Optimizing Immersion: Analyzing Graphics and Performance Considerations in Unity3D VR Development

.

ABSTRACT

Aims: Virtual reality (VR) is a new approach that gives users an immersive experience. VR applications frequently experience performance bottlenecks due to the high processing cost of displaying real-time animations multiple times (for both eyes) and the limited resources of wearable devices, performance optimization serves a significant part in VR application development. The goal of this paper is to investigate approaches and techniques available in Unity3D to improve the immersion of VR in virtual reality. This entails studying visuals and performance elements to design more fluid and engaging VR apps.

Study design: Quantitative analysis

Methodology: Systematic literature reviews, Consensus on the design space for VR-based education would greatly benefit future advances. To do this, we use a systematic mapping technique to the literature, extracting essential data from papers indexed in four academic online databases.

Results: VR applications include extra aspects like graphics rendering and actual time animation, performance improvements for VR apps might differ significantly from those for standard software. Fully immersive virtual environments vary from standard virtual reality environments in that they can record entire body movements. This skill enables users to engage with other avatars, computer-controlled entities, and artifacts in the virtual world using their complete range of physical activities. VR performance optimization, in essence, relates to the methods, tactics, and procedures used to ensure that VR experiences function as smoothly as feasible.

Conclusion: This optimization might be performed on different kinds of technology, particularly low-end devices, without sacrificing visual fidelity or immersive experience. VR is well-known for. It is critical to optimize visuals as well as performance in Unity3D VR apps to produce immersive and engaging experiences. Balancing great aesthetics with seamless performance necessitates a thorough grasp of Unity's features as well as a dedication to constant optimization.

Keywords: Virtual reality, Performance, Graphical, Unity3D, VR applications

1. INTRODUCTION

Humans depend on perception to support their sense of reality. and appealing to one's perception of their surroundings can aid in improving awareness of situations and decision-making abilities. This approach can also promote user participation if the visual depiction is good enough. An interaction approach is the combination of all the technical elements that represent both input and output and allows the user to complete a task. Combining user interface design ideas with user experience principles that are more closely related to the actual 3D environment might generate interesting outcomes [8]. Moore's Law describes how technologies are evolving and becoming more efficient. Virtual Reality (VR) and Augmented

Reality (AR) have recently become a thriving sector on the rise in recent years. Since the Oculus Rift Kickstarter campaign in 2012 [9], visionaries have wanted to bring virtual reality to the business market and consumers. The idea is to use these technologies to improve and build a realistic experience that helps the human condition further. Simulation VR and augmented reality may help in discovery while utilizing spatial and geographic domains. VR has led to significant breakthroughs in game creation, most notably in the reproduction of realistic first-person perspectives [1]. Game engines like Unity3D may create user experiences that mix computer visuals, interactivity, creativity, and so on [22]. They've also been tried out for tactics like situational awareness and information visualization. Unity3D's skills as an efficient visualization tool are being expanded into sectors other than gaming, including research and education [21]. It is also rapidly developing into one of the most important augmented and virtual reality development tools [13].

1.1. Immersion

Computer-simulated immersion is utilized to confuse the genuine and untrue. The VR experience provides consumers with an experience of immersion in "intentions." Everything in the simulation appears and sounds realistic. Users will see themselves as players in the virtual environment rather than spectators. It allows users to completely experience themselves as characters in the virtual world while using it, delivering an immersive experience. Wearing interactive gear like head-mounted smart gadgets and data gloves, participants immerse themselves in the virtual world. Computers produce a true three-dimensional picture according to the user's physiological attributes. At this stage, the user will transition from spectator to participant, immersing themselves in the virtual world and getting immersed in it [6].

1.2. VR and immersion

Immersion, presence, and interaction are recognized as the essential qualities of VR technology. The extent to which a user may alter the VR world in real time is referred to as interactivity. Presence is defined as "the individual's perception of being present in a single location or environment, even when physically situated in another". While scholars generally agree on the term’s interactivity and presence, they disagree on the idea of immersion [7].

