Performance Evaluation of WebGL and WebVR Apps in VR Environments

Renato M. Toasa[1](✉), Paúl Francisco Baldeón Egas[1](✉), Miguel Alfredo Gaibor Saltos[1](✉), Mateo A. Perreño[2](✉), and Washington X. Quevedo[3](✉)

[1] Universidad Tecnológica Israel, Quito, Ecuador
{rtoasa,pbaldeon,canciller}@uisrael.edu.ec
[2] Universidad Politécnica Salesiana, Quito, Ecuador
mparrenoa@est.ups.edu.ec
[3] Universidad de las Fuerzas Armadas ESPE, Sangolqui, Ecuador
wjquevedo@espe.edu.ec

Abstract. This paper presents an evaluation of devices and browsers performance when executing web applications in WebGL and WebVR. For this purpose, the measurement of several computational resources is performed to determine which technology optimizes hardware resources when working with this type of technologies. For the purpose of this study, an application has been developed in Unity 3D that provides the exact values that the computational resources consume. In this context objects and particle systems are used to give a stress behavior on hardware entities. As a result, it was determined that WebGL and WebVR have similar performances in executions on a PC while for WebVR mobile devices it has a remarkable performance compared to WebGL.

Keywords: Virtual reality · WebGL · WebVR

1 Introduction

In the last years, applications in virtual reality (VR) are experiencing a very big revolution, as these increasingly resemble the real world and allow the user to interact with several elements in an artificial 3D environment [1]. In virtual reality a widely used term is spatial immersion, which is defined as the perception of being physically present in a non-physical world [2]. VR applications were initially executed on specialized platforms, new technologies such as WebGL (Web-based Graphics Library) and WebVR (Bringing Virtual Reality to the Web) have been developed, which allow the execution of VR applications on several browsers and devices.

When mentioning a 3D browser, one of the first thoughts of the user is “A Dom (document object model) to the web page”; although it is possible to produce web versions of classic 3D games in the browser, 3D is more powerful and with a wide field of development. The web is not a one-way communication, it allows not only the dissemination of information but also allows collaboration and communication like no other medium. This particularity of the web has made possible the creation of applications that are not possible in autonomous mode [3]. 3D web content is just another

G. Bebis et al. (Eds.): ISVC 2019, LNCS 11845, pp. 564–575, 2019.
https://doi.org/10.1007/978-3-030-33723-0_46

form of this growing set of diverse applications [4]. One of many technologies that bring 3D graphics on web is WebGL. WebGL for VR applications is WebVR that brings a VR experience attached in a webpage.

WebGL is a software library related to graphics which is designed to integrate extended JavaScript programming language with OpenGL (Open Graphics Library) to generate accelerated 3D graphics within any supported web browser. WebGL allows the user to incorporate to the Web a 3D graphic without complements, implemented directly in the browser. The main providers of browsers are Apple (Safari), Google (Chrome), Microsoft (Edge) and Mozilla (Firefox). These providers are members of the WebGL Working Group. WebGL is also widely used in the fields of visualization, 3D modeling and has been implemented on desktop, tablets and smart phones. [5]. Several works have determined that the potential use of WebGL is 3D modeling. For example, the Tinkercad project which allows users to draw Tinkercad 3D models within a web browser and send them to any 3D printer. Also, WebGL has been applied to the visualization of complex data that requires 3D presentation. One of these projects is described in [6], which proposes a new and innovative use of WebGL in medical applications using 3D images for better interpretation of medical examinations. Another field of study that has benefited from this technology is Geospatial. Data sets were introduced and WebGL was used to visualize the result of environmental models in 3D [7].

WebVR is an Internet-based virtual reality web technology which provides support for virtual reality devices, such as virtual reality glasses e.g. Oculus Rift or HTC Vive. WebVR allows developers to translate screen position and motion information into a 3D scene [8]. This has numerous interesting applications, from virtual product tours and interactive training applications to immersive games. Some of the browsers that support this specification are Chrome for Android, Chromium, Firefox, Microsoft Edge, Oculus Browser and Safari. [9]. WebVR affects frame rates considerably when using high quality effects such as shadows, soft lighting and image clarity. It is also a great improvement over Three.js Implementation (it is a light library written in JavaScript to create and display computer animated graphics in 3D in a Web browser). It is natively supported in mobile browsers [10].

