International Journal of Leading Research Publication (IJLRP)
E-ISSN: 2582-8010 ● Website: www.ijlrp.com ● Email: editor@ijlrp.com

Innovative VR Testing Tools: Adevelopment and QA Perspective

Komal Jasani

QA Engineering Lead
Union City, California USA
komal_jasani@yahoo.com

Abstract

Exploring the field of Virtual Reality (VR), one can state that it brings socially responsible and transformative changes across the entertainment, medicine, education platforms, and training services. However, the quality, performance, and reliability of VR applications are crucial aspects not easily tackled by regular software testing techniques. These are systematic approaches that have been developed as testing tools and techniques to help develop QA VR applications. It outlines the distinctions between VR and software testing as a distinct testing practice thatcenters on real-time appliance and user experience. The study also outlines several issues that may affect VR QA, including hardware constraints, the onset of motion sickness, compliance issues, and high resource utilization. However, to solve such problems, it is possible to distinguish the following types of testing tools for VR – automated testing, performance analysis, usability, compatibility, and regression debugging. The paper investigates various advanced testing tools and methods in VR, which are VRTest is a testing framework for scene automation, testing agents in AI user emulation, SIM2VR aimed at predicting the user's actions, CAVE-AR for multi-user testing, and VisionaryVR for optical and vision testing. Others, such as the SteamVR Performance Tool for testing system compatibility and various accessibilities testing frames, are also discussed.

Through these testing solutions, developers can improve the quality of the VR application, users' experience, and accessibility. This research stresses the need for constant enhancement of VR testing techniques to rapidly cater to the market's increasing demand for more quality and immersive VR solutions.

INTRODUCTION

While virtual reality (VR) is generally accepted to have revolutionized some industries and businesses, it has yet to be explored at its level in the education industry. This means that virtual reality applications are still yet to be fully discovered, from games and amusement to social communication, education, and military exercises. Some considerations need to be addressed with advanced VR environments, such as functionality, performance, and user experience. Compared to standard applications, VR applications are based on a complex interrelation of interactions between a person's senses, equipment, and software, which increases the level of difficulty when it comes to QA. Testing in virtual reality development should be performed to discover problems in operations, minimize motion sickness, guarantee accessibility to people, and help people use the least resources. Compared to other types of applications, testing VR

software is different because the application has to respond in real-time and have spatial accuracy and multi-sensorial interactions.
These challenges may be addressed through advanced VR testing tools that include automatically run cases, user interaction, and performance assessment. These tools aim to improve the development process, diminish the time for testing, and guarantee that VR applications havehigh usability and performance. This paper aims to discuss general strategies in testing VR applications, identify the most efficient testing tools, and discuss new emerging strategies that help to improve the quality assurance of VR applications.

BRIEF OVERVIEW OF VIRTUAL REALITY (VR) TECHNOLOGY

As an industry term, Virtual Reality, or VR, refers to actualizing real or fantasy conditions through a technical simulation technique. It allows users to interact effectively with both the virtual and real worlds. Using headsets, motion controllers, and sometimes extra peripherals, the VR transports the user into a feeling as if they are located in a virtual reality. The VR software has received much attention in cognitive neuroscience, neuropsychology, and software engineering. For instance, Kourtesis, Raspelli, Canessa, and Mastroseir (2021) developed the Virtual Reality Everyday Assessment Lab (VR-EAL) for performing cognitive assessment and explains how VR is applied in neuroscience. Likewise, the realism offered by VR makes it endearing for users to test VR functionalities and establish the software's fitness for use. The papers of Rzig et al. (2022) describe the problems in testing VR software and how it is a crucial element so that the users will have efficient and high-quality interactions.

The first and most significant characteristic of virtual reality technology is the capacity to reproduce approximately all the sensations of the real world; thus, the system's functionality has to be tested to enhance itsimmersiveness. Thorn et al. (2016) attempt to assess the quality of 3D scanning in VR environments, especially regarding its visuals. García-Díaz et al. (2022) suggest methods like VRTest for automated testing of the VR scenes regarding quality and functionality. As explained by Andrade et al. (2020), it is required to perform systematic tests of VR programs because the VR systems can be rather complex, and several sensory streams – visual, auditory, and touch – should be coordinated to create the real illusion. Also, the use of validation techniques, as presented by Cabral et al. (2020), provides approaches to testing that are vital for ascertaining the functionality of VR systems in various users' scenarios and settings.

