The Development and Usability of an Extended Reality Aircraft Engine Curriculum

Albie Morrison

College of Aeronautics and Engineering, Kent State University

Mentored by Dr. Stephanie Fussell and Dr. Benjamin Kwasa

July 1, 2024

Abstract

Extended Reality (XR, Augmented Reality or Virtual Reality) applications can make a powerful learning aid and supplement to a traditional curriculum. For subjects such as aircraft propulsion, physical copies of reference for aircraft engines are too large, impractical, and expensive to reasonably use in most educational institutions. In these cases, AR/VR can be used as a source of interaction with the material.

To better understand the benefits and limitations of this solution, we have prototyped a basic AR aircraft engine training curriculum using two different AR development platforms: Scope AR and Unity. We then began testing of the Scope AR application on a variety of common AR-capable devices including mobile devices and AR/VR headsets. For this testing, we used an AR/VR usability heuristic checklist to determine strong and weak points of the Scope AR platform, benefits and disadvantages of each device tested, and a preliminary outlook on the usefulness of this curriculum for education on aircraft engines. Although Scope AR is very limited in its capabilities, we found that it enables relatively simple development and easy compatibility with a variety of commonly used mobile devices.

Introduction

As fields like design, marketing, manufacturing, and education become more complex, the minimum competencies required to be productive are rising as well. This means that education for many fields takes more time, more energy, and more money, and therefore is less accessible and desirable for students (Lucache, 2023). We believe this is where extended reality (AR/VR) techniques can be incredibly valuable as a supplement to traditional curricula (Kaviyaraj, 2021) which involve lectures, textbooks, and often labs, (Sheng, 2014). The benefit of AR/VR learning is that it can provide users with interaction and experiences that would normally be impossible to participate in for safety, practicality, or cost reasons. This can increase understanding, material retention, student motivation, and collaboration (Brenner, 2017).

Specifically, at the Kent State University College of Aeronautics and Engineering, extended reality could be a powerful learning aid to students in our propulsion class. In this class, students learn extensively about aircraft engines, which are too large and expensive to have on-site. So we aim to answer the question: can AR/VR interaction be an effective curriculum supplement for training on aircraft engines? Exploring that question this summer involved developing building blocks of the curriculum itself, then testing our product for usability on common AR/VR-capable devices.

Development

In this section, we will overview the design and development process for creating AR content in Scope AR and Unity.

To create augmented or virtual reality environments and content, developers need to have 3D models of any virtual element they want to display to the users. Getting this content proved to be a challenge for our particular project of a jet engine training curriculum.

For most commercially used jet engines, the complete 3D models for each type of engine are not available because the engine designs are proprietary. To solve this problem, we used some scale models of similar engines that were readily available as collectors' items, and built 3D models of them using an online computer aided design (CAD) system called OnShape. This process was quite tedious and took significant effort, but was deemed to be the most accessible way to move the project forward in a relatively short time.

AR Development

Once we had 3D content, the next step was to move that content into various AR/VR development platforms and determine the difficulties and methods available for each platform. We used two platforms, called Scope AR and Unity. Scope AR is exclusively meant for AR projects, and aims to make it simple to author augmented reality experiences. Unity is a much more flexible and complex game engine with support and resources for almost all types of extended reality hardware, but it is significantly more difficult to learn and use.

Scope AR

The benefit of Scope AR's platform is that it makes content creation fairly simple, with little to no programming or setup. But the major trade off is that Scope AR tends to be inflexible and not customizable for different use cases. The structure of Scope AR projects makes it easy to create projects where the users "flow" through the program in a fairly scripted, linear manner.

The first step of authoring in Scope AR was importing the 3D models and assets. This was much harder than necessary in Scope AR, partially because it is a completely web-based platform that is run in a browser. In addition to difficulties with moving the assets, they also took a very long time to process on Scope AR's servers to become usable. Second, we had to build the model back up from the individual parts in Scope AR because the platform does not have support for CAD assemblies. Third was the content generation for the prototype curriculum, which is where Scope AR really stood out. The WorkLink platform made it extremely simple to add textual content, as well as images and videos, to each step. And fourth was loading it onto devices. While the device support was heavily limited with Scope AR, it did work “out of the box” on the devices they do support.

Overall, Scope AR’s platform was simple to use, but there were many features that we intend to add to our curriculum which make Scope AR unrealistic for much more than a prototype curriculum. These include considerations like accessibility, user settings, data security, and device compatibility. On the whole, Scope AR was a good platform to introduce developers to AR authoring and create simple projects, but is not a full-featured AR development engine.