Today, developers and researchers are exploring the use of multisensorial input and output devices to work with WebVR applications [11]. Based on this, a system was built to help firefighters simulate emergency rescues for VR technology accidents [12]. A project for “Immersive and Collaborative Data Visualization” was presented in 2014 with the aim of maximizing intrinsic human pattern recognition skills through the use of emerging technologies associated with immersive RV [13]. In this context it is planned to test WebGL and WebVR technologies in Virtual Reality environments to show the advantages and disadvantages of each technology running them as web applications on remote servers.

Now with the penetration of mobile devices in much of the world’s population and the rapid adoption of virtual reality applications, it is imperative to determine the most optimal use of web technologies that can be run in virtual reality environments on mobile devices. As already seen WebGL and WebVR are highly used to display 3D graphics on the web and thanks to virtual reality browsers, this content can be consumed in virtual environments and interact with Game Objects. The main aim of this article is to compare the performance of each technology on different devices to determine the best option when developing web applications that run in virtual reality environments.

The rest of the document is organized as follows: Sect. 2 presents a formal description of the problem being studied, Sect. 3 details the structure of the evaluation, Sect. 4 shows the evaluation carried out with different case studies, then Sect. 5 mentions the results obtained and the discussion of these, finally the conclusions and future work are mentioned in Sect. 6.

2 Problem Definition

According to several studies found in literature, the evaluation of performance is a determining factor for the success or failure of a system computer applications. Web and mobile devices have taken over several business processes being used as strategic tools for business success, regardless of the area in which they are located [14]. The use of these applications is not a luxury but a necessity that large companies have to manage their strategic, operational and support processes [15]. However, despite all the technological advances, there are still certain challenges that need to be addressed and solved, for example, browser complements, use of computational resources from the browser, efficient 3D visualization and compatibility with mobile devices [16].

A critical factor in the development of WebGL and WebVR applications is to determine the level of support and performance offered by different technologies (for example: devices, operating systems and browsers) when running these types of applications, as they consume large computing resources (RAM, CPU, GPU). This is a big problem for developers since it is not easily defined which browser or device is adequate to evaluate and validate their applications; depending on the purpose of these, some will run better in one browser or device than on another. As far as browsers are concerned, the best known are: Chrome, Firefox, Safari; as far as the devices are concerned, the ones that are currently setting the trend are: Oculus Rift, GearVR and smartphones [17].

Another problem that arises in this type of applications is related to the management of computational resources that are used, because in some cases resources are wasted by running a simple application on a computer with great features or on the contrary running a complex application on a mobile device that has limited resources to run these applications. In this study, it is proposed to evaluate the different technologies that work with WebGL and WebVR in order to determine which type of application to use and in which technology.

In the following sections, different case studies will be mentioned in which several scenarios are considered, shaped by the different technologies described in the previous image. In addition, different computational resources are evaluated in order to identify the ideal scenario to work with these technologies.

3 Evaluation Structure

The structure of the evaluation process is detailed in Fig. 1, which has 4 consequential stages. The process begins with a methodological stage in which the way of evaluating, the case studies and the variables to be measured will be defined. For the Configuration stage, the entities that will intervene in the entire evaluation process are specified, as well as the indicators of each test. Next in the Test stage, the way in which the data coming from the entities defined in the Configuration stage is obtained and processed will be explained. Finally, the Results stage shows a comparative analysis based on the data obtained in the Test stage. It ends with conclusions about the use of WebGL and WebVR technologies for specific purposes detailed in each one. Next, a detailed explanation of each evaluation stage is made.

