Journal of Applied Informatics and Computing (JAIC)
Vol.10, No.1, February 2026, pp. 718~728
e-ISSN: 2548-6861

Experimental Comparison of Ground Plane Detection Speed Across Mobile Platforms

Laura Ladiesga [1]*, Caraka Aji Pranata [2], Heri Sismoro[3]*****
* Information Systems, Amikom Yogyakarta University
** Information Technology, Amikom Yogyakarta University
*** Informatics Management, Amikom Yogyakarta University
lauraladiesga@students.amikom.ac.id[1], caraka@amikom.ac.id[2], herisismoro@amikom.ac.id[3]

Article Info

Article history:

Received 2025-12-11
Revised 2025-12-31
Accepted 2026-01-08.

Keywords:

Markerless Augmented Reality,
Ground Plane Detection,
Three-Way ANOVA,
Platform Comparison,
Detection Speed.

ABSTRACT

Markerless Augmented Reality (AR) technology has become increasingly important in various applications, yet its performance varies significantly across different platforms. This study conducts a comparative experimental analysis of ground plane detection performance between iOS and Android platforms using the Vuforia-based KreasiFurniture application. The research examines detection speed under varying lighting conditions (indoor and outdoor) and camera distances (50 cm, 100 cm, and 150 cm) through systematic testing with five repetitions per condition. Data were analyzed using Three-Way ANOVA with IBM SPSS Statistics 25. Results demonstrate that iOS achieves significantly faster and more consistent detection (mean = 1.402 seconds, SD = 0.143) compared to Android (mean = 1.541 seconds, SD = 0.235), with a statistically significant difference of 0.139 seconds ($p = 0.003$). The optimal detection distance was found at 100 cm for both platforms ($p = 0.018$). While lighting conditions showed no significant main effect ($p = 0.129$), a significant Platform × Light interaction ($p = 0.038$) was revealed, indicating that iOS maintains stable performance across lighting variations, whereas Android experiences substantial performance degradation in indoor conditions. These findings provide practical recommendations: iOS is preferable for applications requiring consistent indoor performance, 100 cm represents the optimal interaction distance for both platforms, and Android deployments should implement adaptive strategies for variable lighting conditions.

I. Introduction

The development of augmented reality technology is rapidly accelerating and has been applied in various fields, education, construction, public health, manufacturing, and entertainment [1]. One of the widely used augmented reality technologies is marker-based Augmented Reality (AR), an immersive technology that utilizes special markers, such as QR codes or pattern images, as a reference to display virtual objects. Although it is quite reliable under clear conditions, this method is less practical as it requires physical markers that must be present in the user's place [2]. In addition to marker-based AR, a widely used approach in augmented reality is Markerless Augmented Reality (MAR). Unlike marker-based AR, markerless AR enables the display of virtual content without the need for markers, such as QR codes or custom images [3].

Markerless Augmented Reality (AR) makes extensive use of Visual Simultaneous Localization and Mapping (Visual SLAM) technology. An example in this case was the use of ORB-SLAM3 and Visual-Inertial Odometry (VIO). These two methods work by combining data obtained from the camera with additional information from the Inertial Measurement Unit (IMU) sensor, enabling the device to perform real-time position and orientation tracking. This approach allows AR systems to render virtual objects with a high level of accuracy and remain stable, even when the user is moving freely in the real environment without the need for special markers or markers. [4].

Markerless AR has the advantage of high flexibility, as users only need to point the camera at any surface, without the need for a marker to display virtual objects. This is especially useful in applications such as education, promotion, or entertainment that want to create more intuitive interactions. [5]. However, this technology also presents significant challenges, such as jitter instability (hologram) or errors in tracking device position and direction, which often occur when visual conditions are less clear or when the device is moving too fast [3].

The effectiveness of Markerless AR performance also depends on the platform used. The iOS (ARKit) and Android (ARCore) platforms exhibit different performance characteristics in detecting the ground plane. ARKit tends to have more stable and responsive tracking performance under normal lighting conditions, while ARCore shows better capabilities in low-light conditions and has a faster first-plane detection time (less than 0.5 seconds compared to ARKit, which is about 5 seconds). This difference is influenced by the implementation of SLAM algorithms, access to device sensors, and different calibration capabilities on each platform [6].

Previous research indicates that the performance of AR on each platform is not always the same. This presents a challenge: significant differences exist in ground plane detection performance between iOS and Android platforms, especially regarding the speed of ground plane detection [6]. This difference creates challenges for developers in understanding the performance characteristics of each platform for the development of markerless AR applications [6].

Several studies have investigated AR performance across platforms, though most focus on tracking accuracy rather than detection speed. Nowacki and Woda [6] demonstrated fundamental capability differences between ARCore and ARKit but did not quantify detection timing under controlled conditions. Other research has examined markerless AR stability [3] and implementation approaches [5], yet systematic comparisons of ground plane detection speed across environmental conditions remain limited. This research gap is particularly significant for developers who must make platform selection decisions without empirical performance data under realistic usage scenarios. Understanding how platform, distance, and lighting conditions interact affect detection speed is essential for optimizing user experience in markerless AR applications.

To address this challenge, this study employs an experimental comparative analysis method between the iOS and Android platforms, by measuring metrics such as *time-to-first-detection* under different light conditions and altitude distances.

This research is expected to yield scientific and practical recommendations that can help developers understand the characteristics of the detection speed of ground plane markerless AR on the iOS platform, facilitating multi-platform application development.

II. METHODOLOGY

This study uses a comparative experiment method, namely conducting direct testing of the performance of Vuforia's ground plane-based markerless Augmented Reality technology on two mobile operating system platforms, namely iOS and Android. This approach was selected to enable a fair comparison of detection performance under uniform experimental conditions, while allowing the examination of performance differences across platforms and environmental factors [7].

A. Research Object

This study uses the AR application "KreasiFurniture" which was built using the Vuforia Ground Plane and has added internal logs that will help in testing. This application is modified to be able to measure light intensity, distance, angle, and time-to-first detection time with the device specifications as shown in Table I.

