3D Optimization Analysis in Augmented Reality Application

Riwinoto[1], Arul Bagus Nugroho[2]

{riwi@polibatam.ac.id[1], arulbagus21@gmail.com[2]}

Informatics Engineering, Politeknik Negeri Batam, Batam, Indonesia[1,2]

Abstract. Augmented Reality is an effort to combine the real world and the virtual world, in making this application there are several processes, one of which is creating 3D assets and there is a difference in the software used, namely Blender and Rhinoceros. Therefore, this research was conducted to provide a solution in case programmers encounter difficulties in integrating 3D assets from Rhinoceros to Unity using the optimization methods of decimate and join mesh in Blender. The research results showed that using Blender can be one solution to optimize 3D graphics in the development of Unity-based applications and games. Three optimization methods have been identified, namely Decimate, Join Mesh, and the combined method Decimate + Join Mesh. Decimate didn't have significant improvements and changes compared to the join mesh method. In this optimization technique, joining the mesh is done first, if time still allows and the level of time effort and improvement results have been taken, decimate can be done.

Keywords: Augmented Reality Integration, 3D Asset Creation, Blender and Rhinoceros Comparison, Unity Integration Challenges, Mesh Optimization Techniques

1 Introduction

Augmented Reality is a technology for combining the real world with the virtual world created using computer technology, thereby shortening the boundaries between the two. Even though it is still in the development stage, there are many futuristic experts and researchers who hope that AR implementation can be realized in the 2010-2020 time frame [1]. Augmented Reality is an application that connects the virtual world with the real world [2]. In the process of creating this application, there are several processes required, namely creating an application concept, creating a database, and creating assets such as 3D assets. Creating 3D assets is usually done using various types of software, for example to make ship construction measurements, calculations and precision are required using Rhinoceros software, whereas to just create 3D assets without requiring calculations and precision, it can usually be done using Blender. KKCTBN (Kontes Kapal Cepat Tak Berawak Nasional) is a competition that focuses on technological innovation, design, prototyping and prototype performance in the shipping and maritime sector. This competition is held every year by the Indonesian Talent Development Center, as part of the technological innovation competition agenda held by the National

ICAE 2023, November 07, Batam, Indonesia

DOI 10.4108/eai.7-11-2023.2342933

Achievement Center and the Indonesian Ministry of Education and Culture. Participants invited to take part in this competition are students. The main objective of this competition is to encourage the development of talent and knowledge in the field of shipping-maritime technology and increase awareness of the importance of innovation and design in this industry. This competition is an ideal opportunity for students to broaden their understanding and experience in this field [3]. AR technology can display the user's location and direction of movement when the user navigated. Also, natural markers were made and configured in the system. In the future, the researcher will improve the efficiency of any exhibition by introducing the system to the search system to facilitate users [4]. The development of AR technology to optimize a product in the economic field has had many good impacts [5,6,7,8], therefore, the author is interested in making an interactive business card as an implementation of Augmented Reality in the creative industry using Blender [9].

As for the need to create an Augmented Reality application together with Shipping Study Program Students at the Batam State Polytechnic who took part in the 2022 KKCBTN competition, the application was created with the aim of an Augmented Reality-based HSSV (High Speed Support Vessel) ship design and innovation competition. A comparison of the advantages between using the Rhinoceros and Blender applications was found, as well as the disadvantages of Rhinoceros which made it a little difficult for programmers to integrate 3D into Unity software. Factors influencing the difficulty in integrating 3D into Unity software include the large size of the 3D, the polygon grid being too detailed, and the precision of the polygon reduction feature in Rhinoceros being not very accurate because it cannot be monitored while reducing the polygon grid (render preview).

Blender has an option to reduce polygons in an existing model. By adding the decimate modifier to an object, the model designer can reduce the number of faces via the ratio values available in the modification window. It should be noted that the model design process of reducing polygons/triangles from scratch is more efficient than reducing them to existing polygons/triangles. This happens because a small ratio value for the decimate modifier can damage the final object [10]. Therefore, a comparative study of these two methods was carried out which will be one solution if programmers experience difficulties when 3D models created using Rhinoceros cannot be integrated into Unity.