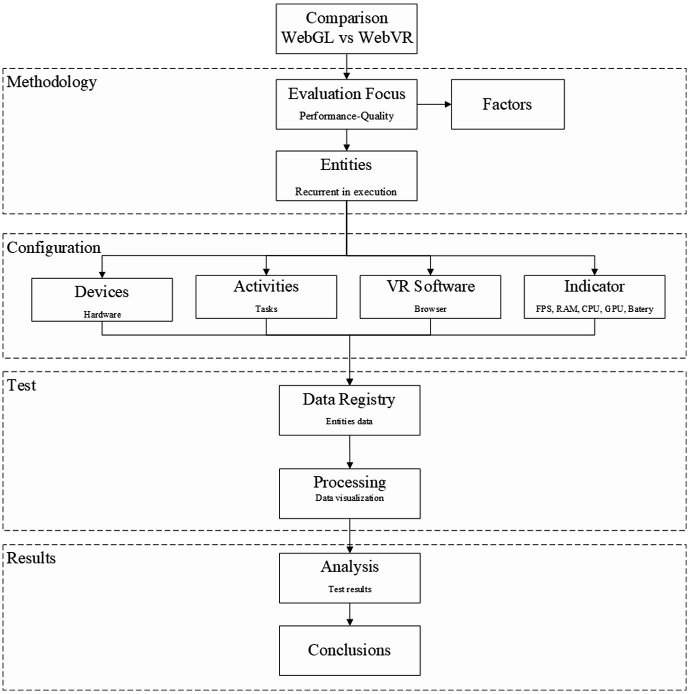


Fig. 1. Structure of evaluation process

In stage *(i) Methodology*, the way in which it will be evaluated is specified, in this case, a focus will be made on performance and quality, principles on which the usability of the system and the user experience are based [18]. In order to use this approach, it is necessary to define the factors that will intervene in the evaluation of the technologies and that will eventually become indicators of the experiment. The block entities specify the instances that will intervene in the experiment such as devices, activities, VR software and indicators.

Next, in stage *(ii) Configuration* the blocks are specified: Devices, Activities, VR Software and Indicators. In the device´s block the hardware to be used is established, considering the most commonly used VR devices in the market and the current trend as a standard in the development of virtual reality [19]. They are divided into 2 groups: PC-Based and Mobile. In PC-Based, HTC VIVE and Oculus Rift devices are used; on mobile devices GearVR and a smartphone are used. In the Activities block, tasks are specified to saturate the system by increasing instances in the environment and their interaction between themselves. Two types of activities have been selected, the first activity is to drop boxes from the top to the ground in order for the system to calculate the collisions between the boxes, and with more instances, the stress level will rise. The second activity consists in instances of a particle system which saturates the system, while more particles and interactions between them are given. For the VR Software block, browsers that run in virtual reality will be used because the applications are hosted on web servers and their access will be through a URL inside virtual environments. In the indicator block the variables to be measured are: frames per second (FPS), RAM consumption in MB, percentage of CPU usage, percentage of GPU use and the percentage of battery consumed in the execution of the test.

In stage *(iii) Test*, the Data Registry blocks are in charge of acquiring and saving the data of each test in a classified and labeled form, while the processing block is responsible for the visualization of data by means of graphs. In the last stage *(iv) Results*, the graphs obtained in the previous stage are interpreted through of trend of values and grouping results, to finalize in the conclusions section to detail the average values of the measurements in each test that determine the best performance of WebGL and WebVR in many devices and initial conditions established.

4 Evaluation Design

The evaluation of technologies for WebGL and WebVR depends on several essential factors: CPU, GPU & RAM consumption and FPS (Frames per second) [22] at the runtime app. These are important computational resources that allow applications to be optimal [20]. For this paper different technologies have been defined for evaluation, using particles and objects.

For the smartphone´s category, the CPU usage is not measured directly, due to it requiring other external elements. For this reason, it was decided to measure the battery level. Tests are designed in 2 phases: WebGL and WebVR with stress level variants in each execution of the same application. The variants of each run are: device, browser and activities (see Fig. 2).

In the devices section, consider desktop (PC needed to run the application) and mobile devices like: HTC VIVE & Oculus Rift and GearVR & Smartphone respectively. The activities contemplate the use of particle systems and game objects according to the saturation level of the test. In the particle system is considered a flamethrower system with 1000 particles and a lifetime of 10 s. In the object system, the use of 10 Unity cube Objects with simple textures for every 10% saturation level was developed.

