Indonesian Journal of Computer Science
ISSN 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam No. 1, Padang, Indonesia
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

Capabilities Comparison Of The Augmented Reality Application Development Frameworks On The Android Platform

Mifta Aprilya Suryani[1], Muhammad Ihsan Jambak[2*], Pacu Putra[3]
miftaaprilya@gmail.com[1], jambak@unsri.ac.id, pacuputra@unsri.ac.id
[1]Department of Informatics, Faculty of Computer Science, Universitas Sriwijaya
[2]Departemt of Informatics Management, Faculty of Computer Science, Universitas Sriwijaya
[3]Departement Information Systems, Faculty of Computer Science, Universitas Sriwijaya

Article Information

Submitted : 19Nov 2023
Reviewed: 1 Jan 2024
Accepted : 19 Feb 2024

Keywords

Augmented Reality, Vuforia, ARCore, Android

Abstract

In the implementation of augmented reality, the natural process of human-computer interaction continues to be a challenge, specifically to reduce the complexity in the effort of using and providing comfortability. Therefore, it is necessary to search for methods and tools to simplify the complex process of building Augmented Reality applications. Vuforia is a multi-platform that has long been used as an Augmented Reality application development framework. Android and iOS platform frameworks are the new alternatives following the development of cellular technology. Furthermore, a comparative capability was conducted between Vuforia and ARCore on the Android platform. The general performance and the ability to understand the environment include working in horizontal and vertical planes, the ability to work based on lighting conditions, and the distance of the camera during the tests. The results showed that ARCore is superior to Vuforia in almost all testing metrics. However, in overexposed or too dim lighting and at very close surfaces, Vuforia is slightly superior but not essential. Therefore, it can be concluded that ARCore's capability is better than Vuforia.

A. Introduction

Augmented Reality (AR) and Virtual Reality (VR) have almost similar characteristics in the field of human-computer interaction. The main goal of VR and AR is to trick the human perception system into accepting rendered objects as real by immersing the user in the created world. The difference is in the role of the real and the created world, and when VR disables the real-world state, the user will be placed in a completely different environment or the world. Conversely, AR connects virtual objects with the real world, which has a more pronounced role since the user's normal view is enhanced following the addition of visual and auditory information by computers [1-3]. Both of these methods require an efficient computer and special equipment, such as the Graphics Processing Unit (GPU) [4, 5]. The rapid development of powerful and reliable mobile devices capable of handling AR and VR in almost real-time should be promoted [6]

AR technology has the potential to be developed, but it presents complex human-computer interfaces and interactions [7-9]. This is because virtual information is added to the real-world environment through the human senses (specifically vision). Therefore, the main challenge is to reduce the complexity of use and provide convenience. The natural process of human-computer interaction has become a trend in the ubiquitous and pervasive computing community, and it imitates the workings of traditional human-human interactions [10].

Several frameworks and software development kits (SDK) are readily available, including ARCore, ARKit, ARToolkit, Kudan, MAXST Wikitude, and Vuforia to facilitate rapid prototyping in the development of AR applications. Qualcomm developed Vuforia, which has long been used as an AR application development platform. Furthermore, Vuforia can be used in all types of operating systems and devices, Linux, Windows, MAC OS, Android as well as iOS, and because of its features, the SDK is the most widely used by developers. In addition, competitive application programming interfaces have emerged to support the creation of AR applications for mobile devices. These include ARCore by Google and ARKit by Apple, which provides new opportunities for mobile device users to create applications and games in depth.

Behind the prediction of the very large size of the AR market, the natural process of human-computer interaction remains a challenge in its implementation, in particular, to reduce complexity and offer convenience when using AR technology. Therefore, it is necessary to search for methods and tools to simplify the complex process of building realistic AR applications.

A comparison of ARKit and ARCore SDKs has been conducted [11], however, behind the advantages and disadvantages, these two technologies require expensive mobile devices, specifically ARKit that is only used on iOS platforms released in 2017 and above. Meanwhile, the devices needed by ARCore are still quite affordable even though they should fulfill the standards of Android output 2017 and above.

Considering that the Android platform is more massive and mass-used in the community, it becomes a strong reason to examine the capabilities of the Vuforia SDK compared to those offered by the ARCore. The results of the comparison of the working mechanism showed the advantages and disadvantages that underlie the selection of a framework with a simpler complexity in developing AR applications.

Indonesian Journal of Computer Science
ISSN 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam No. 1, Padang, Indonesia
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

B. Research Method

ARCore is an SDK used to build AR applications based on Android, and it uses a new concept to recognize and track objects captured on camera in real-time. The main concepts are motion tracking, environmental understanding, and light estimation. Motion tracking can recognize points called features and track them during motion. This point movement is combined with sensors built into Android to produce the required position and orientation in markerless AR.

Meanwhile, Vuforia is an SDK developed by Qualcomm and was widely implemented for interactive learning media, entertainment, news, information, and games. It has two methods in developing AR applications, namely the Marker-Based Tracking and Markerless-Based Tracking methods. In the Markerless-Based Tracking method, the flagship technologies include Face, 3D Object, Motion, and GPS-Based Tracking.

To compare the capabilities of the two frameworks, an AR application with the Android platform has been developed using the features of the Vuforia and ARCore frameworks. The AR application developed is "Door Switching" and it takes the simple concept of a magic door in the Doraemon series. Both frameworks provide functionalities that allow experiments to be conducted as follows: detection of flat surfaces, depicting feature points and planes on detected surfaces, placing virtual objects on the stage, adjusting to actual lighting, and measuring and storing time information [3, 11-13].

These features are used as the basis for testing the comparison of the two frameworks. The required data is in the form of real-world environmental surface areas obtained from cellular camera captures. Furthermore, it uses a rear camera that captures the ground plane, and the virtual object displayed is created through the “Moving Doors” application.

The test criteria established during the experiment are overall performances and the ability to understand the environment [11, 14, 15]. The general performance in question is the time to load (render) the virtual model or object from tapping on the plane until the object appears. The second criterion, is the ability to understand the environment which is a key element of AR because the quality of the surface detection of the plane defines the entire AR application.

