The Reduction of 3D Model Polygon by Applying Imposter Technique to Optimize Real-time Rendering of Metaverse in Hubs and Spoke by Mozilla

Komsan Sukwinya

College of Creative Design and Entertainment Technology, Dhurakij Pundit University, Bangkok, 10210, Thailand
Corresponding author. E-mail address: komsan.suw@dpu.ac.th

Received: 2 April 2023; Revised: 25 July 2023; Accepted: 3 August 2023; Available online: 26 August 2023

Abstract

It is crucial for metaverse visualization that users have a shared experience and can interact in real-time over high-speed internet. However, the significant problem is that the amount of geometry artefacts in 3D models affects the effectiveness of processing performance. If the users want a realistic dimensional image, they must use a 3D model with a high polygon, which results in slow processing speed. For these reasons, the main objectives of this research were 1) To have fewer polygons in 3D models than in the original 3D models, by applying the Imposter Technique, and 2) To increase the efficiency of processing 3D models in the Metaverse system which was designed as a workflow that studies the reduction of polygons of a 3D model from High Polygon to Low Polygon by applying the Imposter Technique and inserting the pattern texture of the 3D model per Metaverse users' viewing angles through the Hubs and Spoke of Mozilla. The results showed that decreasing the polygons of the 3D model by applying the Imposter Technique can improve the real-time rendering performance of the Metaverse in the Hubs and Spoke of Mozilla, within the polygon number requirement of not more than 50,000 triangles, with video RAM usage of 256 MB, a file size no bigger than 16 MB, and a framerate at 60 FPS. The results were summarized as percentages. The mean of polygons of all groups' imposter model was 158.2 triangles. The mean of polygons decreased by 98.67%. The mean of the video RAM usage of the imposter model was 42.7 MB, which was a decrease of 73.60%. The mean file size of the imposter model was 4.893 MB with the standard file size reduced by an average of 71.26%, and the maximum copy of imposter at a framerate of 60 FPS of the imposter model was 107.6, an average increase of 365.80%.

Keywords: metaverse, 3D model, polygon, imposter

Introduction

The Metaverse is a technology in which humans can interact and communicate within virtual environments that combine physical reality with the digital world (Duan et al., 2021). All participants will be part of the communities through a platform that supports the Metaverse system (Sadiya, 2020) The idea was created for the benefit of several applications by relying on the Internet of the future that will change from a 2D world to a 3D world. This transformation will be able to create new and simultaneous experiences seamlessly in many dimensions, such as virtual classrooms, virtual conferences or seminars, viewing virtual art exhibitions, virtual playing of online multiplayer games, going shopping in a virtual minimart, owning virtual geographical locations or real estate, consulting doctor, transactions, and business negotiations, working in a virtual office that reduces the gap and feeling of loneliness from working from home, watching a virtual concert, virtual tourism and travel where the participants are not physically located at the tourism site or geographical location (Nevelsteen, 2016).

Creating a virtual environment requires relevant elements and technologies to help drive the Metaverse world to create a model of communication and interactions that closely simulate the physical world. These technologies consist of Augmented Reality (AR), which is the overlap between the real environment and the virtual world (Mystakidis, 2022). The virtual world consists of digital data, for instance, images, videos, audio, and 3D models which are all processed by computer systems or portable devices. Those digital and simulated data can be viewed and reacted to. Virtual Reality (VR) is a computerized visualization and environment to allow users to access and experience the virtual world in digital form through the various devices used to display images (Cipresso, Giglioli, Raya, & Riva, 2018). Currently, it is popular to use 3D glasses and headgear, such as the Meta Quest Pro or HTC Vive Pro Series for vision and hearing, with the use of the device to interact with the computer system to respond to user commands. The users will be able to perceive and interact as if they are in an actual event (Buhalis & Karatay, 2022) Mixed Reality (MR) is the combination of Augmented Reality and Virtual Reality, enabling us to see and interact with objects and environments in the digital and physical world. This requires a device specially designed to provide users with a rich virtual experience. For example, the Microsoft HoloLens® untethered mixed reality headset displays a 3D image and allows the user to see the virtual image together with images of physical objects while performing physical adjustments to and handling the digital objects. This technology is very useful in the field of training and the business sector as it can reduce both the loss of life and property that can occur from mistakes during the practice in real-life situations where there are risks, for example, fire drill, training in assembly and maintenance of complex and risky machines, and so forth. Extended Reality (XR) is a technology that combines a physical environment with a digital environment or combines AR, VR and MR technologies to create the most immersive experiences. XR enables interaction between users in a uniquely designed environment to achieve real-time, high-quality virtual content over a high-speed Internet network.

The Metaverse on each platform varies on different equipment depending on the format and need for each aspect relevant to the immediate purpose. However, it is equally important for digital content in a virtual environment — image, text, video, audio, and 3D models — to be created for use in the world of the Metaverse, especially realistic graphical representations. These representations require digital objects or 3D models to create a virtual world as a substitute for real-world objects. It is a mathematical process that uses a three-dimensional computer graphics system to create. There are currently two types of 3D objects based on their structural characteristics. The first is NURBS or Non-Uniform rational basis splines, whose structures are 2D curves mutually connected where the surface between the curves is calculated to form a 3D shape. There are parameters to control the curve, and the surface can modify the shape of that curve into a shape according to the design. NURBS applications are often used in engineering design in various applications, including prototypes of cars, ships, aircraft, or vehicles and a variety of other products (Munira, Jaafar, Fazilah, & Nooraizedfiza, 2014). The second type of 3D object is the polygon. Adjacent polygons simulate flat surfaces. If a sufficient number of polygons are pieced together a curved surface object is created. However, this increases storage space and requires more processing time as a result. The polygon-type object is commonly used to create 3D models in both Organic Form and living forms such as humans, plants or animals, or Hard Surfaces which are man-made objects such as houses, buildings, guns, armor, robots, and so forth. The Polygon is very flexible, easy to adjust, and details can be controlled according to needs, with a Vertex component, which is a point for specifying line coordinates, Edges which are lines on each side formed by two or more vertex connections, and Faces which are surfaces formed by at least 3 vertex points or at least 3 Edge lines resulting in a surface of 1 Face. A model object is formed by joining Faces together (Xia, El-Sana, & Varshney, 1997).

With this model, which is created from the polygon structure, a 2D image can be applied to a 3D model surface. The process is to place a 2D image on the UV coordinates, where U and V are referenced along the axes of the surface in the 2D flat surface, as X, Y, and Z are used to represent the axes of the 3D model. This process can only be started after the 3D modeling has been completed. It is a method for 3D models to display colors and patterns. Metaverse therefore essentially relies on creating 3D models with forms and colors that are created in place of real or imaginary objects. The completer and more beautiful the objects are, the more realistic and enjoyable the experience is for users.

