

# Augmented Reality Project - VR Environment

AKASH SHASHIKANT SHIKARE, Trinity College Dublin, Ireland

TONG ZHAO, Trinity College Dublin, Ireland

VARUN KUMAR GUPTA, Trinity College Dublin, Ireland

The project sticks to our proposal document, with minor changes in some features. We are utilizing Unreal Engine 5 to simulate realistic environmental interactions. Our primary target hardware is the Oculus Quest 2. It gives state of the art tracking features such as real-time hand tracking, spatial coherency, helping the user place objects and keeping them there in 3D space.. The project is timely, which can be attributed to a jump in computing prowess and visual fidelity of recent hardware. Drawing inspiration from Half-Life: Alyx [Valve 2020], from Valve, which is the most advanced game in the VR space. And, from Legend Zelda - Breath of the Wild [Wikipedia 2017], which is by far the most notable game in the toonish space. Our environment is largely inspired by toon shaders, to provide a more relaxing feel, an escape from everyone's day to day life. It was not a limitation of hardware, rather a choice to progress in this direction.

## 1 INTRODUCTION

As mentioned before, there have been some changes since the project proposal. These include dropping the mountains, rock climbing, interactive haptic [Games 2024b], and few sound cues. While, other features such as hand gesture movement, and complete independence from the Oculus hand held controllers has been achieved. There are also more diverse areas to explore in the ecosystem, including a waterfall, a cave and new fauna! A cute dragon or dinosaur as some may argue, with a male deer roaming the wild! The dragon chases a butterfly, which adds to the toonish aesthetic. There is also a cozy little shack for the player in the middle. More ambient sounds for the lake, with non-overlapping attenuation and priorities have been added. New blocking volumes have been added to limit the player to reachable areas, and areas of interest. We have also introduced Binaural Audio [Wikipedia 2024] to increase immersion, mixing between binaural and just spatial audio depending on the sound cue.

The work was divided based on an Agile Methodology, with two people working together on a feature to prevent code-blocks and increasing efficiency.

## 2 BACKGROUND

Virtual Reality is a hot topic with more developers releasing titles for many VR platforms. With the introduction of Unreal Engine 5.4 [Games 2024h], major improvements such as better raw input, better tracking and wider array of devices is not supported. With this new technology in the hand of developers, the number of games will increase by a lot in upcoming years.

In the past, often times people have been limited by the computational powers of Oculus devices, but with the release of Meta Quest 3 and Apple Vision Pro [ZDNet 2024], these barriers have been overcome to a certain extent.



Fig. 1. An overview of our landscape and various elements including the flora, fauna and other visual cues.

## 3 IMPLEMENTED FEATURES

- (1) **Tree cutting:** In the previous project proposal, we've decided to use UE5's Procedural Content Generation Framework(PCG) [Games 2024e] to generate forests and cuttable trees. But it turned out that for fracture, we should use UE5's Chaos Physics [Games 2024a] system, which is not compatible with PCG Graphs. Due to time limitations, we created some tree models that were fractured using the Chaos Physics system. In-game whenever a collision event is detected, for example the player uses an axe to hit the tree, the fracture effect is triggered, giving the effect of actually cutting the tree.
- (2) **Animal Life with AI:** We have multiple animals in our scene with some basic AI behaviour. The spider swarm and deer are created and animated ourselves. To simulate the spider swarm effect, we utilized the Niagara system in UE5 along with vertex animation. An advanced technique is used here called Vertex Texture Animation, VTA [Games 2024i]. This allows us to store the animation data in texture files. Traditional animation methods based on rigging and keyframing were used. An add-on for Blender [Foundation 2024] allowing us to export the animation data as VTA was used.
- (3) **Spatial Audio:** Using Unreal's Spatial Audio Temple [Games 2024f] we can add 3D Audio, including environmental sounds such as echo in a cave, binaural audio for narrating experiences, and generic real-life sound effects. As planned earlier, we have achieved our original goal of having different sounds for different actions. Except from Niagara and Unreal's Audio Engine, Blueprints [Games 2024d] were heavily used in order to maximize the control. These allow dynamic control over various elements. These also cut down development times by a lot.
- (4) **Environment:** The environment was made in Unreal itself. It was made using the landscape tool with erosion, hydroson