The ability to understand this environment defines work areas detection in various conditions. This includes recognizing horizontal and vertical planes, recognizing planes in various lighting conditions, as well as recognizing planes at various distances between the camera and virtual objects. The surface area detection capability is measured by several components, namely: a. Coverage of Plane Detection such as the percentage of the area detected successfully from the provided plane; b. First Plane Time, namely the time for the first plane to appear; c. Plane Coverage Time including the total time required for the plane coverage completion.

To measure the ability to detect horizontal (floor) and vertical surfaces (walls), 4 slope conditions are used, namely 0^o, 45^o, 75^o, and 90^o [11] by varying the camera angle. This ability is needed because when it is only good in vertical conditions or only horizontally, it limits the user's movement, inflexible, and stiff. Furthermore, the virtual object is still perpendicular to the floor while the camera is facing the wall. The next criteria are the ability to work in various lighting conditions. The impact on surface detection is observed, and this consists of outdoor (day-light), indoor weak (11 lux), and

Indonesian Journal of Computer Science
ISSN 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam No. 1, Padang, Indonesia
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

normal indoor lighting (52 lux) [11]. It aims to determine the optimal conditions in the use of AR applications later.

The ability to work at a distance between the camera and the surface should also be considered since the criterion is needed concerning the size of the virtual object used. A large virtual object, for example, a door, requires that the surface should be far from the camera to be fully visible. In this experiment, distances of 20 cm and 60 cm are used to represent close distances, while 90 cm and 1 meter+ are used to represent long distances.

In setting the test conditions, the experiment was conducted on a plane with an area of 1x1 m2 and 40 times per framework. To display virtual objects on a flat surface, several steps should be carried out, namely determining feature points as well as estimating position and orientation before using anchors and trackable. The general architecture used to describe the stages of the method is shown in Figure 1.

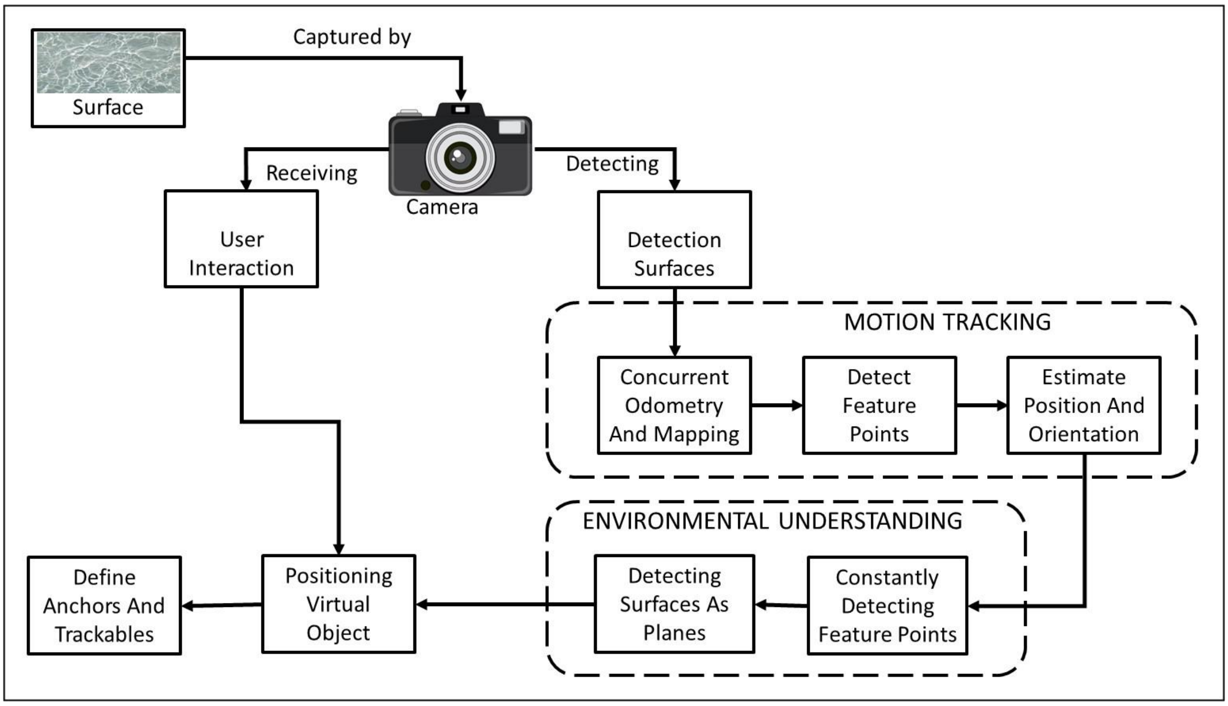

Figure 1. Testing Work Diagram

The user points the camera at the surface (floor surface) captured directly. It is called the processed input, and it can display virtual objects on the planes. The object is a door that can display a virtual room, and in this surface detection process, the textures on the surface are detected by using the camera's distance to the texture.

The system performs motion tracking with the concurrent odometry and mapping (COM) process along with surface detection. A visual representation of a 3-dimensional environment is built by tracing the environment in the real world. Therefore, with the COM process, the location of the feature points can be detected by analyzing the orientation position of the camera and its surroundings. Furthermore, the results of the feature points detection are used to update the camera location continuously.

Virtual objects are observed from various camera angles with motion tracking. This can be conducted since the Inertial Measurements Units (IMU) in smartphones combine and measure the information obtained to produce an estimate of the position and orientation of the camera relative to the real environment in real-time. Meanwhile, the IMU is an electronic device that combines accelerometers and gyroscopes to measure angle levels.

The process of understanding the environment is conducted by converting a set of feature points into planes. Therefore, the plane is the surface detected and tracked continuously following the camera movement. Material plane settings and special shaders adjust the size following the surface area detected. Each plane has a different shader color when the ground has various objects with varying heights. For example, a table and a floor are captured in one scene, then two planes with different colors are depicted. In addition, the plotted area remains in position even when the camera capture range is moved.