The authors provide a different perspective on the advantages of immersion and engagement in learning outcomes. Learners who used an immersive HMD got more interested, spent a longer period on learning tasks, and enhanced their mental in nature psychomotor, and affective capacities. This study, however, indicates various aspects that may act as either reinforces or barriers to immersion and awareness. For example, the visual quality of VR, as well as being conscious while using VR, may reduce the sense of presence. Individual personality traits may potentially be linked to restricted skill development using VR technology [10]. "Immersion" is defined as "the degree in which displays on computers have the ability of offering an accessible, broad, surrounding, and intense illusion of reality." This includes the degree to which physical reality is removed, a diversity of sensory inputs, the length of the surrounding environment, and the resolution and precision of the display [2]. The technological aspects of a VR system, like as its frame rate or panel image quality, influence the amount of immersion perceived by the user. In contrast, immersion is described psychologically as a state of awareness in which the person senses sensory separation from reality. According to this viewpoint, the perceived level of absorption varies from individual to individual and is hardly influenced by technical qualities.

1.3. Unity 3D

Unity is a 3D engine that works across several platforms. The most crucial aspect of Unity 3D is the way it allows engineers and designers to collaborate in one environment. It is a straightforward, easy, and quick development tool. The Unity engine's object engine blends real-life visual effects into the gaming experience. It might include the collision element in the environment's components and use it to detect whether or not the two items contact in real time. Furthermore, Unity 3D has fully integrated code editing as well as a straightforward design and editing interface [15]. Model assets from several 3D modeling software, including modeling components, graphics, and character bone bindings, may be recognized [16].

Fig 1. Application development in Unity3D

Source: medium.com

1.3.1. Features of Unity 3D

Unity 3D, the famous freely available engine that contain the following features [21]:

- It offers real-time work preview, a strong element checker, and a hierarchical framework for development in addition to visual editing.
- Several export platforms. Unity can simply export content such as.exe files for Windows and.dmg files for Mac OS X. Furthermore, assets may be directly exported to the browser for use with the Unity Web Player plug-in.
- Import resources automatically. Drag and drop may be used to directly load model designs or code into the Unity editor. External resource modifications will also be reflected in real-time in the execution of the task. Most current standard 3D modeling formats, including 3DMax and SketchUp, are supported by Unity 3D, and other software development drafts may be smoothly docked, providing developers with more ease when using third-party models.
- Unity takes advantage of the most popular and appropriate graphics libraries, such as Microsoft Direct and OpenGL, as well as built-in compatibility with Nvidia's PhysX physics engine.
- Game script creation is built on the multi-platform freely available version Mono of Microsoft.NET Framework. Developers may execute scripts written in JavaScript or C# thanks to Unity's built-in API support.
- Unity supports widely recognized audio and video formats and can reduce various media assets for game development purposes, Furthermore, Unity has a robust official app store and numerous built-in resources. It enables a multi-person connection to the network function given by a third-party package; numerous possibilities are available, including RakNet, Photon, and SmartFoxServer.

1.3.2 Unity 3D Engine modules

The Unity 3D engine is composed of four core modules, the primary functionalities of which are as follows (Table 1) [17].

Table 1. Unity 3D Engine Core Modules Overview

Module Name	Description
Module Overview	◦ Object module ◦ Event processing module ◦ Camera module ◦ Module for rendering
Object Module	◦ Components, scenarios, and buttons are staging objects ◦ Objects are fundamental units. ◦ Objects can be assigned to a script code
Event Processing Module	◦ Alters the state of stage elements ◦ Modifies properties of the stage object ◦ Used to modify the level object's characteristics
Camera Module	◦ Inspired by 3D stage design ◦ User's device content varies with the camera's content ◦ Scripts determine camera location and angle ◦ User validation of stage interface, situation, or role modifications ◦ Camera switches for transitioning between stage scenarios
Module for Rendering	◦ Calculates real-time effects ◦ Includes models, animations, lighting, and special effects ◦ Displays effects on the screen ◦ High skill level impacts stage output quality