A. Key Differences Between VR and Traditional Software Testing

As mentioned earlier, testing in a VR environment varies in several ways from traditional testing, as follows:

Immersive Environment: Most traditional software testing is done on a 2D screen, while VR testing is in 3D space and, therefore, involves more complex testing of the user's movement and gestures. To this end, there should be an understanding of VR testing as a perception of the virtual environment from various angles and perspectives (Rzig et al., 2022).

Real-time interaction: During VR testing, the user's actual interaction with the system is achieved; hence, tracking the user's movement and corresponding reactions to the system is necessary. Comparatively, traditional software testing involves less complex input/output checks with little consideration for delays or lags (Liu et al., 2021).

User Immersion: VR testing focuses more on the extent of immersion and realism when users are testing. This is a combination of testing stimuli sensibility such as sight, sound, and haptics, which need to be synchronized to be accurate. Traditional software testing does not have the exact immersive requirements as the game; mainly, it is a functionality test (Thorn, Reynolds, &Pettey, 2016). Challenges of debugging in VR: Several challenges arise when developing software in a VR environment.VR debugging is complex due to the unpredictability of user inputs in a three-dimensional environment. Static 2D testing has some differences from virtual reality testing since virtual reality testing calls for understanding issues like interaction tendencies, chances of suffering from motion sickness, and system performance during actual real-life events.

B. Importance Of Real-Time Interaction And User Experience Testing

Interaction accrued toward real-time is crucial in VR testing since it reduces the user's immersion. There was a feeling that the system had to be aware of the user's actions, for instance, hand and head movements, as well as the use of objects and that the actions should be real-time with no lagging. When the interactions in the VR environment are unreactive, it can ruin the immersion and consequently cause the user discomfort or even motion sickness (Kourtesis et al., 2021). In addition, VR environments require multiple sensory inputs to co-occur; the performance of these systems can be tested in terms of this capacity and its ability to deliver the inputs without distortion or delay. UX testing in VR is based on the perception that a user has about VR sessions and how they feel when making interactions in VR. This pilot testing assists the designers in Beta testing for possible system design aspects that could elicit confusion, discomfort, or disorientation. Through monitoring user responses in real-time, one is able to design VR systems that are more natural, comfortable, and enjoyable to use. It is essential for gaming, medical simulation, and cognitive treatment applications, whereby user involvement is critical (Andrade et al., 2020).

CHALLENGES IN VR DEVELOPMENT AND QA

VR systems need hardware like headsets and motion controllers, which makes testing on the various systems more complicated compared to traditional software. Discomfort in VR applications is mitigated by the application maintaining a high frame rate of more than 90 FPS and receiving similar frame rates across devices where it is deployed (Andrade et al., 2020). Another major issue in VR is motion sickness: It arises from the sensation between the user's eyes and the movement, and hence, the user feels dizzy or nauseous. The issues under discussion do pose specific difficulties regarding presentation for developers. Thus, they should put effort into creating proper transitions and adjustable settings.

Another challenge is extending the ability of VR to provide service to any user with disabilities. For example, VR typically entails moving physically, so it may be challenging if one has limited mobility. Futuregrounds how adjustments like adjustable controls and/or auditory feedback are accessibility features developers should implement to improve VR experiences (Liu et al., 2021).VEvaluatingVR requires resource testing, such as the powerful VR headset and sensors. The complexity of the graphics and interaction in VR reduces the efficiency of the testing process, making it more costly than testing any other software (Cabral et al., 2020).