2 Research Method

Optimization Techniques

This research method uses an optimization method discovered by the author during the application development process. The author has the result that optimization is carried out only when the 3D size exceeds 1GB, the optimization process is required in Blender, decimate is carried out to reduce the polygons in the 3D network so that detail is reduced and the 3D file size is reduced. The following is the process flow used.

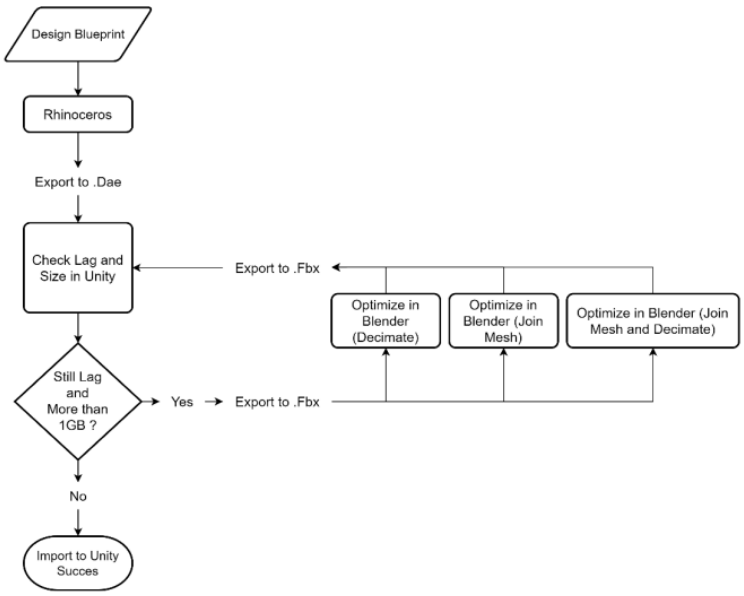

Fig. 1. Optimation technic

Optimization is carried out when the file size exceeds 1GB and when it feels heavy (lagging) during editing in Unity. The first step is to export to .Dae from Rhinoceros. Then, import it into Unity to check if the 3D model feels heavy (lagging). If it still feels heavy, export to .Fbx and import it into Blender for optimization, such as decimating and joining objects. Then, check if there are any inverted faces and whether the 3D model still feels heavy or not. If everything is correct, export from Blender to .Fbx and import it back into Unity. Then, profiling is performed before and after optimization to examine the process data in detail.

Optimization in Blender is done by decimating objects, joining objects, and simultaneously decimating and joining objects. A comparison is made to see the number of triangles obtained in Blender and Unity, whether there are differences or similarities, and why this occurs and what factors affect the difference or similarity in the number of vertices and triangles in an object studied using two different software, Blender and Unity.

The minimum percentage level in 3D optimization using Blender for a Unity game application can vary depending on the complexity of the initial model, game performance requirements, and the optimization goals to be achieved. However, as a general guideline, one reference point is the number of polygons. The common objective in 3D optimization is to reduce the number of polygons (triangles) as much as possible without significantly sacrificing visual quality. As a reference, some game development practitioners set a target of reducing the number of polygons by around 50-70% from the original model [11].

Decimate

Because of limitations in the hardware, the initial models would have produced a low frame rate. Hence, the models underwent a decimation process utilizing the blender modeling

software. This process involves reducing the vertex and face count of a mesh while preserving the essential shape with minimal alterations. Without implementing this procedure, achieving the desired performance with our current models would have been impossible. Decimating the models enables us to retain crucial details while eliminating unnecessary polygons that do not contribute to the essential model features [12].

Models in game development consist of polygons. A high polygon count makes assets more realistic, but requires a lot of resources when rendering. The GPU can only render objects in triangular form, so polygons must be converted to triangles. Models with a low polygon count are easier to draw on the screen. Unity recommends a polygon count of between 300 and 1500 for mobile devices, but this number depends on the number of models in the game scene. Blender has a “decimate” feature to reduce polygons in an existing model, but designing a model with low polygons from scratch is more efficient [10]. The ratio used in this research ranges from 0.5 to 0.75 depending on the quality level of the object. The choice of ratio value is also taken into as not to damage the model you want to decimate.