Therefore, 3D modeling in Polygon format is the main method used to create Metaverse environment. The Metaverse's 3D objects consist of several objects, including many 3D model's Polygon which is complex and can affect the processing time and discrete data transmission if there are limitations in internet speed and processing device performance when navigating the Metaverse world. There is also a requirement on each platform that designers create virtual spaces to allow people to mutually interact and join in activities. In addition, simulating oneself as an avatar character with a 3D graphic format. The display scenario should be optimized by including photos and videos and processing performance optimized by image and video files, and optimizing 3D model displays by reducing the number of Polygons and the size of the surface. However, there is a tradeoff between having high display optimization and experiencing lower aesthetics and graphics integrity. Therefore, for the most part, every platform provides recommendations for creating a metaverse environment with the proper number of polygons being used without affecting the appearance of the model. Instead of hundreds of thousands of polygons (triangles), the number of polygons in the model should be reduced to tens of thousands of polygons (triangles). There are several techniques and methods for reducing the number of polygons and the size of 3D model files, including the Imposter Technique, which is one of the interesting options the researchers have chosen. This Technique has been tested with a variety of original 3D models based on basic geometric shapes with a high number of polygons in the original and reduced in size as much as possible while maintaining the model's appearance and texture as close to the original at most. The research conceptual framework is shown in Fig. 1.

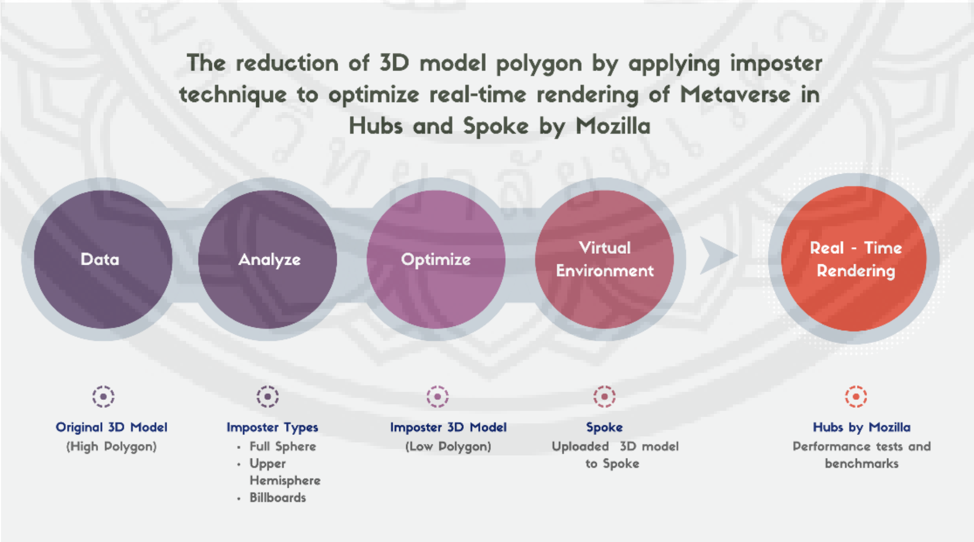


Figure 1 The process of reducing the number of polygons to optimize real-time rendering

The reduction of the 3D model polygon by applying the Imposter Technique is an effort by computer graphics professionals who want to create 3D models with rendering costs without affecting their aesthetics (Nuraliev, Hamidov, & Giyosov, 2020). In early 1976, James Clark stated that the purpose for using dimensional models in flight simulators at various resolutions was to guarantee that 3D aircraft models would keep frame rates consistent (Trapp & Dollner, 2019). The creation of 3D geographic visualization requires new techniques to guide the accurate representation of geographic images. It uses a large dataset method to reduce millions of polygon segments in real-time and has low memory requirements but high performance. The authors (Xu, Chen, Moreno-Noguer, Jeni, & Torre, 2020) has proposed a framework for optimizing 3D human surface estimation from a single image for use in virtual reality (VR), augmented reality (AR), gaming, robotics, and clothes try-on. The challenge is to create a texture image that looks like a 2D image by creating a framework to estimate the texture by compensating for missing images to get another view and blending it with the 3D model to be more accurate. However, in this study, multi-view images were used to create textures around objects in a consistent perspective for Metaverse users by studying and analyzing the original 3D model and converting that model to an imposter model to reduce Render costs and performance optimization for the best virtual reality experience in the Metaverse.

Materials and Methods

In this research, the researcher conducted a study based on the concept of research and development. The first step was to study the style of the original 3D model, which is high polygon and analyses the feasibility of reducing the number of polygons with Imposter types, after that optimization the model to low polygon before creating a virtual environment with Spoke and running performance tests and benchmarks in Hubs by Mozilla. The details and operating procedures are as follows:

A : Study the simplification and number of 3D model polygons with the Imposters Technique.

Imposters are techniques and methods that can be used to replace original 3D models which are complicated and which contain a large number of polygons. Impostors come in several forms but the main method consists of polygons whose numbers are reduced as much as possible and which are arranged so that the surface of the polygon is facing at an angle to the angle of the camera from the viewer's perspective through the display screen. Viewers can still experience realism that is no different from viewing the original 3D model because the 2D image calculation is pre-rendered at various angles and degrees: this is referred to as a Billboard mesh. It has been applied to the polygon surfaces that have been reduced in number, resulting in faster processing speed than using the original 3D model, but the aesthetics and realism are still the same as the original.

The Imposter format of each software has the same basic principles, but with differing terminology. Although the overall functionality is similar, there are some variations. For example, In the Unity program, there is a plugin called Amplify Imposters that divides the Imposter into three types: Spherical, Octahedron, and HemiOctahedron (Amplify, 2022). In addition to these software tools, there is a concept of capturing objects from various perspectives to store them as a surface instead of the 3D model according to expectations of different results and this depends on the features of the original 3D model regardless of its complexity. The proper way to use Imposters should not be based on one form or another. Instead, it should be a combination of multiple formats to achieve the most realistic image display which is created by the Imposters technique when changing camera angles or at any angle of view.

In terms of memory or file size to save 3D models and 2D textures, consider the conditions and limitations under which Platform Metaverse intermediates to publish 3D data. The researcher used Hubs by Mozilla with Spoke, an online 3D scene editor, which is primarily responsible for importing 3D models, and uploading them into a Virtual Environment before publicizing them to be available for users' participation through the Metaverse. Therefore, the common factor affecting the application performance is the difference in the device of users, especially of those using smartphones which may not be able to process images thereby providing a less immersive experience than using a computer. Internet speed is also another crucial factor in the Metaverse user experience because the media used in the Metaverse is not exclusively 3D models. It also consists of a variety of media, for instance, images, videos, and audio, or even an emoticon to express emotions, which is a factor that affects the user's response to the entire Metaverse world.

