National Conference on Advancements in Computer Science and Engineering
In association with International Journal of Scientific Research in Science, Engineering and Technology
Print ISSN: 2395-1990 | Online ISSN : 2394-4099 (www.ijsrset.com)

FoodDroid AR : A Wholesome Approach to Your Food Ordering Experience

Varun Rajesh Makhija[1], Vishnu Vardhan Vemuru[1], Shreyas Chandrashekar[1], Dr. Pamela Vinitha Eric[2]
[1]Department of Computer Science and Engineering, New Horizon College of Engineering, Bangalore, Karnataka, India
[2]Assistant Professor, Department of Computer Science and Engineering, New Horizon College of Engineering, Bangalore, Karnataka, India

ABSTRACT

Augmented Reality creates an experience using the user's real-time environment that allows them to interact with it. Today's rapidly developing technologies have made smartphones, laptops, and wearable devices more accessible, thus enabling AR-based applications.

Keywords — Augmented Reality, 3D Models, FoodDroid AR, AR Core, Photogrammetry

I. INTRODUCTION

This paper will focus on two things, the first being the problem that FoodDroid AR attempts to solve, and the second being how it solves the problem.

The food ordering and delivery industry has consistently been growing over the years; however, the customer experience hasn't evolved much since the advent of such services. While ordering, people tend to do either of two things: one, they order food items they are already familiar with, or two, they order new (From the customer's perspective) food items on an experimental basis by judging the items in question via a picture (If it exists) or a description (If it exists). We are going to focus on the second aspect wherein many times, you would find neither a picture nor a description for some food items, and upon ordering them, you may not be satisfied with either the food item itself or the quantity of the food item that you received. FoodDroid AR aims to solve this problem using the marvel that is Augmented Reality. By making use of 3D rendered models of the food items available within food ordering and delivery app services, FoodDroid AR allows the user to view any food item they desire in the form of a fully rendered 3D model and can also choose to project said model onto their real-time environment by making use of their Android phone's camera system. This would help the user interact with their order more efficiently but also help them gauge the quantity of food that would be delivered to them.

II. REQUIREMENTS

A. Hardware Requirements:

Android mobile device: An ARCore supported Android Nougat 7.0 with API level 24 or higher is required to run Scene Viewer to display AR models that are generated. The AR models are displayed within the environment using the mobile camera, which is accessed by Scene Viewer.

GPU: A CUDA-enabled Nvidia GPU is required for Meshroom to generate AR models.

B. Software Requirements:

1) ARCore: Google's platform for building augmented reality experiences. ARCore enables your phone to sense its environment, understand the world and interact with information. ARCore uses three capabilities to allow virtual content to interact with the real world as specified in [1]. They are:
   - Motion tracking allows the phone to understand and track its position relative to the world.
   - Environmental Understanding allows the phone to detect the size and location of all types of surfaces: horizontal, vertical, and angled surfaces like the ground, a coffee table, or walls.
   - Light Estimation allows the phone to estimate the environment's current lighting conditions.
2) Java: Java is a general-purpose class-based, object-oriented programming language that is intended to let application developers write once and run anywhere while having as few implementation dependencies as possible. The language is computer architecture-free, and the applications created are typically compiled to bytecode that can run on any Java Virtual Machine as mentioned in [2].
3) Meshroom: Meshroom is a software developed by the AliceVision Association that uses a collection of images to create 3D models using a technique called Photogrammetry. In order to perform said task, Meshroom requires an NVIDIA Cuda-enabled GPU with CUDA-10 compatibility and compute capability between 3.0 to 7.5.
4) Scene Viewer: An immersive viewer used by an Android application to facilitate 3D and AR experiences. Many Google partners have implemented it to support AR reliably. FoodDroid AR uses Scene Viewer. to allow the food AR models created on Meshroom. Details about Scene Viewer are in [3].
5) Android Studio: It is the official integrated development environment for Google's Android Operating system. It was jointly developed by JetBrains and Google. The IDE supports Kotlin, but here the preferred language of choice is Java.

III. MAKING THE 3D MODELS

Making 3D models can be an extremely complex process; however, using Photogrammetry technology, we were able to create 3D models of food from scratch. Photogrammetry is used to create a map or 3D model of real-world objects or scenes from a collection of photographs of said object/scene. We used AliceVision's Meshroom, a software specifically designed to render 3D Models from photographs using Photogrammetry. An example of using Meshroom for photogrammetry can be found in [4]. The process of creating a 3D model of a food item, as followed for FoodDroid AR, can be broken down into x steps. They are as follows:

A. The Photography Environment:

1) The room in which you click the photographs must be as clutter-free as possible. This means that you keep only the essential items in the room: no unnecessary real-world objects or scenes must be there as this will confuse the software while building the model.
2) An ideal case scenario would be a room with a single table for the food item to be placed and absolutely nothing else in the room. If the room has white walls and flooring, that will only enhance the quality of the model being rendered.

B. The Environment Lighting:

1) Once the room has been set up, it is imperative that the lighting is even across the object from all angles.
2) The ideal case would be to have a powerful light source on top of the table that would light up the object appropriately.