Unity

From the outset, the difference between Unity and Scope AR was clear; it was much harder to get results in Unity because of its flexibility and extensive configuration. Unlike in Scope AR, every part of the user’s experience is changeable, which also means Unity requires programming in C#. It also means there are many fewer pre-built UI and functionality templates.

The first major challenge was finding the appropriate packages to use for extended reality applications with Unity. For each device, different packages were needed for the target operating system. Second was the configuration; getting even one device to work as intended took significant time and effort. And third, as before mentioned, Unity is very controllable, but that comes at the price of developing much of your own behavior for virtual objects.

In summary, Unity gives the developer the freedom and ability to create almost any project they can think of, but it can be challenging to configure for each device. There are so many different options and so much flexibility in the platform that it can be detrimental to those just beginning to learn the platform and those attempting new use cases for the game engine.

Methods

This summer, we had time to perform one major testing session. Two participants tested each of 3 devices with the developed initial Scope AR curriculum: an iPad, a small iPhone, and the OnePlus Pad. These were chosen because the iPad was a good baseline, the iPhone gave insight into the usability of a small device, and the OnePlus Pad was an Android device to compare against the Apple operating system on the others.

Our testing made use of a set of usability heuristics for AR and VR applications and devices (Derby, 2023), which was developed by Dr. Jessyca Derby and Dr. Barbara Chaparro from Embry-Riddle Aeronautical University. A heuristic checklist like this is not meant to give the user “hard data,” but rather it is a collection of “rules of thumb,” which can help developers to see areas in which their application needs improvement and areas in which it excels. The heuristics are organized into 12 categories and 109 total items, each of which is rated with either Yes, Somewhat, No, or N/A.

Results

After completing the ratings for each of the three devices, we performed some basic analysis of the results, presented below. The charts on the left are aggregates of the total scores from the entire heuristic checklist for each device, and the charts on the right are scores grouped by each of the 12 heuristic categories. In each chart there are two sets of data (on the left the inner and outer rings, on the right the left and right bars of each category) that correspond to the results of each of the two participants in the test session.

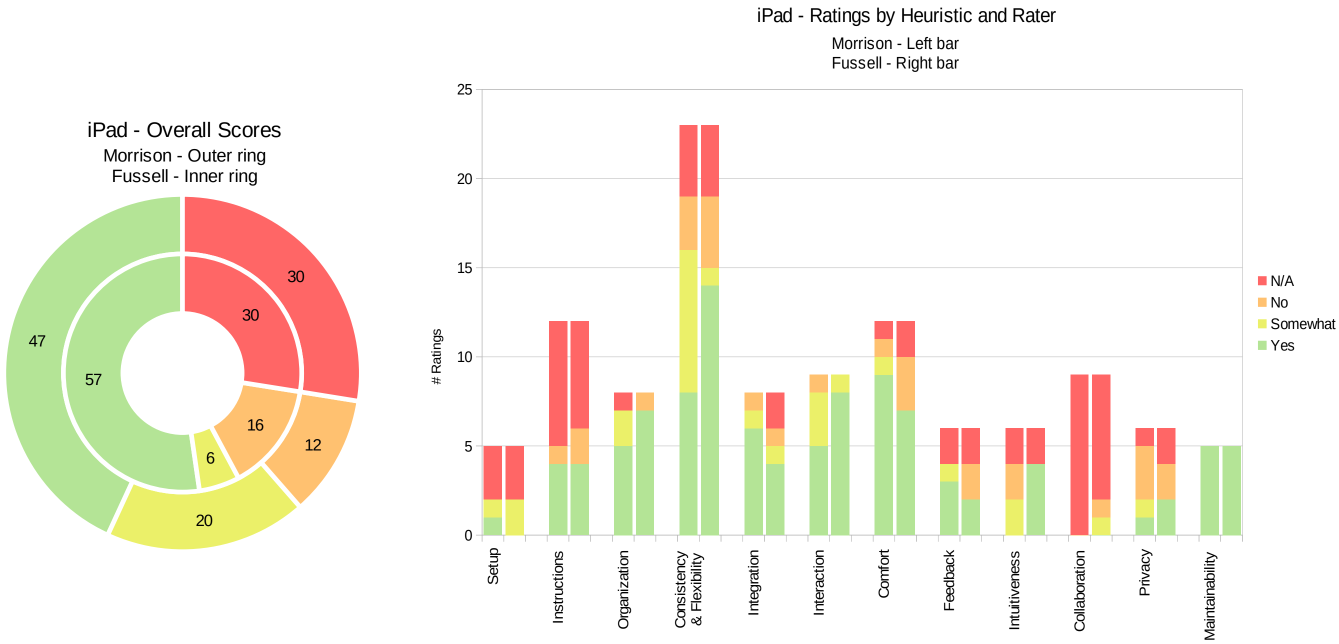


Figure 1. Test results for iPad.