Source : author's development

This timeline demonstrates that VR technology is not new, but with the introduction of low-cost VR headsets like the Oculus Rift in 2014 and the HTC Vive in 2016, it has become more widely available. As a result, today's VR applications benefit significantly from visuals, such as interactivity and sensor tracking via Head-Mounted Displays (HMDs), that boost the degree of immersion [18]. Hruby et al. (2020) conducted the first quantitative comparison of HMD vs screen display [12]. The following conditions must be followed in order to use this sort of electronic devices for an immersive experience: An IDE (such as a game engine) must be used to create and texture a virtual 3D environment, a head-mounted display (HMD) must be used to connect the created and executed VR application to the HMD, and a controller and HMD must be used to control and track the user's movements. Immersion is an essential feature of VR since it defines how an immersive environment affects a user, leading the feeling of having been treated to false stimulation disappear into background noise to the point where the VR can be confused for the real one. However, it is recommended that the number of frames per second (fps) be at least 90 to enable a smooth

and accurate representation of the VR application when using the VR headset. The lower the frame rate of the VR software in the HMD, the greater the likelihood that the user would feel motion sickness (or cyber sickness) owing to latency, which can cause dizziness, nausea, or overall discomfort when using the headsets for virtual reality [12].
As a result, designing a VR application calls for managing visual representation with real-time performance. As VR systems with massive processing capability are developed, much research on rendering algorithms, data transfer, and processing in real time has been conducted. The performance parameters are classified as "system size", "rendering method" (software), and "VR hardware". Even with small systems in past years, several challenges developed due to inadequate VR hardware performance. VR hardware problems, for example, including "motion tracking" and "communication to distributed parallel-rendering computers," but these challenges have now been handled, even for high-end systems like CAVE. Algorithms have been accepted and commodified while using HTC-VIVE, Oculus Rift, and other technologies using programs like Unity and Unreal Engine. The development of fast and effective rendering techniques is crucial, demanding system size consideration when considering VR for CMS used by no specialists or beginners. The "data transfer" problem from the CPU to the GPU is considered as an issue inside the framework of the size variable [14].

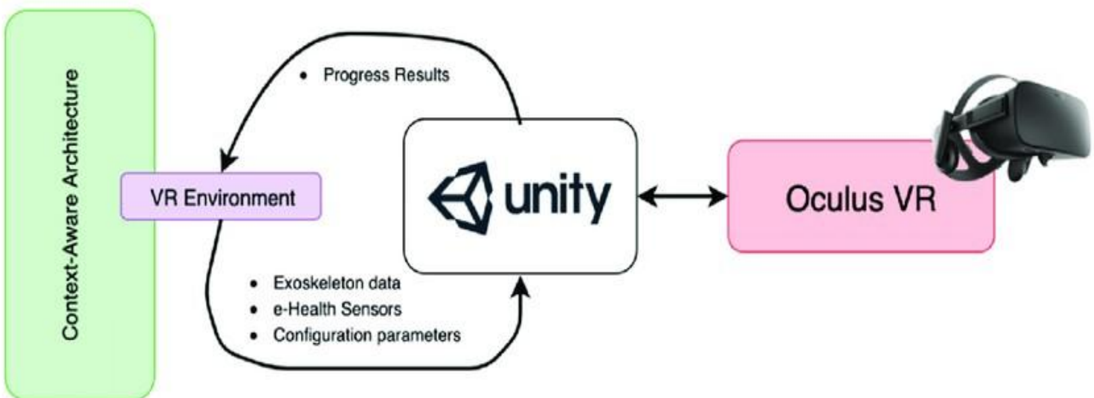

Fig 2. The Unity 3D engine and the Oculus system serve as a base for the virtual reality world [5].

The goal of this paper is to investigate approaches and techniques available in Unity3D to improve the immersion of VR in virtual reality. This entails studying visuals and performance elements in order to design more fluid and engaging VR apps.

2. METHODOLOGY

Consensus on the design space for VR-based education would greatly benefit future advances. To do this, we use a systematic mapping technique to the literature, extracting essential data from papers indexed in four academic online databases. Our research aims to do the analysis of immersive virtual reality and their performance and graphical optimization in unity 3D. Systematic literature reviews, such as those given by Radianti, Majchrzak, Fromm, & Wohlgenannt (2020), have frequently been used to get comprehensive insights into a certain research topic [19]. Furthermore, the author advocates doing mapping research, which is a sort of systematic research review, to answer concerns about how to organize a wide region, critical topics in this subject, and research trends. A mapping study, as opposed to a traditional systematic review, evaluates a bigger topic and categorizes primary papers for research within the specific region under consideration [3].

