Comparative Study of 3D Assets Optimization of Virtual Reality Application on VR Standalone Device

Sritrusta Sukaridhoto [a,*], Amma Liesvarastranta Haz [b], Evianita Dewi Fajrianti [b],
Rizqi Putri Nourma Budiarti [c]

[a] Department of Information and Computer Engineering, Politeknik Elektronika Negeri Surabaya, Surabaya, 60111, Indonesia
[b] Department of Information and Communication Systems, Okayama University, Tsushimanaka, Kita Ward, 700-8530, Okayama, Japan
[c] Department of Information System, Universitas Nahdlatul Ulama Surabaya, Surabaya, 60237, Indonesia
Corresponding author: *dhoto@pens.ac.id

Abstract—The progress of VR technology is undeniably rapid and reaches many sectors unrelated to what it first came out of entertainment. Today, many educational, health care, company training, etc., use and utilize VR in one way or another as their first step to familiarize a concept or procedure with their members or workers. The advantages of implementing educational content with VR are ease of development, cheap operational cost, and safety. This kind of approach is a good step considering the impact of VR technology on those cases. However, because the leading device for VR is a standalone VR, some things to consider are performance and visualization. Some early adaptations have these problems, performance issues, and the realism of visualization shown on the VR application. We can minimize those problems by meticulously optimizing 3D assets used in VR applications. The optimization method improved the average FPS on CBIVT by 14.02% on Quest 1 and 8.99% on Quest 2. The GPU utilization level percentage of Quest 1 decreased by 6.73%, and the Quest 2 GPU utilization level percentage decreased by 11.72%. On the other metrics, the user's comfortability also increases because of the enhancement of performance on the CBIVT application. These changes are marked by the increase in Important and Satisfactory levels to 4.26 and 4.16, respectively.

Keywords— 3D asset; optimized performance; virtual reality; immersive technology; standalone VR.

I. INTRODUCTION

The rapid growth of immersive technology today serves as a two double-sided sword. Like all new technology, immersive technology faces many reactions to its early development years. Some think it will bring new opportunities for entertainment, education, and social interaction development. Many also think it can create an isolated behavior against what humans are, social beings [1]–[5]. Whatever the reason, immersive technology has become a topic of discussion in technological development in recent years. Immersive technology can be divided into three interlock topics as Augmented Reality (AR), Virtual Reality (VR), and Mixed Reality (MR) [6]–[12]. The development of VR technology to help human activities have been rolling long before Facebook announced that they would launch Metaverse in 2021, which is essentially an MMO set in a VR Environment to do social activities with other through a predefined device. However, to Facebook's announcement, VR technology has risen exponentially. The term VR rapidly changes to Metaverse. But the main purposes of VR technology remain the same, entertainment, education, and social activities [10], [13], [14].

To use VR technology, one needs a specific device to run the VR application, and some people call it a VR headset or Head Mounted Display (HMD). Some prolific VR headset developers are Oculus and HTC, one of the first to develop marketable VR headsets [15], [16]. VR technology can also be classified based on the VR headset, whether a standalone or PCVR. Likewise, one has its advantages and disadvantages. To explain briefly, a standalone VR head is based on an Android OS, a smartphone with limited computational ability but providing cable-free VR experiences. On the other hand, PCVR, which uses a PC as its main computational source, has a bigger computational ability assisted by its GPU, but PCVR is limited to the need to use cable. Moreover, today, the trend is shifting to the

standalone side, which is cable-free with its limited computation [17].

Some VR applications that use standalone devices face the same problems: performance and immersion. Some previous VR applications use a standalone device, such as [7]–[9]. Those VR applications roughly face the same problems, performances, and immersion. To overcome these problems, one needs to apply such an optimization method to minimize the application load so that the standalone VR headset can run the application smoothly without causing a lagging effect which can cause motion sickness. Considering the above limitations, we take the VR application [8] to do an optimization method and compare the before and after optimization to understand the effect of optimization when developing a VR application.

To satisfy the need for achieving a low application load in VR applications, one needs to know where and what can be minimized so that performance can be maximized without sacrificing the immersion of VR technology which is the main thing of VR. So, the primary contributions of this paper are as follows:

- Create an optimization method so that the application load can be minimized.
- Performance comparison between Quest 1 and Quest 2 using the CBIVT application [8].
- User satisfaction and the significance of before and after optimization of the CBIVT application is analyzed.

II. MATERIAL AND METHOD

A. Human-Computer Interaction

The connection between humans and computers in the present advanced computing era presents challenges in combining human and machine abilities. Ubiquitous Computing, or a computation process that can be performed everywhere, increasingly depends on human-computer interaction. Historically, human-computer interaction was limited to stationary computers/laptops or desktops; however, today, nearly everyone uses their smartphone as the main source of information and entertainment, which is impossible without the improved computation that has grown in recent years. Due to the increase of connectivity that humans need in their daily activities, the interaction of recent human-computer relations must be molded following the needs of humans and not the other way around [18]. Moreover, following the improvement of computational power in a smaller package, many applications which were impossible in the past are popping out today. The most recent is the emergence of Virtual Reality. A small size with comparable computational power to a laptop is a huge advancement to make a Virtual Reality device with an unobtrusive and comfortable form factor much closer. The current technology offers a VR standalone HMD, bringing a virtual experience and virtual environment closer to people.

B. Critical Role in Assisting Robot

A game usually means for entertainment purposes only. Considering the development of video games today becoming more common, the development of games that serve as learning and training is more often found and developed. The advantages of serious games are that they serve as a method to implement the first concept using engaging content and interaction. Also, what makes serious games is the options for a pedagogical approach when teaching students [19], [20]. Serious games also negated the safety risk when doing some first-time training in hazardous environments with almost similar understanding [21]. Erickson et al. [22] developed a VR platform to explore to possibility of a simulation medium for interaction between humans and a Virtual Assistive Service Robot (VASR). The goal of this research is to revise the policy of Virtual Assistive Service Robots (VASR). Originally there was just one policy for a real robot, but to better simulate the movement, Erickson et al. [22] developed a virtual reality simulation to revise the policy. The simulation is done for doing a common life activity such as feeding, drinking, etc. Participants were instructed to sit in a wheelchair like real ASR patients. According to the participant's responses, the outcomes from the virtual reality simulation of the original policy are unsatisfactory and unreliable. Then, when the revised policy is used in virtual reality simulation, the participant's response is reliable and quite satisfactory.