B. Research Variables

In this study, there are three variables, namely independent variables, dependent variables, and control variables. Independent variables are variables that affect or cause changes in dependent variables, in contrast to dependent variables that can change in response to independent variables. Control variables are variables that are controlled or made constant so that the relationship between independent variables and dependents is not influenced by external factors that are not being studied [8].

1) Independent Variables: The Independent variable in the research to be conducted is the type of mobile operating system platform used to run applications, namely iOS and Android. The specifications of the two platforms to be used will be shown in Table I.

TABLE I
INDEPENDENT VARIABLE SPECIFICATION

Platform	Device	Specifications
iOS	iPhone 11	iOS 26, 12 MP Camera, 4GB RAM.
Android	Poco X6	Android 15, 64 MP Camera, 12 GB RAM.

Device selection was based on market representativeness and availability. The iPhone 11 represents mid-range iOS devices widely used in educational and commercial AR applications, while maintaining compatibility with current ARKit features. The Poco X6 was selected as a representative mid-range Android device with comparable price point and capable hardware specifications, enabling fair cross-platform comparison within the same market segment. Both devices support the Vuforia Ground Plane feature and provide sufficient computational resources for markerless AR applications.

2) Dependent Variables: The dependent variables in the research to be conducted are the results of the performance of each platform tested, in the form of ground plane detection speed (in seconds). Values are taken from the app's internal logs during testing. The specifications of the dependent variables will be shown in the following Table II.

TABLE II
VARIABLE DEPENDENT SPECIFICATIONS

Variable	Unit	Definition	Metric
Detection Speed	Second(s)	The time from the camera is directed to the surface until surface detection is recorded.	Measured using the app's internal logs (start & detect timestamp)

3) Control Variables: In the research to be conducted, the control variables include the type of lighting, the distance of the height of the camera to the surface detection. Control of these variables is done to ensure that the differences in performance results that arise are actually due to differences in the AR platform, not due to environmental conditions or other external factors. The specifications of the control variables are shown in the following Table III.

TABLE III
VARIABLE CONTROL SPECIFICATIONS

Variable	Criterion	Metric
Lighting Type	Indoor (7-8 pm)	Measured using a light meter/lux sensor on the device with internal app logs.
	Outdoor (12-13 pm)	
Camera Height Distance to the Floor	Short Distance: 50 cm	Measured using the app's internal logs.
	Medium Distance: 100 cm	
	Distance: 150 cm	

C. Experimental Design

The experimental design in this study employs a comparative within-subjects experiment, where each platform is tested under the same conditions, allowing for direct comparison of the measurement results [9]. Experiment was conducted by running the AR application "KreasiFurniture" on two different platforms, namely iOS and Android. Each platform was tested using two control variables, namely lighting and camera distance to the floor.

The measurement is made by recording the time it takes for the system to detect the ground plane from the moment the camera is pointed at the surface, with five repetitions under each condition to obtain a standard mean and deviation. Control variables are ensured by ensuring that all platforms use the same surface type, the same lighting conditions, the same camera and floor distances, and the same hardware specifications, so that the measurement results can be compared fairly.

The data obtained will be analyzed descriptively to see the performance pattern of each platform, then analyzed inferentially using *Analysis of Variance* (ANOVA) to determine the influence of platform, light conditions, and distance on the speed of ground-plane markerless AR detection.

D. Data Collection Techniques

The data in this study were collected through direct observation of the performance of the AR application "KreasiFurniture" which is run on two different platforms, namely iOS and Android. Observations were made by paying attention to the variation in light conditions, specifically indoor and outdoor lighting conditions, as well as the distance of the camera from the ground plane surface. Measurements include the speed of ground plane detection under various lighting conditions and camera altitudes. The detection speed is calculated from the time the camera is pointed until the system successfully recognizes the ground plane. Each combination of conditions is tested five times to obtain valid data and allow for the calculation of averages and standard deviations. All data were recorded systematically using observation sheets and then analyzed using IBM SPSS Statistics 25 for both descriptive and inferential analysis.

E. Data Analysis Methods

The collected data were analyzed in two ways. First, descriptive statistics were used to summarize and describe the main characteristics of the dataset, including mean, minimum, and maximum values, variability, and overall distribution patterns, to provide an initial overview of the detection speed of AR applications across different platforms, lighting conditions, and distances [10]. Second, inferential analysis was conducted using three-way Analysis of Variance (ANOVA), a statistical method that simultaneously evaluates the main effects of three independent variables (platform, distance, and lighting conditions) and their interaction effects on the dependent variable (detection speed) [11][12]. This approach enables identification of not only which factors significantly influence detection speed, but also whether the effect of one factor depends on the levels of other factors—information critical for understanding performance in realistic multi-variable environments. When ANOVA indicated significant effects, post hoc tests were performed to identify which groups differed significantly. All analyses used a significance level of $p < 0.05$.

III. RESULTS AND DISCUSSION

A. Test Results

Tests were carried out on *the KreasiFurniture* AR application on two mobile operating system platforms, namely iOS and Android, with indoor and outdoor lighting conditions, as well as the distance of the camera height to the surface, distances of near: 50 cm, medium: 100 cm, and far: 150 cm sizes. The test results data are obtained from the app's internal logs, which automatically record the detection time.

1) iOS

After testing the KreasiFurniture AR application on the iOS platform, the data presented in Table I was obtained.

TABLE I
TEST RESULTS DATA ON IOS PLATFORM

Distance	Light	1	2	3	4	5
Near: 50 cm	Indoor	1,486	1,352	1,251	1,385	1,419
	Outdoor	1,484	1,585	1,319	1,451	1,384
Medium: 100 cm	Indoor	1,251	1,318	1,751	1,284	1,285
	Outdoor	1,251	1,385	1,285	1,419	1,252
Far: 150 cm	Indoor	1,318	1,319	1,352	1,486	1,584
	Outdoor	1,518	1,384	1,352	1,485	1,352

2) Android

Testing of the KreasiFurniture AR application was conducted on the Android platform, as shown in Table II.