The documenting of significant works and assessment of the collected materials is basically presented in Fig 3. Besides this, Fig 3 is created using the PRISMA approach, which is illustrated below. As a result, the literature survey has been regarded as one of the most important components of the research report because it assists both the writers and the readers in learning and assessing many.

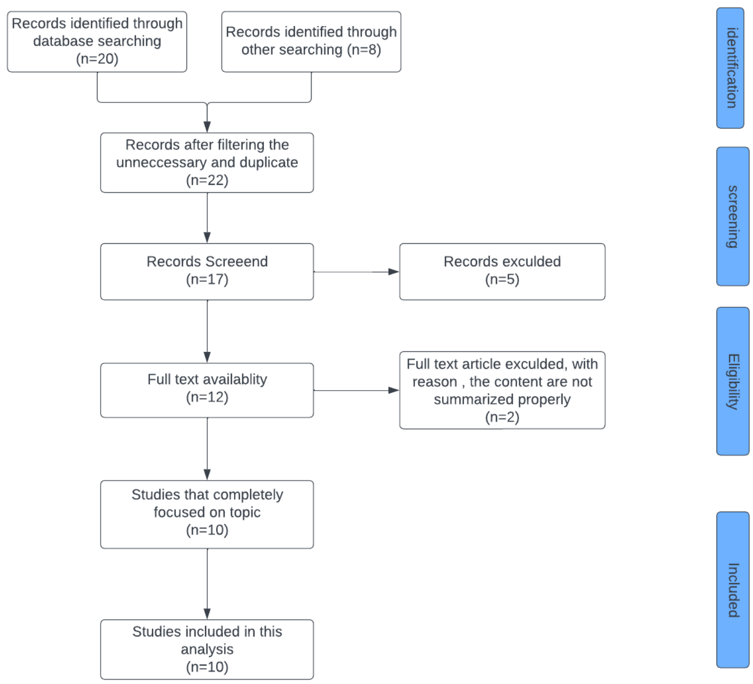

Fig 3. PRISMA Chart

2.1. Research Question

From the previous research we have computed three research questions:

- RQ1: What are the primary performance requirements and possible concerns with virtual reality (VR) games and applications that designers must address in order to provide a realistic and immersive user experience?
- RQ2: What are the problems and issues that VR developers have when optimizing their apps for a variety of hardware devices, and how do they strive to strike the correct balance between computational needs and device capabilities?
- RQ3: Why is user experience speed optimization important in VR apps, and how does it affect user engagement and overall satisfaction?

Why is user experience speed optimization important in VR apps, and how does it affect user engagement and overall satisfaction?

3. RESULTS AND DISCUSSION

While there have been a lot of empirical studies on general/mobile device performance bugs and optimizations that explain their common patterns and offer recommendations to developers/researchers, there have only been a small number of research papers on VR performance optimizations, and as a result, our understanding of them is very limited. Because VR applications contain complex real-time animations, heavy GPU utilization, and strong implications of asset/scene design (apart from source code) on rendering costs, issues impacting VR performance might be quite distinct from those impacting traditional software performance [20]. Over the last few decades, a wide range of fully immersive devices have been tested as media for showing abstract 3D representations and contrasted to traditional, non-immersive analytical contexts. While research consistently identifies benefits in immersive settings, such as modifying the capabilities of visualization components bypassing intermediary input modalities, interface issues are frequently highlighted as a barrier to effective analytical techniques. Various user interactions are complicated by high degrees of freedom and interface limitations (e.g., text input, coding) [22]. Constant technological advancement leads to the continual creation of new interaction modes for immersive environments, each of which must be examined separately. Furthermore, advancements in immersive device technology may impact the efficiency of certain representations and erase evaluation data from past research using obsolete technologies. Previous study indicates that parameters like as multisensory stimulation, display resolution, and the fidelity/photo-realistic of the virtual world have a substantial impression on the amount of felt immersion. Increased levels of immersion can have an impact on visual analysis tasks. For example, allowing the user to touch, feel, or even smell information via haptic VR gloves or HMD attachments enhances the sense of really dealing with real items [23]. Another prevalent rationale for poorly working VR/AR situations is that most participants are inexperienced with immersive settings and associated input methods. As a result, if users have become more proficient and more accustomed to the new locations, VR/AR environments may already be more successful [4]. However, this may have a negative impact on other aspects such as enthusiasm and engagement, which may be enhanced in new AR/VR situations due to low levels of familiarity. Novel interaction paradigms beckon further investigation. Virtual teleportation, for example, is a common strategy for compensating for the restricted physical area in VREs and should be carefully weighed against physical walking or other options such as VR treadmills or directed strolling [11].