C. Collaborative Robot

Collaborative robots, often called Co-Bots, function as operational partners alongside humans [23]. In addition to its industrial applications, Co-Bot is a so-called Assistive Service Robot (ASR) that assists people with their daily tasks. ASR helps elderly people at home by easing the burden of human activities or aiding with routine tasks. The most common tasks of an ASR include obtaining liquids or food, providing walking assistance, and providing walking support [24]. Co-Bot can do operational tasks typically performed simultaneously exclusively by humans or autonomous robots. The current predicament illustrates that the product manufacturing process has shifted from focusing on a few items with lengthy product lifetimes to many with short product lives. To enable this product transition, production and assembly lines must be modified [25]. Currently, Co-Bot is being used to aid human operators in assembling the product to suit market demand. When using this method, the mobility and trajectory of the Co-Bot provide difficulty. They offer human operators a whole new problem. The human operator working alongside Co-Bots could experience anxiety if the trajectory of the Co-Bots is erratic and not smooth in its movement and if the Co-Bots cannot stop their trajectory or continue working when a barrier is found in the way of the trajectory. Bilberg and Malik [25] used simulation and Digital Twin (DT) to develop and manage the Co-Bot's trajectory. This was done to determine whether the suggested and planned trajectory passes obstacles such as human operators or other objects. Bilberg and Malik [25] can imitate both the Co-bot and human DT methods to replicate linear- motor installation procedures. When a human DT enters the area of the DT Co-Bot trajectory, the Co-Bot can halt the trajectory since it has completed the assembly for the allocated motor linear sections.

On the other hand, human DTs only moved and functioned in response to a simulation that was specifically meant to prevent additional interaction between DT Co-Bots and human DTs. This was done to prevent human DTs from

becoming too dependent on DT Co-Bots. Such DT was also implemented by Falah et al. [26], where the early implementation of DT in the immersive virtual environment was done on a conveyor. It uses lightweight and flexible scripting program language, and it was deployed on a web browser to ease users' access, as they are not needed to download the application. However, due to the nature of its early development, this implementation is lacking on the real-time side of visualization and data.

D. VR Applications

Some of the recent VR applications that use a standalone VR headset that utilize the flexibility of movement in standalone VR headset are [7]–[9]. Where Al Hafidz et al. [27] use VR technology to create a simulation of a Normal Childbirth practicum for Nursey students. The system created can follow the scenario designed from the guidelines the Nursery gave. The interaction present on the application is also sufficient, where users can grab, hold and interact with objects. However, the immersion side of Hafidz et al. works is still lacking; the Figure shows that the environment still looks bland and monotone. Haz et al. [8] utilize VR technology to interact with a Virtual Assistive Service Robot (VASR), or we can name it a Collaborative Robot. While interacting with the Co-Bot in Virtual Environment, the user must also complete the scenario with the help of the Co-Bot. Co-Bot acts as a helper to bring tools for completing the scenario, such as a screwdriver, hammer, or power drill. But from the Figure shown, we know that the performance is bad because needing to reach near 96% GPU utilization level for the application to run. Even though the immersion is not as lacking as Hafidz et al. [27], the performance is worse, and this can happen because Haz et al. [8] use Quest 1 instead of Quest 2. The research was done by Muntahir et al. [9] to learn the instructions for infusion installation for Midwifery students by utilizing all forms of immersive technology, such as AR, MR, and VR. Muntahir et al. [9] compared each platform's performance to its user's importance and satisfaction level. However, the GPU Utilization level percentage is reaching 99% marks, which is undesirable and can cause motion sickness.

E. Proposed System

The system is intended to help users get familiar with Co-Bot interaction. To do this, the system needs an AI that enables user participation and a scenario for the user to complete with the aid of the Co-Bot, as seen in **Error! Reference source not found.**. The design system presented in **Error! Reference source not found.** is grouped into two boxes to simplify the system explanation.

Here at CBIVT, we separated the development process into two boxes, namely Co-Bot AI and the VR Scenario box. Where on Co-Bot AI is tasked to manage and operate the AI of Co-Bot, such its movement and behavioral decisions. On the other hand, the VR Scenario box manages and operates all the parameters to render a Virtual Environment and perform scenarios in order. In this box, many optimizations can be done to the scenario itself or the 3D assets used to build the Virtual Environment and the objects present in the Virtual Environment to minimize the application load and thus increase the performance.
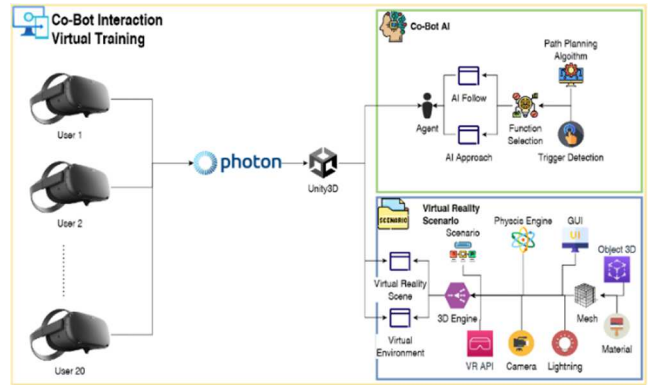

Fig. 1 System Design of CBIVT

1) Performance Optimization: Many 3D assets used in previous implementations, such as healthcare, educational, or company training, lack visualization and excess on the performance load [7], [13], [27]. What happens right now is we need to lower the performance load caused by the excess 3D asset and increase the visualization of 3D assets by improving the textures and adding baked lighting to enhance the visualization further. Optimization is needed when developing a VR Application supported by the limited capabilities of the deployment device.