The process of user interaction is to place virtual objects by tapping on the smartphone screen and give commands to display virtual objects on top of the plane. The position of the virtual object is determined according to the user's will, and in this study, a door leading to the virtual room is used. Furthermore, the anchor and trackable process are conducted, and the virtual object's position and orientation are maintained in their original place even though the camera's position and angle are moved.

To find out the comparison results of ARCore and Vuforia capabilities as an Augmented Reality Application framework, the mean (average) and standard deviation of 40 trials should be calculated for each predetermined condition to evaluate the accuracy and precision (framework stability). In comparing the results of the framework accuracy and stability, a higher mean value is considered better. Meanwhile, the smaller standard deviation value indicates better precision or stability. Statistical tests were applied in data analysis using SPSS version 22, and the equipment used was a T-test for comparison of two data groups, ANOVA (Analysis of Variance) for comparison of more than two groups of data, and Tukey Post-Hoc for multiple comparison tests [16].

C. Result and Discussion

The first test was conducted on various conditions of camera slope to the floor surface, and at a slope of 0°, it faces straight and parallel to the floor. Furthermore, the camera is angled at 45°, 75°, until 90° parallel to the wall. Experimental observations obtain the coverage value (in %), time of appearance of the first plane (FP) (in seconds), and the time to complete the coverage plane (PC) (in seconds). The test results are shown in Tables 1 and 2.

Table 1. Mean Value of Test Criteria on Slope Conditions

Angle	ARCore			Vuforia		
	Coverage (%)	FP (seconds)	PC (seconds)	Coverage (%)	FP (seconds)	PC (seconds)
0°	96.0352	4.6677	10.2520	94.0380	5.9825	10.6270
45°	95.1835	5.1220	11.0997	93.8497	6.4500	11.3147
75°	94.7072	5.6297	11.4982	92.8512	7.3337	13.5540
90°	93.9872	6.0625	13.1835	92.0592	7.4602	14.0507

Table 2. Standard Deviation Value of Test Criteria on Slope Condition

Angle	ARCore			Vuforia		
	Coverage (%)	FP (seconds)	PC (seconds)	Coverage (%)	FP (seconds)	PC (seconds)
0°	2.6834	1.179	1.703	4.4820	1.670	2.526
45°	2.9932	1.341	2.820	4.5376	1.361	2.315
75°	2.8219	1.073	2.765	3.0038	1.591	2.537

90°	3.1474	1.288	2.614	4.1972	1.455	2.571

Statistical tests using ANOVA and Post-hoc Tukey explain that determining the type of SDK and the camera tilt position against the floor surface are significant factors for the results. This should be considered in terms of accuracy (% coverage), the first time the plane appears, and the total time the area detection completes. Furthermore, the slope angle and the results obtained are inversely proportional to each other, therefore, the best results are obtained at 0° concerning the floor.

Figure 2 showed that ARCore is superior to Vuforia on all slope degrees. At all angles tested, the mean accuracy (% coverage) outperformed Vuforia. Statistical test with t-test showed that there was a significant difference between ARCore and Vuforia with a difference of 1.79% for accuracy. Similarly, the mean time needed to display the first plane based on the t-test showed a significant difference, where ARCore is 21.09% faster than Vuforia. The time required to complete area detection also has a significant difference, where ARCore is also 7.09% faster than Vuforia.

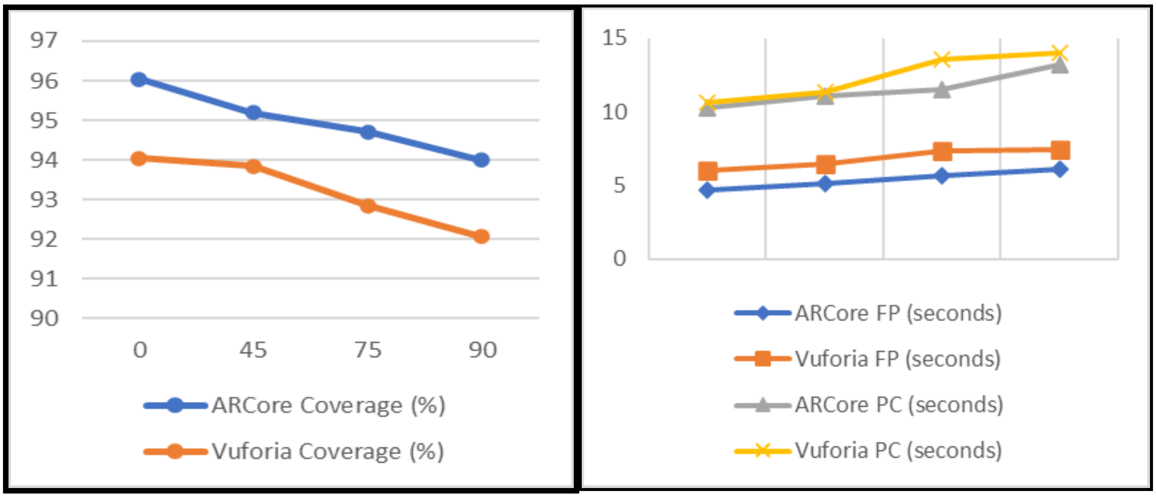

Figure 2. Comparison of ARCore and Voforia Mean Values on Slope Conditions

The comparison of the standard deviation values in the slope experiment is shown in Figure 3, where ARCore tends to have a smaller value in all tested slope conditions, therefore, it is believed to be more stable than Vuforia. However, Vuforia displays slightly smaller values than ARCore considering the plane coverage at slope degrees of 45°, 75°, and 90°.