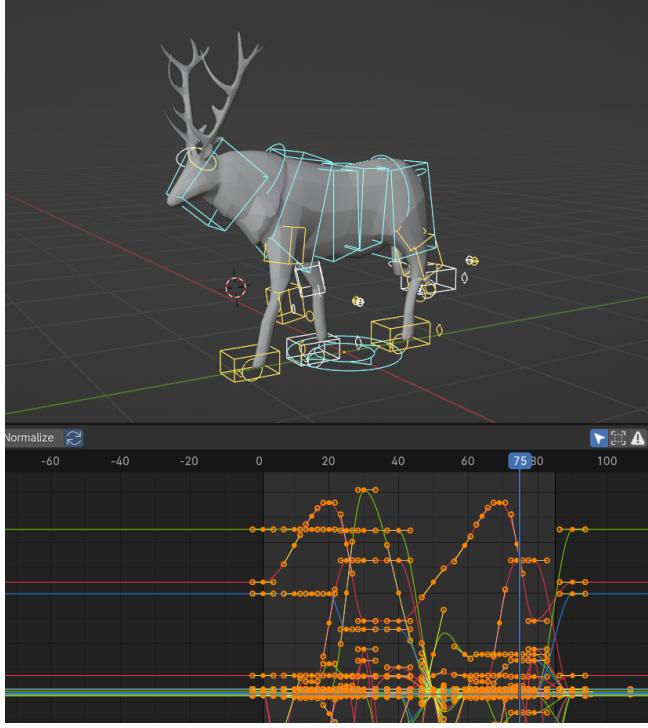


Fig. 2. Male Deer Rig, made manually in Blender using Auto Rig Pro [Market 2024]. The animation has been carried out using a reference sheet for a male deer.



Fig. 3. Spatial Audio visualized using a spherical attenuation boundary. The inner circle has more strong effect, while the other circle acts as a feather, blending various sounds together.

and various other weathering tools! The texturing has been carried out using procedural texturing, where-in the grass is automatically instanced over the grass textures and removed from the soil/stone textures. These are also computed dynamically based on the slope of the terrain. The foliage and stylized models have been taken from the Dreamscape: Stylized Environment Meadows [Studio 2020]. Sketchfab and Free3D were also used for other assets.

- (5) **Hand Tracking:** Instead of using controllers, this project implements full control logic using hand tracking, greatly improving the player's immersion. To enable hand tracking,

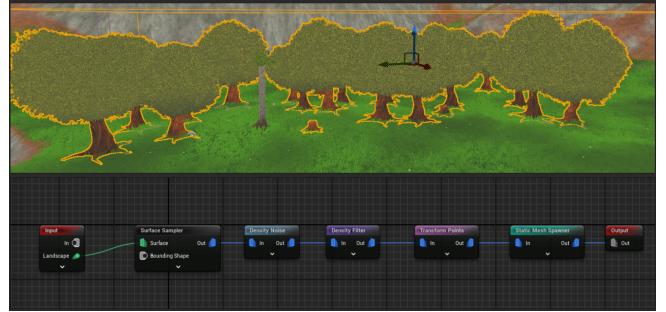


Fig. 4. PCG, showcasing the result and the driving graph below. This is just touching the surface and basics of the PCG graph in Unreal!