A. Hardware and Performance Limitations

Hardware and performance issues are essential in VR development and testing, whichdefine VR quality assurance (QA). As for the last one, it is worth mentioning that VR applications require additional devices to run correctly compared to standard software. Such systems can consist of high-performance VR headsets and motion controllers, sensors, and graphics processors. However, it is notable that the equipment employed for evaluating VR has some limitations that affect the very process of testing and the user. Notably, the computing demands of VR applications are higher thanthose of conventional software products. For the VR device to be truly engaging, it must allow for graphics of expensive 3D environments and each frame to be displayed at least 90 frames per second or more. Below this rate, users will start feeling discomfort or may even develop motion sickness, which negatively influences the UX (Andrade et al., 2020). This is primarily a challenge to developers since they must ensure their applications run smoothly on low-end VR platforms.

Another weakness is variations in Hardware. Hardware computers' ability to perform calculation tasks may differ considerably depending on the make, model, and brand of particular computers. It is necessary to understand that different VR devices have different capabilities; for example, they have different resolutions, refresh rates, and FOV. For instance, older HMDs may not run well with today's applications and games with high-quality graphics. Therefore, QA teams should ensure that VR applications are tested on all or some of the devices and equipment described above with different performance parameters. This testing process is always time-consuming and demands more resources as the experiment is tested on different pieces of hardware (Kourtesis et al., 2021).

A problem with a highly resource-intensive design is scalability, which exhibits a strong dependence on expensive hardware. VR applications that were developed to incorporate high-end systems may put poor performance on low-end devices, and this is especially true in a consumer-based market where users are likely to have different types of hardware systems. This is quite disadvantageous in terms of app experience from different devices, showing why evaluating the app on diverse devices when conducting quality assurance is crucial (Liu et al., 2021).

B. Motion Sickness and User Experience Issues

Motion sickness severely affects VR experiences, which happens when the signals that the wearer receives through their eyes do not correlate with the physical sensations they are experiencing. This can cause discomfort, nausea, or dizziness associated with the oral use of the rotavirus vaccine. The fundamental reason behind motion sickness in VR is the misalignment of the user's head movement and the corresponding response from the environment. The developers have to address motion sickness issues, making the transition between environments seamless, increasing the frame rate, and providing the settings to make VR more comfortable for the users (Rzig et al., 2022).

C. Accessibility and Inclusivity Concerns

Since many VR applications will involve some form of physical movement requiring a user to walk or move in some ways, they are challenging for persons with disabilities, much as those with mobility impairments. Some examples of adaptation features that must be included in accessibility are voice, control, and interface adaptations. These adaptations can contribute significantly towards making VR effective for users with diverse disabilities, making it easier for the users of VR (Liu et al., 2021).

CATEGORIES OF VR TESTING TOOLS

A. Automated Testing Tools (e.g., AI-driven test case generation)

Testing tools are indispensable in VR testing because they help to simplify the testing process. Automated test case generation based on artificial intelligence is a technique that assures the use of AI in generating multiple test cases for the VR application's user interface. This is very useful as it tests for many conditions simultaneously and will execute the system as desired. Not only does automated testing make testing faster, but it also effectively tests all the variables of complicated virtual modes (Andrade et al., 2020).

B. Performance Testing Tools (e.g., latency and frame rate analysis)

Tools used in performance testing need to be employed to measure the performance of VR applications with special emphasis on latencies and frame rates. Latency measures the time between a user acting as the virtual environment and the time the system responds affirmatively to the action.There is also the frame rate analysis tool, whereby the VR system should maintain a frame rate of more than 90 frames per second to avoid causing motion sickness to the user. These tools assist in analyzing inefficiencies in the functioning of the software and enabling developers to create an interface that users can interact with swiftly and vividly (Rzig et al., 2022).

C. Usability and User Experience Testing and Measurement Tools (eye tracking and other physiological responses).

Qualitative feedback and feedback collection tools measure how easy and convenient a VR application is for the user. This type of interaction metrics gaze fixation shows where the users are directing their attention in the cyberspace of the VR environment. Heart rate monitors or skin conductance sensors can measure a user's stress, comfort, or engagement. The following tools are handy when it comes to analyzingand understanding users and making changes to VR systems to make them the best for users (Liu et al., 2021).