In order to fully utilize these chances, developers must correctly design their sceneries and set up their projects in VR frameworks like Unity, which often offer numerous optimization opportunities for their off-stream software applications. setting on static/dynamic occlusion culling, for example, can prevent the display of game objects obstructed by other game objects while setting on light-baking allows you to pre-calculate lighting effects during compile time. Setting game objects to static avoids activating several lifecycle procedures, and grouping objects together reduces the number of draw calls since many things may be drawn within a single draw call [24].

First, VR apps and games are intrinsically more demanding than classic video games. This requirement arises from the fact that VR demands a high, consistent frame rate (usually about 90 frames per second per eye), minimal latency, and high-resolution stereoscopic rendering to provide a realistic, pleasant, and immersive user experience. Failure to adhere to these criteria can result in a variety of problems, such as a jerky or sluggish environment, subpar visual quality, and even user uneasiness or motion sickness this answer the RQ1.

Second, VR is run on a variety of hardware devices with varying performance capabilities. The performance gap between high-end gaming PCs with strong GPUs and independent VR headsets with mobile-class CPUs is significant. As a result, it is critical for VR developers to optimize their apps in order to guarantee that they work smoothly across this broad spectrum. At its foundation, VR performance optimization is all about finding the right balance. It's all about achieving the ideal balance between the amount of computation we put on the gadget and its capabilities. It's all about finding the proper combination of visual performance and durability this answered RQ2.

Furthermore, speed optimization is critical to the user experience. An optimized VR application delivers a smooth, immersive experience to users, increasing engagement and overall happiness. Poor performance, on the other hand, can interrupt immersion, resulting in a substandard and sometimes annoying user experience. Because of its direct effect on user engagement and overall happiness, speed optimization is critical for the user experience in VR apps. Maintaining a steady high frame rate while minimizing latency is critical in the immersive environment of virtual reality. Users are completely immersed in the virtual world when VR apps perform properly, increasing their feeling of presence and immersion. Poor performance, which results in shaky graphics or latency, on the other hand, may break immersion, leading to irritation and pain. Users are more likely to like and return to virtual reality experiences that provide fluid, responsive interactions, highlighting the crucial importance of speed optimization in delivering a great and engaging VR trip. this answer RQ3.

3.1. Graphics Optimization

In the realm of VR development, achieving optimal graphics performance is crucial for providing users with an immersive and comfortable experience. The strategies outlined in the table offer valuable insights into graphics optimization.
Firstly, maintaining a high frame rate of at least 90Hz and adjusting resolution to match hardware capabilities are fundamental for preventing motion sickness and enhancing immersion. Unity's dynamic scaling provides a means to achieve this.
Secondly, the use of Level of Information (LOD) systems and Unity's LODGroup component enables efficient rendering of complex 3D objects, particularly those at a distance, thereby reducing rendering demands and improving speed.

Table 2. Strategies for Graphics Optimization in VR Development

Optimization Strategy	Description
1. Resolution and Framerate	Aim for a consistent frame rate of 90Hz or greater to prevent motion sickness and enhance immersion. Modify resolution to match target hardware. Utilize Unity's dynamic scaling for frame rate preservation.
2. Level of Information (LOD)	Implement LOD systems for rendering complex 3D objects with varying detail levels based on proximity. This reduces rendering demand, especially for distant objects. Unity's LODGroup component simplifies LOD management.
3. Texture Compression	Reduce VRAM usage while preserving visual quality using texture compression techniques like ASTC or ETC2. Unity provides texture compression parameters for optimal textures on different platforms.
4. Shaders and Materials	Create efficient, visually appealing shaders that maintain speed using Unity's Shader Graph. Use complex shaders sparingly and avoid real-time reflections whenever possible for VR applications.
5. Occlusion Culling	Employ occlusion culling to avoid rendering objects obstructed from the user's view. Unity's integrated occlusion culling technology automatically optimizes scene rendering in complex environments.

