A Collaborative Augmented Reality Platform for Interactive and Immersive Education

Owen Sweeny Zhanchen Dong Bin Li
School of EECS, Pennsylvania State University, University Park, Pennsylvania, USA
{oms5141, zbd5062, binli}@psu.edu

***Abstract*—Augmented reality (AR) integrates virtual objects with the physical world. This technology allows us to create immersive experiences that can enhance our lives, especially through education. In combination with networking, we are able to create collaborative AR experiences which can be used to enhance the learning experience for students or people of any age. The goal of this demonstration is to show one of these systems in action. The purpose is to show students how we are able to calculate the Astronomical Unit (AU) by using the position of Venus relative to Earth and the Sun. We use AR Foundation to realize basic AR functions and rendering. The server for this multi-user demonstration is the teacher, where the teacher is able to change different parameters of the demo, which will appear on the user's device. This demonstration is made for mobile devices to reach the broadest audience. We measure the frame rates of multiple users and the round-trip time between the server and users. The frame rate is about 60 frames per second during the whole lifetime of the demo, providing a smooth and stable visual performance. The round-trip time increases from 13ms to 24ms as the number of connected users grows, which means the communication latency is less than 10ms most of the time.**

I. Introduction

Augmented reality (AR) can have many applications in an education setting for children. One of the most attractive applications of AR is that it can give students a more visual intuition for subjects that might not be as understandable. This is an especially interesting idea for teaching concepts that would be difficult for students to visualize, such as learning about astronomy [1].

The purpose of this demo was to show how a teacher and a group of students could interact with one demo. The instructor is able to control the demo using their mobile device while the students are able to view the demo on their own desks. This demo covers the topic of measuring the Astronomical Unit (AU), which is defined as the distance between the Earth and the Sun. This can be done by finding the distance between the Earth and Venus when Venus is at its point of greatest elongation, where a right triangle is formed between the Sun, Earth, and Venus [2] as shown in Figure 1. After this, students will be able to make a simple calculation using trigonometry to find the distance between the Sun and the Earth.

Multi-user AR does not come without its own set of challenges, however. One of the main issues with this technology would be scalability. With a larger classroom size, it may be more difficult for a network to handle such traffic without significant slowdowns. Another challenge is synchronization, where the difference between what each student can see on their device should not differ too much.

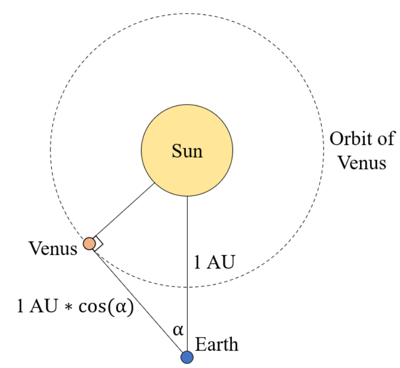


Fig. 1: Solar System Model

II. System Architecture

In our design, we have a server that is represented as the teacher for our demo. The clients for the server are the students. In this server-client architecture, a very classic and flexible architecture, especially in a multi-user game [3], the server will send data about the demo to the students across a router. This data has to do with the position and the time parameters of the solar system in the demonstration. Next, we will go into detail about the role of the user and server in this system.

Users: Each user will have their own version of the solar system with all adjustments to the demonstration hidden from them. They can place their virtual demo in their own physical space and adjust the scale of the demo to their preference. We use ARFoundation's AR Plane Manager [4] to detect surfaces in the physical world. After detecting a plane in the direct line of sight of the mobile camera, the demo can then be placed on it. The user will be able to join the server that was created by the instructor to adjust its local model. The users will then receive information about the demo from the server.

Server: The instructor chooses to create a room for the demo. This will attempt to create a TCP connection to all clients using the IP address of the instructor. Once their virtual demo is placed, a client list will be created through which information about the server's demo will be sent to each user. This information includes the position of each component of the demo and the time scale of the demonstration. When a change is made to the server's demonstration, an update will be sent to each user regarding the demonstration.