Fig. 5. Cave Biome, near the lake. This includes echo's, waterfall sounds and ambient lake sounds. All sounds are blended based on attenuation regions.

we went through a series of settings - Turning on the developer settings in Oculus application and enabling the relative plugins in UE5. After these settings were turned on, we had to import skeletal meshes for hands to recognize various gestures. UE5 provides basic gesture recognition within the enhanced input system, such as the pinch between every finger's tip with the thumb's tip. We also implemented a basic grabbing gesture, using the same idea as our AR Assignment 2, that is calculating the average distance between every fingertip with the wrist bone. Then, comparing the value with a threshold. This provides us with a more reliable tracking system.

- (6) **Arms tracking with Body:** To ensure the feeling of better immersion, we implemented an arm tracking system, using the Inverse Kinematics System [Games 2024c] in UE5. A hoodie model from Metahuman's from Epic Games was used. It is pre-rigged with many bones, making it ideal for a smooth locomotion system. We then attach it to our Player Actor and set up the IK rig for the hand-arm system. The arm tracking system fits perfectly in-game. Conversion from Euler to Quaternion space had to be carefully considered here.
- (7) **Simple Widget Interaction:** During the development stage, we found that moving in the scene while carrying items like axe was difficult. Hence, we implemented a simple widget system which is hand intractable. There is a simple re-spawn



Fig. 6. Image of fauna - A dragon/dinosaur in the game. He chases the butterflies around, an example of basic AI.

button which can generate an axe in front of the player with the correct translation and rotation. This was done through some basic vector calculations.

- (8) **Models and Digital Content Creation Software (DCC):** We are using low to mid poly assets and the player area is limited. This helps us overlook potential performance and efficiency issues. The trade-off between is lower "realistic" quality and higher performance. We are using open-source software, Blender [Foundation 2024] to aid in development of assets and UE5's level-of-detail (LOD) management.

#### 4 EXPERIMENTAL RESULTS

- (1) **Visual Quality** We used a stylized rendering [Games 2024g] instead of realistic rendering. There are still some broken pieces in the game which can be observed from the video. We haven't figured out the reason and we should look into it in the future in comparison with other VR games and possibly other VR headsets. The multi-light banding that happens is due to system specifications.
- (2) **Performance** Although we didn't put too many assets into this project, it is still lagging a bit during the game running. The possible reason is stylized rendering. Even though it may look toonish, it is often as heavy as Ray Tracing, due to many limitations in the mathematical modelling of the way light behaves. The smoothness in games, especially in VR games is important since there are people who are motion-sick and can not stand with low framerates. Also, we noticed that the hand tracking system was not as accurate as we expected, it requires a constant lighting setup.

#### 5 DISCUSSION AND CONCLUSION

This project still has a lot of room for improvement. We have developed a good project that includes many concepts, especially hand

tracking and gesture recognition. Not many open source and free projects such as ours do the same. This is a huge contribution to the open source community, as people can take this project and develop on top of it.

- (1) We dropped the rock climbing features. This was quite a difficult task, much more than we expected. Especially, for a hand-tracking project. Due to time limitations and some discussions like hand-tracking not being accurate enough, and we couldn't figure out a good way to lift up the body as desired.
- (2) Part of the features were implemented in a simple way, we didn't test the program thoroughly. For object interaction, we only have the grabbing logic for the right hand. For the avatar, we did not include the lower body's movement since it's hard to track whether a user is playing the game while standing or sitting. For the animal AI, we didn't include a complete behaviour tree to make a more interactive animal system, and we didn't have the time to make more animation sequences for them; We used PCG for forests generation, but we didn't use crowd simulation technique to simulate crowds.
- (3) This project is a combination of rendering, animation and virtual reality. We are very happy that most of the knowledge we learned from this semester was used in this project. It was a very good practice to help us better understand the modern game development workflow using Unreal Engine 5.