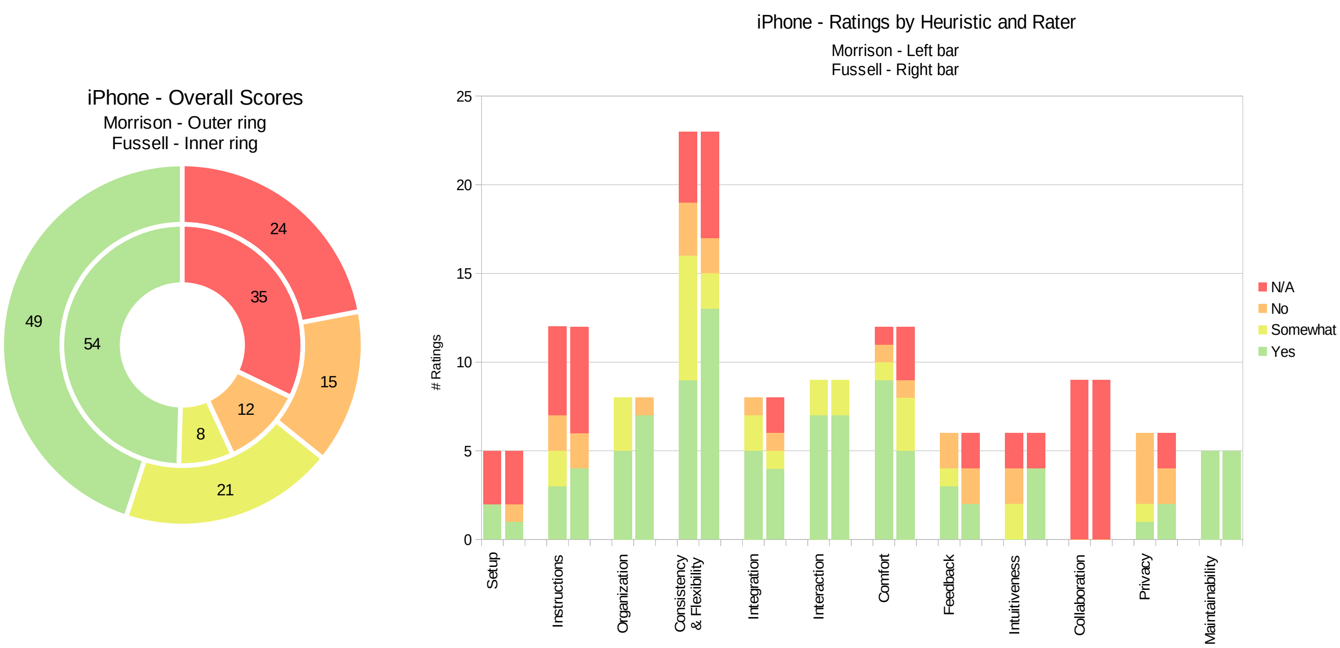
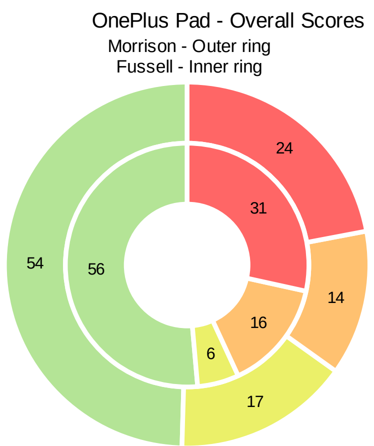


Figure 2. Test results for iPhone.



