The Implementation of an Augmented Reality Display and a Virtual Reality Controller on Glasses

Jiuhe Tian
2022 - 2023
Princeton International School of Mathematics and Science
General Engineering

Description

A near-eye augmented reality display with simple user control using lens and ESP. A combination of augmented reality display and virtual reality control is to be designed.

Content

Project Goals and Significance

Augmented Reality (AR) is a technology that enhances a user's perception of the real world by overlaying digital information on top of it. Some of the key features of AR include the ability to interact with digital content in real-time, and the ability to overlay digital content on top of the real world in a way that enhances the user's understanding of the environment around them. But nowadays, all AR sets have some disadvantages. For example, Nreal's Air Glasses just project a device screen in front of people's eyes, and C-Thru's AR helmet for firefighters relies on the real environment to generate the scene, it cannot create extra contents.

For Virtual Reality (VR) control, there are different ways like controllers and body tracking. In this project, users will use a controller to control the scene projected on AR glasses. To track the position of VR controllers, people have to set up at least two cameras to calculate the controller's position, which means current VR can just be realized in a closed space.

In other words, current AR have freedom of movement, but cannot generate a completely virtual world. People can act as what they want in VR, but they are restricted in a closed area.

This project will find a way to combine AR's and VR's advantages together by using offline position tracking on a near-eye projection.

An example of projection-based AR glass. Information is projected on the glasses, and the user is able to see the real through at the same time. From URL: https://www.stambol.com/2020/09/28/ar-glasses-for-consumer-enterprise-users/

Objectives & Approach

The first objective is to successfully display the graphical information from a server (laptop) on a large scale model. It can be in a single color, but it will be better if it is in multiple colors. To approach this objective, I will use a projector, lenses, and a thin, transparent board to project my laptop screen on the board. I will build this model on an optical table. When a clear image is formed on the screen, the first objective is achieved.

The second objective is to build a scale-downed model for the glass. The proper scale of this objective is in the size of regular glasses, and weigh at most 250g. This requires a tiny projector. I will order a half-hand-sized projector online to complete the object. To hold the structure, I will use a 3D printer to print some parts and a laser cutter to cut some plywood to build the frame. The sign of success to this objective is I can see a clear projection on the glasses when the projector is connected to the server.

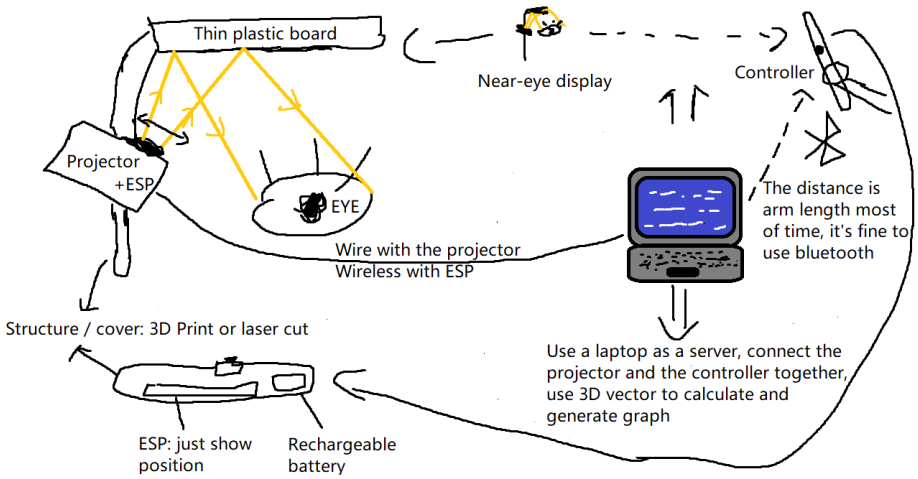
The third objective is to know the position and the angle the projector is facing. A piece of ESP and an accelerometer will be used here. Attach the ESP and accelerometer with the frame, then connect the ESP with the server wirelessly, and receive the data from the accelerometer. The acceptable lag should be less than 0.1 seconds, and I'll start the display with 60 FPS, a resolution of 1920 $\times$ 1080, the same with my laptop. Considering the effective range and situation, I'll set the limit of rotational speed to 166.7 rpm and acceleration to ±8g. The third objective is accomplished when the correct displacement and rotation is shown on the server.

The fourth objective is to build a simple user controller. Do the same thing as the second and the third objectives. The only difference is to build a cylinder container which contains ESP and accelerometer inside. Success condition is still showing the displacement and rotation on the server.