Source : author's development

Thirdly, texture compression techniques like ASTC and ETC2 help conserve VRAM without compromising visual quality. Unity's texture compression parameters facilitate optimization across different platforms. Fourthly, the creation of efficient shaders using Unity's Shader Graph is emphasized. Balancing visual appeal and speed is crucial, with a recommendation to use sophisticated shaders sparingly and to avoid real-time reflections in VR applications. Lastly, occlusion culling, facilitated by Unity's integrated technology, aids in the efficient rendering of scenes by preventing the drawing of objects obstructed from the user's perspective.
Incorporating these graphics optimization strategies into VR development can lead to smoother and more immersive experiences while effectively managing hardware resources. Developers should consider these guidelines when striving for top-notch VR graphics performance.

3.2. Performance Optimization

Performance optimization plays a pivotal role in the domain of virtual reality (VR) development, serving as a fundamental prerequisite for the attainment of a seamless and immersive user experience. The present discourse delineates a comprehensive set of strategies, enumerated in the subsequent table, that offer invaluable insights into the art and science of performance optimization within the context of VR.
The practice of routine profiling of both Graphics Processing Units (GPU) and Central Processing Units (CPU), abetted by the capabilities inherent in Unity's embedded profiler, equips developers with the requisite tools to discern and address performance bottlenecks with precision, thereby facilitating a streamlined and efficacious approach to optimization.
Subsequently, the amalgamation of game objects through the utilization of Unity's batching mechanisms, in conjunction with the harnessing of Graphics Processing Unit (GPU) instancing functionality, engenders a palpable reduction in the computational overhead associated with rendering. This, in turn, augments rendering efficiency, thereby concomitantly fostering a perceptibly smoother VR experiential milieu.
Thirdly, the strategic deployment of dynamic loading mechanisms and judicious resource management, facilitated by Unity's AssetBundle technology, serves the dual purpose of curtailing the latency associated with initial resource loading and ensuring an expeditious commencement of user engagement.

Table 3. Strategies for Performance Optimization in VR Development

Optimization Strategy	Description
1. GPU and CPU Profiling	Regularly profile VR applications to identify performance issues. Unity's embedded profiler provides insights into GPU and CPU usage, enabling targeted optimization efforts.
2. Batching and Instancing	Group game objects to reduce draw requests and overhead. Utilize Unity's batching and GPU instancing to significantly improve rendering efficiency.
3. Dynamic Loading	Load resources and scenes as needed to reduce initial load times. Employ Unity's AssetBundle technology for efficient runtime resource management and loading.
4. Multithreading	Utilize Unity's multithreading features to distribute tasks across CPU cores. This enhances performance in physics, AI, and other compute-intensive processes, boosting overall efficiency.

5. UI Optimization	Simplify VR user interface components for efficiency. Avoid complex UI actions that strain the CPU. Utilize Unity's UI Toolkit, which includes VR-specific components and design principles.
6. Audio Enhancement	Enhance audio for VR immersion using spatialization and 3D audio sources. Unity's audio system offers options for improving audio quality in VR environments to create realistic soundscapes.

Source : author's development

Fourthly, capitalizing upon the concurrency afforded by Unity's multithreading capabilities, developers can judiciously distribute computational tasks across multiple CPU cores. This stratagem culminates in a notable enhancement of performance across resource-intensive computational domains, including, but not limited to, physics simulations and artificial intelligence algorithms.

Fifthly, the conformance to a doctrine of simplicity and efficiency in the realm of VR user interface design, reinforced through the integration of Unity's UI Toolkit, insulates User Interface (UI) components against inordinate CPU strain, thereby fostering a comprehensive elevation in overall system performance.