2) 3D Asset Optimization: The most basic step to enhance the performance of VR applications by optimizing their 3D assets is using low- Poly objects. The low-Poly object did not mean to lower the fidelity or quality of existing 3D assets, which may have a larger poly count, but to minimize some unused vertices or edges which increase the poly count. The smallest examples of optimizing the poly count on a 3D asset can be seen in **Error! Reference source not found.**. In **Error! Reference source not found.**, a primitive cube is smoothed on 2 of its edges to create a round corner. The two objects above look similar from a distance, but they have a huge difference in the poly count. One on the front is a high-poly version of the one in the back. We can observe the difference in poly in **Error! Reference source not found.**.
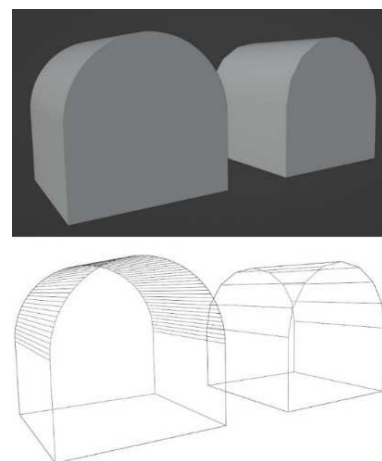

Fig. 2 Looks at How 3D Objects high-Poly Compared to low-Poly

The other step is to bake the detail of an existing High Poly object into the Low Poly object. This step is optional and can help make small details on the existing High Poly

into the Low Poly as a Normal map texture. Small dents like surface detail can be baked easily to save poly count.

3) Materials Optimization: Minimize the materials used in one 3D asset or combine multiple 3D assets as one material. Considering Unity 3D as our Development Software and Quest acting as our deployment device, we need to minimize the performance load. Another factor that generates a huge application load is the draw calls caused by a large count of materials. We can minimize the number of materials used by baking the textures of various objects into one huge texture or Atlas Map Textures to create one material that is used by multiple 3D assets, as we can see in **Error! Reference source not found.**. On **Error! Reference source not found.**, it is composed of multiple small textures from each 3D assets. This, in practice, can decrease the materials count, but we need a much bigger texture size not to lose detail from the original textures map. Another thing to consider is that not all objects can be made into the same materials and baked into one texture map. Objects that need to emit lights or objects that are designed to be transparent need to be separated from normal objects so that no texture error occurs, thus destroying the original texture map.

Fig. 3 One of the examples of Atlas Map Textures

4) Texture Visualization: Besides optimizing the geometry of a 3D object, we need to optimize its looks or so-called textures map. Because no matter how good the geometry or how small the poly count of a 3D object is, if the texture map looks bland or lacks detail, the 3D object will look bad. This can cause the overall VR experience to go downhill. The current standard of textures mapping that can make the 3D objects look very detailed is PBR (Physically Based Rendering) map pipeline. This rendering pipeline tries to imitate realism when generated through computer graphics. PBR map also enables developers to add more detail without adding more poly to optimize 3D objects. It can also add roughness or metalness, which works by imitating the bump on real-life objects located on a normal map generated using the PBR map pipeline.

Standard 3D objects commonly use PBR maps as their textures. PBR map generates many textures maps, but what is commonly used are Base Color/Albedo, Normal, Roughness, Emission, and Opacity. To quickly describe how each map affects the 3D objects, we can see Fig. 4. As shown in **Error! Reference source not found.** and these are some examples of PBR textures maps that are commonly used. Other texture maps may be used for other cases, but for the most part, 3D objects will use these four texture maps. Starting from **Error! Reference source not found.** (top left) is the Base Color, Albedo, or Diffuse texture map. This map affects the color of 3D assets and shows the base color, as the name suggests, of the 3D assets.
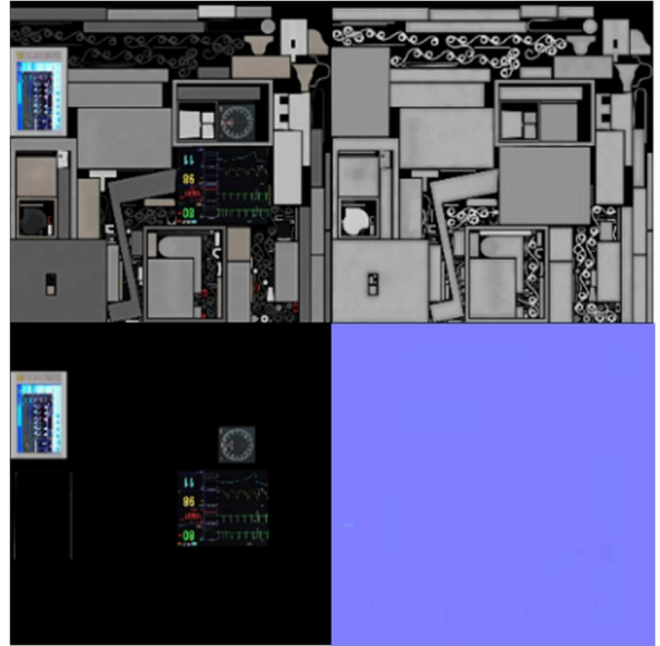

Fig. 3 Commonly used PBR Texture Map

Error! Reference source not found. (top right) is the roughness texture map. This map affects the smoothness of 3D objects. Some use this map to show how rough a surface is or display the shiniest metal. **Error! Reference source not found.** (bottom left) is the emission texture map. This map affects the 3D object which is designed to emit lights, such as lights, led, or screens. Then the last is **Error! Reference source not found.** (bottom right), which is the Normal texture map. This texture map affects the small detail that is designed when creating the textures. The normal map also increases realism by giving lights an uneven scatter effect, as if the 3D object's surface area is rough and bumpy without changing the geometry.

5) Lighting Baking: One of the most crucial parts when dealing with 3D assets used for VR applications is that we also need to optimize it inside the software development. In this case, we happen to use Unity 3D. Some basic optimizations that can be done inside Unity 3D are baking lights and static objects. Some 3D assets in the Virtual Environment are rarely used for interaction. Some of it is only used as a prop. However, Unity 3D won't know which object is used for interaction and which is not. So, Unity 3D saves some space for calculations related to its location and rotation. We can eliminate that by setting the prop unused to Static

Objects. Baking lights can be useful to minimize the calculation load when dealing with lights used inside 3D Environments. Some early adoptions of VR applications used in the healthcare sector [7] have not utilized this feature. Thus, the environment looks bland and monotone despite the engaging scenario. The process of baking lights in Unity 3D also involves the static objects that have been set before because Unity 3D knows that those objects will not interact. The effect of applying these methods can improve the overall looks of a Virtual Environment, which can be seen in

6)

7)

8) Fig. 4.