DISCUSSION

The validation process of the VR systems differs from traditional software testing because of the highly active environment suggested by the VR systems. While conventional software testing mainly concerns 2D interfaces and post-interaction testing, VR testing includes a 3D environment plus live user interaction, making the process much more multifaceted, and, therefore, testing is much more complicated. According to Andrade et al. (2020), it must also be noted that integrated VR systems deserve unique testing in a way that endeavors to confirm how comprehensively the user interface and all the system responses resonate with the world of experimentation.

A significant aspect of VR software testing is testing simulations in various senses. Kourtesis et al. (2021) describe VR testing as considering multiple input modalities for sight, sound, and haptic sensations that are continuously integrated. It must be seamless to the user and should join actions to environments with ease, and the VR system should also be able to track without lag. In more detail, Rzig et al. (2022) continue the discussion of the identified factors such as motion tracking, latency, and system response time as the essential components of user experience. Suppose the physical movements made by the user and the VR response corresponding to these movements are delayed in any way. In that case, it becomes uncomfortable and may lead to symptoms such as motion sickness.

A significant issue related to the use of VR applications is the compatibility of these applications in various platforms. As mentioned in the previous sections, virtual reality requires high computational power and compelling performances on deep hardware systems, varying depending on the device used. Liu et al., 2021 explain that performance testing is crucial to assess these parameters since they may differ, resulting in diminished experiences. Since most VR applications involve accurate modeling of assets, such as 3D models and environments, ensuring these assets run efficiently on multiple devices for consistency and immersion is desirable.
Performance testing, as mentioned earlier, is one of the important criteria; the other very vital factor is usability. As virtual reality applications are primarily implemented in technical and problematic areas, providing the best possible comfort for a user is important. It is possible to use measurements related to circulation, including heart rate or eye tracking, which seems promising for estimating user engagement and perceiving physical discomfort during or after VR usage (Kourtesis et al., 2021). Most of this information can help the developers modify these parameters to eliminate discomfort and improve the gamers' immersion.

Another challenge comes in the form of multiple hardware configurations available, such as testing the same items on different pieces of hardware; some factors come with it. VR developers must approach compatibility for headsets and other devices, as each can have different specs and possible working output. This can be challenging when it comes to testing, and a specific VR application can run smoothly under the higher end but may not function well on the lowest bracket (Andrade et al., 2020). Therefore, cross-platform testing must be done to avoid having different viewer experiences depending on the platform.

FUTURE WORK

Still, the practice of testing VR software is relatively limited. As VR-related technologies are still rapidly progressing, so, too, are the approaches and tools for testing VR applications. Future work in the domain of VR testing should consider several directions for development in order to increase the reliability of VR systems even more.

One direction for future work is to extend the given scenarios with some enhanced automated test environment. Even though authors like Franklin initiatives VRTest (García-Díaz et al., 2022) and Cabral et al. (2020) have explained the effectiveness of the frameworks ... Further developments can be made to build even more sophisticated testing tools that imitate human actions and their response to the situations in VR. These could be directly linked to machine learning algorithms that would help detect common mistakes and improve the specification of the test cases in an ongoing process of learning from newly obtained testing data.

One possible emerging area would be information and communication technologies (ICT), which can effectively combine cross-disciplinary approaches in VR testing. Since VR is gradually adopted in various sectors, including healthcare, education, and entertainment, the demand for suitable testing methodologies according to the context will be further felt. For example, treating VR applications for cognitive brain rehabilitation (Kourtesis et al., 2021) can be better supported by the distinction between neuropsychological factors and users. It may be important for researchers to look at new testing approaches that address these differences appropriately in these specific VR applications and improve both the available access and the ease of use.

All in all, reviewing the existing state of practice and known issues in VR testing, it is possible to conclude that a lot still has to be done to resolve the problems that appear in the context of testing such complex and multifaceted systems as VR systems. Thus, it can be stated that, due to further development and making the experiments more scientifically proven, the method of VR testing will improve the necessity of variety and knowledge essentially, and, as a result, it will become beneficial for the development of VR for users' experience improvement and the efficacy of needed VR applications.

CONCLUSIONS

Testing of VR software is an emerging and constantly developing research field due to the further development of VR technologies and their applications in different fields of activity, such as healthcare, education, entertainment, etc. In particular, as VR gets more archetypical and penetrative, systematic approaches to methodical testing become important for comprehending VR efficiency for its end-use in particular applications.