Finally, the orchestration of audio enhancement techniques, encompassing spatialization and the employment of 3D audio sources, engenders a multisensory immersive landscape within VR applications. This auditory augmentation augments the holistic VR experience, lending it an unprecedented level of realism and engagement.
In summation, the judicious amalgamation of these performance optimization paradigms into the tapestry of VR development equips developers with the capability to deliver applications that operate with an exemplary degree of fluidity, responsiveness, and immersive appeal. Consequently, this potent confluence of technical acumen ultimately redounds to the triumphant fruition of VR projects, securing their efficacy and resonance within the contemporary VR landscape.

4. CONCLUSION

It is critical to optimize visuals as well as performance in Unity3D VR apps to produce immersive and engaging experiences. Balancing great aesthetics with seamless performance necessitates a thorough grasp of Unity's features as well as a dedication to constant optimization. Developers may construct VR apps that take users to new environments without compromising their comfort or enjoyment by considering resolution, LOD, texture compression, and other graphics-related issues, as well as profiling, batching, multiple threads, and optimizing audio for performance. As virtual reality technology advances, understanding these methods for optimization will become increasingly important for developers looking to push the frontiers of virtual reality. In conclusion, further study is required to determine the true influence of technological variations (resolution, fidelity, multimodal stimulation) and user familiarity on user efficiency in immersive visual evaluation tasks. Furthermore, tests on obsolete devices and technologies may need to be performed on newer equipment that result in better levels of success.

COMPETING INTERESTS

Author have declared that no competing interests exist.

AUTHORS' CONTRIBUTIONS

Maksym Tytarenko designed the study, performed the statistical analysis, wrote the protocol, and wrote the first draft of the manuscript, managed the analyses of the study, managed the literature searches.

REFERENCES

[1] Abich J IV, Parker J, Murphy JS, Eudy M. A review of the evidence for training effectiveness with virtual reality technology. Virtual Real [Internet]. 2021;25(4):919–33. Available from: http://dx.doi.org/10.1007/s10055-020-00498-8

[2] Barteit S, Lanfermann L, Bärnighausen T, Neuhann F, Beiersmann C. Augmented, mixed, and virtual reality-based head-mounted devices for medical education: systematic review. JMIR Serious Games [Internet]. 2021;9(3):e29080. Available from: http://dx.doi.org/10.2196/29080

[3] Borges AFS, Laurindo FJB, Spínola MM, Gonçalves RF, Mattos CA. The strategic use of artificial intelligence in the digital era: Systematic literature review and future research directions. Int J Inf Manage [Internet]. 2021;57(102225):102225. Available from: http://dx.doi.org/10.1016/j.ijinfomgt.2020.102225

[4] Chong HT, Lim CK, Ahmed MF, Tan KL, Mokhtar MB. Virtual reality usability and accessibility for cultural heritage practices: Challenges mapping and recommendations. Electronics (Basel) [Internet]. 2021;10(12):1430. Available from: http://dx.doi.org/10.3390/electronics10121430

[5] De La Iglesia DH, Mendes AS, González GV, Jiménez-Bravo DM, De Paz JF. Connected elbow exoskeleton system for rehabilitation training based on virtual reality and Context-Aware. Sensors (Basel) [Internet]. 2020;20(3):858. Available from: http://dx.doi.org/10.3390/s20030858

[6] Dincelli E, Yayla A. Immersive virtual reality in the age of the Metaverse: A hybrid-narrative review based on the technology affordance perspective. J Strat Inf Syst [Internet]. 2022;31(2):101717. Available from: http://dx.doi.org/10.1016/j.jsis.2022.101717

[7] Fan X, Jiang X, Deng N. Immersive technology: A meta-analysis of augmented/virtual reality applications and their impact on tourism experience. Tour Manag [Internet]. 2022;91(104534):104534. Available from: http://dx.doi.org/10.1016/j.tourman.2022.104534

[8] Foxman M. Making the virtual a reality: Playful work and playbour in the diffusion of innovations. Digit Cult Soc [Internet]. 2022;7(1):91–110. Available from: http://dx.doi.org/10.14361/dcs-2021-0107