Fig. 4 The Effects of Applying Baking Lights and Static Object

The upper image in

Fig. 4 is before applying Texture Visualization and Baking Lights, and the lower image in
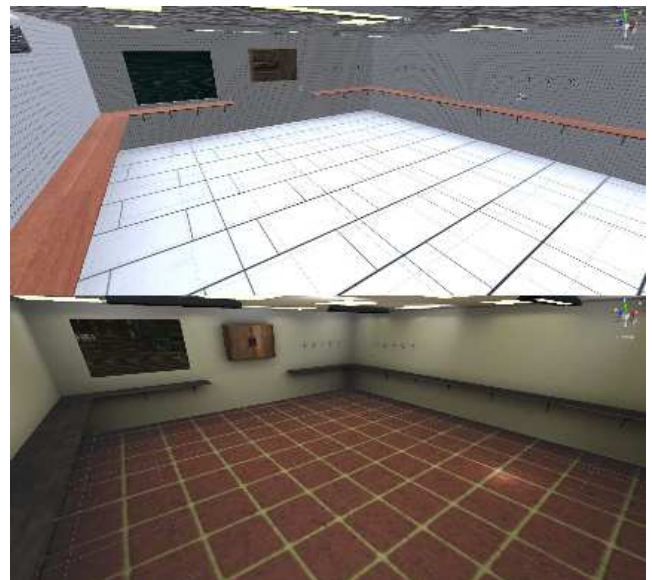

Fig. 4 is after applying all methods above. We can see the difference these methods offer the virtual environment can be more realistic, close to mimicking a real-world environment with its lighting, shadow, surface detail, and roughness.

F. Device Limitation

Considering the direction of VR technology is headed, a standalone VR headset is gaining more and more attention and recognition. Even with its performance limitations, since it is based on Android OS and uses a phone like System-on-a-Chip (SoC), which pales in comparison to the PCVR, which uses PC computing power with its GPU, the standalone VR headset still gains more interest. So, we must pay more attention to the standalone VR headset's performance limitation. Some brand that popularized the standalone VR headset is Oculus with their Quest 1 and 2. Those two devices will be deployed to see the effect of 3D assets optimization on its performance, and by knowing the specification of Quest 1 and 2 we can see the computing power difference Quest 2 has over the later Quest 1 in Table 1.

Component	Specification	
Device	Oculus Quest 1	Oculus Quest 2
OS	Android 7.1.1	Android 10
Chipset	Snapdragon 835	Snapdragon XR2
GPU	Adreno 540	Adreno 650
RAM	4GB	6GB
Refresh Rate	72 Hz	90 Hz
LCD	1440x1600 @eye	1832x1920 @eye
Max Tris Count	< 500.000 tris	< 1.000.000 tris
FPS Limit	72 FPS	72 FPS

Related to the limitations presented by the deployment device, we need to take advantage of the abovementioned method to have a pleasant experience when operating the standalone VR headset. By understanding the limitations of the deployment device, we can design the 3D assets used in the systems to not be more than the limitations stated in TABLE I1.

G. Design Scenario

This system's scenario is intended to be common work to improve the user experience when operating VR applications. Because VR is a novel experience for the user, an assembly of clothes wardrobe is chosen as the assembly scenario. We use the official IKEA assembly instructions for the HEMNES wardrobe as a guide for constructing the wardrobe [28]. The guide provides a detailed, step-by-step process for assembly, as well as the necessary tools. Using the guidelines as the basis for designing the scenario, we can simulate assembling the wardrobe, thereby increasing the system's realism. Each assembly of the drawers and frame requires nine and thirty-one steps, respectively, as shown in Table 2.

TABLE II
ASSEMBLY STEPS FOR DRAWER AND FRAMES

Steps	Drawers	Frame
1	Input pins on the top, middle, and bottom drawer panels	Attach the pins to the front left, rear left and front right, rear right
2	Install the left and right panels in the middle and bottom drawers	Connect the front left foot, rear left, front right, and rear right on the outer left and right panels of the wardrobe
3	Tighten the pin-fixing bolts on the left and right panels for the top middle, and bottom drawers	Tighten the retaining pins on the left and right panels of the wardrobe.
4	Insert the bottom panel into the middle and bottom top drawer	Attach the rubber base retainer on the front left foot, rear left, front right, and rear right
5	Install the back panel and tighten the bolts on the top middle and bottom drawer	Connect the innermost left and right panels to the outer left and right panels of the cabinet using the retaining pins
6	Install the drawer railings on the left and right panels on the middle and bottom drawers.	Install the drawer rails on the left and right innermost panels of the wardrobe using the bolts provided
7	Install the upper middle and lower drawer	Tighten the railing bolts on the cabinet's innermost left

handles by tightening the provided bolts.   and right panels using a plus screwdriver.

Table 2 is some of the steps used in the CBIVT. The scenario must be completed to complete. Users will be shown highlights around 3D objects related to the step. Another interface that helps the user to finish the scenario we utilize the type of meta-UI where a HUD (Head-Up Display) will appear in front of the user [29]. The HUD shows the current task and the objective to finish the current task, as we can see in **Error! Reference source not found.**.

Fig. 5 Highlighted Current Object Used (Left) Meta-UI HUD (Right)

In Fig. 6 (Left), the current task is displayed on the meta-UI HUD so the users can track the current steps better and quickly locate the used objects as stated on the HUD. As for, **Error! Reference source not found.** (Right), the current 3D objects used in the current task is shown in highlighted color around the 3D objects, and the 3D object destination is shown with transparent color giving a hint to the user that the 3D object is meant to be placed right there.