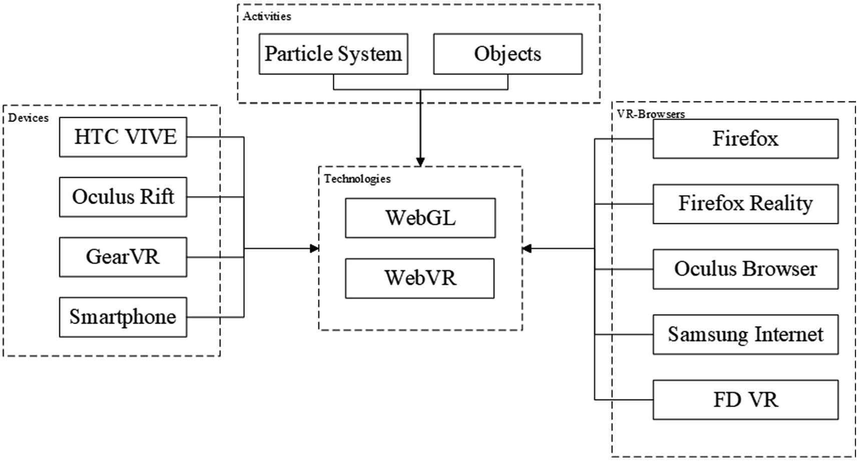
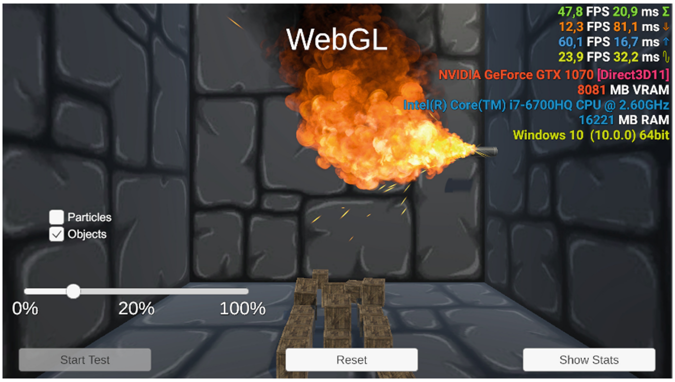


Fig. 2. Variants for evaluation

For the test purposes in the browser′s component, the most used software and only those that run in VR environments were selected for each device. On the PC-based devices, Firefox used in HTC VIVE & Oculus Rift; Firefox Reality & Oculus Browser used in GearVR, and for smartphones Android & iOS, FD VR and VR Browser were selected respectively. For the development of application test the IDE Unity were used and to achieve the least impact on the factors that we want to measure the app only uses gameobjects and native resources (cubes, particle systems and lights) [21]. The target platform has been configured as WebGL, and for WebVR has been used with the WebVR unitypackage powered by Mozilla.

For both WebGL and WebVR executions, the app consists of an interface that allows the user to select the type of test between the particle and object options, as well as a slider that sets the level of stress that will be applied to the test (from 0% to 100%). In addition, it has a Start Test button that allows user to begin with the test. The Reset button allows the user to delete all the objects that are part of the test and enables the Start Test button to perform a new test. Finally, there is an indicator of computational device resources such as: FPS, CPU, RAM, VRAM (Video Random Access Memory), GPU. As shown in Fig. 3

The execution of each test allows the user to collect the data according to the following table. Through the combination of devices, technologies, activities and browsers, 28 tests are determined and each of them considers 5 cases of measurement: 20%, 40%, 60%, 80% and 100% corresponding to the level of stress applied in each activity. This methodology is determined by the devices target to the measure in which they are PC-based and mobile (see Table 1).



```
47,8 FPS 20,9 ms Σ
12,3 FPS 81,1 ms ⇓
60,1 FPS 16,7 ms ⇑
23,9 FPS 32,2 ms ∿
NVIDIA GeForce GTX 1070 [Direct3D11]
8081 MB VRAM
Intel(R) Core(TM) i7-6700HQ CPU @ 2.60GHz
16221 MB RAM
Windows 10  (10.0.0) 64bit
```

Fig. 3. Consumption of computing resources

Table 1. Data for evaluation

Devices	Browser	Activities	Technologies
-HTC VIVE -Oculus Rift	-Firefox	-Particles -Objects	-WebGL -WebVR
-GearVR	-Firefox Reality -Oculus Browser -Samsung Internet		
-Smartphone (iOS y Android)	-Full Dive VR -VR Browser		

5 Results and Discussion

The results obtained through the different case studies detailed in the previous chapter, provide information on the performance of the different VR technologies used. To simplify the 28 tests carried out in total, a compilation is presented in each category. The VR App is published on a webserver and its necessary to access through a link. The duration of each test is unique due to the loading and representation time of each browser.

5.1 PC

In this category an Alienware 17 R4 computer has been used with the following specifications: Intel Core i7-6700HQ @ 2.60 GHz Processor, 16 GB of RAM and a Nvidia GeForce-GTX 1070 graphics card with 8 GB of memory. For the first result, HTC VIVE headset (as a virtual reality device) and the Firefox browser have been used, running the test application in an online server, as shown in Fig. 4. It can be seen that the performance in FPS in the WebVR application is slightly higher as is the consumption of CPU and GPU resources, which indicates that, faced with a greater consumption of resources, there is no significant difference in performance in the FPS number.