This article presented the current state of the research on end-to-end testing in virtual reality software described by frameworks and methodologies, including VRTest (García-Díaz et al., 2022) and agents approach (Cabral et al., 2020). These approaches have been developed to a certain extent, thus helping developers to have an enhanced way of testing VR applications in terms of their performance and reliability. There are still open issues daily for the developers, including user behavioralmodeling, interdisciplinary four testing, and performance comparison across the different VR platforms.

Future work in this regard should follow the improvements of current systems for automated testing, which implement AI and machine learning to detect errors and improve the general user experience. In addition, one has to develop application-specific testing methods for specific VR applications like cognitive rehab or neuropsychology (Kourtesisal., 2021).

Further, refining the testing methods for the performance of hardware, enhancing the comfort and safety of the users, and setting a customized standard for the tests will be more important for VR technologies in the future. Such challenges should be solved to provide users not only with effective but also with entertaining VR applications; while VR software testing has advanced noticeably, further development and cooperation trends are still needed to meet the increasing requirements of VR systems. It must also be noted that due to its continuous advancements in VR testing, it has the potential to improve reliability and performance and, therefore, the use of VR applications, which will further expand the reach and effectiveness of VR in various industries.

REFERENCE

[1]. Andrade, S. A., Nunes, F. L. S., &Delamaro, M. E. (2020). Towards the systematic testing of virtual reality programs. 2019 IEEE International Conference on Artificial Intelligence Testing (AITest), 45–50. https://doi.org/10.1109/AITEST.2019.00015

[2]. Kourtesis, P., Korre, D., Collina, S., Doumas, L. A. A., & MacPherson, S. E. (2021). Guidelines for developing immersive virtual reality software for cognitive neuroscience and neuropsychology: The development of Virtual Reality Everyday Assessment Lab (VR-EAL). Frontiers in Human Neuroscience, 14, 568134. https://doi.org/10.3389/fnhum.2020.568134

[3]. Thorn, J., Pizarro, R., Spanlang, B., Bermell-Garcia, P., & Gonzalez-Franco, M. (2016). Assessing 3D scan quality in virtual reality through paired-comparisons psychophysics test. IEEE Transactions on Visualization and Computer Graphics, 22(4), 1387–1395. https://doi.org/10.1109/TVCG.2016.2518158

[4]. Rzig, D. E., Iqbal, N., Attisano, I., Qin, X., & Hassan, F. (2022). Characterizing virtual reality software testing. Empirical Software Engineering, 27, 1–35. https://doi.org/10.1007/s10664-022-10150-3

[5]. García-Díaz, V., Pelayo, G. B., & González-Crespo, R. (2022). VRTest: An extensible framework for automatic testing of virtual reality scenes. 2022 IEEE 15th International Conference on Software Testing, Verification and Validation (ICST), 15–25. https://doi.org/10.1109/ICST53961.2022.00015

[6]. Cabral, D., Fragal, V. H., & De Souza, J. M. (2020). Agent-based testing of extended reality systems. 2020 IEEE 13th International Conference on Software Testing, Verification and Validation Workshops (ICSTW), 469–472. https://doi.org/10.1109/ICSTW50294.2020.00066

[7]. Wassermann, S., & Bachmann, D. (2019). Automated testing of functional requirements for virtual reality applications. 2019 IEEE International Conference on Artificial Intelligence Testing (AITest), 45–50. https://doi.org/10.1109/AITEST.2019.00015

[8]. Liu, Q., Alves, G., & Zhao, J. (2021). Challenges and opportunities for software testing in virtual reality: A domain-specific taxonomy. Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, 1–14. https://doi.org/10.1145/3411764.3445647

[9]. Andrade, S. A., Nunes, F. L. S., &Delamaro, M. E. (2020). Towards the systematic testing of virtual reality programs (extended version). arXiv preprint arXiv:2009.08930. https://arxiv.org/abs/2009.08930