Join Mesh

" Join" is a function that consolidates multiple named particulars into the final chosen active item. It establishes a connection between all object data and the active item, which needs to be the one presently named. For this operation to work, all objects must partake the same abecedarian type, similar as mesh, wind, face, orarmature. However, each wind will maintain its original subtype, whether it's NURBS or Bézier, If you're combining several angles. When incorporating, colorful attributes of object data, including accoutrements , vertex groups, UVs, and Vertex layers, will be unified. still, modifiers, constraints, groups, and parent- child connections are disregarded during this process and will not be transferred to the active object [13].

Batches (Dynamic Batching)

Unity forwards every object the user intends to showcase in a game scene to the GPU for rendering, employing the graphics API specified for the given platform [14].This action is known as a draw call [14]. Draw calls must be applied to every object used for gaming purposes in order to give applicable vertices, shaders, textures, and more. still, in cases where there's a high number of models, these calls can put a heavy cargo on the GPU, especially on mobile bias. The result to this problem is handed through batch reduction [15]. More specifically, models that have the same materials can be combined in a batch to be rendered together [14]. Thus, it is possible to minimize memory consumption at runtime, along with CPU and GPU workloads [15]. Basically, batching is a way to find and correct the modeling material misuse mentioned earlier.

Recommended Framerate Settings in Unity

The Vuforia Engine API for Unity provides the `CameraDevice.GetRecommendedFPS()` method to obtain and set the recommended rendering framerate based on your rendering needs and application performance. To set the target framerate, call these methods during application initialization. The `getRecommendedFPS()` method returns the recommended framerate setting based on device configurations [16].For video see-through devices, the table below shows the available `GetRecommendedFPS()` values for various device types. Generally, rendering on

HoloLens should always be performed at 60 fps, while AR rendering on smartphones can achieve framerates of either 30 fps or 60 fps, depending on the device being used [16].

Table 1. Recommended Framerate By Vuforia

Eyewear/Mobile Device	Target Render FPS
HoloLens	60
Magic Leap	60
Android Devices	30
iOS Device	60 (on most devices)

Profilling

The best choice for implementing augmented reality technology in software is the Unity development environment. It serves as a top-tier platform for creating applications and games, equipped with a robust editor featuring a user-friendly graphical interface and an optimized graphical engine specifically tailored for mobile devices [17].

Profiling allows for in-depth examination of issues related to video game performance, such as the percentage of memory used, CPU or GPU usage by each process, and more. These actions can help developers more easily identify the parts that have the highest resource costs and take appropriate action to reduce them. For this, the main tool used is the profiler which is integrated in Unity [10].

Unity Profiler is a pretty good tool to start the process of monitoring and evaluating the performance of a game. The second tool, which is actually a script, is used to calculate FPS during the game. Although the profiler has this kind of functionality, it can only monitor games running on the computer. Therefore, the measurement of game FPS on mobile devices is done through a script that counts frames per second over a certain time interval [10].

3 Result and Analysis

For this experiment, two versions of the same game were developed. One uses optimization techniques, while the other does not. The main purpose of this comparison is to show how big a difference rendering results in a simple Augmented Reality can make in an application, especially one based on Android

Analysis of Tris, Vertis, Batch and FPS

The analysis started from collecting Tris and Vertis data in Blender using statistics. Then it is imported into Unity and collecting Tris and Vertis data again using the Unity profiler. To carry out profiler data analysis, we can use a package called Profile Analyzer.