B: Designing and creating Virtual Spaces by Spoke and Hubs by Mozilla.

Hubs by Mozilla is the tool which the researcher chose to create virtual spaces as a hub for people to interact and do activities together. The design of virtual environments is based on a tool called Spoke which is an online 3D scene editor. It is responsible for bringing digital content such as image files, video files, audio files, particles, and 3D model files into the scenario. Spoke has tools that can create and customize 3D scenes by the program itself. Moreover, it can still add 3D content from Sketchfab, or if there is a 3D model created from a 3D design program such as Blender, Autodesk Maya, 3D StudioMax, Nomad, Sketch UP, and etc (Bredikhina, Sakaguchi, & Shirai, 2020). It can be used in Spoke by converting model files to .glb or .glTF format files, and they can also be uploaded to Spoke's asset library. Furthermore, it is recommended that the total size of all files implemented in this Virtual Spaces should not exceed 16 MB. In addition to 3D model files, Spoke can be added a 2D projection or a 360-degree equirectangular projection by uploading such images or linking files to extract images directly online as well. For the video content, videos can be streamed via YouTube and Twitch URLs by configuring to auto-loop or pause the video. However, it is worth noting that the associated content or URL cannot be changed after this virtual space is published to Hubs. Therefore, to share screenshots from a Web Cam or sharing a Desktop to Live Video, they must be done within the Hubs. These cannot be done directly from Spoke. Sound is indispensable when designing and building a Metaverse Environment. Spoke supports .mp3 files, nonetheless it does not directly control audio playback. If users want to access the volume controls, audio files must be uploaded to Hubs instead, but noise issues should be considered when using Metaverse through Hubs. The users of the Metaverse through Hubs hear multiple audio sources simultaneously. Therefore, Spoke provides an interesting feature to deal with the problem of sound interference perfectly - Audio Zones. These appearances are 3D Volumes that can be adjusted the softness of the sound according to the distance between the listener and the sound source. This is real-world behavior duplication. For example, if the listeners or avatars are close to the sound source, the sound level will be clear. If they are farther away, the volume will become so low that it will eventually become inaudible. Once the virtual spaces have been created (Fig. 2), the next step is to link the Spoke to the Hubs so that other users can participate in the activities in the created room.

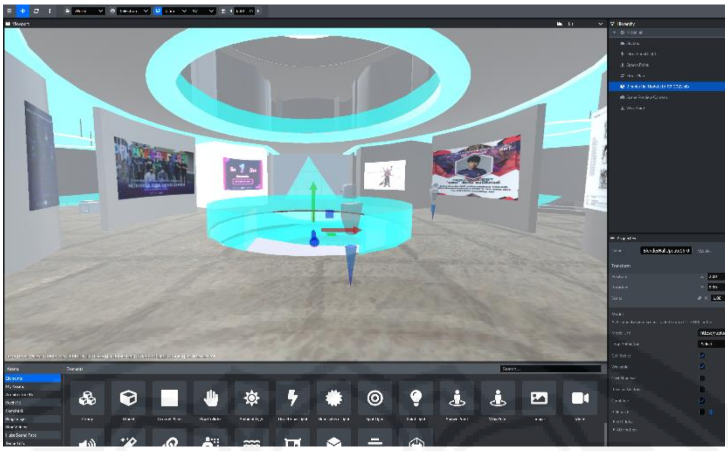


Figure 2 Virtual Spaces by Spoke and Hubs by Mozilla

C: 3D Model and Texture

The 3D modeling is a method in computer graphics to create digital objects (Tang & Ho, 2019). It requires the science of art and computer mathematics to create surfaces that mimic the shape of real or imaginary objects. 3D modeling requires specialized software such as Blender, Autodesk Maya, 3Ds Max, Nomad, Zbrush, SketchUp, and others. There are many methods of 3D modeling; however, the main design method is Polygon Meshes or NURBS, depending on the shape or nature of the object being created. The entire model in the Object Mode can be adjusted position, rotation, and scale, can be duplicated, and can be switched from Object Mode to Edit Mode to access subcomponents. When the model type is Polygon Meshes, which includes Vertices, Edges, Faces, or if it is a NURBS model, it contains Control points and Surfaces to modify the shape and shape of the digital object to be complete and beautiful.

Once the 3D model has been created, UV Unwarping is the process of mapping a 2D surface onto a 3D object by converting the surface position of a 3D model (Fig. 3). The dimensions on the X, Y and Z coordinates are spread or unfolded into a 2D plane on the U and V coordinates. For comparison and explanation for understanding in this regard, examples are given with models such as soft drink cans. The model is gray, which is usually the default color in 3D design programs. Because gray is a comfortable color and reduces eyestrain from staring during work. This gray color will eventually be replaced by a 2D image, which is a product label that will be placed around the soda can. However, with the 3D model being on the X, Y and Z coordinates, the 2D label image cannot be pasted directly. The surface of the polygon that is in 3D coordinates must first be converted to UV coordinates, just like a spherical map of the world. It is unfolded into a 2D image presenting the continents, oceans, and territorial areas of each country in one image. When the UV Unwrapping process is completed, an image with clear borders will be obtained to be colored, drawn, or filled with colorful images usually using 2D design programs such as Adobe Photoshop or Gimp. This has a tool brush head type, Layer system and color adjustment that are more flexible than drawing or painting than using a 3D design program. However, today there are 3D programs designed to paint and draw patterns directly with 3D models. For example, Adobe Substance 3D offers the same tools and layering system as a 2D design program that allows artists to create their work very freely and flexibly because the program allows them to switch between 2 modes, 3D and 2D at the same time. The benefit of this method is that the artists can see the results in real-time. However, regardless of the method used, the result will be a 2D image file that will be assigned to the 3D model. The process is to cover the labels of soda cans with colorful and beautiful colors instead of the gray shades, giving this piece a realistic shape and appearance.