#### 6 RESOURCES

We are using UE5, Blender and Quixel [Quixel 2024] for developing the game and assets. Resource-wise, we are working with average system specifications following:

- (1) **RAM:** 16 GB
- (2) **CPU:** i7 9750H - 2.6GHz (4.5GHz Turbo)
- (3) **GPU:** GTX 1660Ti - 6GB VRAM
- (4) **Hardware:** Meta Quest 2 - 128 GB

The game is supposed to run locally on our system, while being projected to the headset using a cable. *Our expected performance is 60 FPS, which provides a good visual and auditory feedback to the player.*

#### REFERENCES

- Blender Foundation. 2024. Blender Repository. <https://projects.blender.org/blender/blender.git>. Accessed: March 27, 2024.
- Epic Games. 2024a. Chaos Physics. <https://docs.unrealengine.com/4.26/en-US/InteractiveExperiences/Physics/ChaosPhysics/Overview/>. Accessed: March 27, 2024.
- Epic Games. 2024b. Haptic Feedback. <https://docs.unrealengine.com/4.26/en-US/BlueprintAPI/Game/Feedback/PlayHapticEffect/>. Accessed: March 27, 2024.
- Epic Games. 2024c. IK Setups. <https://docs.unrealengine.com/4.27/en-US/AnimatingObjects/SkeletalMeshAnimation/IKSetups/>. Accessed: March 27, 2024.
- Epic Games. 2024d. Introduction to Blueprints. <https://docs.unrealengine.com/4.27/en-US/ProgrammingAndScripting/Blueprints/GettingStarted/>. Accessed: March 27, 2024.
- Epic Games. 2024e. Procedural Content Generation Overview. [https://dev.epicgames.com/documentation/en-us/unreal-engine/procedural-content-generation-overview?application\\_version=5.2](https://dev.epicgames.com/documentation/en-us/unreal-engine/procedural-content-generation-overview?application_version=5.2). Accessed: March 27, 2024.
- Epic Games. 2024f. Spatial Audio. <https://docs.unrealengine.com/4.27/en-US/Resources>Showcases/SpatialAudioTemplate/>. Accessed: March 27, 2024.

- Epic Games. 2024g. Stylized Rendering. <https://docs.unrealengine.com/4.27/en-US/Resources>Showcases/Stylized/>. Accessed: March 27, 2024.
- Epic Games. 2024h. Unreal Engine 5.4 is now available. <https://www.unrealengine.com/en-US/blog/unreal-engine-5-4-is-now-available>. Accessed: April 24, 2024.
- Epic Games. 2024i. Vertex Animation Tool. <https://docs.unrealengine.com/4.27/en-US/AnimatingObjects/SkeletalMeshAnimation/Tools/VertexAnimationTool/>. Accessed: March 27, 2024.
- Blender Market. 2024. Auto-Rig Pro. <https://blendermarket.com/products/auto-rig-pro>. Accessed: April 24, 2024.
- Quixel. 2024. Quixel. <https://quixel.com/>. Accessed: March 27, 2024.
- Polyart Studio. 2020. Dreamscape: Stylized Environment Meadows. <https://www.unrealengine.com/marketplace/en-US/product/dreamscape-nature-meadows>. Accessed: April 24, 2024.
- Valve. 2020. Half-Life: Alyx. <https://www.half-life.com/en/alyx>. Accessed: March 27, 2024.
- Wikipedia. 2017. The Legend of Zelda: Breath of the Wild. [https://en.wikipedia.org/wiki/The\\_Legend\\_of\\_Zelda:\\_Breath\\_of\\_the\\_Wild](https://en.wikipedia.org/wiki/The_Legend_of_Zelda:_Breath_of_the_Wild). Accessed: April 24, 2024.
- Wikipedia. 2024. Binaural Recording. [https://en.wikipedia.org/wiki/Binaural\\_recording](https://en.wikipedia.org/wiki/Binaural_recording). Accessed: March 27, 2024.
- ZDNet. 2024. Meta Quest 3 vs. Apple Vision Pro. <https://www.zdnet.com/article/meta-quest-3-vs-apple-vision-pro/>. Accessed: April 24, 2024.