Table 2. Optimization method based result of Tris, Vertis, Batch and FPS

Variable	Non Optimize	Decimate	Join Mesh	Decimate + Join Mesh
Blender Tris	1.6M	1.3M (18,75%)	1.6M (=)	1.3M (18,75%)
Blender Vertis	920K	730K (20,65%)	920K (=)	730K (20,65%)
Unity Tris	1.7M	1.3M (23,52%)	1.7M (=)	1.3M (23,52%)
Unity Vertis	2.3M	1.9M (17,39%)	2.3M (=)	1.9M (17,39%)
Unity Batches	3145	3108 (1%)	48 (98,48%)	48 (98,48%)
FPS	10 FPS	20 FPS (100%)	30 FPS (200%)	30 FPS (200%)

The data in Table 2 is taken from each software, namely blender tris and vertis using blender statistics, Unity tris, vertis and batches using Unity Statistics and Profile Analyzer, and FPS is calculated from the Mean FPS which is still worth ms. From the Tris, Vertis, Batch and FPS comparison table based on the optimization method above, it can be analyzed as follows:

● The decimate method has an optimization percentage that is not very significant for Tris and Vertis blender, namely 18.75% and 20.65%. When imported into Unity, the number of Tris and Vertis increased, but the percentage of Tris optimization increased to 23.52%. The number of batches has decreased but is not significant.

● The join mesh method does not have changes to the tris and vertices, both blender variables and unity variables. However, the number of batches had a drastic increase, namely 98.48%.

● The combined decimate and join mesh methods have the same optimization presentation as the decimate method for tris and vertices. However, the number of batches has a drastic increase in optimization percentage of 98.48%.

● The framerate produced by the decimate method has an increase of 100%, the join mesh mesh method has an increase of 200% and the combined decimate + join mesh method has an increase of 200%.

● The join mesh method is intended for batch reduction, while the decimate method is intended for tris and vertical reduction.

● Based on experience, using decimate takes twice as long as joining mesh. So if the joining mesh has been carried out, and additional optimization using decimate will be carried out, then you have to consider the time whether it is still possible or not because the effort and results are far away.

3.2 Analysis of the increase in tris and vertis values

The analysis was carried out to find out the reason why the increase in tris and vertices occurred when importing into Unity from Blender, especially files in FBX format. So research was carried out using Blender's default cube, exported to an FBX file and then imported into Unity

Table 3. Comparison of tris and vertis cubes when in blender and after being imported into unity

Software	Tris	Vertis
Blender	12	8
Unity	12	24

From the comparison data of Tris and Vertis Cube when blended and after being imported into Unity according to Table 3, it can be concluded that Tris has not increased. However, if you look at the data in Table 2, there are some optimizations that experienced an increase in tris and some that did not experience an increase in tris. This happens because when exporting to an FBX file, the quads are divided into triangles/tris and this can increase the number of vertices/vertis [18]. Today's GPUs are designed to render tris because tris makes things easier for users. Because tris makes it easier for users, modern GPUs are ultimately designed to focus more on rendering tris only. So if a modern GPU is asked to render a quad, it will divide it into 2 tris and render both. This is one of the reasons why sometimes there is an increase in tris and vertis when importing blender into FBX and then importing into unity [19].

4.Conclusion

From the analysis above, it is concluded that using Blender can be one solution to optimize 3D graphics in the development of Unity-based applications and games. Three optimization methods have been identified, namely Decimate, Join Mesh, and the combined method Decimate + Join Mesh.

Optimization can be carried out by developers who experience a decrease in the application's frame rate due to resource-intensive 3D rendering or using devices with low specifications. By optimizing 3D rendering, the workload on the device becomes lighter compared to using non-optimized 3D models, which could otherwise force the device to work harder during rendering.

The Decimate method doesn't show significant improvement or change when compared to the Join Mesh method. In this optimization process, mesh joining is performed first. If time permits and after considering the level of effort and the improvement results obtained, only then the Decimate process is carried out.

References

[1] A. Winatra, R. Khair, I. Idris, and A. Santosa, “Aplikasi Augmented Reality (Ar) Sebagai Media Edukasi Pengenalan Bentuk Dan Bagian Pesawat Berbasis Android,” *Jurnal Teknologi Informasi*, vol. 3, no. 2, 2019.

[2] M. A. Lesmana, I. F. Astuti, and A. Septiarini, “Penerapan Augmented Reality Sebagai Media Pengenalan Pesawat Udara Berbasis Android,” *Informatika Mulawarman : Jurnal Ilmiah Ilmu Komputer*, vol. 16, no. 2, p. 71, Oct. 2021, doi: 10.30872/jim.v16i2.3744.