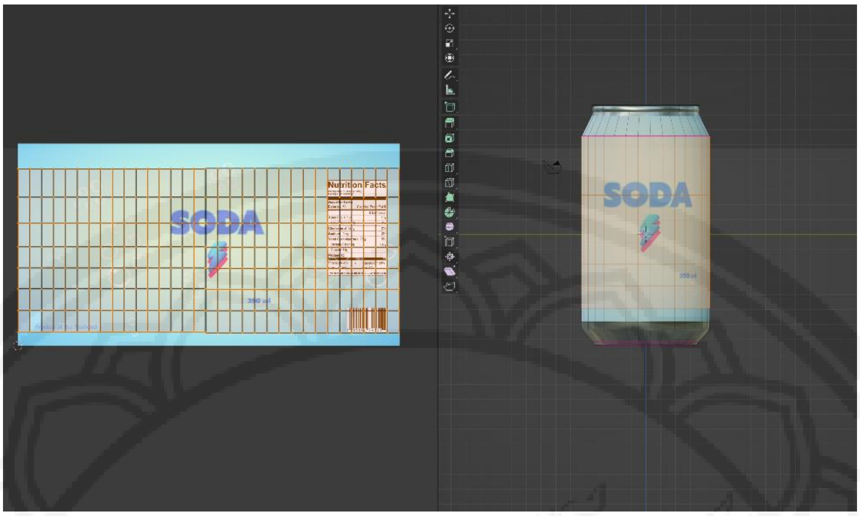


Figure 3 3D Model and UV Unwrapping Process

If users want to make 3D models that represent real-world objects as realistic as possible, photogrammetry is used. The basic principle is to take photos of real objects from many different angles to be processed in geometry with a program that can calculate the common points and determine the position of the object through the photo, such as Agisoft Photoscan, Reality Capture, and etc. The program will process the data from the photograph to create a 3D model with a surface image that can be used immediately. However, the drawback is that the 3D models created in this way, although realistic, have a very large number of objects or poly blocks, and at the cost of the computer's resources. Both in terms of memory and processing is very high. That is why designers need to reduce the size of the polygons derived from the process. Photogrammetry by 3D programs with various methods to recreate that model to reduce the details or the number of Polygon to be suitable for use in various fields, whether it is a 3D game or as a prop in an animation, a film, or visual effects. However, if the models are used in the metaverse, they are considered too much and not suitable for use anyway.

D: Metaverse rendering optimization by the imposter technique

According to the information above, it can be seen that the amount of data transmitted through the Internet is high. As soon as a room is connected and shared as a guest, each person will have access to image, audio, video, text, particle, or 3D model data during simultaneous playback. It will have an impact on display performance in the metaverse causing the image to be intermittent. The sound was missing at some point, or, in the worst case, it was not usable at all. Most of the system resources used will be spent on the 3D model. Rendering optimization is achieved using the Imposter technique, a traditional approach to 3D game design and development, to implement Level of Detail (LOD) reductions by replacing 3D models with 2D images. That image is an illusion that it is like a real 3D model or object, but it is to rotate the 2D image to an angle that corresponds to the angle of the camera (Misiak, Fuhrmann, & Latoschik, 2021). This process requires an understanding of Linear algebra or one of the mathematical fields related to the theory of linear equations and has the property to change the form of linear equations in either form Linear maps, Matrix, and Vector Spaces. The things that can be done to render an imposter in an original 3D game when the position and view of the camera can change every frame require CPU processing to update the coordinates to pinpoint the imposter's position to the GPU. The GPU preserves the texture, light, and shadow image, simulating 3D objects in the frame buffer, and then renders the image. It is many times faster and more efficient than using the original 3D model. Realism also depends on how users choose to apply the image instead of the subject's perspective on the camera. If the user selects a reasonable number of multi-angle images and seamlessly blends different textures, the more realistic it becomes. This can be considered the most difficult thing in choosing the Imposter Technique because, in the game system, the character or the camera that represents the player's eyes will move closer to the object or 3D model that will make the player notice. It can be said that the object is a disguised object and is not a real 3D model. In game design and development, it is inevitable to switch the display from Imposter to a real 3D model so that players cannot detect graphic anomalies when approaching the position of that object.

Type of Imposter

Currently, a variety of Imposter styles have been developed, the most widely used and least complicated is the Traditional Billboard (Fig.4). The nature of this method is to take the rendered 2D images of each surrounding view from the original model and paste that image onto the card. In other words, flat polygon planes are arranged in a cross-over manner. The number of cards depends on the expectation of the result to present the view of the object as closely as possible to the original 3D model. How close it is to the original 3D model? A typical game engine has a function called "Dithering Transparency" that hides the cards that are not facing the camera and shows only those cards facing the camera. If the view is changed, it will show - hide the card alternately. As an example, in this experiment, the complexity of a polygon from 23,545 tris can be reduced to 22 tris, which will reduce the processing time by several seconds.

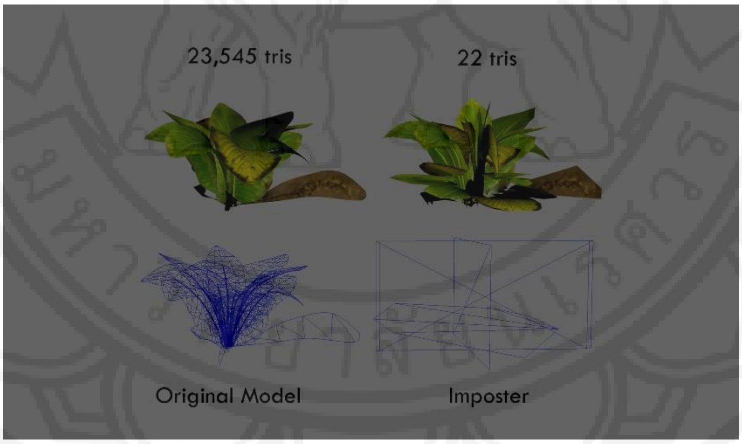


Figure 4 Traditional Billboard

Upper Hemisphere is the creation of an imposter only in the upper half of the sphere. It is a conversion between 2D and 3D space. It takes the processed image around the object but cuts out unnecessary parts. The image will remain only the top half of the object. The image will remain only the top half of the object. If the user chooses to use this method, the result must be considered whether the view conveyed must be a top-down view or an eye level that is parallel to the object only. It will not display properly if users viewed from below. For the most part, gamers or Metaverse users do not need to see the bottom of the model that is next to the ground. The advantage of this method is that it is twice as accurate as a full sphere due to the same number of images but only half of the sphere (Fig.5).