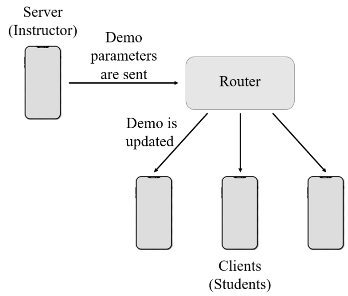


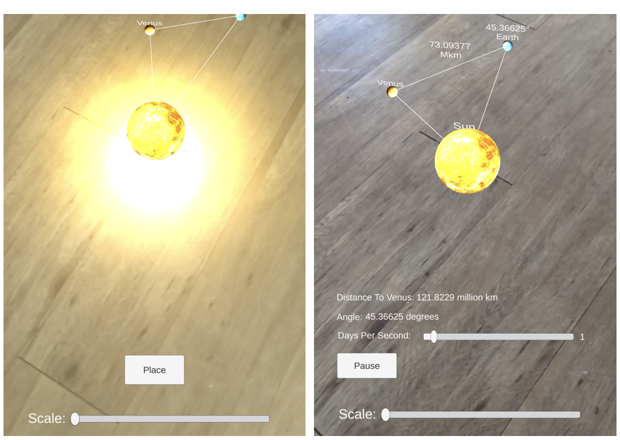
Fig. 2: System Architecture

III. Demonstration

In our system, we deploy the server on a Dell Precision 7920 workstation with Intel Xeon Gold 6242R CPU @ 3.10Ghz, 128 GB memory, and running Windows 10 Enterprise. All users are deployed on commercial Android phones, including 15x Google Pixel 6. Each Google Pixel 6 displays Sun, Earth, and Venus. The demo video is available in [5].

In Fig. 3, we show the placing of the virtual model in the physical space with Fig. 3a. In Fig. 3b, we show the user interface for the instructor. The instructor is able to change the time scaling as shown with the Days per Second slider. The instructor is also able to pause the demo using the Pause button below the slider. Finally, the instructor is able to change the scaling of the local demo using the Scale slider below the Pause button.

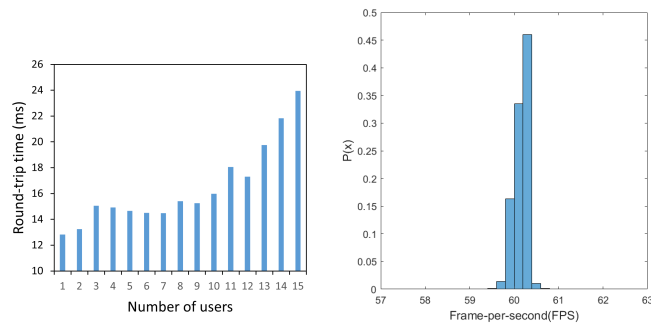
Since we use TCP as the communication standard, to examine the responsiveness of the system, we measured the round-trip time from the server to the user and back to the server. We measured the communication delay every 100 frames for the server where the server is set to run at approximately 100 frames per second. We conducted fifteen tests, with each having a different number of users, starting with one user and letting each test run for 30 seconds. We also tested the frame rate for the users to ensure that the rendering of the demo was smooth and to facilitate immersive collaboration.



(a) Placement of demo (b) Demo user interface for instructor

Fig. 3: Demonstration

The evaluation of our system is shown in Figure 4. From Fig. 4a, we can observe that the round-trip time roughly increases with the number of users, with the average for one user being approximately 13ms and the average delay for fifteen users being approximately 24ms. This is because users have to compete for wireless access and thus more users result in a larger access time. The delay performance that does not concur with this intuition in some cases is probably due to the wireless channel fluctuation. Another possibility is that there was significant delays when initially connecting the client to the server which causes increased round-trip times for the first few seconds. From Fig. 4b, we can observe that the average frame rate is around 60fps.



(a) Round-trip time comparison (b) PDF of frame rate

Fig. 4: Performance evaluation

IV. Conclusion

In this paper, we develop a collaborative AR-based educational system for calculating the AU. In the system, the server communicates with multiple users through a router by using TCP as the communication standard. We have one instructor who acts as the server sending position and time scale information to the users of the application. Users make use of the received information to calculate the AU and render virtual objects through ARFoundation. We then show that the number of users causes the latency to increase slightly from 13ms to 24ms. The frame rate is very stable around 60fps.

V. Acknowledgements

This work has been supported in part by NSF under the grants CNS-2221897 and CNS-2152610.

References

[1] K. Lee, "Augmented reality in education and training," *TechTrends*, vol. 56, no. 2, pp. 13–21, Mar 2012. [Online]. Available: https://doi.org/10.1007/s11528-012-0559-3

[2] "How do you measure the distance between earth and the sun?" [Online]. Available: http://curious.astro.cornell.edu/about-us/41-our-solar-system/the-earth/orbit/87-how-do-you-measure-the-distance-between-earth-and-the-sun-intermediate

[3] M. C. Angelides and H. Agius, *Synchronization in Multiplayer Online Games*, 2014, pp. 175–196.

[4] "Ar plane manager: Ar foundation: 4.0.12." [Online]. Available: https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@4.0/manual/plane-manager.html

[5] "Video demonstration." [Online]. Available: https://pennstateoffice365-my.sharepoint.com/:f:/g/personal/oms5141_psu_edu/EmK7hknVKe9Cr0Mj0dAcaQoBd-K-7poN-fOfLoOITXyPUw?e=BCxk1F