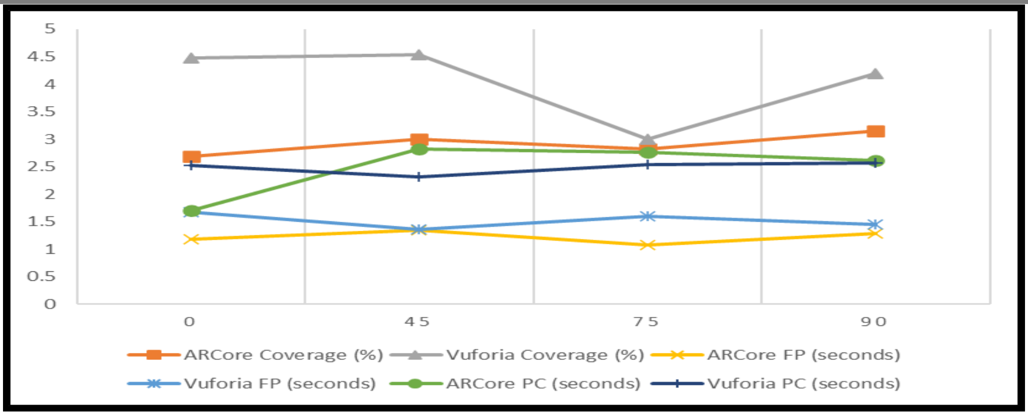

Figure 3. Comparison of Standard Deviation between ARCore and Vuforia on Slope Condition

The second test was conducted to examine the ability to work in various lighting conditions, namely outdoor (day-light), and indoor lighting, both dim and bright. Analysis of the test results is also conducted on the mean value of accuracy (% coverage), the time for the appearance of the first plane of FP (seconds), and the time to complete the PC plane coverage (seconds). The test results are shown in Tables 3 and 4.

Table 3. Mean Value of Test Criteria in Lighting Conditions

Light	ARCore			Vuforia		
	Coverage (%)	FP (seconds)	PC (seconds)	Coverage (%)	FP (seconds)	PC (seconds)
daylight	95.6055	8.131	13.865	94.1657	7.047	11.579
11 lux	91.7667	10.064	17.348	90.9905	12.970	21.0135
52 lux	96.2637	4.7366	12.586	93.9980	7.265	14.081

Table 4. Standard Deviation Value of Test Criteria in Lighting Conditions

Light	ARCore			Vuforia		
	Coverage (%)	FP (seconds)	PC (seconds)	Coverage (%)	FP (seconds)	PC (seconds)
daylight	2.30801	1.453	1.404	3.41548	1.005	1.151
11 lux	3.41224	2.249	3.032	5.37267	3.474	4.755
52 lux	2.49942	1.264	1.3705	4.23460	1.859	1.764

Statistical tests with ANOVA and Post-Hoc Tukey explained that the determination of the SDK type and the lighting level was significant factor affecting the results. Therefore, there is an interaction factor between SDK type and lighting level that affects the first time and the total plane detection completion time. Generally, both SDKs require optimum lighting or normal conditions, where excessive and very dim lighting degrades the performance of both.

The t-test statistical test in the various lighting conditions showed that there is a significant difference between ARCore and Vuforia in the aspect of accuracy (% coverage) with a difference of 1.49%, and the appearance time of the first plane ARCore is superior and faster than Vuforia 22.74%. However, in the aspect of total plane coverage completion time, there is no significant difference, therefore, it can be believed that the capabilities of both are the same.

Indonesian Journal of Computer Science
ISSN 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam No. 1, Padang, Indonesia
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

The outdoor conditions are the better time to display the first plane when explored per lighting conditions and to complete the entire plane Vuforia excels because it is faster. Meanwhile, for low light conditions, the % coverage of both is similar, and in the comparison of standard deviation values, ARCore has a smaller value. Therefore, it can be believed to be more stable as well as superior to Vuforia in low and normal light conditions. Vuforia outperformed ARCore in outdoor lighting conditions in terms of time to display the first and complete the overall plane view. Figures 4 and 5 showed a comparison of test results under various lighting conditions, both on the mean value and the standard deviation.