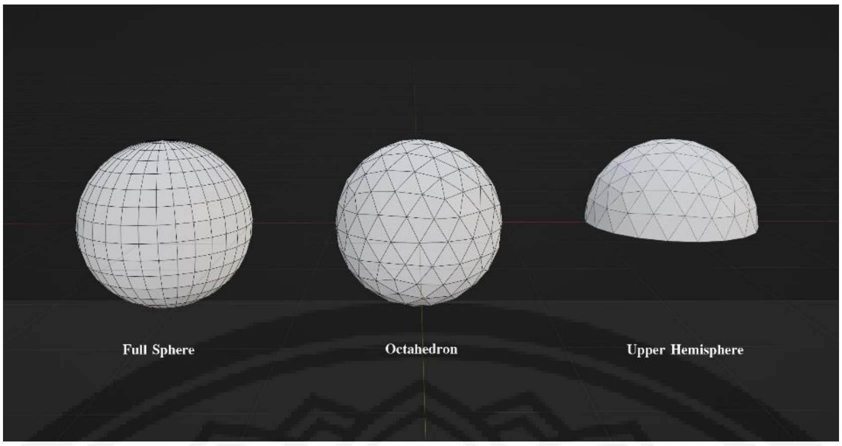


Figure 5 Spherical and Octahedron Imposter Techniques

A Full Sphere is a form of construction that shows objects all around whether the broadcast view goes above or below. This has an advantage when the camera is at close range but has disadvantages if the perspective of the subject or camera changes quickly. The movement that appears between frames is quite noticeable. This can be addressed with the use of Octahedron Imposter. This is a technique that can continuously transition between different views, but at the cost of higher shader complexity, when observed up close, it will be noticed that this is an imposter object (Amplify, 2022).

E: Imposter application to the limitations of Metaverse on Mozilla Hubs

Hubs by Mozilla is a real- time communications platform for users to experience the social virtual environments (SVEs) through their computer, smartphone, or tablet (Hagler, Lankes, & Gallist, 2022). Hubs act as a medium that allows users to send digital content such as text, images, audio, video, 3D models and scenes to share with other users. Mozilla is responsible for processing such data on servers located outside of Thailand. Users must consider the legal requirements for importing digital data according to their country of residence. Hubs operate through a web browser without the need to install any apps or software and they can work cross- platform or can be used in conjunction with a VR headset. All users will have an avatar like a representative of them in the same room when entering the Virtual Space or activity area. Users can view and explore via mouse and keyboard controls or by touch screen if they use a smartphone or tablet. In the case of VR devices such as the Oculus and Vive, users can control via the controller. They can change scenes and environments using an online 3D scene creation and editing program- Spoke. Creating and customizing content offers a wide variety of different assets. However, 3D models are available through Sketchfab or Google Poly, or by importing custom 3D model files from 3D design programs such as Blender. It needs to be converted to gLTF(glb) format before being used in Spoke. This glTF (glb) file contains all Mesh, UV, Materials, Animation, Shape keys and Skinning values in one file. Finally, parameterization is essential to the eventual rendering of 3D models in the Metaverse.

When transferring content generated via Spoke to Hubs, it is limited to 16MB. This means that every element in the scene must have the lowest file size possible. Recommendations for all Polygon Counts in a scene should not exceed 50,000 triangles. Moreover, the texture type should use the glTF standard recommended texture image resolution at 2048 x 2048 or lower, and texture file size should not exceed 256 MB of video RAM for mobile devices. Hubs are real-time processing web applications that support both computers and mobile devices, if users use large files, but negatively affect performance, especially on mobile devices with less processing power than a PC. The Mobile devices have additional restrictions on Light that cannot be dynamically enabled lights. On computers, the number of Lights should be no more than three. In the Materials section, there are texture specifications to support the handling of different textures: Base Color, Metallic, Roughness, Normal, Ambient Occlusion and Emissions. All Materials in a Scene should not exceed 25 Materials.

Pattern texture, Base Color, Normal and Emission are saved as separate image files, preferably in PNG format to avoid data loss. Ambient Occlusion, Roughness and Metallic are saved in Chanel Red, Green and Blue, whereas Red (R) stores ambient occlusion textures, Green (G) stores roughness textures, and Blue (B) stores metallic textures. When all three channels are packed together, they are known as the ORM Texture, which is derived from the initial letters of Occlusion, Roughness and Metallic.

Content optimization is necessary to reduce the complexity of the scene by reducing the number of 3D models or the number of poly cubes and reducing the texture file size. Optimizing content can increase display performance. Moreover, considering the above limitations, if the Imposter technique is applied the file size can be reduced even more without affecting the appearance of the model too much.

Based on the imposter model used in this test, the researcher applied the advantages of Traditional Billboard with Camera Projection Mapping techniques around the original 3D model (Bruneton & Neyret, 2012). The researcher then rendered the image according to the position of the camera positioned around the object as a still image and puts that still image onto the model that undergoes the retopology process. Retopology's principle is to bring the High Polygon model to arrange in fewer polygons for a low polygon 3D model. And this principle uses the surface images obtained from the rendering according to the position of each camera angle to paste it, resulting in a parallax effect that allows viewers to see the object in 3D, even if it is a 2D image (Fig.6).

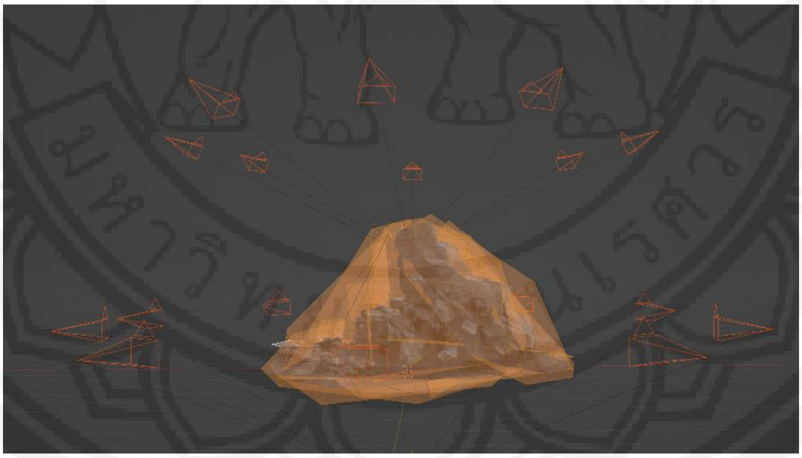


Figure 6 Traditional Billboard with Camera Projection Mapping Techniques

However, the figure and shape of the 3D object must also be considered per the imposter's form. For example, if an object is rectangular or has a flat surface, the camera capturing 2D each side of the object will be accurate when the model is an imposter. However, if the object is spherical or cylindrical, the characteristics of the surface and other components must be considered whether when reducing the number of polygons, the appearance of that object can be maintained or not (Amplify, 2022).

Performance tests and benchmarks

Realtime rendering is the process of rendering images using 3D data typically used in video games or the Metaverse. This research requires high-speed data processing so that many 3D models in the scene can interact promptly with the user consistent with human perception (Wang , S., & Zhang , J.(2021). Typically, frame rate (FPS) is an important measure of the metaverse browsing experience. Because the human line can perceive changes in image detail quickly, for example, if the image is slow to respond to the player's response, the experience will be uneven and motion sickness may occur (Sukwinya, 2022). This test compared the rendering performance between the Original 3 D model and the Imposter, which consists of the following visual performance metrics.