TABLE II
TEST RESULTS DATA ON ANDROID PLATFORM

Distance	Light	1	2	3	4	5
Near: 50 cm	Indoor	1,991	1,807	1,573	1,406	1,757
	Outdoor	1,673	1,506	1,557	1,206	1,573
Medium: 100 cm	Indoor	1,523	1,690	1,389	1,891	1,406
	Outdoor	1,440	1,272	0,905	1,256	1,323
Far: 150 cm	Indoor	1,540	1,389	1,807	1,673	1,506
	Outdoor	1,506	1,974	1,607	1,456	1,640

The raw data from the two platforms were then processed using the Three-Way ANOVA method using IBM SPSS Statistics 25 to determine whether the difference between the platforms was statistically significant.

B. Descriptive Statistics

Descriptive statistical analysis was performed to provide an overview of the mean, standard deviation, and number of samples for each combination of platforms, distances, and lighting conditions [10].

TABLE III
DESCRIPTIVE STATISTICS

Dependent Variable: Time					
Platform	**Distance**	**Light**	**Mean**	**Std. Deviation**	**N**
Ios	50 cm	Indoor	1.37860	.086887	5
		Outdoor	1.51020	.212868	5
		Total	1.44440	.168241	10
	100 cm	Indoor	1.37780	.209966	5
		Outdoor	1.31840	.078459	5
		Total	1.34810	.152675	10
	150 cm	Indoor	1.41180	.118474	5
		Outdoor	1.41820	.078033	5
		Total	1.41500	.094636	10
	Total	Indoor	1.38940	.137957	15
		Outdoor	1.41560	.151716	15
		Total	1.40250	.143100	30
Android	50 cm	Indoor	1.70680	.224562	5
		Outdoor	1.50300	.176730	5
		Total	1.60490	.218704	10
	100 cm	Indoor	1.57980	.211440	5
		Outdoor	1.23920	.200244	5
		Total	1.40950	.264415	10
	150 cm	Indoor	1.58300	.161005	5
		Outdoor	1.63660	.202731	5
		Total	1.60980	.174888	10
	Total	Indoor	1.62320	.195790	15
		Outdoor	1.45960	.247656	15
		Total	1.54140	.234600	30
Total	50 cm	Indoor	1.54270	.235984	10
		Outdoor	1.50660	.184486	10
		Total	1.52465	.206987	20
	100 cm	Indoor	1.47880	.225384	10
		Outdoor	1.27880	.149330	10
		Total	1.37880	.212488	20
	150 cm	Indoor	1.49740	.160938	10
		Outdoor	1.52740	.184993	10
		Total	1.51240	.169459	20
	Total	Indoor	1.50630	.204526	30
		Outdoor	1.43760	.203032	30
		Total	1.47195	.204994	60

Table III shows that the iOS platform has an overall average of 1,403 seconds (SD = 0.143) with the fastest performance at a distance of 100cm (mean = 1,348 seconds). In contrast, the Android platform shows an average of 1,541 seconds (SD = 0.235) with higher variability. Detection on Android is fastest at a distance of 100 cm (mean = 1,410 seconds) and at the slowest at a distance of 50 cm (mean = 1,605 seconds). The larger standard deviation on Android indicates that its performance is less consistent than iOS.

C. Levene's Test of Equality of Error Variances

The Levene test was conducted to test the assumption of homogeneity of variance, namely to check whether the variance of detection time between test groups is homogeneous or the same. This test is an important prerequisite before performing an ANOVA analysis [13][14].

TABLE IV
LEVENE'S TEST OF EQUALITY OF ERROR VARIANCES

		Living Statistic	**df1**	**df2**	**Sig.**
Time	Based on Mean	.893	11	48	.553
	Based on Median	.404	11	48	.947
	Based on Median and with adjusted df	.404	11	33.489	.944
	Based on trimmed mean	.801	11	48	.638

The results of the Levene test showed a value of F = 0.893 with a significance of 0.553 ($p > 0.05$), which means that the assumption of variance homogeneity was met. This indicates the data is eligible for ANOVA analysis and the results of the analysis are reliable.

D. Three-Way ANOVA: Tests of Between-Subjects Effects

Three-Way ANOVA analysis was conducted to test the influence of platform, distance, light, and the interaction between variables on the ground plane detection time. This test determines whether the observed difference is statistically significant [12][15].

TABLE V
TESS OF BETWEEN-SUBJECTS EFFECTS

Dependent Variable: Time						
Source	**Type III Sum of Squares**	**Df**	**Mean Square**	**F**	**Sig.**	**Partial Eta Squared**
Corrected Model	1.052a	11	.096	3.219	.002	.424
Intercept	129.998	1	129.998	4373.151	.000	.989
Platform	.289	1	.289	9.735	.003	.169
Distance	.262	2	.131	4.404	.018	.155
Light	.071	1	.071	2.382	.129	.047
Platform * Distance	.048	2	.024	.807	.452	.033
Platform * Light	.135	1	.135	4.544	.038	.086

Distance *Light	.140	2	.070	2.359	.105	.089
Platform * Distance * Light	.107	2	.054	1.802	.176	.070
Error	1.427	48	.030			
Total	132.478	60				
Corrected Total	2.479	59				

The analysis revealed a significant main effect of platform (F = 9.735, p = 0.003, partial $\eta^2 = 0.169$), indicating a large effect size according to Cohen's (1988) guidelines ($\eta^2 \geq 0.14$), with iOS achieving faster detection times than Android. Distance also showed a significant effect (F = 4.404, p = 0.018, partial $\eta^2 = 0.155$), similarly indicating a large effect, meaning camera distance explains 15.5% of variance in detection speed. Light did not show a significant main effect (F = 2.382, p = 0.129, partial $\eta^2 = 0.047$), representing a small effect size. However, the Platform × Light interaction was significant (F = 4.544, p = 0.038, partial $\eta^2 = 0.086$), indicating a medium effect size, where the two platforms responded differently to lighting conditions. Overall, the model explained 42.4% of the variance in detection time ($R^2 = 0.424$), suggesting that platform, distance, lighting, and their interactions substantially account for the observed performance differences.

E. Normality

To verify that the ANOVA model assumptions were satisfied, the Shapiro-Wilk test was conducted on the residuals extracted from the Three-Way ANOVA model.