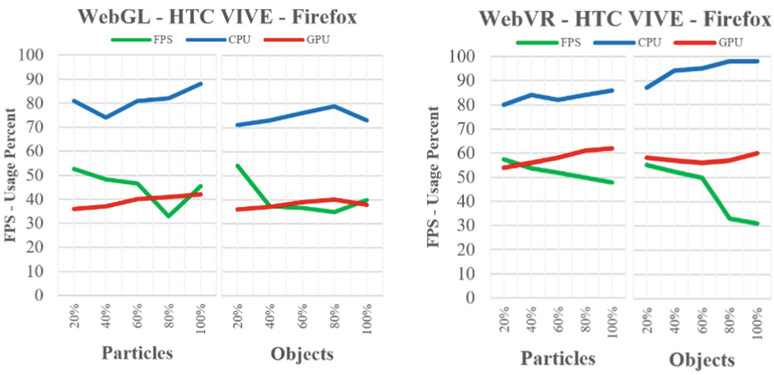


Fig. 4. HTC VIVE test results

For memory consumption VRAM is 512 MB for both technologies, but differs in 256 MB RAM for WebGL and 64 MB for WebVR.

The following result has been used with the Oculus Rift and Firefox browser, as can be seen in Fig. 5. The graphics behave similarly to the execution in the HTC VIVE headset which entails the same description. There are no relevant differences between the performance in FPS compared to the computational and graphic cost caused in the use of WebVR. For VRAM consumption it is the same for the two 512 MB technologies, but differs in 256 MB RAM for WebGL and 64 MB for WebVR.

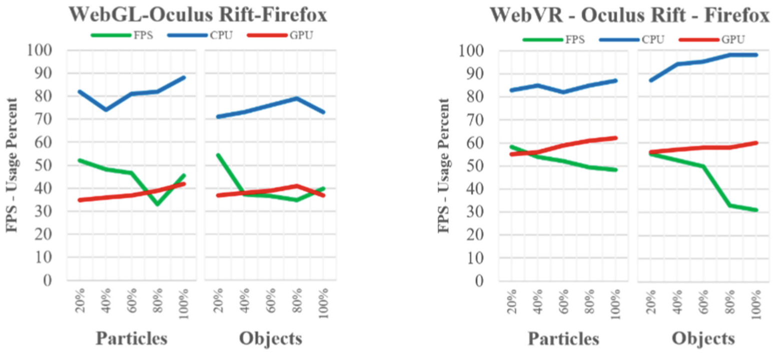


Fig. 5. Oculus Rift test results

5.2 Mobile

This section considers the use of a smartphone Samsung Galaxy S7 Edge with Android 8.0 OS and the following features: Exynos 8890 Octa 2.3 Ghz processor, 4 GB of RAM, Mali-T880 GPU and 3600mAh battery. In the first result, the GearVR glasses were used with the Oculus Browser, as can be seen in Fig. 6. The graphs have an improvement in FPS performance between the WebGL and WebVR technologies, giving the latter the best option for the execution of 3D content in web browsers inside virtual environments. In the battery section, a similar consumption has been recorded that ranges between 6%–8% for the execution of all the tests, which does not determine a competitive advantage between both technologies. For RAM consumption it is 512 MB for both, but it differs in 256 MB of VRAM for WebGL and 64 MB for WebVR.

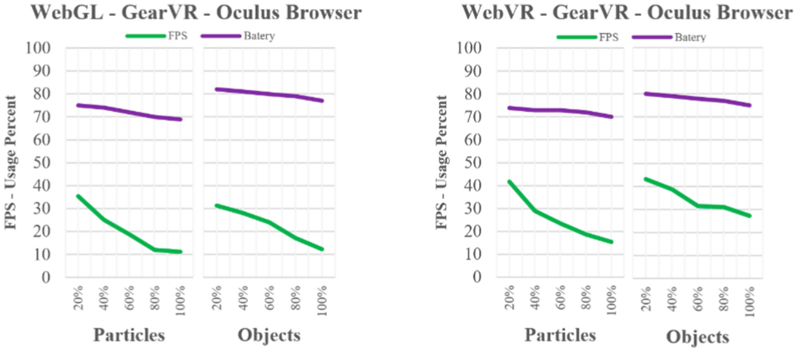


Fig. 6. Test results in GearVR running in OculusBrowser

In the second result we have used the Firefox Reality browser designed precisely to run remote applications in WebVR inside virtual environments. It should be mentioned that the application in WebGL was not executed by the browser, therefore there are no results as shown in Fig. 7. WebVR shows a lower performance than that achieved in the Oculus Browser, which indicates that the browser performs the remote application optimally. The consumption of RAM is 512 MB and for VRAM is 64 MB.