Polygon Count

Compare the number of polygons of the original 3 D model with the sorted imposter and reduce the number of textures from High Polygon to Low Polygon, where polygons are triangles.

Video RAM Usages (MB)

Video RAM, abbreviated as VRAM, is a portion of RAM (RAM) that is used to store images created by a computer to display on the screen with the Hubs and Spoke by Mozilla specification allowing texture files no more than 256 MB.

The comparison of the file size of the original 3D model with the imposter consisting of Mesh, UV & Texture and Materials that have been converted to gLTF (glb) format before being used in the Spoke, in KB or MB.

Frame Rate

Frame rate is a benchmark to assess display performance in real-time. which has the value The typical standard frames per second (FPS) display is 30FPS, but for use via the Metaverse, a frame rate of 60FPS or higher is recommended for the best experience. The test replicates the 3 D model over and over until the frame rate is less than 60FPS. This number is counted as the maximum copy value obtained by comparing the original 3D model with the imposter.

Percentage Change Formula

Percentage change equals the change in value divided by the absolute value of the original value, multiplied by 100

$$\text{Percentage Change} = \frac{\Delta V}{|V_1|} \times 100 = \frac{(V_2 - V_1)}{|V_1|} \times 100 \qquad \text{Eq.1}$$

Results and Discussion

The 3 D models tested were taken from the website: https://quixel.com. This was a website that collects 3D content obtained from scanning 3D objects in the real world into a digital format. It was divided into several categories to choose and optimize the content for each project and can be exported to 3 D applications or Game Engine as needed via Quixel Bridge program under a variety of file formats. Examples include FBX, USDZ, USD, ABC, MDL, and glTF. In this test, the model is grouped into 5 groups of 2, each with a total of 10 models: 1. Cube Models 2. Sphere Models 3. Cylindrical Models 4 .Vegetation Model 5. Free-form Models. The comparison of the rendering performance between the Original 3D model and the Imposter model was shown in Table 1.

Table 1 The comparison of the rendering performance between the Original 3D model and the Imposter model

Geometry	Model	Polygon (Triangles)		Video RAM Usages (MB)		File size (MB)		The maximum number of copy objects Framerate at 60FPS	
		original model	Imposter	original model	Imposter	original model	Imposter	original model	Imposter
cube	Wooden Crate	22,786	25	85	11	10.4	1.36	25	116
	Rusty Jerrycan	12,330	116	341	5	36.1	1.38	14	138
sphere	Pumpkin	9,086	122	85	5	4.44	0.80	11	71
	Construction Helmet	10,712	292	85	5	8.10	0.98	17	59
cylinder	Clay Pot	8,132	240	85	5	4.05	0.59	16	57
	Gas Tank	19,996	110	341	11	30.8	1.24	18	86
vegetation	Ferns	2,750	23	85	11	7	0.67	33	201
	Palm	1,929	484	341	352	50.3	38.9	36	63
free-form	Rotten Tree Stump	22,545	131	85	11	11.6	1.54	40	105
	Female Statuette	9,256	39	85	11	7.54	1.47	21	180

The conclusion and discussion of the reduction of a 3D model polygon by applying the Imposter Technique to optimize real-time rendering of Metaverse in Hubs and Spoke by Mozilla based on five basic geometric groups as follows:

Geometry Group 1: The tested cube models were a wooden crate and a rusty Jerrycan.

The polygons of the original wooden crate model were 22,786 triangles. When transforming this model into an imposter model, there were 25 triangles. The polygons decreased by 99.89%. The video RAM usage memory of the original wooden crate model was 85 MB, whereas the video RAM usage of the imposter models was 11 MB. The video RAM usage memory decreased by 87.05%. The file size of the original wooden crate model was 10.4 MB, whereas the file size of the imposter model was 1.36 MB. The file size decreased by 86.92%. Regarding the number of copies, the maximum framerate at 60 FPS for the original wooden crate model was 25 copies, whereas the imposter model was 116 copies. The number of copy objects was a 364% increase.

The polygons of the original rusty Jerrycan model were 12,330 triangles. When transforming this model to the Imposter model, there were 116 triangles. The polygon decreased by 99.05%. The video RAM usage memory of the original rusty Jerrycan model was 341 MB, whereas the video RAM usage memory of the imposter model was 5 MB. The video RAM usage memory decreased by 98.53%. The file size of the original rusty Jerrycan model was 36.1 MB, and the file size of the imposter model was 1.38 MB. The file size decreased by 96.17%. Regarding the number of copies, the maximum framerate of 60 FPS for the original rusty Jerrycan model was 14, whereas the imposter model was 138 copies. The number of copy objects was an 885.71% increase.

Geometry Group 2: The tested sphere models were pumpkin and construction helmets.

The polygons of the original Pumpkin model were 9,086 triangles. When transforming this model to the Imposter model, there were 122 triangles. The polygons decreased by 98.65%. The video RAM usage memory of the original Pumpkin model was 85 MB and the video RAM usage of the imposter model was 5 MB. The video RAM usage memory decreased by 94.11%. The file size of the original pumpkin model was 4.40 MB and the file size of the imposter model was 0.80 MB. The file size decreased by 81.81%. Regarding the number of copies, the maximum framerate at 60 FPS for the original pumpkin model was 11 copies, whereas the imposter model was 71 copies. The number of copy objects was a 545.45% increase.

The polygons of the original Construction Helmet were 10712 triangles. When transforming this model into the Imposter model, there were 292 triangles. The polygons decreased by 97.27%. The video RAM usage memory of the original construction helmet was 85 MB and the video RAM usage of the imposter model was 5 MB. The video RAM usage memory decreased by 94.11%. The file size of the original construction helmet model was 8.10 MB and the file size of the imposter model was 0.98 MB. The file size decreased by 87.90%. Regarding the number of copies, the maximum framerate at 60 FPS for the original construction helmet model was 17 copies, whereas the imposter model was 59 copies. The number of copy objects was a 247.05% increase.

Geometry Group 3: The tested cylinder models were a clay pot and a gas tank.