TABLE VI
TEST OF NORMALITY

	Saphiro-Wilk		
	Statistic	df	Sig.
Residual for Time	.973	60	.204

The test yielded T = 0.973, df = 60, p = 0.204, indicating that the residuals were normally distributed ($p > 0.05$). These results, combined with Levene's test confirming homogeneity of variance (F = 0.893, p = 0.553), confirm the reliability of the ANOVA results presented in Table V.

F. Estimated Marginal Means

Estimated Marginal Means (EMMs), also called least-squares means, are the model-based means for each factor level, averaged over or “marginalized” across other factors in the model. EMMs provide adjusted group means from the fitted model, useful when the design is unbalanced or covariates are present and are typically reported with SE and confidence intervals. [16]

1) Platform

Estimated marginal means display the average detection time for each platform after controlling for the influence of other variables and comparing the performance of the two platforms directly.

TABLE VII
ESTIMATES PLATFORM

Depenedent Variable: Time				
			95% Confidence Interval	
Platform	**Mean**	**Std. Error**	**Lower Bound**	**Upper Bound**
Ios	1.402	.031	1.339	1.466
Android	1.541	.031	1.478	1.605

TABLE VIII
PAIRWISE COMPARISONS PLATFORM

(I) Platform	**(J) Platform**	**Mean Difference (I-J)**	**Std. Error**	**Sig.b**	**95% Confidence Interval for Difference[b]**	
					Lower Bound	**Upper Bound**
Ios	Android	-.139*	.045	.003	-.228	-.049
Android	Ios	.139*	.045	.003	.049	.228

TABLE IX
UNIVARIATE TESTS PLATFORM

Dependent Variable: Time						
	Sum of Squares	**Df**	**Mean Square**	**F**	**Sig.**	**Partial Eta Squared**
Contrast	.289	1	.289	9.735	.003	.169
Error	1.427	48	.030			

Table VI shows iOS has a mean of 1,402 seconds, while Android has 1,541 seconds. In Table VII, this difference of 0.139 seconds is statistically significant (p = 0.003), confirming that iOS is consistently faster at detecting the ground plane. Table VIII confirms this significance (F = 9.735, p = 0.003) with a partial eta squared of 0.169, indicating a moderate effect size.

2) Distance

Marginal means analysis for the distance factor aims to identify the optimal distance that results in the fastest detection time.

TABLE X
DISTANCE ESTIMATES

Depenedent Variable: Time				
			95% Confidence Interval	
Platform	**Mean**	**Std. Error**	**Lower Bound**	**Upper Bound**
50 cm	1.525	.039	1.447	1.602
100 cm	1.379	.039	1.301	1.456
150 cm	1.512	.039	1.435	1.590

TABLE XI
PAIRWISE DISTANCE COMPARISONS

(I) Distance	**(J) Distance**	**Mean Difference (I-J)**	**Std. Error**	**Sig.b**	**95% Confidence Interval for Difference[b]**	
					Lower Bound	**Upper Bound**
50 cm	100 cm	.146*	.055	.031	.011	.281
	150 cm	.012	.055	1.000	-.123	.148
100 cm	50 cm	-.146*	.055	.031	-.281	-.011
	150 cm	-.134	.055	.054	-.269	.002
150 cm	50 cm	-.012	.055	1.000	-.148	.123
	100 cm	.134	.055	.054	-.002	.269

TABLE XII
UNIVARIATE DISTANCE TESTS

Dependent Variable: Time						
	Sum of Squares	Df	Mean Square	F	Sig.	Partial Eta Squared
Contrast	.262	2	.131	4.404	.018	.155
Error	1.427	48	.030			

Table IX shows that a distance of 100 cm results in the fastest time (mean = 1.379 seconds), followed by 150 cm (1.512 seconds) and 50 cm (1.525 seconds). In Table X, the difference between 50 cm and 100 cm is significant (p = 0.031), indicating that a distance of 100 cm is the optimal distance for ground plane detection. Table XI confirms that distance has a significant influence on detection time (F = 4.404, p = 0.018) with a partial eta squared 0.155.

3) Light

This interaction analysis tests whether there is an influence of different lighting conditions on iOS and Android.

TABLE XIII
LIGHT ESTIMATES

Depenedent Variable: Time				
Platform	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
iOS	1.506	.031	1.443	1.570
Android	1.438	.031	1.374	1.501

TABLE XIV
PAIRWISE COMPARISONS OF LIGHT

(I) Light	(J) Light	Mean Difference (I-J)	Std. Error	Sig.b	95% Confidence Interval for Difference[b]	
					Lower Bound	Upper Bound
Indoor	Outdoor	.069	.045	.129	-.021	.158
Outdoor	Indoor	-.069	.045	.129	-.158	.021

TABLE XV
UNIVARIATE LIGHT TESTS

Dependent Variable: Time						
	Sum of Squares	Df	Mean Square	F	Sig.	Partial Eta Squared
Contrast	.071	1	.071	2.382	.129	.047
Error	1.427	48	.030			

Table XII and Table XIII show that indoor conditions have a mean of 1.506 seconds and outdoor 1.438 seconds. Although outdoor is slightly faster, this difference is not significant (p = 0.129). However, significant interaction between Platform × Light shows the influence of light differs between iOS and Android.

G. Interaction Effects

An interaction effect occurs when the effect of one factor on the outcome depends on the level of another factor. In a three-way ANOVA, you must examine two-way and three-way interactions; a significant interaction means main effects alone do not fully describe the pattern, and interpretation should focus on the interaction, often visualized via profile/interaction plots, and be followed by simple-effects or post-hoc comparisons. [17]

*1) Platform*Distance*

This interaction analysis evaluates whether the effiect of distance on detection time differs between the two platforms.

TABLE XVI
PLATFORM * DISTANCE

Dependent Variable: Time					
Platform	Distance	Mean	Std. Error	95% Confidence Interval	
				Lower Bound	Upper Bound
iOS	50 cm	1.444	.055	1.335	1.554
	100 cm	1.348	.055	1.238	1.458
	150 cm	1.415	.055	1.305	1.525
Android	50 cm	1.605	.055	1.495	1.715
	100 cm	1.410	.055	1.300	1.519
	150 cm	1.610	.055	1.500	1.719

Table XV shows that on iOS, the detection time is relatively stable at all distances. In contrast, Android shows greater variability, with optimal performance at a distance of 100 cm. Although in Table V the interaction was not statistically significant (p = 0.452), this pattern shows that iOS is more consistent across different distances.