H. Co-Bot AI

This system's Co-Bot collaborates with the user using a path-finding algorithm. Follow user and approach user are two examples of how Co-Bot interacts with humans. The CO-Bot AI works in CBIVT can be seen in **Error! Reference source not found.**.
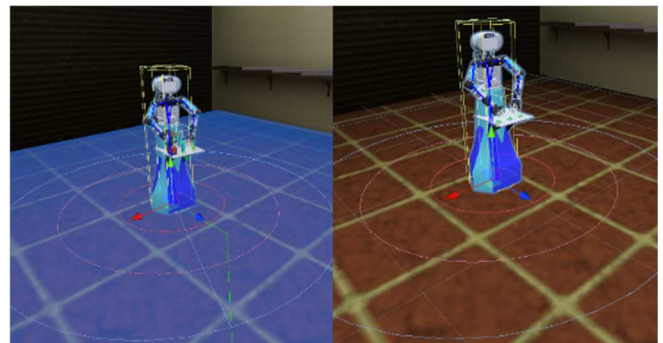

Fig. 6 Co-Bot AI in CBIVT

In **Error! Reference source not found.**, we can see some circles below the Co-Bot. Those circles act as a guide to measure the shortest path from the initial position to the end position, and as the range to where pick the next waypoint in case there is a new end position when the Co-Bot is moving. The green lines that appeared in **Error! Reference source not found.** (Left), are the path segment of the path of Co-Bot

AI that will traverse from its initial position to its end position. Co-Bot is programmed to assume its natural state is to follow the user at a certain predefined distance. This behavior enables the Co-Bot to follow the user and be ready whenever the user gives a signal to approach and gives tools needed for the assembly procedure. The approach to user behavior allows Co-Bot to move closer to the user to provide the user with the required scenario-building tool. In this approach, user behavior appears when a predetermined condition is met. The condition is when the user raises his or her hand, the system's designed trigger will activate. Co-Bot can also resume its follow behavior in response to a user-initiated touch.

III. RESULT AND DISCUSSION

In this section, the results of the system's experimental testing are detailed. Table I explains the experimental apparatus. While utilizing the system, we must be aware of its degree of significance, level of satisfaction, and performance. It is necessary to conduct an evaluation to determine the technical and operational proficiency of the system. This experiment employed the PIECES framework, which stands for Performance, Information and Data, Economy, Control and Security, Efficiency, and Service, to evaluate the aforementioned factors.

Each participant will be assigned to use the VR device and complete the scenario presented by the system as part of the process of evaluating performance and usage. This experiment's participants consist of 47 people combined from High School, Vocational High School, and its teacher. All participants stated that they had not tried any Virtual Reality experience.

To collect data from experiment participants, questionnaires are used. First, the performance of the Co-Bot AI's two user-following, and user-approaching features will be evaluated. The second evaluation consists of verifying the system-designed scenario's continuity. And finally, it is necessary to evaluate the system's overall performance, comprised of the parameters mentioned previously. The questionnaires use the five-Likert scale method illustrated by Equation 1.

$$AS = \frac{TSQ}{NQ} \qquad (1)$$

Where:
AS   is the Average Score of Parameters
TSQ is the Total Score of the Questionnaire
NQ   is the Number of Questionnaire

An evaluation was done to determine the system's performance utilizing the Unity Developer's OVR Metrics, an official statistic monitoring application[30]. This application can determine the system's average frame rate when deployed on a device, as shown in Table 1. Additionally, we can assess the system's load for future development by measuring the GPU utilization levels. We measure the FPS and the GPU utilization level percentage from each Oculus Quest 1 and 2 before and after the optimization to see and assess the difference it makes and how efficient it is if we apply the optimization method to all future VR application development processes.
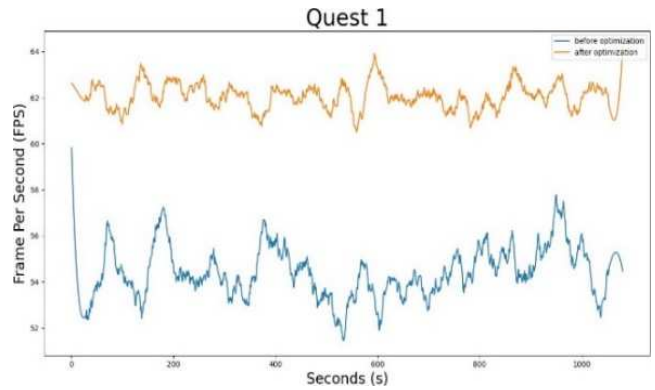

Fig. 8 FPS on Quest 1

From the first metric depicted in **Error! Reference source not found.**, the FPS level dropped low for the before optimization graph. It appears that because the application load is too high, the application took a sudden drop in FPS. Moreover, after applying the optimization method, we can see the numbers are greatly increased on the orange graph. Even though the Oculus standard application needs to have a steady 72 FPS, we understand that the device computation level is a limitation we cannot overcome. Thus, we need to try on the newer Oculus Quest 2.
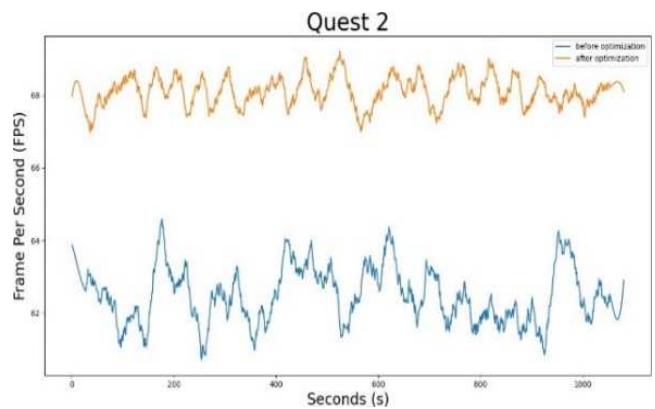

Fig. 7 FPS on Quest 2

Fig. 9 shows the graph for the Quest 2 FPS measurement on the before and after optimization of CBIVT. Because the specifications are different and better than Oculus Quest 1, the overall FPS is better even before optimization. Moreover, after optimization, the FPS measurement on Oculus Quest 2 is nearing the 72 FPS mark accepted by the Oculus Developer. The FPS measurement on Quest 1 increased by 14.02% percent after applying the optimization method, and on Quest 2, the FPS measurement increased by 8.99%.
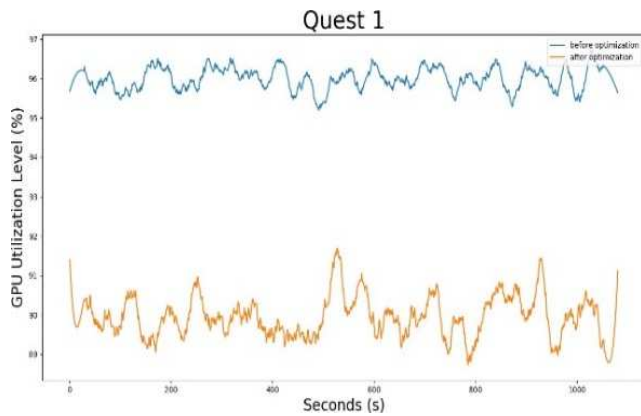
Fig. 8 GPU Utilization Percentage on Quest 1