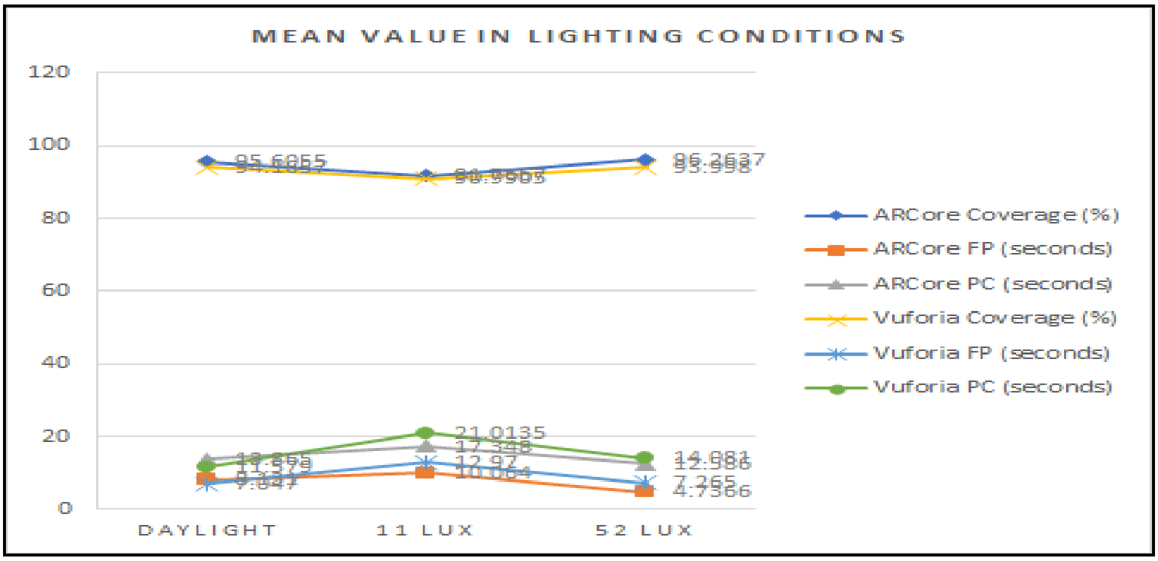

Figure 4. Comparison of ARCore and Vuforia Mean Values in Lighting Conditions

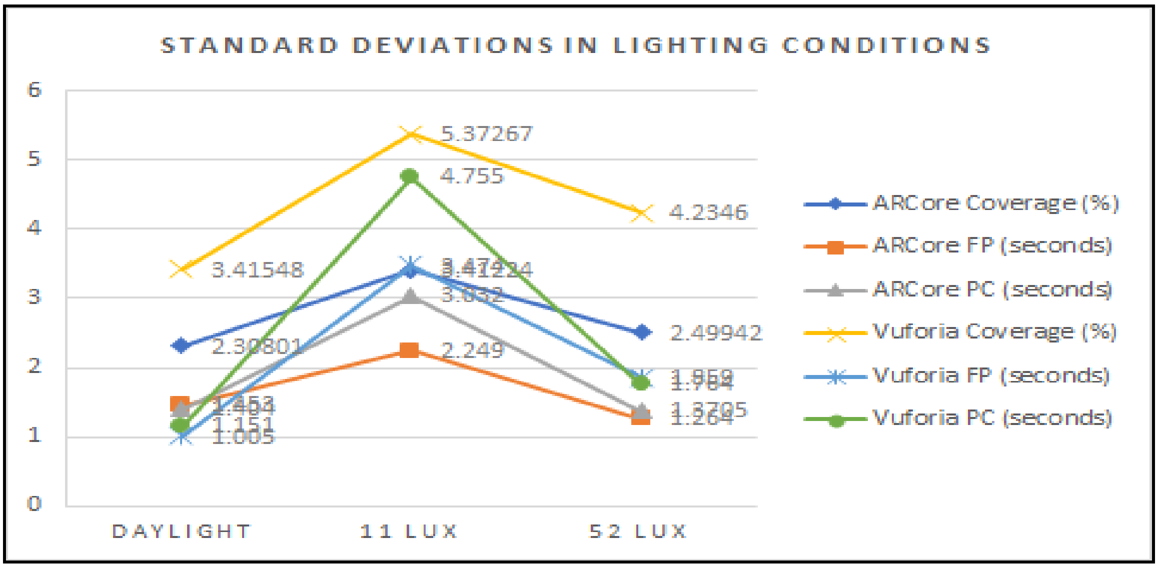

Figure 5. Comparison of ARCore and Vuforia Standard Deviations in Lighting Conditions

The third experiment was conducted to test the ability to work at a distance from the camera and the surface, both near and far. Close distances were tested at a distance of 20 cm and 60 cm. Meanwhile, long distances were tested at 90 cm and more than 1 m, and the results of the observations are shown in Tables 5 and 6.

Indonesian Journal of Computer Science
ISSN 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam No. 1, Padang, Indonesia
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

Table 5. Mean Value of Test Criteria on Camera Distance Conditions

Distance	ARCore			Vuforia		
	Coverage (%)	FP (seconds)	PC (seconds)	Coverage (%)	FP (seconds)	PC (seconds)
20 cm	95.0275	3.182	4.59225	93.9175	2.427	4.312
60 cm	94.917	4.002	6.06	93.511	3.49	5.348
90 cm	96.9385	4.8315	10.8175	93.00175	6.997	13.122
1 M +	97.88275	6.143	13.4675	90.13075	7.977	14.788

Table 6. Standard Deviation Value of Test Criteria on Camera Distance Conditions

Distance	ARCore			Vuforia		
	Coverage (%)	FP (seconds)	PC (seconds)	Coverage (%)	FP (seconds)	PC (seconds)
20 cm	2.408	1.2505	0.965	4.6901	1.097	1.129
60 cm	2.7107	1.1442	1.2656	4.8712	1.067	1.636
90 cm	2.0483	1.3626	1.114	4.4853	1.113	1.961
1 M +	1.9326	0.9699	1.097	3.0494	1.114	2.104

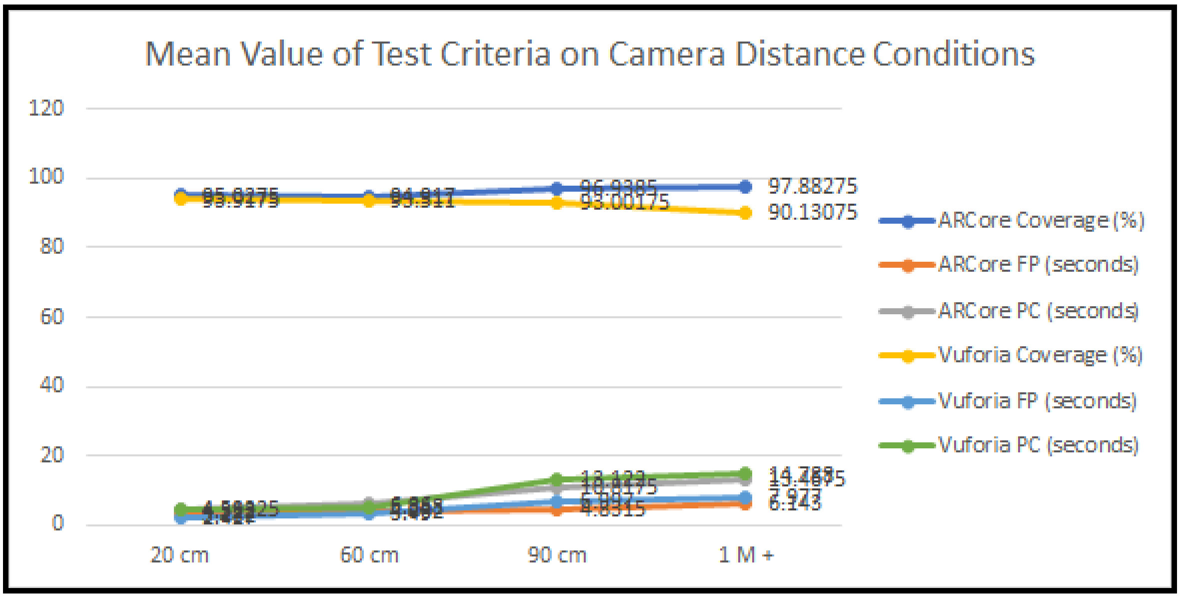

Figure 6. Comparison of ARCore and Vuforia Mean Values on Camera Distance Conditions