*2) Platform*Light*

This interaction analysis tested whether the effect of lighting conditions differed between iOS and Android, which proved to be significant (p = 0.038) in Table V.

TABLE XVII
PLATFORM * LIGHT

Dependent Variable: Time					
Platform	Light	Mean	Std. Error	95% Confidence Interval	
				Lower Bound	Upper Bound
iOS	Indoor	1.389	.045	1.300	1.479
	Outdoor	1.416	.045	1.326	1.505
Android	Indoor	1.623	.045	1.534	1.713
	Outdoor	1.460	.045	1.370	1.549

iOS shows a small difference between indoor (1,389 seconds) and outdoor (1,416 seconds), just 0.027 seconds. In contrast, Android shows a big difference: indoor (1,623 seconds) is much slower than outdoor (1,460 seconds), with a difference of 0.163 seconds. This indicates that Android is more sensitive to changes in lighting conditions.

*3) Distance*Light*

This analysis evaluates the effect of different distances on different lighting conditions.

TABLE XVIII
DISTANCE * LIGHT

Dependent Variable: Time					
Distance	Light	Mean	Std. Error	95% Confidence Interval	
				Lower Bound	Upper Bound
50 cm	Indoor	1.543	.055	1.433	1.652
	Outdoor	1.507	.055	1.397	1.616
100 cm	Indoor	1.479	.055	1.369	1.588
	Outdoor	1.279	.055	1.169	1.388
150 cm	Indoor	1.497	.055	1.388	1.607
	Outdoor	1.527	.055	1.418	1.637

In table V, the interaction was not significant (p = 0.105), but the pattern showed that at a distance of 100 cm, the difference between indoor (1.479 seconds) and outdoor (1.279 seconds) was greatest (0.200 seconds), indicating that the medium distance was most sensitive to lighting conditions.

*4) Platform*Distance*Light*

This three-way interaction analysis provides a comprehensive picture of how the three factors interact simultaneously.

TABLE XIX
PLATFORM * DISTANCE * LIGHT

Dependent Variable: Time						
Platform	Distance	Light	Mean	Std. Error	95% Confidence Interval	
					Lower Bound	Upper Bound
iOS	50 cm	Indoor	1.379	.077	1.224	1.534
		Outdoor	1.510	.077	1.355	1.665
	100 cm	Indoor	1.378	.077	1.223	1.533
		Outdoor	1.318	.077	1.163	1.473
	150 cm	Indoor	1.412	.077	1.257	1.567
		Outdoor	1.418	.077	1.263	1.573
Android	50 cm	Indoor	1.707	.077	1.552	1.862
		Outdoor	1.503	.077	1.348	1.658
	100 cm	Indoor	1.580	.077	1.425	1.735
		Outdoor	1.239	.077	1.084	1.394
	150 cm	Indoor	1.583	.077	1.428	1.738
		Outdoor	1.637	.077	1.482	1.792

Although not significant in Table V (p = 0.176), the pattern shows that Android is much more sensitive to lighting conditions than iOS, especially at close and medium distances. iOS shows stable performance in a wide range of combinations of conditions.

H. Post Hoc Test

Post hoc tests are multiple-comparison procedures run after a significant ANOVA to identify which specific group means differ. In this study, we used the Tukey HSD and Bonferroni methods. [18]

TABLE XX
MULTIPLE COMPARISONS I

Dependent Variable: Time					
	(I) Distance	(J) Distance	Mean Difference (I-J)	Std. Error	Sig.
Tukey HSD	50 cm	100 cm	.14585*	.054522	.027
		150 cm	.01225	.054522	.973
	100 cm	50 cm	-.14585*	.054522	.027
		150 cm	-.13360*	.054522	.046
	150 cm	50 cm	-.01225	.054522	.973
		100 cm	.13360*	.054522	.046
Bonferroni	50 cm	100 cm	.14585*	.054522	.031
		150 cm	.01225	.054522	1.000
	100 cm	50 cm	-.14585*	.054522	.031
		150 cm	-.13360	.054522	.054
	150 cm	50 cm	-.01225	.054522	1.000
		100 cm	.13360	.054522	.054

TABLE XXI
MULTIPLE COMPARISONS II

Dependent Variable: Time				
	(I) Distance	(J) Distance	95% Confidence Interval	
			Lower Bound	Upper Bound
Tukey HSD	50 cm	100 cm	.01399	.27771
		150 cm	-.11961	.14411
	100 cm	50 cm	-.27771	-.01399
		150 cm	-.26546	-.00174
	150 cm	50 cm	-.14411	.11961
		100 cm	.00174	.26546
Bonferroni	50 cm	100 cm	.01059	.28111
		150 cm	-.12301	.14751
	100 cm	50 cm	-.28111	-.01059
		150 cm	-.26886	.00166
	150 cm	50 cm	-.14751	.12301
		100 cm	-.00166	.26886

In Table XIX and Table XX the results show. At a distance of 50 cm vs 100 cm, it was significant (p = 0.031) with a distance of 100 cm faster 0.146 seconds. At 100 cm vs 150 cm, it was close to significance (p = 0.054). While 50 cm vs 150 cm, it does not look significant (p = 1,000). This confirms the distance of 100 cm is the optimal distance for both platforms.

I. Homogeneous Subsets

Homogeneous subsets are groups formed by some post-hoc procedures, where group means that do not differ significantly are placed in the same subset. The output helps summarize which levels form non-significant clusters at the chosen α[18].

TABLE XXII
HOMOGENEOUS SUBSETS

Time				
	Distance	N	Subset	
			1	2
S.S. S.S.	100 cm	20	1.37880	
	150 cm	20		1.51240
	50 cm	20		1.52465
	Sig.		1.000	.973

The results showed two subsets, namely the distance of 100 cm forming a separate subset (the fastest), while the distance of 50 cm and 150 cm were in one subset (no significant difference).