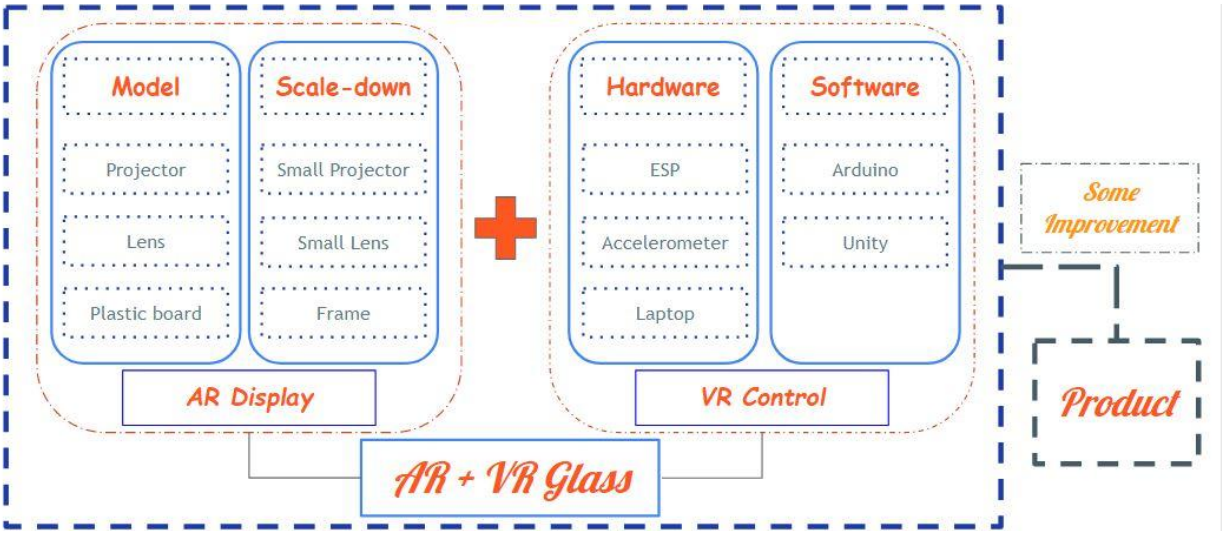
The fifth objective is to create a simple scene to give user feedback when he is moving head or the controller around. I will use Unity to create a new scene with only a colored cylinder and a camera. The data from two accelerometers before will be implemented in the Unity objects. The data received from the two ESPs should be in different ports. I'll write a C# script to deal with it. The camera shows what the user should see, and the cylinder should be accurately in the position of the controller. (excursion less than 1 mm) The goal of this phase is to allow users to both see what's in the real world and the colored cylinder on the controller.

The sixth objective is to build another set of controllers and glasses, which makes 2 different users see the models in different positions. This objective will enable multi-user to use the same scene at the same time, which means different sets of equipment will interact with each other. The approach of this objective is just repeating the steps before.