Indonesian Journal of Computer Science
ISSN 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam No. 1, Padang, Indonesia
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

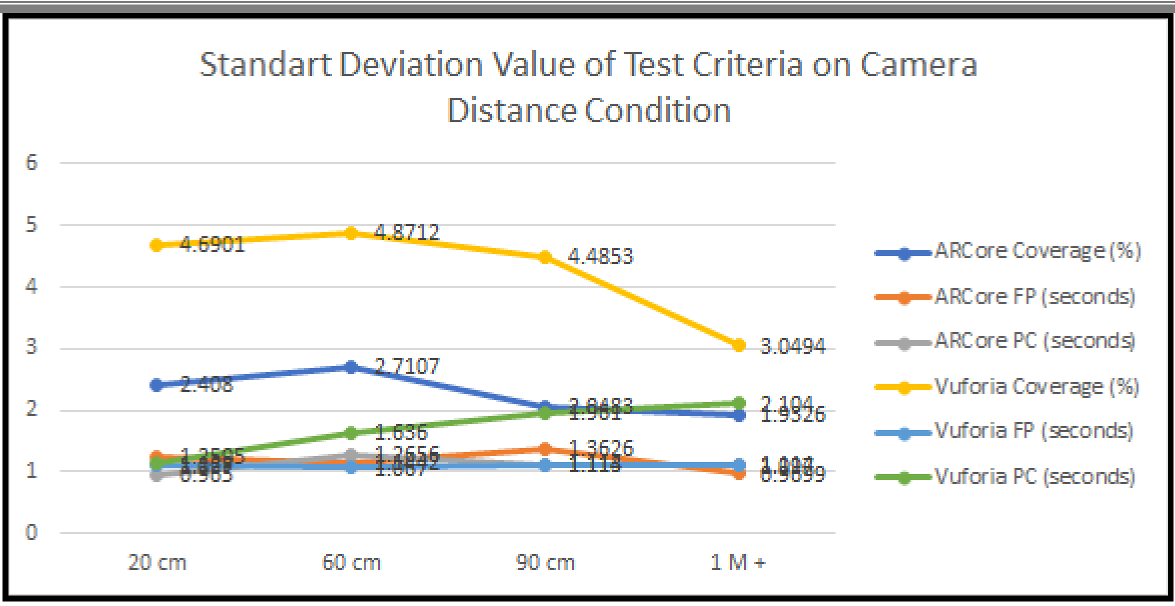

Figure 7. Comparison of ARCore and Vuforia Standard Deviations on Camera Distance Conditions

The camera distance setting in Figure 6 showed that ARCore is superior to Vuforia in the value of % coverage. Vuforia outperformed the mean speed for displaying the first and the total plane detection completion time, specifically while handling close-up objects.

Statistical tests with ANOVA and Post-Hoc Tukey explained that determining the SDK type and the distance of the camera to virtual objects were significant factors for the results, except for the accuracy (% coverage). The difference in % coverage by distance is not significant or considered the same. Therefore, distance only affects the first plane appearance time and the total plane detection completion time.

The accuracy tends to get better while Vuforia gets worse as ARCore gets further away. There is a tendency that the farther the distance, the longer it will take to display the first plane and also the longer the total plane detection completion time.

For comparison on distance variations, ARCore significantly outperformed Vuforia, with an accuracy difference of 3.55% and 13.07% faster while displaying the first plane. However, in the total plane detection completion time, there is no significant difference. It can be believed that ARCore's capabilities are the same as Vuforia's. Figure 7 showed the difference in the standard deviation value of % coverage, where ARCore has better stability at all distance settings. However, there is a tendency for stability % coverage as the camera distance increases in Vuforia

The final test measures the rendering time of the virtual door object. Table 7 and Figure 8 showed the mean and standard deviation of the 40 times virtual object rendering time required by ARCore and Vuforia. The statistical t-test showed that ARCore has significantly faster rendering capabilities than Vuforia, with a difference of 38.54%, and in terms of stability indicated by a smaller standard deviation value. Therefore, for General Performance testing, it can be believed that ARCore is better than Vuforia following the speed of the virtual object rendering time.

Table 7. Comparison of Mean Value and Standard Deviation of Virtual Object Rendering Time

	ARCore (seconds)	**Vuforia (seconds)**
Mean	2.594975	5.07825
std.dev	1.404156969	1.741487557

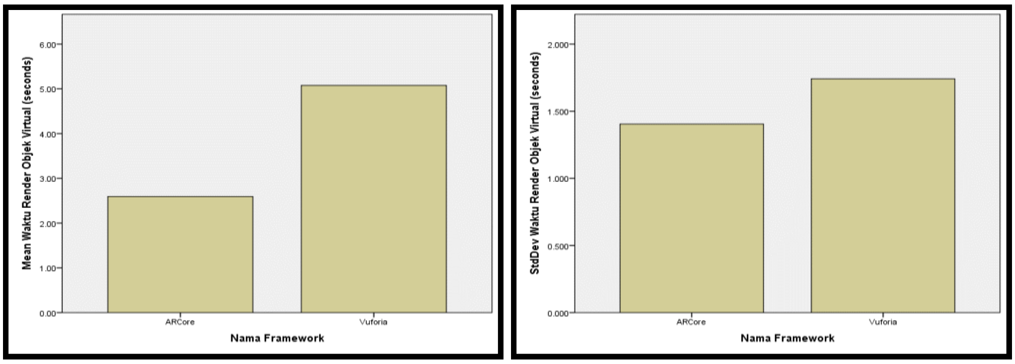

Figure 8. Differences in Mean Value and Standard Deviation of Virtual Object Rendering Time

The simultaneous analysis determines the general ability to understand the environment based on plane detection accuracy, first plane time speed, and total plane detection completion time. In the plane detection accuracy on all conditioning factors, ARCore is completely superior to Vuforia as shown in Figure 9. For the first plane time speed on most conditioning factors, ARCore is superior, but on the condition of camera distance to near virtual objects, Vuforia is better, as shown in Figure 10. Furthermore, for the total plane detection completion time on most conditioning factors, ARCore is better than Vuforia, except for the outdoor light factor and the close distance between the camera and the virtual object, as shown in Figure 11.