[3] Pusat Prestasi Nasional, “Kontes Kapal Cepat Tak Berawak Nasional 2022.”

[4] P. Jomsri, “Implementing Virtual 3D Model and Augmented Reality Navigation for Library in University,” *International Journal of Modeling and Optimization*, vol. 8, no. 6, pp. 315–317, Dec. 2018, doi: 10.7763/IJMO.2018.V8.671.

[5] Du Z, Liu J, Wang T. Augmented Reality Marketing: A Systematic Literature Review and an Agenda for Future Inquiry. Front Psychol. 2022 Jun 16;13:925963. doi: 10.3389/fpsyg.2022.925963. PMID: 35783783; PMCID: PMC9244620.

[6] Gasmi A, Benlamri R. Augmented reality, virtual reality and new age technologies demand escalates amid COVID-19. Novel AI and Data Science Advancements for Sustainability in the Era of COVID-19. 2022:89–111. doi: 10.1016/B978-0-323-90054-6.00005-2. Epub 2022 Apr 8. PMCID: PMC9068980.

[7] Xu, Lan & Zhang, Lei & Cui, Nan & Yang, Zhilin. (2019). How and when AR technology affects product attitude. Asia Pacific Journal of Marketing and Logistics. ahead-of-print. 10.1108/APJML-03-2019-0221.

[8] Arena, F.; Collotta, M.; Pau, G.; Termine, F. An Overview of Augmented Reality. *Computers* **2022**, *11*, 28. https://doi.org/10.3390/computers11020028

[9] A. Rosemalatriasari, “Implementation of Augmented Reality Technology on Business Cards Using Blender and OpenSpace 3D,” *Jurnal Inovatif : Inovasi Teknologi Informasi dan Informatika*, vol. 6, no. 1, pp. 49–56, 2023, doi: 10.32832/inovatif.

[10] G. Koulaxidis and S. Xinogalos, “Improving Mobile Game Performance with Basic Optimization Techniques in Unity,” *Modelling*, vol. 3, no. 2, pp. 201–223, Mar. 2022, doi: 10.3390/modelling3020014.

[11] Unity Technologies, “Optimizing Graphics Performance.” [Online]. Available: https://docs.unity3d.com/2020.1/Documentation/Manual/OptimizingGraphicsPerformance.html

[12] P. Delacruz, J. Gibson, D. Howard, J. Peacock, and K. Robbins, “Medical Manikin Augmented Reality Simulation (M2ARS),” 2022, doi: 10.25776/82c4-fn62.

[13] Blender Foundation, “Join - Blender.” [Online]. Available: https://docs.blender.org/manual/en/latest/scene_layout/object/editing/join.html#join

[14] Unity Technologies, “Draw Call Batching.” [Online]. Available: https://docs.unity3d.com/Manual/DrawCallBatching.html

[15] A. Zhang, K. Chen, H. Johan, and M. Erdt, “High Performance Texture Streaming and Rendering of Large Textured 3D Cities,” in *Proceedings - 2020 International Conference on Cyberworlds, CW 2020*, Institute of Electrical and Electronics Engineers Inc., Sep. 2020, pp. 17–24. doi: 10.1109/CW49994.2020.00011.

[16] Vuforia, “Device Performance Optimization.” Accessed: Jun. 14, 2023. [Online]. Available: https://library.vuforia.com/platform-support/device-performance-optimization#setting-performance-mode-and-framerate-in-unity

[17] T. A. Vakaliuk and S. I. Pochtoviuk, “Analysis of tools for the development of augmented reality technologies,” 2021. [Online]. Available: http://ceur-ws.org

[18] Unity Discussion, “Number of vertices of a model is not the same in blender as in unity,” 2017, Accessed: Jul. 16, 2023. [Online]. Available: https://discussions.unity.com/t/number-of-vertices-of-a-model-is-not-the-same-in-blender-as-in-unity/190409

[19] Stack Overflow, “Are triangles a gpu restriction or are there other rendering pathways?” Accessed: Jul. 15, 2023. [Online]. Available: https://stackoverflow.com/questions/12495511/are-triangles-a-gpu-restriction-or-are-there-other-rendering-pathways