The polygons of the original clay pot model were 8,132 triangles. When transforming this model into the imposter model, there were 240 triangles. The polygons decreased by 97.04%. The video RAM usage memory of the original clay pot model was 85 MB and the memory of the video RAM usage of the imposter model was 5 MB. The video RAM usage memory decreased by 94.11%. The file size of the original clay pot model was 4.05 MB and the file size of the imposter model was 0.59 MB. The file size decreased by 85.43%. Regarding the number of copies, the maximum framerate at 60 FPS for the original clay pot model was 16 copies and the imposter model was 57 copies. The number of copy objects was a 256.25% increase

The polygons of the original gas tank model were 19,996 Triangles. When transforming to the imposter model, there were 110 triangles. The polygons decreased by 99.44%. The video RAM usage memory of the original gas tank model was 341 MB, whereas the video RAM usage memory of the imposter model was 11 MB. The video RAM usage memory decreased by 96.77%. The file size of the original gas tank model was 30.8 MB and the file size of the imposter model was 1.24 MB. The file size decreased by 95.97%. The maximum framerate at 60 FPS for the original gas tank model was 18 copies, whereas the maximum framerate of the imposter model was 86 copies, an increase of 377.77%.

Geometry Group 4: The tested vegetation models were fern and palm.

The polygons of the original fern model were 2,750 triangles. When transforming this model into the imposter model, there were 23 triangles. The polygons decreased by 99.16%. The video RAM usage memory of the original fern model was 85 MB and the video RAM usage memory of the imposter model was 11 MB. The video RAM usage memory decreased by 87.05%. The file size of the original fern model was 7 MB and the file size of the imposter model was 0.67 MB. The file size decreased by 90.42%. Regarding the number of copies, the maximum framerate at 60 FPS for the original ferns model was 33 Copies, whereas the imposter model was 201 copies. The number of copy objects was a 509.09% increase.

The polygons of the original palm model were 1,929 triangles. When transforming this model into the imposter model, there were 484 triangles. The polygons decreased by 74.90%. The video RAM usage memory of the original Palm model was 341 MB and the video RAM usage memory of the imposter model was 352. The video RAM usage memory increased by 3.22%. The file size of the original palm model was 50.3 MB whereas the imposter model was 38.9 MB. The file size was reduced by 22.66%. The maximum framerate at 60 FPS for the original palm model was 36 copies, whereas the imposter model was 63 copies. The number of copy objects was a 75% increase

Geometry Group 5: The tested free-form models were a rotten tree stump and a female statuette.

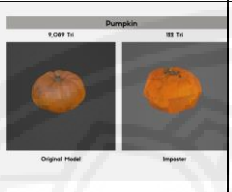
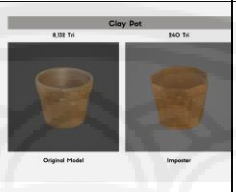
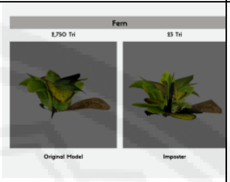
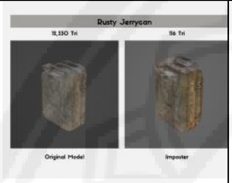
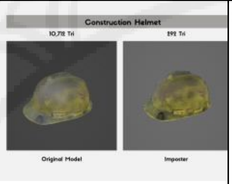
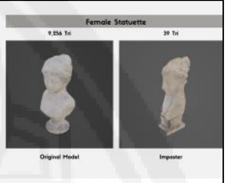
The polygons of the original rotten tree stump model were 22,545 triangles, when transforming this model to the imposter model, there were 131 triangles. The polygons decreased by 99.41%. The video RAM usage memory of the original rotten tree stump model was 85 MB, whereas the video ram usage memory imposter model was 11 MB. The video RAM usage memory decreased by 87.05%. The file size of the original rotten tree stump model was 11.6 MB whereas the file size of the imposter model was 1.54 MB. The file size was reduced by 86.72 %. Regarding the number of copies, the maximum framerate at 60 FPS for the original tree stump was 40 copies and the imposter was 105 copies. The number of copy objects was a 162.5% increase.

The polygons of the original female Statuette were 9, 256 triangles. When transforming this model into the imposter model, there were 39 triangles. The polygons decreased by 99.57%. The video RAM usage memory of the original female statuette model was 85 MB, whereas the imposter model was 11 MB. The video RAM usage memory decreased 87.05%. The file size of the original female statuette model was 7.54 MB, whereas the imposter model was 1.47 MB. The file size decreased by 80.50%. The maximum framerate at 60FPS for the original female statuette model was 21 copies, whereas the imposter model was 180 copies. The number of copy objects was a 757.14 % increase.

The mean of all groups of the original polygon models was 11,952 triangles, whereas the mean of all groups of the imposter model was 158.2 triangles. The mean of polygons decreased by 98.668%. The mean of the video RAM usage of the original model was 161.8 MB, whereas the mean of the video RAM usage of the imposter model was 42.7 MB. The mean of the video RAM usage decreased by 73.60 %. The mean of the original model file size was 17.03 MB whereas the mean of the imposter model was 4.893 MB .The mean file size decreased by 71.26 %. The number of copies at the maximum framerate at 60 FPS for the original model was 23.1 copies, whereas the imposter model was 107.6 copies. The mean of the maximum copies of the model framerate at 60 FPS was a 365.80% increase.

Analysis of the accuracy of the imposter's characteristics of figures and shapes compared to the original 3D model was shown in Table 2.

Table 2 Comparison of figures, shapes and textures between the original 3D model and the imposter

Group 1	Group 2	Group 3	Group 4	Group 5
				
				

Characteristics of figures, shapes, and texture between the original 3D model and the first group of the imposters: Cube is a wooden crate and rusty Jerrycan is a rectangular shape with all 6 sides. The virtue of the imposter is that the surrounding sides are as accurate as the original model, but when it is viewed from above, it is noticeable that the wooden crate has missing textures due to the 2D capture. The top is a slight deviation from the original and, on the part of the rusty Jerrycan, there will be problems in the circular bezel area and top handle where the model has been compromised. However, overall, the users can understand what it is.

Characteristics of figures, shapes, and textures between the original 3D model and the second group of the imposters: Sphere is the pumpkin and construction helmet, which is a spherical shape with a full sphere and upper hemisphere surface, respectively. The overall integrity of the imposter pumpkin resembles the original model, but capturing 2D images to assemble it into a spherical model requires a lot of imagery. As a result, the surface of the model has square edges that may not look realistic when viewed up close. The construction helmet is hemispherical and has a protruding brim at the front, requiring twice the number of shots of the Pumpkin, although it has a smaller volume to keep its shape close to the original model.

Characteristics of the figures, shape and texture between the original 3D model and the third group of Imposters: The cylinder is a clay pot and the gas tank has a cylindrical surrounding surface. The overall Imposter finish of the Clay Pot is similar to the original, but the surrounding model surfaces also have square edges, similar to Pumpkin, which may look unrealistic when viewed up close. As for the gas tank, there will be a problem with the valve on the gas tank being compromised because it is a small and detailed component.