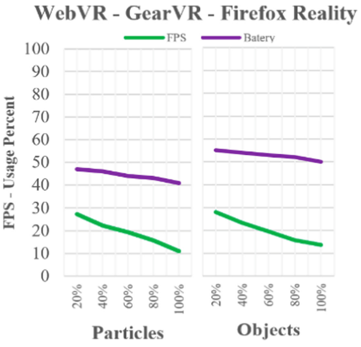


Fig. 7. Test results in GearVR running on Firefox Reality

In the third result, the Cardboard glasses were used in conjunction with the FullDive VR navigator, resulting in lower performance in FPS performance even with WebVR, as shown in Fig. 8. This determines that the best execution in the testing context has been performed in the Oculus Browser with the GearVR glasses. For the battery level, a similar value of consumption is observed in the execution of the tests of around 7% and 9%, which differs intimately in comparison to the energy consumed in the tests carried out with the GearVR glasses. For RAM consumption it is 512 MB for both technologies, but it differs in VRAM, 256 MB for WebGL and 64 MB for WebVR.

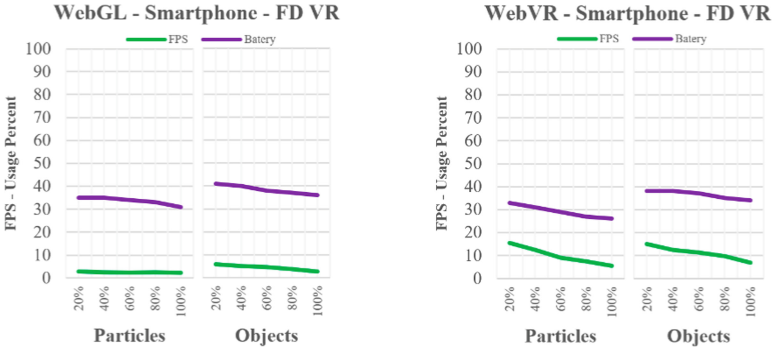


Fig. 8. Test results in Gear VR running on Firefox Reality

For iOS smartphones, no results have been registered because no existing browser has managed to run the remote application in either WebGL or WebVR.

6 Conclusions

The main objective of this work is to evaluate the performance of the devices and applications mentioned, for this purpose 28 experiments have been developed with the same test approach, in which the measurement of indicators such as: FPS, RAM, VRAM, CPU, GPU and battery have been recorded. The results vary slightly in resource consumption for PC platforms, and a marked difference in FPS performance for mobile devices (GearVR and Smartphones) can be seen.

Based on the tests carried out, we affirm that: WebVR technology performs efficiently on the mobile system due to lower VRAM memory consumption and a higher level of stable SPF compared to the same execution in a WebGL environment. Whereas WebGL performs stably but not more efficiently than WebVR based on the results obtained in the tests performed in this work.

From the mobile approach GearVR was identified as the technology that performs best executing the test. WebVR had the best performance in FPS on average, which determines an efficient consumption of computing resources in mobile devices, whose purpose is the efficiency in the execution of software.

From the PC point of view for HTC VIVE and Oculus Rift similar results have been seen in FPS performance on average. A slight difference in computational resource consumption between WebGL and WebVR determines that the use of these technolo-gies will depend on the application approach you plan to develop.

Finally, it is important to mention that the proposed evaluation was made by the need of the research group, to determine which devices and technologies are optimal for the development of applications in WebGL and WebVR. Future work is intended to evaluate the performance of the battery of the different devices, focused on factors that affect their proper functioning.