[9] Golding D. Far from paradise: The body, the apparatus and the image of contemporary virtual reality. Converg Int J Res New Media Technol [Internet]. 2019;25(2):340–53. Available from: http://dx.doi.org/10.1177/1354856517738171

[10] Hamilton D, McKechnie J, Edgerton E, Wilson C. Immersive virtual reality as a pedagogical tool in education: a systematic literature review of quantitative learning

outcomes and experimental design. J Comput Educ [Internet]. 2021;8(1):1–32. Available from: http://dx.doi.org/10.1007/s40692-020-00169-2

[11] Han L, Zheng T, Zhu Y, Xu L, Fang L. Live semantic 3D perception for immersive augmented reality. IEEE Trans Vis Comput Graph [Internet]. 2020;26(5):2012–22. Available from: http://dx.doi.org/10.1109/TVCG.2020.2973477

[12] Hruby F, Sánchez LFÁ, Ressl R, Escobar-Briones EG. An empirical study on spatial presence in immersive Geo-environments. PFG – J Photogramm Remote Sens Geoinf Sci [Internet]. 2020;88(2):155–63. Available from: http://dx.doi.org/10.1007/s41064-020-00107-y

[13] Iatsyshyn AV, Kovach VO, Romanenko YO, Deinega II, Iatsyshyn AV, Popov OO, et al. Application of augmented reality technologies for preparation of specialists of new technological era. [Internet]. 2020. 181-200. Available from: https://lib.iitta.gov.ua/720108/1/paper14.pdf

[14] Kersten T, Drenkhan D, Deggim S. Virtual reality application of the Fortress Al Zubarah in Qatar including performance analysis of real-time visualization. KN-Journal of Cartography and Geographic Information. 2021;71:241–51.

[15] Makransky G, Mayer RE. Benefits of taking a virtual field trip in immersive virtual reality: Evidence for the immersion principle in multimedia learning. Educ Psychol Rev [Internet]. 2022;34(3):1771–98. Available from: http://dx.doi.org/10.1007/s10648-022-09675-4

[16] Murray JW. C# game programming cookbook for Unity 3D. CRC Press; 2021.

[17] Okita A. Learning C# programming with Unity 3D. AK Peters/CRC Press; 2019.

[18] Petersen GB, Petkakis G, Makransky G. A study of how immersion and interactivity drive VR learning. Comput Educ [Internet]. 2022;179(104429):104429. Available from: http://dx.doi.org/10.1016/j.compedu.2021.104429

[19] Radianti J, Majchrzak TA, Fromm J, Wohlgenannt I. A systematic review of immersive virtual reality applications for higher education: Design elements, lessons learned, and research agenda. Comput Educ [Internet]. 2020;147(103778):103778. Available from: http://dx.doi.org/10.1016/j.compedu.2019.103778

[20] Rappa NA, Ledger S, Teo T, Wai Wong K, Power B, Hilliard B. The use of eye tracking technology to explore learning and performance within virtual reality and mixed reality settings: a scoping review. Interact Learn Environ [Internet]. 2022;30(7):1338–50. Available from: http://dx.doi.org/10.1080/10494820.2019.1702560

[21] Xiong Z. Development of VR teaching system for engine dis-assembly [Internet]. arXiv [cs.HC]. 2022. Available from: http://arxiv.org/abs/2207.05265

[22] Zhao X. Application of 3D CAD in landscape architecture design and optimization of hierarchical details. Comput Aided Des Appl [Internet]. 2020;18(S1):120–32. Available from: http://dx.doi.org/10.14733/cadaps.2021.s1.120-132

[23] Zhou Y, Tian L, Zhu C, Jin X, Sun Y. Video coding optimization for virtual reality 360-degree source. IEEE J Sel Top Signal Process [Internet]. 2020;14(1):118–29. Available from: http://dx.doi.org/10.1109/jstsp.2019.2957952

[24] Maziriri ET, Mashapa MM, Nyagadza B, Mabuyana B. As far as my eyes can see: Generation Y consumers' use of virtual reality glasses to determine tourist

destinations. Cogent Bus Manag [Internet]. 2023;10(3). Available from: http://dx.doi.org/10.1080/23311975.2023.2246745