Characteristics of the figures, shapes, and texture between the original 3D model and the fourth group of the Imposters: Vegetation is Ferns and Palm using the technique of placing billboard sheets crossed over each other. The integrity of the Imposter, and the surrounding flanks, are as accurate as the original model. Ferns are more complex than the original. Because the 2D capture reproduces the texture in the top view and the palm, the detail in the leaves is tapered. Some are slightly lacking in integrity.

Characteristics of the figures, shape and texture between the original 3D model and the fifth group of the Imposters: Free Form is the Rotten Tree Stump and the Female Statuette has free surrounding surfaces. The integrity of Rotten Tree Stump's overall imposter is very similar to the original. The female statuette is somewhat lacking in 3D, giving it a flat, unrealistic feel. Due to the shape of a woman's face requiring details from every angle, it makes quite a lack of perfection.

Conclusion and Suggestions

Based on the analysis of the accuracy of the figure, shape and the Imposter's accuracy characteristics compare to the original 3D model, in conjunction with its real-time rendering performance. It is found that the advantage is that the average of the polygon model can be reduced by at least 98%. This is sure to greatly improve real-time performance. However, the trade-off is the accuracy of the 3D model, as are the components in the main model, such as gas tank valves, tank caps and handles, protruding bonnets, and intricately tapered bushes. Female Statuettes require detailing to enhance their aesthetic beauty. They also require new techniques and methods to help the model achieve full detail and be able to replace the original model.

This research not only enables the introduction of 3D models into the Metaverse world but is also provides a very useful in-game engine, animation and visual effects applications that require hundreds of thousands of models to be assembled into 3D scenes to be displayed. result in real-time or dramatically reduced rendering time.

References

Amplify. (2022). *Amplify Impostors*. Retrieved from http://wiki.amplify.pt

Bredikhina, L., Sakaguchi, T., & Shirai, A. (2020). *Web 3D Distance live workshop for children in Mozilla Hubs*. Retrieved from https://shs.hal.science/halshs-03781969/document

Bruneton, E., & Neyret, F. (2012). Real - time Realistic Rendering and Lighting of Forests. *Computer Graphics Forum, 31*, 373-382. https://doi.org/10.1111/j.1467-8659.2012.03016.x

Buhalis, D., & Karatay, N. (2022). Mixed Reality (MR) for Generation Z in Cultural Heritage Tourism Towards Metaverse. In: J. L. Stienmetz, B. Ferrer-Rosell, D. Massimo (Eds.), *Information and Communication Technologies in Tourism 2022. ENTER 2022.* (pp. 16-27) Springer: Cham. https://doi.org/10.1007/978-3-030-94751-4_2

Cipresso, P., Giglioli, I. A. C., Raya, M. A., & Riva, G. (2018). The Past, Present, and Future of Virtual and Augmented Reality Research: A Network and Cluster Analysis of the Literature. *Frontiers in Psychology*, *9*, 2086. https://doi.org/10.3389/fpsyg.2018.02086

Duan, H., Li, J., Fan, S., Lin, Z., Wu, X., & Cai, W. (2021). Metaverse for Social Good: A University Campus Prototype. *Proceedings of the 29th ACM International Conference on Multimedia, 20-24 October, 2021* (pp. 1-9). China: Virtual Event.

Hagler, J., Lankes, M., & Gallist, N. (2022). "Behind the Curtains: Comparing Mozilla Hubs with Microsoft Teams in a Guided Virtual Theatre Experience," *2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, (pp. 19-22). Christchurch: New Zealand. https://doi.org/10.1109/VRW55335.2022.00011.

Misiak, M., Fuhrmann, A., & Latoschik, E. M. (2021). Impostor-based Rendering Acceleration for Virtual, Augmented, and Mixed Reality. *27th ACM Symposium on Virtual Reality Software and Technology (VRST'21), 8-10 December 2021* (pp. 1-10) Osaka, Japan: ACM. https://doi.org/10.1145/3489849.3489865

Munira, A., Jaafar, N. N., Fazilah, A. A., & Nooraizedfiza, Z. (2014). Review on Non Uniform Rational B-Spline (NURBS) : Concept and Optimization. *Advanced Materials Research*, *903*, 338-343. https://doi.org/10.4028/www.scientific.net/amr.903.338

Mystakidis, S. (2022). Mataverse . *Encyclopedia 2022*, *2*, 486-497. https://doi.org/10.3390/encyclopedia2010031

Nevelsteen, K. (2016). Virtual world, defined from a technological perspective and applied to video games, mixed reality, and the Metaverse. *Computer Animation and Virtual Worlds*, *29*(1), e1752.

Nuraliev, F. M., Hamidov, V. S., & Giyosov, U. E. (2020). Placing A Custom 3d Object In The Virtual World Environment. *Journal of Advances in Engineering Technology*, 2(2), 1-8. https://doi.org/10.24412/2181-1431-2020-2-3-8

Sadiya, S. (2022). Humans Emerging beyond Universe with Metaverse, is it possible? How? *International Journal for Research in Applied Science & Engineering Technology (IJRASET) 2022*, 1479-1485. https://doi.org/10.22214/ijraset.2022.40044

Sukwinya, K. (2022). A Production of Virtual Reality Technology Learning Materials: Driving for Preparator of Practical Driving License Test. *Naresuan University Journal: Science and Technology*, *30*(1), 1-17.

Tang, Y. M., & Ho, H. L. (2019). *3D Modeling and Computer Graphics in Virtual Reality, Mixed Reality and Three-Dimensional Computer Graphics, Branislav Sobota and Dragan Cvetković.* Retrieved from https://www.intechopen.com/books/mixed-reality-and-three-dimensional-computer-graphics/3d-modeling-and-computer-graphics-in-virtual-reality

Trapp, M., & Dollner, J. (2019). Real-time Screen-space Geometry Draping for 3D Digital Terrain Models. *23rd International Conference Information Visualisation (IV)*, (pp. 281-286). Paris, France: IEEE.

Wang, S., & Zhang , J.(2021). Research and Implementation of Real-time Render Optimization Algorithm Based on GPU. Journal of Physics: Conference Series, Volume 2136, *The International Conference on Electronics, Electrical and Information Engineering 2021 (ICEEIE 2021)* 13-16 August 2021, Changsha, China

Xu, X., Chen, H., Moreno-Noguer, F., Jeni, L., & Torre, F. (2020). *3D Human Shape and Pose from a Single Low-Resolution Image with Self-Supervised Learning.* Retrieved from https://arxiv.org/pdf/2007.13666.pdf

Xia, J. C., El-Sana, J., & Varshney, A. (1997). Adaptive Real-Time Level-of-detail-based Rendering for Polygonal Models. *IEEE Transactions on Visualization and Computer Graphics*, *3*(2), 171-183. Retrieved from https://ieeexplore.ieee.org/document/597799