In the deepening on the test of the ability to detect horizontal and vertical planes, the Anova test showed that there were significant differences between the ARCore slope angle groups in the three test components. Furthermore, it can be believed that the camera angle affects the final result. However, this is not the case for Vuforia, where the whole group of test data is different but not significant and can be assumed to be similar. Therefore, the final result on the Vuforia is not affected by the camera angle.

The Anova test on the ability to detect planes with light conditioning showed that both ARCore and Vuforia were significantly affected by lighting quality factors. They have the same pattern, which provides optimal results in normal indoor and outdoor lighting as a second alternative. In the test on plane detection conditioned by the distance of the camera to the virtual object, the Anova test showed different characteristics between ARCore and Vuforia. In ARCore, the close-range and long-range test scenarios were dichotomized, but in Vuforia, the significant difference was more or less than 1 meter. In Vuforia, there was a normative tendency since the result is better while the camera is closer to the virtual object. Contradictory conditions occur in ARCore, where the farther the virtual object is from the camera, the better the accuracy. Therefore, size can be a factor to be considered in setting the camera distance to the virtual object.

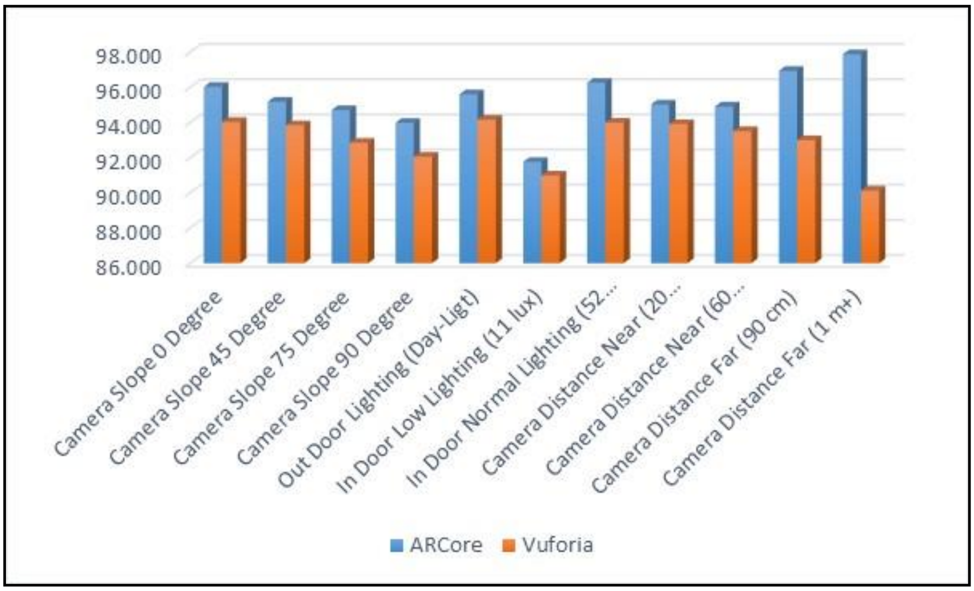

Figure 9. Differences in Mean Accuracy Value between ARCore and Vuforia Based on Conditioning Factors

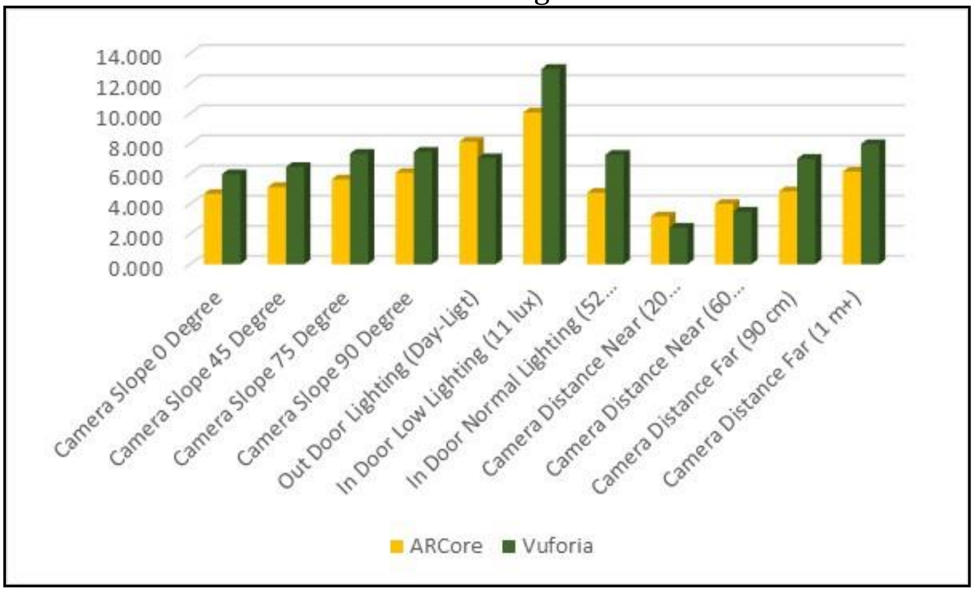

Figure 10. Differences in the Mean Value of the First Time Plane Appears between ARCore and Vuforia Based on Conditioning Factors