C. Clicking Pictures of the subject:

1) Meshroom supports almost any smartphone camera, as most smartphones encode picture metadata within the photographs themselves.
2) Hence, Meshroom can identify things like the camera aperture for depth of field, the ISO for picture granularity, and the shutter speed for sharpness.
3) It is essential to avoid any motion blur and depth blur, which is why it is ideal to use a tripod stand for the smartphone.
4) Finally, use the smartphone camera to click at least 60 pictures of the subject by going 360 degrees around the subject and getting pictures from eye level as well as above-eye level of the subject.
5) The more the pictures, the more details will be rendered in the 3D model.

D. Importing the pictures into Meshroom:

1) The photographs need to be imported into Meshroom, which is a pretty straightforward process: you can drag and drop.
2) Once imported, save the project as a Meshroom file and head over to the workflow.

E. Rendering the 3D Model:

1) You can alter any settings in the workflow if you would like, concerning MeshFiltering, Meshing, etc.
2) In order to begin rendering the model, head over to the Texturing tab and right-click and click compute.
3) The colours on each tab in the workflow indicate the state of the computation. Blue signifies that it has been submitted for computation, Red indicates an error, Orange indicates the computation is going on, Green indicates that the computation is completed.

F. Exporting the results:

1) The final model will be a Textured Mesh that will be saved as an OBJ file with the corresponding MTL and Texture files.

G. Converting to GLTF 2.0:

1) Since SceneViewer can only display 3D models in GLB/GLTF 2.0 format, we must convert the OBJ file to the required format.
2) Any online 3D Converter can convert the OBJ file and corresponding MTL and Texture files to a single GLTF 2.0 file.
3) This file can then be hosted to display in the FoodDroid AR app.

IV. DISPLAYING MODELS IN AN AR ENVIRONMENT

Scene Viewer from Google's ARCore is used to display the models in an AR environment. The main reason for choosing Java as the preferred language of development instead of Kotlin is because Scene Viewer can only be used in Java. Scene Viewer requires the Google Play Services for AR in order to display the models in AR. Scene Viewer has certain strict model requirements, and they are as follows:

- The file format must be GLB or GLTF 2.0.
- The model size should not be over 10 MB.
- It is recommended not to bake shadows for models as Scene Viewer creates hard shadows for the models.
- The maximum texture resolution must be 2048x2048.

In FoodDroid AR, all the AR models used are hosted on GitHub so they can be loaded by Scene Viewer using a link provided to the model's raw content.

Scene Viewer is launched using an explicit intent on Android. It requires three different intent parameters as follows:

- A file URI must be passed in order to load the AR model into Scene Viewer.
- A title for the model should also be passed.
- A mode parameter must also be specified. The mode parameter allows Scene Viewer to determine which mode the model must be loaded in. The modes are as follows:

A. 3D preferred:
Displays models in a 3D viewer by default but has a button specified to allow AR mode.

B. 3D only:
Displays models in a 3D viewer only and does not allow an AR viewer.

C. AR preferred:
Displays models in an AR viewer but has a button specified to allow 3D mode.

D. AR only:
Displays models in an AR viewer only and does not allow a 3D viewer. This mode is generally discouraged as a mobile may not support AR mode.

The mode used in FoodDroid AR is 3D preferred since it is safer to load a 3D viewer in case a mobile phone does not support AR. The model can then be viewed in AR mode by clicking the view in your space button. Scene Viewer allows resizing, moving, and zooming into the models using finger gestures to increase the user experience. Optional parameters like sound, link, and resizable can also be used but are not used in this project.

V. CONCLUSION

The food ordering and delivery industry is an evergreen one that is constantly growing. In order to make a mark in the vast scope of this industry, one is required to give more value to the user with innovative solutions. While AR is yet to enter the food industry completely, a few restaurants have employed an AR menu experience for diners. The process to generate models for each of the food items in a food delivery app is a tedious process, but the right amount of workforce can turn it into a much smoother process. The results we saw in our application proved to represent the original food item quite well, although it does not appear exactly as in real life. Eventually, the aim should be to make the food items look as authentic as possible and make them available as a possible feature in food delivery applications.

A. Benefits:
1) FoodDroid AR is a food ordering app that allows you to view food items in a 3D format.
2) FoodDroid AR can also project these 3D models in real-time environments.
3) FoodDroid AR aims at reducing complaints of any sort and thus improving customer satisfaction.
4) 3D models projected by the android system AR core are used to show the customers exactly what they pay for or how much quantity they get.

B. Disadvantages:
1) Due to the limitations of pre-existing android smartphones, not all of them can render and view 3D models.
2) Rendering and projecting 3D models need a stable internet connection and suitable system hardware.
3) To be able to build 3D models, you require powerful software, which may not be cost-effective.

VI. REFERENCES

[1]. "ARCore overview | Google Developers", Google Developers, 2021. [Online]. Available: real-world objects or sceneshttps://developers.google.com/ar/discover. [Accessed: 08- May- 2021].

[2]. H.Schildt, “Java: The Complete Reference, Eleventh Edition,” McGraw Hill.

[3]. "Using Scene Viewer to display interactive 3D models in AR from an Android app or browser", Google Developers, 2021. [Online]. Available: https://developers.google.com/ar/develop/java/scene-viewer. [Accessed: 08- May- 2021].

[4]. "Tutorial: Meshroom for Beginners — Meshroom v2021.0.1 documentation", Meshroom-manual.readthedocs.io,2021. [Online]. Available: https://meshroom-manual.readthedocs.io/en/latest/tutorials/sketchfab/sketchfab.html. [Accessed: 10- May- 2021].