References

1. Maisto, M., Pacchierotti, C., Chinello, F., Salvietti, G., De Luca, A., Prattichizzo, D.: Evaluation of wearable haptic systems for the fingers in augmented reality applications. IEEE Trans. Haptics **10**(4), 511–522 (2017)
2. Slater, M., Wilbur, S.: A framework for immersive virtual environments (FIVE): Speculations on the role of presence in virtual environments. Presence Teleoperators Virtual Environ. **6**, 603–616 (1997)
3. Carrozzino, M., Bruno, N., Bergamasco, M.: Designing interaction metaphors for Web3D cultural dissemination. J. Cult. Herit. **14**(2), 146–155 (2013)
4. Leung, C., Salga, A.: Enabling WebGL. In: Proceedings of the 19th International Conference on World Wide Web - WWW 2010, p. 1369 (2010)
5. Chen, B., Xu, Z.: A framework for browser-based Multiplayer Online Games using WebGL and WebSocket. In: 2011 International Conference on Multimedia Technology, pp. 471–474 (2011)
6. Mobeen, M.M., Feng, L.: High-performance volume rendering on the ubiquitous WebGL platform. In: 2012 IEEE 14th International Conference on High Performance Computing and Communication and 2012 IEEE 9th International Conference on Embedded Software and Systems, p. 381 (2012)
7. Tinkercad-Team, Tinkercad|Create 3D digital designs with online CAD. https://www.tinkercad.com/. Accessed: 19 Jul 2019
8. Congote, J., Segura, A., Kabongo, L., Moreno, A., Posada, J., Ruiz, O.: Interactive visualization of volumetric data with WebGL in real-time. In: Proceedings of the 16th International Conference on 3D Web Technology - Web3D 2011, p. 137 (2011)
9. Kopeć, A., Bała, J., Pięta, A.: WebGL based visualisation and analysis of stratigraphic data for the purposes of the mining industry. Procedia Comput. Sci. **51**, 2869–2877 (2015)
10. Lv, Z., Yin, T., Han, Y., Chen, Y., Chen, G.: WebVR-web virtual reality engine based on P2P network (2011)
11. Fominykh, M., Prasolova-Førland, E., Morozov, M., Smorkalov, A., Molka-Danielsen, J.: Increasing immersiveness into a 3D virtual world: motion-tracking and natural navigation in vAcademia. IERI Procedia **7**, 35–41 (2014)
12. Neelakantam, S., Pant, T.: Introduction to VR and WebVR. In: Learning Web-based Virtual Reality, pp. 1–4. Apress, Berkeley (2017)
13. Letic, M., Nenadic, K., Nikolic, L.: Real-time map projection in virtual reality using WebVR. In: 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), pp. 1439–1443 (2018)

14. Butcher, P.W.S., Roberts, J.C., Ritsos, P.D.: Immersive Analytics with WebVR and Google Cardboard. In: Posters of IEEE VIS, pp. 30–32 (2016)
15. Lee, V.W., et al.: Debunking the 100X GPU vs. CPU myth. ACM SIGARCH Comput. Archit. News **38**(3), 451 (2010)
16. Smith, S.P., Trenholme, D.: Rapid prototyping a virtual fire drill environment using computer game technology. Fire Saf. J. **44**(4), 559–569 (2009)
17. Donalek, C., et al.: Immersive and collaborative data visualization using virtual reality platforms, October 2014
18. Vermeeren, A.P.O.S., Law, E.L.-C., Roto, V., Obrist, M., Hoonhout, J., Väänänen-Vainio-Mattila, K.: User experience evaluation methods. In: Proceedings of the 6th Nordic Conference on Human-Computer Interaction Extending Boundaries - NordiCHI 2010, p. 521 (2010)
19. Gonçalves, Y.A., Boas, V.: Overview of Virtual Reality Technologies. In: Interactive Multimedia Conference, pp. 20–26 (2013)
20. Asano, S., Maruyama, T., Yamaguchi, Y.: Performance comparison of FPGA, GPU and CPU in image processing, In: 2009 International Conference on Field Programmable Logic and Applications, pp. 126–131 (2009)
21. Yang, K., Jie, J., Haihui, S.: Study on the virtual natural landscape walkthrough by using Unity 3D. In: 2011 IEEE International Symposium on VR Innovation, pp. 235–238 (2011)
22. Lugrin, J.L., Cavazza, M., Charles, F., Le Renard, M., Freeman, J., Lessiter, J.: Immersive FPS games: user experience and performance. In Proceedings of the 2013 ACM International Workshop on Immersive Media Experiences, October 2013, pp. 7–12. ACM (2013)