feedbacks during meetings. The codes of the project will be saved in a DVD for submission.

5. Communication

Meeting with Timothy Richards, the Committee Chair shall occur biweekly for a half hour. The expectations of these meetings are to report progress, discuss solutions to problems, make changes to the project plan if needed, and become of aware of any difficulties. Meeting with Shane Mecklenburger, the other Committee Member shall occur monthly for a half hour. The expectations is to evaluate the aesthetic aspect of the project and discuss improvements. The expected commitment of work towards will be ten hours per week between meetings.

6. Timeline

- 01/26: Finish use cases
- 02/02: Finish maze design
- 02/09: Finish a preliminary game prototype
- 02/16: Finish game map decoration
- 03/01: First game prototype ready for playtest. Finish playtest plan
- 03/09: Finish playtest report and analysis
- 03/23: Finish modified game prototype
- 04/13: First draft of thesis submitted to advisor
- 04/20: Second draft of thesis submitted to HDP
- 04/20: Oral Defence (subject to change)
- 04/30: Final Submission to CHC

7. References

1. Anthes, C., Hernandez, R., & Kranzlmuller, D. (n.d.). State of the Art of Virtual Reality Technologies. doi:10.1109/AERO.2016.7500674
2. Learn - Modules. (n.d.). Retrieved December 04, 2017, from https://unity3d.com/learn/tutorials
3. Autodesk. (n.d.). Getting Started. Retrieved December 04, 2017, from https://knowledge.autodesk.com/support/maya/getting-started?sort=score
4. GoogleVR Guide. (n.d.). Retrieved December 04, 2017, from https://developers.google.com/vr/unity/guide
5. Möller, T., Haines, E., & Hoffman, N. (2008). *Real-time rendering*. Wellesley, MA: A.K. Peters.