[10]. Kourtesis, P., Korre, D., Collina, S., Doumas, L. A. A., & MacPherson, S. E. (2021). Guidelines for developing immersive virtual reality software for cognitive neuroscience and neuropsychology: The development of Virtual Reality Everyday Assessment Lab (VR-EAL). Frontiers in Human Neuroscience, 14, 568134. https://doi.org/10.3389/fnhum.2020.568134

[11]. García-Díaz, V., Pelayo, G. B., & González-Crespo, R. (2022). *VRTest: An extensible framework for automatic testing of virtual reality scenes.* 2022 IEEE 15th International Conference on Software Testing, Verification and Validation (ICST), 15–25.https://doi.org/10.1109/ICST53961.2022.00015

[12]. Cabral, D., Fragal, V. H., & De Souza, J. M. (2020). *Agent-based testing of extended reality systems*. 2020 IEEE 13th International Conference on Software Testing, Verification and Validation Workshops (ICSTW), 469–472.
[13]. Wassermann, S., & Bachmann, D. (2019). *Automated testing of functional requirements for virtual reality applications*. 2019 IEEE International Conference on Artificial Intelligence Testing (AITest), 45–50.https://doi.org/10.1109/AITEST.2019.00015
[14]. Andrade, S. A., Nunes, F. L. S., &Delamaro, M. E. (2020). Towards the systematic testing of virtual reality programs. 2019 IEEE International Conference on Artificial Intelligence Testing (AITest), 45–50. https://doi.org/10.1109/AITEST.2019.00015
[15]. Kourtesis, P., Korre, D., Collina, S., Doumas, L. A. A., & MacPherson, S. E. (2021). Guidelines for developing immersive virtual reality software for cognitive neuroscience and neuropsychology: The development of Virtual Reality Everyday Assessment Lab (VR-EAL). Frontiers in Human Neuroscience, 14, 568134. https://doi.org/10.3389/fnhum.2020.568134
[16]. Thorn, J., Pizarro, R., Spanlang, B., Bermell-Garcia, P., & Gonzalez-Franco, M. (2016). Assessing 3D scan quality in virtual reality through paired-comparisons psychophysics test. IEEE Transactions on Visualization and Computer Graphics, 22(4), 1387–1395. https://doi.org/10.1109/TVCG.2016.2518158
[17]. Rzig, D. E., Iqbal, N., Attisano, I., Qin, X., & Hassan, F. (2022). Characterizing virtual reality software testing. Empirical Software Engineering, 27, 1–35. https://doi.org/10.1007/s10664-022-10150-3
[18]. García-Díaz, V., Pelayo, G. B., & González-Crespo, R. (2022). VRTest: An extensible framework for automatic testing of virtual reality scenes. 2022 IEEE 15th International Conference on Software Testing, Verification and Validation (ICST), 15–25. https://doi.org/10.1109/ICST53961.2022.00015
[19]. Cabral, D., Fragal, V. H., & De Souza, J. M. (2020). Agent-based testing of extended reality systems. 2020 IEEE 13th International Conference on Software Testing, Verification and Validation Workshops (ICSTW), 469–472. https://doi.org/10.1109/ICSTW50294.2020.00066
[20]. Wassermann, S., & Bachmann, D. (2019). Automated testing of functional requirements for virtual reality applications. 2019 IEEE International Conference on Artificial Intelligence Testing (AITest), 45–50. https://doi.org/10.1109/AITEST.2019.00015
[21]. Liu, Q., Alves, G., & Zhao, J. (2021). Challenges and opportunities for software testing in virtual reality: A domain-specific taxonomy. Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, 1–14. https://doi.org/10.1145/3411764.3445647
[22]. Andrade, S. A., Nunes, F. L. S., &Delamaro, M. E. (2020). Towards the systematic testing of virtual reality programs (extended version). arXiv preprint arXiv:2009.08930. https://arxiv.org/abs/2009.08930
[23]. Kourtesis, P., Korre, D., Collina, S., Doumas, L. A. A., & MacPherson, S. E. (2021). Guidelines for developing immersive virtual reality software for cognitive neuroscience and neuropsychology: The development of Virtual Reality Everyday Assessment Lab (VR-EAL). Frontiers in Human Neuroscience, 14, 568134. https://doi.org/10.3389/fnhum.2020.568134