Device

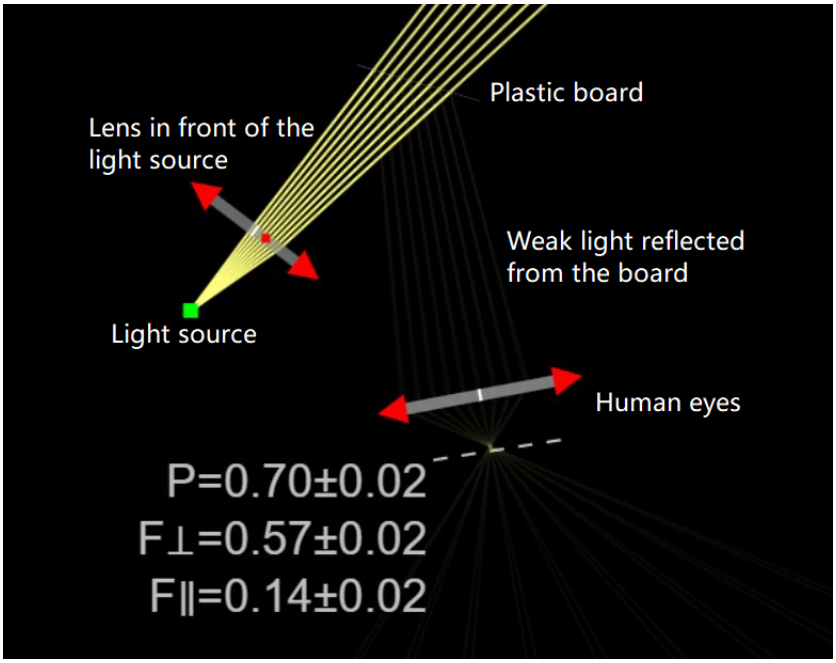


Operations



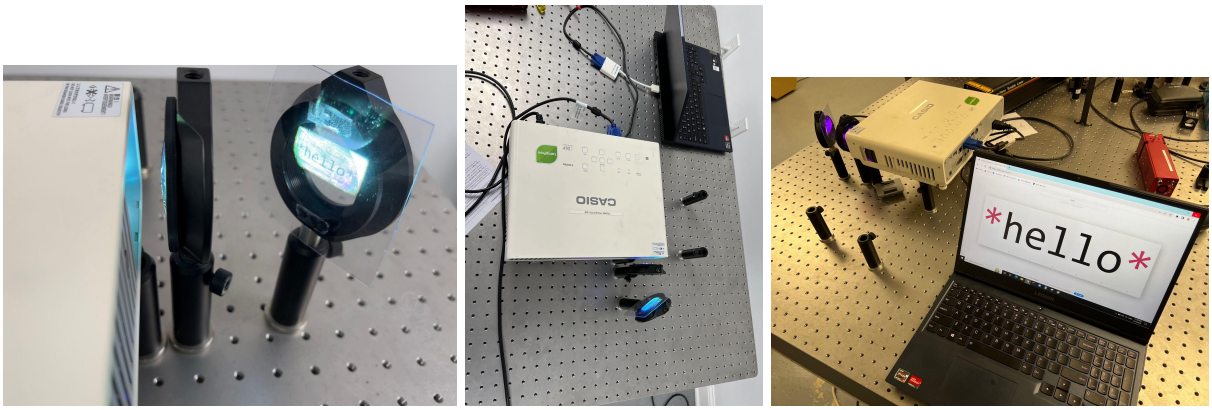
Methods

Method 1: Geometrical optics calculations. The reflection and refraction of light on the lens and plastic board will be simulated.



The design of the first objective. There's a piece of lens in front of the projector to collect the light. Then some of the light is reflected back from the second glass (it is not obvious from the screenshot). The weak light on the screen will have a less chance to harm human eyes. At last, reflected light goes into an eye-like structure to show what the figure looks like in human eyes. Simulator url: https://ricktu288.github.io/ray-optics/simulator/

Product Model:



The photos show the actual model for method 1, the laptop screen is displaying a large word "hello", the projector's ray is reflected from the plastic board, and goes into the camera. Large text url: https://large-type.com/#*hello*

Method 2: Calculate displacement x from acceleration a with integration.

$$v = \int a dt, \qquad (1)$$

$$x = \int v dt, \qquad (2)$$

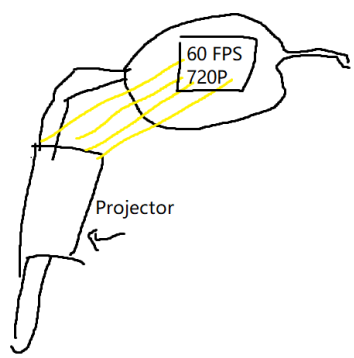
When $\Delta t \to 0$, we can express (1) and (2) as

$$\lim_{\Delta t \to 0} \sum_{i=0}^{n} \frac{1}{2} \overline{a}_i \Delta t^2 \qquad (3)$$

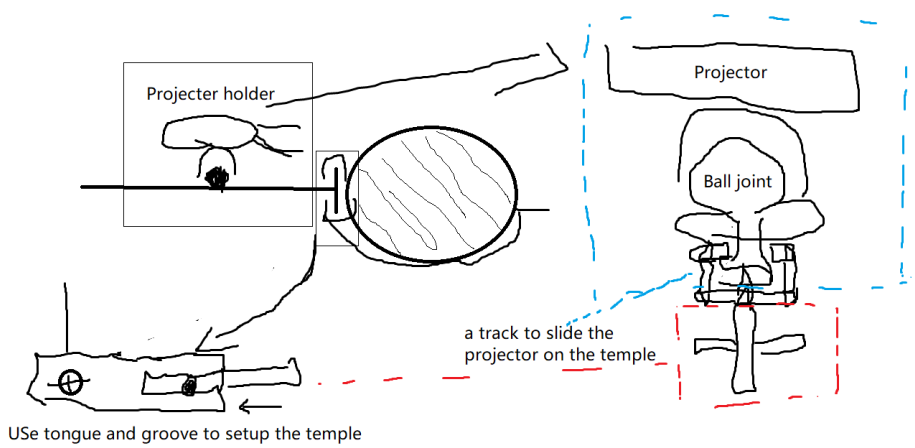
$$where\ \overline{a}_i = \frac{a_i + a_{i+1}}{2}$$

With formula (3), the displacement information can be calculated from the acceleration data sent from the accelerometer.

Method 3: Build a frame for the projector, the lens, and the plastic board.