The third metric is shown in Fig. 8, the GPU utilization percentage level on Quest 1 is nearing 100%, and it is best to avoid developing an application where the GPU utilization per- centage level is too high because it can cause a lagging effect which can make the users feel nausea or what is commonly called motion sickness. Then after applying the optimization method, we can lower the GPU utilization level even if it is still on the 90% mark, where it is still not recommended and can still cause a lagging effect.
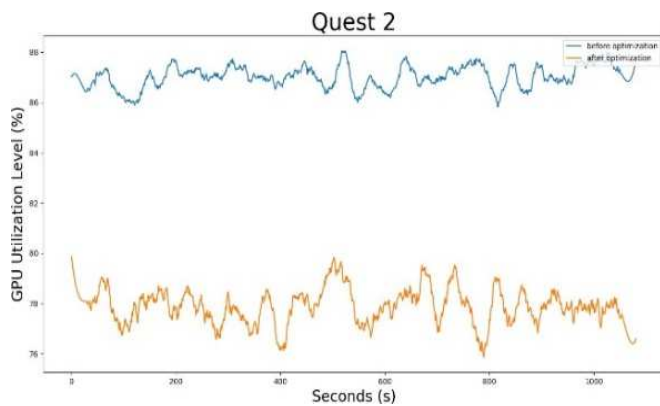

Fig. 9 GPU Utilization Percentage on Quest 2

In Fig. 9, we can see the graph for the Quest 2 GPU utilization level percentage on the before and after optimization of CBIVT. The Oculus Quest 2, before optimization, reached the upper 80% mark, and after applying the optimization method, the GPU utilization level percentage dropped to near the upper 70% mark. Helped by the Oculus Quest 2's better computing performance and optimization method, the GPU utilization level is at a safe level and does not cause a lagging effect that can lead to motion sickness. The GPU utilization level percentage on Quest 1 decreased by 6.73% after applying the optimization method. Moreover, in Quest 2 the GPU utilization level percentage decreased by 11.72% after applying the optimization method. The average FPS reflects the capabilities of the device during rendering and gameplay. The greater the number, the more frames per second are rendered, thus the name frame per second or FPS. A system with low FPS produces a lagging effect, causing motion sickness or dizziness in the user [30]. The GPU and CPU usage levels show the device's GPU and CPU capabilities to calculate and render the 3D object and materials necessary to generate a Virtual Reality experience.

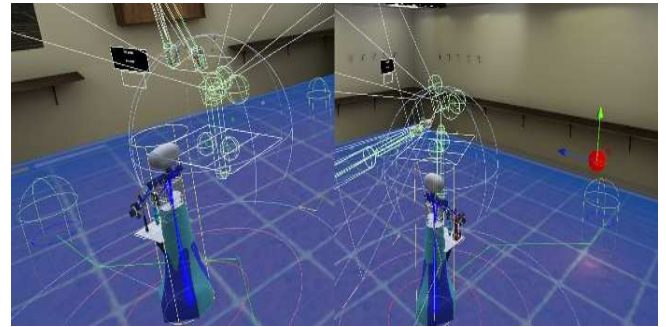As a result of the system's low CPU consumption, future enhancements may be made to improve the system further.

Fig. 10 C-Bot Approach User (Left) Co-Bot Follow User (Right)

As shown in Fig. 10, the function of the CO-Bot AI that follows the user's location and approaches the user using a path-finding algorithm was evaluated to determine the user's experience when interacting with the Co-Bot AI. The green lines in Fig. 10 represent the Co-Bot's route, with the path altering in real time depending on the user's distance from the Co-Bot. And the approach behavior leads the Co-Bot to approach the user before halting until an input is received, at which time the Co-Bot resumes its normal behavior, which is to follow the user. In addition, a typical work from the real world is created to be the system's scenario to reduce the strain on participants still adjusting to the VR experience. The scenario mimics the procedure of building IKEA furniture. The step-by-step instructions for the assembly scenario may be found on the official website for the product. The PIECES framework was used to determine the significance and acceptability of each parameter mentioned above, with an analysis of each point provided in Table 3.

TABLE III
EVALUATION OF THE PIECES MODEL

Analysis Type	Analysis Results
Performance	The CBIVT performance slightly got better after applying the optimization method and did not cause nausea after using it for a prolonged duration.
Information and Data	The information required for completing the scenario is well-informed and well-annotated, so the user can easily complete the scenario.
Economics	The virtual environment and 3D assets shown in the CBIVT become more realistic and more computationally friendly, marked by better performance, as shown in the Figure above.
Control and Security	The user can easily control which option of locomotion they can use. Users can select stationary boundaries or room-scale boundaries for limited movement space if a large space is present. These options can reduce the user's stress when experiencing Virtual Reality applications for the first time.
Efficiency	Depending on CBIVT, the experience is made simpler by the user and Co-intuitive Bot's interaction. The CBIVT is made easier to use by the deployment device selection. Additionally, CBIVT offers users a variety of movement options to aid in navigating the

Service	virtual environment. The CBIVT is easy to learn because the next object is highlighted, and the HUD shows what users need to do to complete the current task.

Moreover, the average score for each basic and acceptable level according to the PIECES framework is shown in Table 4. The analysis results are found in Table 3, in addition to the range value of the five-Likert scale in Table 4.

TABLE IV
SCORE RANGE OF THE IMPORTANT AND SATISFACTION LEVEL

Value Range	Important	Satisfactory
1 – 1.79	Very Important	Very Dissatisfied
1.8 – 2.59	Not Important	Not Satisfied
2.6 – 3.39	Sufficiently Important	Sufficiently Satisfied
3.4 – 4.91	Important	Satisfied
4.92 – 5	Very Important	Very Satisfied

The outcomes of applying the PIECES approach to determine the significance and degree of user satisfaction are shown in Table 5.