J. Profile Plots

Profile plots graphically display estimated means for factor levels across another factor; they are the standard visual tool to inspect interactions [16].

*1) Distance*Platform*Light*

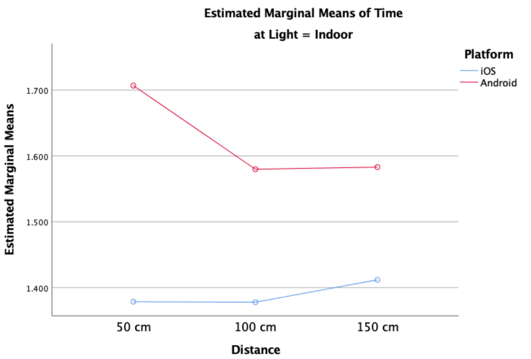

Figure 1. EMM of Time at Indoor

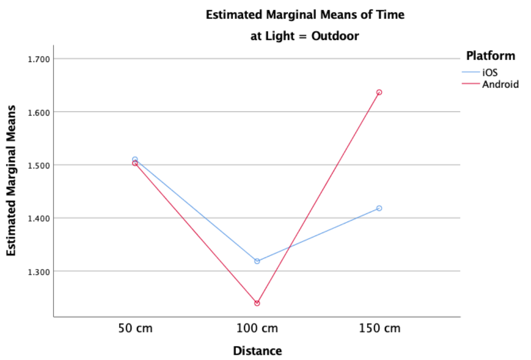

Figure 2. EMM of Time at Outdoor

The graphics at Figures 1 and 2 show iOS having a relatively flat (consistent) line across all distances, while Android shows large fluctuations. The difference in platforms is more noticeable in indoor conditions, with the two platforms converging in outdoor conditions, especially at a distance of 100 cm.

*2) Distance*Light*Platform*

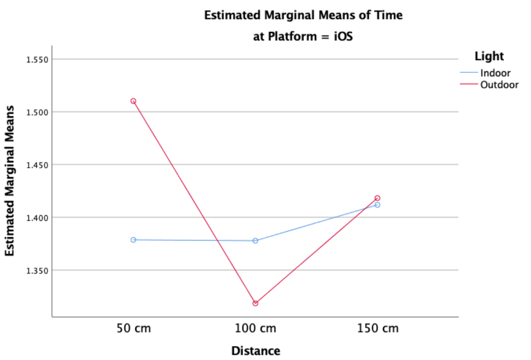

Figure 3. EMM of Time at iOS

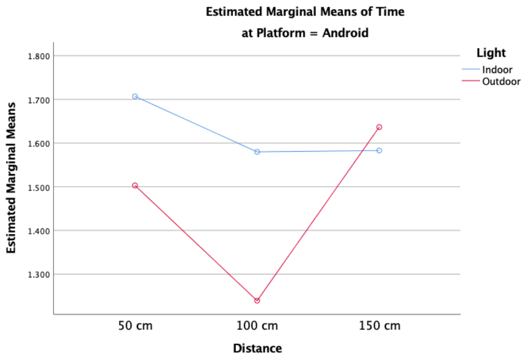

Figure 4. EMM of Time at Android

In Figures 3 and 4, the graphs show Android's much higher sensitivity to lighting changes than iOS's, with the largest indoor-outdoor gap visible at 100 cm on Android.

*3) Platform * Light * Distance*

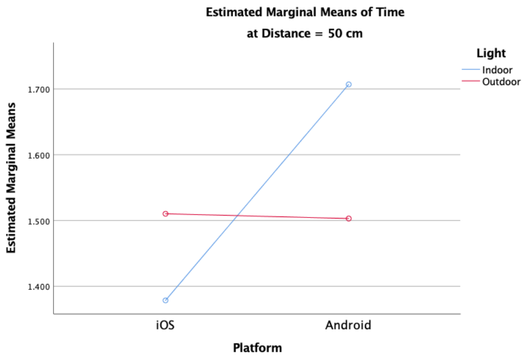

Figure 5. EMM of Time at 50 cm

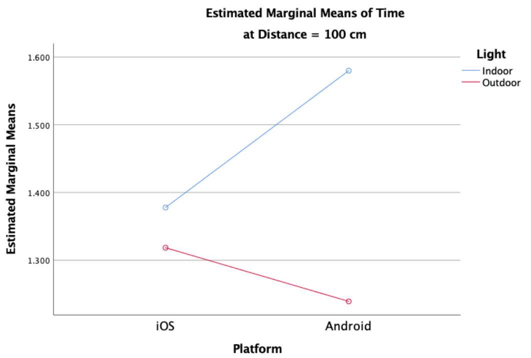

Figure 6. EMM of Time at 100 cm

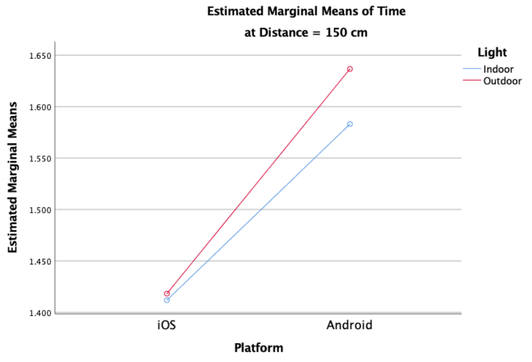

Figure 7. EMM of Time at 150 cm

Figure 5, Figure 6, and Figure 7 show that at all distances, iOS has more consistent performance between indoor and outdoor. The gap between iOS and Android is largest in indoor conditions at a distance of 50 cm.

4) Platform

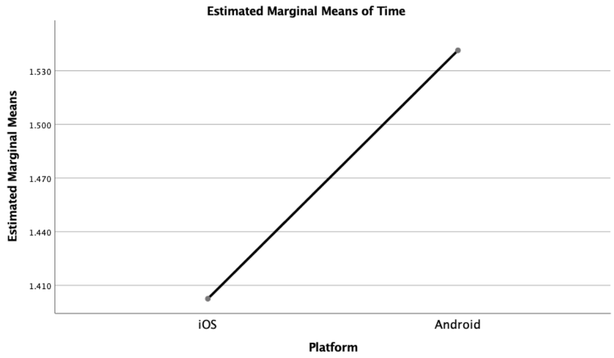

Figure 8. EMM or Platform

Figure 8 shows the overall mean difference between iOS (1,402 seconds) and Android (1,541 seconds), confirming the consistent superiority of the iOS platform.

