Tech Review Document
Team 20: xRLucid

AUSTIN LIANG, CHRIS COOPER, DAVID OKUBO, JONATHAN CHEN,
MINGYU ZHANG
Oregon State University
November 10, 2018

Abstract

In this document, Team 20 also known as xRLucid will be introducing the ideas behind how positioning will be considered when developing the Mixed reality for Infrastructure Maintenance. Specifically targeting and mentioning topics such as 3D Rendering, BIM, Toolkits to handle 3D rendering, API for 3D rendering, and the possible conversion of BIM to mobile 3D engine. These topics will be further explained and defined through out this document.

Keywords: BIM, 3D Rendering, Vulkan, OpenGL, Metal, API

1

Contents

1   Introduction

With the computer science industry quickly expanding in every possible way, one of the hot topics that are frequently talked about within the field is virtual reality, and augmented reality. A new idea briefly explored is mixing both of these realities together forming into something known as mixed reality, which is basically virtual reality with some augmentations included into the software.

According to realitytechnologies.com, the definition of mixed reality is to classify the spectrum of realities that in technology.= [1]. Overall is a fancy way of saying that it is a mix between in this case virtual reality and augmented reality. In context to this project, team 20 will be implementing a plan and design that will create a mixed reality to allow simulation of construction maintenance.

2   Toolkit for 3D Rendering

According to Medium.com, the top 15 best tools for 3D Rendering software available currently are TinkerCAD, 3D Slash, Voxel Builder, MagicaVoxel, Autodesk123d, Sculptris, Sketchup, Onshape, Fusion360, SolidWorks, Blender, Cinema 4D, Maya, 3ds Max, and Z Brush [2].

With these tools to our selection, Team 20 has decided to brainstorm together and select one of these free methods and learn how to work the software. Once learning how to operate the software we will work together to develop 3D renderings to maintenance grounds and apply them there. To further understand what the process to expect when using these software, I will explain the general idea behind each software.

All of the 3D rendering tools that we will need are all time consuming and takes a lot of experience to manipulate and use. I personally have experience with Blender and Autocad,and from this experience I have learned that both software are hard to create 3D models if you do not follow the tutorials. Also it is very easy to get lost and confused if you do not have a plan or design to follow when using these software.

3D rendering will need to operated with sensors, GPS, cameras, and others. Rendering is the ability to process generated images given using a camera, many generated images that should be detected and picked up are objects, light sources, and much more. With these tools we are using, this will allow us to manipulate and work with 3D objects in augmented reality.

In context to our tools that we are using, we are going to use them to visually create 3D objects for construction maintenance purposes such as if we are developing a bridge we would use the GPS to find our position

and location to the bridge and then we would be able to use the camera to point at the bridge and generate 3D models that will appear on our tablet. With these 3D models, our software will be able to distinguish where ever the location the user is in. For example, if we wanted the overlay and the entire bridge to fit into the frame of the tablet then we would have a 3D model created for the entire bridge but have small separate segments. If the user would like to look at what small materials are used to stabilize and fix the bridge, all they would need to do is walk closer to the 3D model object that is already created for the bridge and look at the tablet to check what tools and materials were used for that specific section of the bridge. This ability will be enabled when the users position is close enough to where the bridges position range between the GPS will be small enough to change and alter the visual display into a more detailed virtual picture. This will enable to user to be able to click on which materials are used to work on the bridge. According to Matthias Teschner, a professor of the Computer Science department of the University of Freiburg states that rendering is possible through many options such as pipeline rendering, vertex, and others [1].

These options allow our team to understand what hardware and positioning works when using the tablet to operate our software. The team is still brainstorming possible solutions and options of working out the topic of positioning using rendering and 3D objects. So far BIM and autodesk are the best options to creating digital and virtual objects for our software.

3   APIs for 3D Rendering

APIs are crucial and important to 3D rendering pipeline, it may be one of the hardest problems team 20 is facing. Understanding how we will deliver signals and messages efficiently from the tablet that stores the users position to the API which will deliver that signal and data to a storage which contains the bridges position to calculate the range between the two and alternate perspectives the user may want. To be even more specific, the software is going to allow the user to have an interface where there is an entire layout on the bridge and if the user is interested, they can approach the bridge and change their positions to have a different point of view and perspective on how materials are used on different spots of the bridge. The API will be the way for our data to be sent back and forth to determine user position and bridges position. Once figuring the perspective options, we can determine which options the user may want from the user interface. Some options that our software will offer is the ability to ask the user if they would like to view from another point of view or select which viewing options they want to see the tools and materials that are used to maintain the bridge.

Having the API options to know how which rendering options the user wants is going to allow the user to choose options. Some options that we may consider including are OpenGL, Mantle, OpenGL ES, or Glide, these are all 3D rendering APIs that would allow the user to choose whichever option they may desire. When selecting an API for 3D rendering there is not a default choice at the moment, however as the software is slowly getting planned out and finalized. Some options for which APIs we are considering are Vulkan, OpenGL, and Metal.

3.1 Vulkan

When analyzing the VUlkan website for the API, I instantly noticed that there multiple downloadables with an array of options to download from. These options Vulkan offer are different download versions of the software. I also noticed that Vulkan was created by AMD, a prestigious and well known company that are known for their advancements in computer science and software development. According to AMD, Vulkan is the only open source standard and cross platform API with high graphics and applications [3]. With this mentioned they also state that Vulkan is revolutionary to games and 3D development. This software is one of the most considered options in using the API because it offers so much to our needs in implementation. Another important thing Vulkan offers us is the ability to be cross platform which allows us to further advance this project from strictly using android mobile phones to potentially apple products and computers.

3.2 OpenGL

The initial impression that I had when analyzing the API website for OpenGL, I noticed the website was very well put and organized. The website listed in an orderly fashion for what OpenGL is and a few different versions of downloadable files. According to OpenGL, this software follows the standards of the combination of virtual reality and augmented reality [4]. With this notice, team 20 is still considering this as the main software to use for the APIs for the mixed reality project. Some additional features on this website is that they present news information and updates on their developer end.

3.3 Metal

Metal is a low level API developed by Apple. According to Apple, Metal is a render advanced 3D graphics and perform data-parallel computations using graphics processors [5]. With this description, team 20 has decided that this may also be a very good selection and option for including as the

API for our project. When using this software, it would be easier to gain experience because Apple products tend to have more reviews and tutorials online. Therefore, would be very helpful in understanding the software for implementation.

4   Conclusion

In conclusion, there are many options for which selections of software we can approach with but for the most part we will settle with either Metal or Vulkan because of the potential it has in the long run. With a major selection for which software option to take, we are still at the beginning stages of testing out each option to find the best and most efficient option for implementation and creating precision in 3D models and positioning perspective that will help us determine how the tablet will be able to observe and send data.

References

1. https://www.realitytechnologies.com/mixed reality/. Reality Technologies. *Journal title*, November 2018.

2. https://medium.com/@ngai.yt/15-best-tools-for-3d-modeling-software a7939fded451/. Medium. *Journal title*, November 2018.

3. https://www.amd.com/en/technologies/vulkan/. Vulkan. *Journal title*, November 2018.

4. https://www.opengl.org/. Opengl. *Journal title*, November 2018.

5. https://developer.apple.com/documentation/metal. Metal. *Journal title*, November 2018.