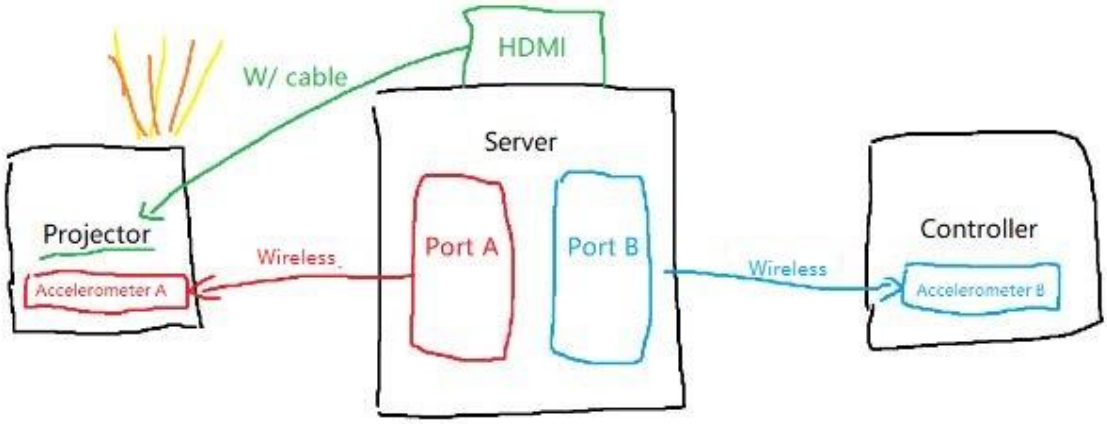


Rough frame design



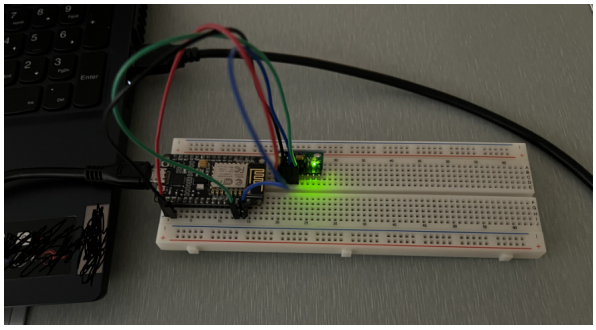
Detailed design

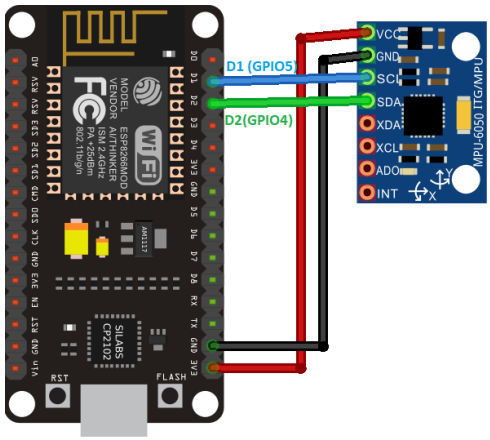
Method 4: Receive data from two esps with different ports.



As the graph shows above, the two accelerometers A and B will connect wirelessly with the server on different ports (port A and port B), which do not disturb each other. This feature enables a server to interact with different clients.

Method 5: Connect the accelerometer with ESP8266.





The connection of the ESP8266 with the MPU6050 accelerometer is VCC to 3V3, GND to GND, SCL to D1, and SDA to D2. The photo on right is from url https://randomnerdtutorials.com/esp8266-nodemcu-mpu-6050-accelerometer-gyroscope-arduino/

References

1. Seokil Moon, Chang-Kun Lee, Seung-Woo Nam, Changwon Jang, Gun-Yeal Lee , Wontaek Seo, Geeyoung Sung, Hong-Seok Lee & Byoungho Lee. (2019). Augmented reality near-eye display using Pancharatnam-Berry phase lenses. Nature, 9:6616.
2. Wikipedia, Augmented Reality. https://en.wikipedia.org/wiki/Augmented_reality
3. Wikipedia, Virtual Reality. https://en.wikipedia.org/wiki/Virtual_reality
4. Stambol, AR Glasses for Consumer & Enterprise Users. 2020. https://www.stambol.com/2020/09/28/ar-glasses-for-consumer-enterprise-users/
5. Nreal Air's AR Glass Amazon shopping page, https://www.amazon.com/Glasses-Massive-Micro-OLED-Augmented-iOS-Consoles-Compatible/dp/B0BF5LKP5Q/
6. C-Thru's AR helmet for firefighters from CBS mornings, https://www.youtube.com/watch?v=D-t7h6hukiA
7. Ray Optics Simulation, https://ricktu288.github.io/ray-optics/simulator/
8. Large Type, https://large-type.com/#*hello*
9. Random Nerd Tutorial for MPU6050 Accelerometer setup, https://randomnerdtutorials.com/esp8266-nodemcu-mpu-6050-accelerometer-gyroscope-arduino/