5) Distance

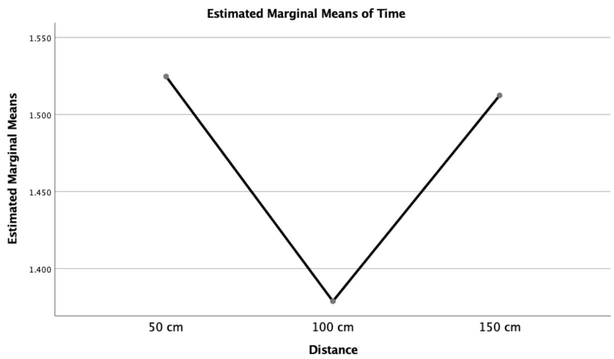

Figure 9. EMM of Distance

In the graph in Figure 9, visualized that a distance of 100 cm results in the fastest detection time (1.379 seconds), indicating the optimal distance for ground plane detection at a medium distance.

6) Light

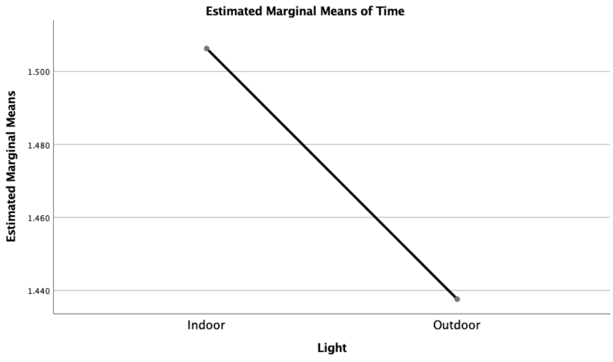

Figure 10. EMM of Light

Judging from Figure 10, the graph shows a small difference between indoor (1,506 seconds) and outdoor (1,438 seconds), which is not statistically significant. However, this graph does not capture significant Platform × Light interactions, demonstrating the importance of analyzing the effects of interactions.

IV. Discussion

A. Technical Interpretation

Although this study used Vuforia Ground Plane, the observed performance differences reflect the underlying native AR platforms. Vuforia relies on ARKit for iOS and ARCore for Android to perform actual plane detection, acting as a cross-platform wrapper.

The superior iOS performance (1.402 s vs. 1.541 s) stems from ARKit's tight hardware–software integration. Apple co-designs cameras and SLAM algorithms for a limited device range, enabling aggressive optimization. In contrast, ARCore must accommodate hundreds of Android devices from various manufacturers, requiring more conservative algorithms that prioritize compatibility over speed, which explains both the slower detection time and higher variability observed on Android (SD = 0.235 vs. 0.143).

Importantly, the Platform × Light interaction ($p = 0.038$) highlights that detection performance is not determined by a single factor in isolation. Android performance degraded significantly under indoor lighting conditions (1.623 s) compared to outdoor conditions (1.460 s), whereas iOS remained relatively stable (1.389 s vs. 1.416 s). This interaction suggests that ARKit may employ more sophisticated image preprocessing algorithms optimized for variable lighting conditions, potentially including advanced noise reduction and contrast enhancement specifically tuned for Apple's camera hardware. In contrast, ARCore's cross-manufacturer compatibility requirements necessitate more conservative algorithms that perform reliably across diverse sensor configurations but may sacrifice performance in challenging lighting scenarios. However, the specific algorithmic implementations are proprietary and warrant further investigation through controlled hardware-level experiments.

Similarly, the optimal detection distance of 100 cm reflects shared SLAM constraints across both platforms, representing a balance between field-of-view coverage and feature resolution required for reliable plane detection. Overall, these findings demonstrate the importance of interaction analysis, as the influence of one factor on detection performance depends on the level of other factors and cannot be fully explained by main effects alone.

B. Generalizability and Limitations

This study provides empirical findings on the comparison of *ground plane* detection speed in markerless Augmented Reality across iOS and Android platforms. However, the generalizability of the results remains limited. The experiments were conducted using a single markerless AR application (KreasiFurniture) and a single AR framework (Vuforia); therefore, the findings primarily reflect the characteristics of this specific implementation and may differ when using other frameworks such as native ARKit or ARCore.

Furthermore, the evaluation was performed on a limited number of devices, with one representative iOS device and one Android device. Variations in hardware specifications, including camera quality, sensor configuration, and processing capability, may influence ground plane detection performance across platforms. The number of repetitions per experimental condition (five trials) was exploratory and may be insufficient for broader statistical generalization.

The experiments were also conducted under controlled environmental conditions, including predefined lighting settings and camera distances. More dynamic real-world usage scenarios were not fully captured in this study. Future research is encouraged to incorporate multiple applications, AR frameworks, devices, and more diverse testing conditions to enhance the generalizability of the findings.

C. Practical Recommendations

Based on the observed performance characteristics, developers of cross-platform markerless AR applications should account for platform-specific behavior despite the use of abstraction frameworks such as Vuforia. iOS demonstrates more consistent ground plane detection, particularly under indoor lighting conditions, suggesting that it is better suited for applications where environmental control is limited. In contrast, Android implementations require greater attention to lighting variability, especially in indoor scenarios where detection latency increases.

The identification of an optimal detection distance of approximately 100 cm provides a practical reference for spatial interaction design. Applications should encourage user interaction within this range to ensure stable plane detection, particularly for object placement tasks such as furniture visualization. Deviations from this distance may reduce detection reliability due to limitations in feature distribution and field-of-view coverage inherent to SLAM-based approaches.

For Android-focused deployments, adaptive design strategies are recommended, including user guidance for improving lighting conditions and extended detection time thresholds to accommodate higher latency. Additionally, testing and quality assurance procedures should incorporate platform-specific benchmarks rather than uniform acceptance criteria, reflecting the inherent differences between ARKit and ARCore observed in this study.

