Room Scanner

An application to show if a room is available.

Ron Gebauer Maximilian Walter

2016-01-10

Scanning a QR-Code with room information with a device is possible using the Vuzix M100. Even though the developer community of Vuzix is small and the access is limited, the development for Android with OpenGL ES resulted in a suitable proof of concept for doing so.

1 Introduction

Problems occur when a docent tries to find a free room for the next hours. The information whether a room is available or not can not be seen directly at the desired place. In this project the human eye will be extended with information about the environment. This will have done using the Vuzix M100 Smart Glasses. When scanning a QR-Code that contains the room number, information about the availability of the room is displayed on the Vuzix and with that into the environment.

2 Technologies Used

Several technologies for virtual and augmented reality are available. All these technologies have different positive aspects and drawbacks.

2.1 Google CardBoard

The initial idea was to use Googles CardBoard for displaying the Element that indicates wether a room is available. Before the start of the development, it occured that Google CardBoard is good for virtual reality where there is no connection to the outside world. Augmented reality, where still a part of the *real world* is hardly possible with Google CardBoards since the camera is one sided and no overlaying is possible.

Another drawback of the CardBoard is the fact that both eyes are needed to see the 3D Elements.

2.2 Vuzix M100

The vuzix M100 Smart Glasses can be used for augmented reality. Using one eye for the *real World* and one eye for the application, it is possible to display additional information to the users enviroment.

2.3 Android and OpenGL ES

The operating system on the smart glasses is Android. Apps are developed

in Java. OpenGL Elements are displayed using the OpenGL ES technique.

The Android SDK and the IDE Eclipse are needed to develop for Android.

The Vuzix OS 2, which is based on Android is limited due to the

3 Architecture

The architecture relies on mainly three classes. Despite the *MainActivity* that starts the internal *ScannerApp* that reads the QR-Code, The *TimeTableReader* class is needed to give information about the occupation of the rooms. Finaly the *RoomScannerRenderActivity* with a *RoomScannerRenderer* that display the information in form of OpenGL elements on the screen.

The *TimeTableReader* class that reads information from the TimeTable API has been mocked to provide a realistic datas set.

4 Discussion

The current project is a prototype that reads a QR-Code and based on that displays two different forms. Getting access to the real timetable is needed to put this application to use. Furthermore it is adviseable to extend the Renderer to display information not only about the current state of the room but also to give information about the next hours, days and weeks.

The lack of suitable, up-to-date libraries for augmented reality resulted in the fact that there is no augmented reality but only a Android Device that shows an OpenGL Element.

5 Conclusion

The use case of the project does not offer much possibilites to extense the skills learned in the OpenGL Course. Even though OpenGL elements are used within the application, the hardest part was to the get the Vuzix and its emulator uptodate. Information is rare and a big developer community does not exist for this kind of hardware.
Luckily Vuzix runs Android applications, which can be emulated with every other device. That is why the first steps took place using the smartphone phone emulator provided by Eclipse.

5.1 Max

- researched different technologies
- worked on the inital project structure
- created *ElementFree*, the element that is displayed when the room is free.

Problems occured in setting up the Vuzix for development. After the reorganisation of the Vuzix-Website not all information have been accessicable. Since the OpenGL ES technqiue is very simular to the JOGL and OpenGL exercises in the module *OpenGL*, the drawing itself caused no problem,depite the fact that rendering an image as texture is not possible. Since this was no requirement, no image is used in the current state of the application. One negative aspect was the fact, that the emulator does not work, therefore the complete developer team had to rely on one Vuzix. This slowed down the development process.

A big problem was the limited time that was given for the project. With more time, a nicer in more detailed solution would have been produced.

5.2 Ron

TODO

Appendix

needed? QR-Codes ? or class diagram?