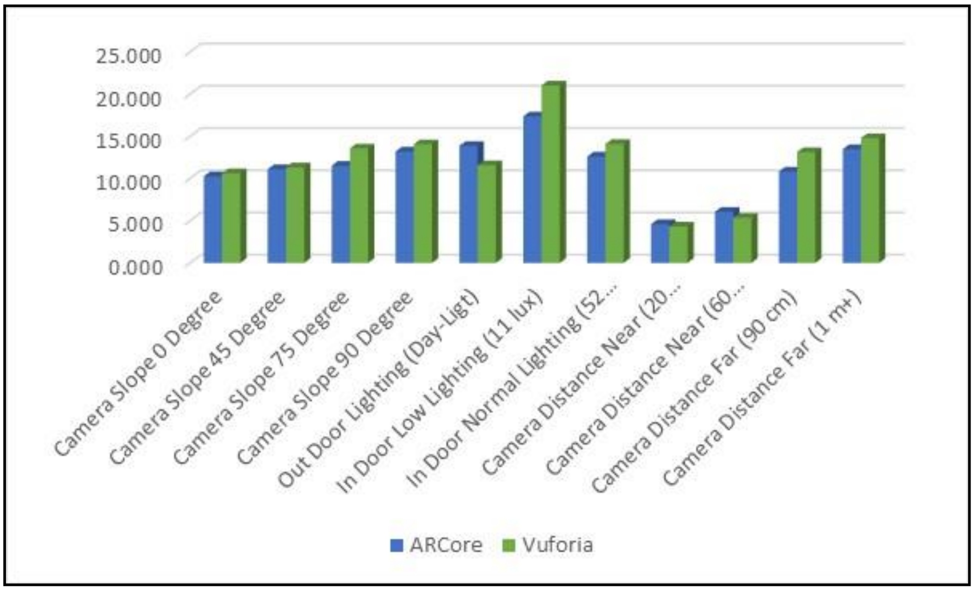

Figure 11. Differences in the Mean Value of the Total Plane Detection Completion Time between ARCore and Vuforia Based on Conditioning Factors

The simultaneous analysis showed the general ability to understand the environment based on the accuracy of plane detection, the first plane appearance speed, and the total plane detection completion time. In the detection accuracy on all conditioning factors, ARCore is completely superior to Vuforia as shown in Figure 9. For the speed of first plane appearance time, on most conditioning factors, ARCore is superior, but the camera distance to near virtual objects is better as shown in Figure 10. Most ARCore conditioning factors are better than Vuforia's for the total plane detection completion time, except for the outdoor light factor and the close distance between the camera with virtual objects.

D. Conclusion

This study examines the comparison of ARCore's capabilities with Vuforia as an AR application developer framework on the Android platform. The results of all test metrics on several camera position slope conditions, lighting variations, camera distance variations, virtual door object rendering time showed that ARCore is better than Vuforia. The advantage is in the ability to understand the surrounding environment.

Vuforia is faster in displaying the plane for very bright and too dim lighting conditions. It is also faster in displaying planes in very close object conditions. However, it cannot be compared with the capabilities of ARCore. This is because the virtual object used in this application is a door, which in the AR application is placed under normal indoor lighting conditions. Furthermore, the door as a large virtual object requires a distance of more than 1 meter to feel the advantage at close range.

E. References

[1] R. T. Azuma, "A survey of augmented reality," *Presence: Teleoperators & Virtual Environments,* vol. 6, no. 4, pp. 355-385, 1997.

[2] R. Duraiswami and A. O'Donovan, "CREATING SCIENTIFICALLY VALID SPATIAL AUDIO FOR VR AND AR: THEORY, TOOLS AND WORKFLOWS," presented at the AUDIO FOR VIRTUAL AND AUGMENTED REALITY CONFERENCE, Los Angeles, USA, 30 Sep 2016, 2016.

[3] P. Soltani and A. H. Morice, "Augmented reality tools for sports education and training," *Computers & Education,* p. 103923, 2020.

[4] R. Abboud, "Architecture in an Age of Augmented Reality: Opportunities and Obstacles for Mobile AR in Design," *Construction, and Post-Completion,* 2014.

[5] P. Willemsen, W. Jaros, C. McGregor, E. Downs, M. Berndt, and A. Passofaro, "Memory task performance across augmented and virtual reality," in *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, 2018: IEEE, pp. 723-724.

[6] M. Halpern, Y. Zhu, and V. J. Reddi, "Mobile cpu's rise to power: Quantifying the impact of generational mobile cpu design trends on performance, energy, and user satisfaction," in *2016 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, 2016: IEEE, pp. 64-76.

[7] G. Bhorkar, "A survey of augmented reality navigation," *arXiv preprint arXiv:1708.05006,* 2017.

[8] C. Lytridis and A. Tsinakos, "Evaluation of the ARTutor augmented reality educational platform in tertiary education," *Smart Learning Environments,* vol. 5, no. 1, p. 6, 2018/08/06 2018, doi: 10.1186/s40561-018-0058-x.

[9] M. Billinghurst, A. Clark, and G. Lee, "A Survey of Augmented Reality," *Foundations and Trends® in Human–Computer Interaction,* vol. 8, no. 2-3, pp. 73-272, 2015, doi: http://dx.doi.org/10.1561/1100000049.

[10] R. M. Carvalho, R. M. de Castro Andrade, K. M. de Oliveira, I. de Sousa Santos, and C. I. M. Bezerra, "Quality characteristics and measures for human–computer interaction evaluation in ubiquitous systems," *Software Quality Journal,* vol. 25, no. 3, pp. 743-795, 2017.

[11] P. Nowacki and M. Woda, "Capabilities of ARCore and ARKit Platforms for AR/VR Applications," in *International Conference on Dependability and Complex Systems*, 2019: Springer, pp. 358-370.

[12] R. Bergquist and N. Stenbeck, "Using Augmented Reality to Measure Vertical Surfaces," ed, 2018.

[13] J. Glover, *Unity 2018 augmented reality projects: build four immersive and fun AR applications using ARKit, ARCore, and Vuforia*. Packt Publishing Ltd, 2018.

[14] A. J. Davison, "Real-time simultaneous localisation and mapping with a single camera," in *Computer Vision, IEEE International Conference on*, 2003, vol. 3: IEEE Computer Society, pp. 1403-1403.

[15] G. Reitmayr *et al.*, "Simultaneous localization and mapping for augmented reality," in *2010 International Symposium on Ubiquitous Virtual Reality*, 2010: IEEE, pp. 5-8.

[16] H. S. Wibowo, *Analisa Statistik I & II* (Modul Pelatihan Statistik). Depok: P3M Universitas Indonesia, 2016.