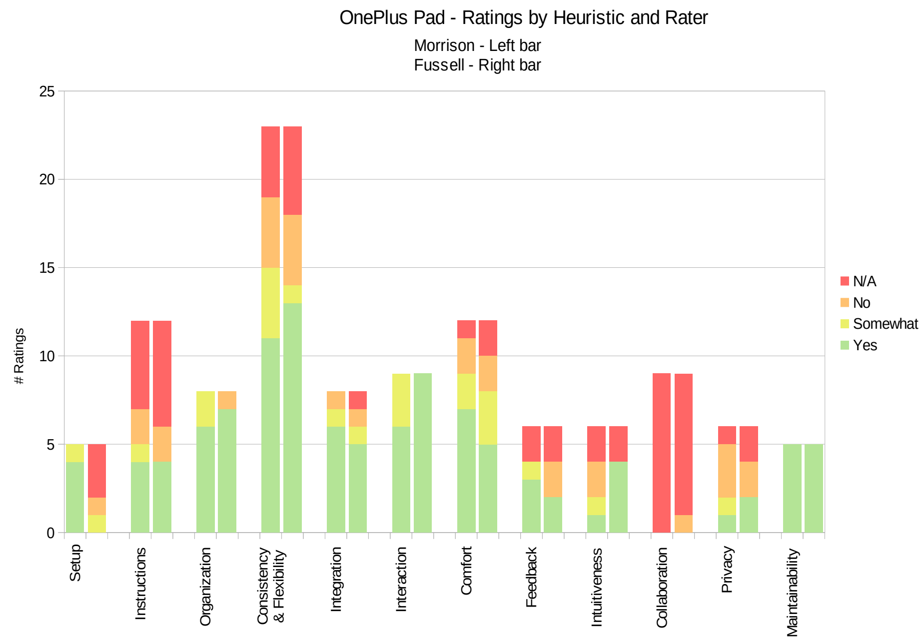


Figure 3. Test results for OnePlus Pad.

Discussion

With the Scope AR platform, each of the three mobile devices tested performed very similarly. In general, they all had a majority of answers of “Yes” on the heuristic, also with a significant portion of the items being inapplicable for our research. The user experience for the two participants was very similar overall, and it was generally positive. This shows that Scope AR enables relatively easy compatibility with a variety of commonly used devices. In the debriefing from the testing session, the biggest theme of the discussion was on the important topic of accessibility.

Throughout the Scope AR development and user cycle, it was found that in the effort to make development easy, the platform appeared to put little consideration into user accessibility. This included problems like there being no support for per-user settings to change things like contrast, font size, and speed and pause-play controls for animations. In addition to the lack of user configuration, there were several accessibility-related bugs. In general, there are many improvements that could be made to the Scope AR platform, which currently makes Unity a better option for developing a curriculum-style extended reality project.

Conclusion

In summary, we have found that AR/VR interaction can likely be an effective curriculum supplement for aircraft engines. While the development of these supplements will be challenging, it is a process that should lead eventually to a better experience for Kent State University students by enhancing their understanding and retention of the material.

We also found that the Scope AR platform enables easy compatibility and building of basic AR experiences for common devices, but it is unlikely to be the best solution for an extended reality experience development platform. Given the progress and capability we have seen during the beginning of the Unity development in this project, we believe that Unity will be a better candidate for future AR/VR curriculum development despite the more complex development process.

Future Work

Further testing and iteration of the software itself on both authoring platforms (Scope AR and Unity) is needed. This research will be important to more comprehensively explore the development process and challenges, end-user device compatibility, and the initial user-friendliness of programs created by each platform. And once the AR/VR aircraft engine training application has matured to the point of usability in a class setting, there will be another swath of testing available to perform. This will likely include feedback from students on usability as well as from professors on ease of implementing this application into their regular curriculum.

References

- Brenner, A. (2017). CREATING ENGAGING INSTRUCTION AND STUDENT PROJECTS UTILIZING AUGMENTED REALITY (AR). *9th International Conference on Education and New Learning Technologies*, 1387. https://doi.org/10.21125/edulearn.2017.1292
- Derby, J., & Chaparro, B. (2023). The Development and Validation of an AR/MR Usability Heuristic Checklist. *HCII 2022. Lecture Notes in Computer Science*. Springer, Cham. https://doi.org/10.1007/978-3-031-05939-1_11
- Kaviyaraj, R., & Uma, M. (2021). A Survey on Future of Augmented Reality with AI in Education. *2021 International Conference on Artificial Intelligence and Smart Systems*. https://doi.org/10.1109/ICAIS50930.2021.9395838
- Lucache, D. D., Haba, C. G., Anton, M. E., & Tiţă, C. (2023). Using AR in Technical Education. The ARphymedes case. *2023 International Conference on Electromechanical and Energy Systems*. https://doi.org/10.1109/SIELMEN59038.2023.10290145
- Sheng, B., Xia, L., & Li, P. (2014). Potential for Augmented Reality in Education: An Overview. *Ambient Intelligence and Smart Environments*. https://doi.org/10.4108/fiee.1.2.e2