TABLE V
RESULTS OF PIECES FRAMEWORK ON IMPORTANT AND SATISFACTION LEVEL

Analysis Type	Important	Satisfactory
Performance	4.28	4.13
Information and Data	4.35	4.21
Economics	4.41	4.25
Control and Security	4.18	3.98
Efficiency	4.31	4.29
Service	4.21	4.14

Fig. 11 Participants Utilizing CBIVT Application with VR Headset

As seen in Fig. 11, the PIECES framework analysis was derived from participant surveys. Participants were then asked to complete a questionnaire after using CBIVT. It demonstrates that users are happy with the performance of the CBIVT and recognize its significance. In addition, customers report no lag effects while using the CBIVT that might induce motion sickness. The functionality of Co-Bot AI's features is also error-free and does not cause the user any annoyance. The reaction to the information and data is a sense of satisfaction and significance since the information provided in CBIVT is essential for the users to complete the needed scenario, allowing them to comprehend the Co-Bot interaction further. To understand a Co-Bot interaction, users must engage with a Co-Bot, which necessitates a certain level of competence.

Consequently, users experience a sense of significance and fulfillment while utilizing CBIVT. Because there are options for traversing the virtual environment and the control for interacting with the item is easy, users feel pleased regarding control and security. When using the CBIVT, the users recognize the significance of the safety area since it protects them from objects that might cause them to trip and fall. Users are happy with the chosen deployment device due to the absence of cables for controlling or powering the VR gadget. In addition, it conveys to consumers the significance of utilizing the proper deployment equipment for VR experiences. Users are pleased with the service since CBIVT offers an easy user interface and provides hints to assist them in completing the scenario. Users recognize the significance of CBIVT because of its tiny application size and minimum system requirements, which reduce the cost of purchasing a VR device.

IV. CONCLUSION

To familiarize humans with Co-Bot interaction, research was undertaken to establish a simulation of interaction with Co-Bot, which was subsequently integrated with ordinary daily life activities on a Virtual Reality device known as CBIVT. The CBIVT has two sides: the Co-Bot AI, which manages the Co-Bot's behavior, and the Virtual Reality Scenario, which controls the system's operation and the scenario's continuity. Participant information was collected through questionnaires to justify the study. The user's significance and degree of pleasure were estimated using the PIECES approach as the foundation. The findings of the surveys indicate that users perceive the importance of CBIVT with an average score of 4.26 and that users perceive the pleasure of CBIVT with an average score of 4.16. This indicates that consumers believe that CBIVT is essential for familiarizing the relationship between humans and Co-Bot in a virtual world and that they are pleased with its intuitive interface and smooth performance, which eliminates motion sickness.

Moreover, the performance issue that commonly appears when dealing with standalone VR headsets, which are low FPS value and high GPU utilization level percentage, which can cause a lagging effect, has been reduced substantially. A comparative experiment was done for each Quest 1 and 2 version using the before and after optimized version of CBIVT to know the improvements in FPS and GPU utilization levels. From the experiments, we know that after applying the optimization method, the Quest 1 FPS value increased by 14.02%, and the Quest 2 improved by 8.99%. Quest 1 decreased by 6.73% for GPU utilization level percentage, and Quest 2 decreased by 11.72%. From these numbers, we can understand that we can develop better performance VR applications with better visualization texture by performing the optimization method.

REFERENCES

[1] S. S. Oyelere, N. Bouali, R. Kaliisa, G. Obaido, A. A. Yunusa, and E. R. Jimoh, "Exploring the trends of educational virtual reality games: a systematic review of empirical studies," *Smart Learning Environments*, vol. 7, no. 1, pp. 1–22, Dec. 2020, doi: 10.1186/S40561-020-00142-7/TABLES/8.

[2] J. Radianti, T. A. Majchrzak, J. Fromm, and I. Wohlgenannt, "A systematic review of immersive virtual reality applications for higher education: Design elements, lessons learned, and research agenda," *Comput Educ*, vol. 147, p. 103778, Apr. 2020, doi: 10.1016/J.COMPEDU.2019.103778.

[3] H. W. Lee, S. Kim, and J. P. Uhm, "Social Virtual Reality (VR) Involvement Affects Depression When Social Connectedness and Self-Esteem Are Low: A Moderated Mediation on Well-Being," *Front Psychol*, vol. 12, p. 5574, Nov. 2021, doi: 10.3389/FPSYG.2021.753019/BIBTEX.

[4] X. Wang, G. W. Young, C. Mc Guckin, and A. Smolic, "A Systematic Review of Virtual Reality Interventions for Children with Social Skills Deficits," *TALE 2021 - IEEE International Conference on Engineering, Technology and Education, Proceedings*, pp. 436–443, 2021, doi: 10.1109/TALE52509.2021.9678808.

[5] Y. K. Dwivedi *et al.*, "Metaverse beyond the hype: Multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy," *Int J Inf Manage*, vol. 66, p. 102542, Oct. 2022, doi: 10.1016/J.IJINFOMGT.2022.102542.

[6] E. D. Fajrianti *et al.*, "Design and Development of Human Anatomy Learning Platform for Medical Students Based on Augmented Intelligence Technology," *International Electronics Symposium 2021: Wireless Technologies and Intelligent Systems for Better Human Lives, IES 2021 - Proceedings*, pp. 195–202, Sep. 2021, doi: 10.1109/IES53407.2021.9594053.

[7] J. Khatib Sulaiman *et al.*, "Implementasi Modul Pembelajaran Berbasis Virtual Reality untuk Pendidikan Kedokteran," *Indonesian Journal of Computer Science*, vol. 11, no. 2, Aug. 2022, Accessed: Nov. 30, 2022. [Online]. Available: https://ijcs.stmikindonesia.ac.id/ijcs/index.php/ijcs/article/view/3059

[8] A. L. Haz, Muhtadin, I. Ketut Eddy Purnama, M. H. Purnomo, and S. Sukaridhoto, "Virtual Reality Application for Co-Bot Training," *IES 2022 - 2022 International Electronics Symposium: Energy Development for Climate Change Solution and Clean Energy Transition, Proceeding*, pp. 644–650, 2022, doi: 10.1109/IES55876.2022.9888286.

[9] M. I. Muntahir et al., "Implementation of Immersive Technology on Medical Education," IES 2022 - 2022 International Electronics Symposium: Energy Development for Climate Change Solution and Clean Energy Transition, Proceeding, pp. 651–657, 2022, doi: 10.1109/IES55876.2022.9888379.