Sample size considerations should also be addressed in future implementations. While this study employed five repetitions per condition, adequate for exploratory factorial analysis, production applications should conduct more extensive testing (minimum 10-15 repetitions per condition) to establish reliable performance baselines. Additionally, developers should implement runtime performance monitoring to detect and adapt to device-specific variations, particularly on Android, where hardware diversity may produce performance characteristics not captured in limited device testing.

V. CONCLUSION

This study provides empirical evidence that ground plane detection performance in markerless AR differs significantly across mobile platforms, even when using cross-platform frameworks. Through systematic three-way ANOVA analysis, we demonstrated that iOS achieves faster and more consistent detection than Android (mean difference = 0.139 seconds, $p = 0.003$, $\eta^2 = 0.169$), while an interaction distance of 100 cm yields optimal performance on both platforms ($p = 0.018$, $\eta^2 = 0.155$). Critically, the significant Platform × Light interaction ($p = 0.038$, $\eta^2 = 0.086$) reveals that lighting conditions affect platforms differently: iOS maintains stable performance across lighting variations, whereas Android experiences substantial degradation in indoor conditions.

These findings have important implications for AR application development. First, platform selection should consider deployment environment—iOS is preferable for indoor applications requiring consistent performance, while Android requires adaptive design strategies for variable lighting. Second, interaction design should target the 100 cm optimal distance to maximize detection reliability. Third, cross-platform frameworks like Vuforia do not fully abstract platform-level differences; developers must account for native AR capabilities (ARKit vs ARCore) when optimizing user experience.

Future research should extend these findings by incorporating additional platforms, multiple AR frameworks, diverse device models, and larger sample sizes to enhance generalizability. Investigation of runtime adaptive algorithms that dynamically adjust detection parameters based on environmental conditions would provide practical solutions for the performance variations identified in this study.

BIBLIOGRAPHY

[1] H. Kolivand, A. El Rhalibi, M. Tajdini, S. Abdulazeez, and P. Praiwattana, "Cultural Heritage in Marker-Less Augmented Reality: A Survey," *Adv. Methods New Mater. Cult. Herit. Preserv.*, 2019, doi: 10.5772/intechopen.80975.

[2] K. S. Kartini, N. Luh, P. Labasariyani, M. Irvan, S. Abenk, and I. N. Tri, "Analisis Perbandingan Efektivitas Augmented Reality Marker-Based dan Markerless pada Media Pembelajaran Struktur Tumbuhan," vol. 5, no. 1, pp. 301–309, 2025.

[3] T. Scargill, J. Chen, and M. Gorlatova, *Here To Stay: Measuring Hologram Stability in Markerless Smartphone Augmented Reality*, vol. 1, no. 1. Association for Computing Machinery, 2021. [Online]. Available: http://arxiv.org/abs/2109.14757

[4] C. Campos, R. Elvira, J. J. G. Rodriguez, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial, and Multimap SLAM," *IEEE Trans. Robot.*, vol. 37, no. 6, pp. 1874–1890, 2021, doi: 10.1109/TRO.2021.3075644.

[5] M. Abdinejad, C. Ferrag, H. S. Qorbani, and S. Dalili, "Developing a Simple and Cost-Effective Markerless Augmented Reality Tool for Chemistry Education," *J. Chem. Educ.*, vol. 98, no. 5, pp. 1783–1788, 2021, doi: 10.1021/acs.jchemed.1c00173.

[6] P. Nowacki and M. Woda, "Capabilities of ARCore and ARKit Platforms for AR / VR Applications," pp. 358–370, 2020, doi: 10.1007/978-3-030-19501-4.

[7] R. S. Munte, R. Risnita, M. S. Jailani, and I. Siregar, "Jenis penelitian eksperimen dan non eksperimen (design klausal komparatif dan design korelasional)," *J. Pendidik. Tambusai*, vol. 7, no. 3, pp. 27602–27605, 2023.

[8] J. Hikmah, "Paradigm," *Comput. Graph. Forum*, vol. 39, no. 1, pp. 672–673, 2020, doi: 10.1111/cgf.13898.

[9] G. Charness, U. Gneezy, and M. A. Kuhn, "Experimental methods: Between-subject and within-subject design," *J. Econ. Behav. Organ.*, vol. 81, no. 1, pp. 1–8, 2012, doi: 10.1016/j.jebo.2011.08.009.

[10] Y. Dong, "Descriptive Statistics and Its Applications," vol. 47, pp. 16–23, 2023.

[11] D. Lakens and A. R. Caldwell, "Simulation-Based Power Analysis for Factorial Analysis of Variance Designs," 2021, doi: 10.1177/2515245920951503.

[12] N. Celik, "Three-way Analysis of Variance for Functional Data," vol. 11, no. 1, pp. 1–7, 2023, doi: 10.13189/ujam.2023.110101.

[13] Y. Zhou, Y. Zhu, and W. K. Wong, "Statistical tests for homogeneity of variance for clinical trials and recommendations," vol. 33, no. September 2022, 2023.

[14] K. Sritan and B. Phuenaree, "A Comparison of Efficiency for Homogeneity of Variance Tests under Log-normal Distribution," vol. 9, no. 4, pp. 254–259, 2021.

[15] M. Zhou, "Journal of Innovation," vol. 7, 2022.

[16] D. Makowski, M. S. Ben-shachar, M. Brenton, I. Patil, R. Thériault, and D. Lüdecke, "modelbased : An R package to make the most out of your statistical models through marginal means , marginal effects , and model predictions," *J. Open Source Softw.*, vol. 10, pp. 1–8, 2025, doi: 10.21105/joss.07969.

[17] S. G. Id, S. Giovagnoli, M. Orsoni, and F. S. Id, "Interaction effect : Are you doing the right thing ?," pp. 1–19, 2022, doi: 10.1371/journal.pone.0271668.

[18] G. A. Morgan, N. L. Leech, G. W. Gloeckner, and K. C. Bannett, *SPSS for Introductory Statistics*.