[10] K. Hanfati, S. Sukaridhoto, D. K. Basuki, R. P. N. Budiarti, E. D. Fajrianti, and I. A. al Hafidz, "Design and Implementation of WebXR Health Learning Module Application," *IES 2022 - 2022 International Electronics Symposium: Energy Development for Climate Change Solution and Clean Energy Transition, Proceeding*, pp. 632–637, 2022, doi: 10.1109/IES55876.2022.9888325.

[11] D. M. Barry and H. Kanematsu, "Virtual reality enhances active student learning," *Procedia Comput Sci*, vol. 207, pp. 408–415, Jan. 2022, doi: 10.1016/J.PROCS.2022.09.075.

[12] R. J. García-Hernández and D. Kranzlmüller, "NOMAD VR: Multiplatform virtual reality viewer for chemistry simulations," *Comput Phys Commun*, vol. 237, pp. 230–237, Apr. 2019, doi: 10.1016/J.CPC.2018.11.013.

[13] A. L. Haz *et al.*, "Study of Factor Analysis on Game-Based Learning for Scaffolding Installation Activities in Education and Training for Indonesian State Electricity Company Employees," *International Journal of Information and Education Technology*, vol. 12, no. 11, pp. 1126–1136, Nov. 2022, doi: 10.18178/IJIET.2022.12.11.1730.

[14] E. D. Fajrianti *et al.*, "Application of Augmented Intelligence Technology with Human Body Tracking for Human Anatomy Education," *International Journal of Information and Education Technology*, vol. 12, no. 6, pp. 476–484, Jun. 2022, doi: 10.18178/IJIET.2022.12.6.1644.

[15] Oculus documentation | oculus developers. (n.d.). https://developer.oculus.com/documentation/ (accessed Dec. 06, 2022).

[16] Vive Wave. VIVE Wave - Developer Resources. (n.d.). https://developer.vive.com/resources/vive-wave/ (accessed Dec. 06, 2022).

[17] N. Rendevski et al., "PC VR vs Standalone VR Fully-Immersive Applications: History, Technical Aspects and Performance," 2022 57th International Scientific Conference on Information, Communication and Energy Systems and Technologies, ICEST 2022, 2022, doi: 10.1109/ICEST55168.2022.9828656.

[18] A. M. Ghanbari, S. Ghanbari, and Y. Norouzi, "A new approach to architecture of human-computer interaction," *2017 IEEE International Conference on Smart Instrumentation, Measurement and Applications, ICSIMA 2017*, vol. 2017-November, pp. 1–4, Mar. 2018, doi: 10.1109/ICSIMA.2017.8311991.

[19] M. Urgo, W. Terkaj, M. Mondellini, and G. Colombo, "Design of serious games in engineering education: An application to the configuration and analysis of manufacturing systems," *CIRP J Manuf Sci Technol*, vol. 36, pp. 172–184, Jan. 2022, doi: 10.1016/J.CIRPJ.2021.11.006.

[20] D. Bagus, K. Setiawan, P. Arisaputra, J. Harefa, and A. Chowanda, "Designing Serious Games to Teach Ethics to Young Children," *Procedia Comput Sci*, vol. 179, pp. 813–820, Jan. 2021, doi: 10.1016/J.PROCS.2021.01.069.

[21] F. Yang and Y. Miang Goh, "VR and MR technology for safety management education: An authentic learning approach," *Saf Sci*, vol. 148, p. 105645, Apr. 2022, doi: 10.1016/J.SSCI.2021.105645.

[22] Z. Erickson, Y. Gu, and C. C. Kemp, "Assistive VR Gym: Interactions with Real People to Improve Virtual Assistive Robots," *29th IEEE International Conference on Robot and Human Interactive Communication, RO-MAN 2020*, pp. 299–306, Jul. 2020, doi: 10.48550/arxiv.2007.04959.

[23] M. Knudsen, and J. Kaivo-Oja, "Collaborative Robots: Frontiers of Current Literature," *Journal of Intelligent Systems: Theory and Applications*, vol. 3, no. 2, pp. 13–20, Sep. 2020, doi: 10.38016/JISTA.682479.

[24] B. Teke, M. Lanz, J. K. Kämäräinen, and A. Hietanen, "Real-time and Robust Collaborative Robot Motion Control with Microsoft Kinect ® v2," *2018 14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications, MESA 2018*, Aug. 2018, doi: 10.1109/MESA.2018.8449156.

[25] A. Bilberg and A. A. Malik, "Digital twin driven human–robot collaborative assembly," *CIRP Annals*, vol. 68, no. 1, pp. 499–502, Jan. 2019, doi: 10.1016/J.CIRP.2019.04.011.

[26] M. F. Falah, S. Sukaridhoto, M. U. H. al Rasyid, and H. Wicaksono, "Design of Virtual Engineering and Digital Twin Platform as Implementation of Cyber-Physical Systems," *Procedia Manuf*, vol. 52, pp. 331–336, Jan. 2020, doi: 10.1016/J.PROMFG.2020.11.055.

[27] I. A. al Hafidz et al., "Design of Collaborative WebXR for Medical Learning Platform," International Electronics Symposium 2021: Wireless Technologies and Intelligent Systems for Better Human Lives, IES 2021 - Proceedings, pp. 499–504, Sep. 2021, doi: 10.1109/IES53407.2021.9593951.

[28] "HEMNES chest of 3 drawers, white stain, 108x96 cm | IKEA Indonesia." https://www.ikea.co.id/en/products/chests-and-other-furniture/chest-of-drawers/hemnes-art-10355689 (accessed Dec. 06, 2022).

[29] Firmanda, A., Sukaridhoto, S., Rante, H., Miranto, C., Fajrianti, E. D., & Prasetyaningsih, S., "Meta-Interface Analysis for Interaction in Mobile Augmented Reality," *Indonesian Journal of Computer Science*, vol. 11, no. 2, Aug. 2022, Accessed: Dec. 06, 2022. [Online]. Available: http://3.8.6.95/ijcs/index.php/ijcs/article/view/3058

[30] Monitor Performance with OVR Metrics Tool: Native/android | Oculus Developers. (n.d.). Monitor Performance With OVR Metrics Tool: Native/Android|Oculus Developers. https://developer.oculus.com/documentation/native/android/ts-ovrmetricstool (accessed Dec. 06, 2022).