Asia-Pacific Journal of Information Technology and Multimedia 12(2) 2023: 286 - 297
https://doi.org/10.17576/apjitm-2023-1202-07

Performance Evaluation of Marker Recognition Algorithm for Mobile Augmented Reality in The Real Environment

Penilaian Prestasi Algoritma Pengesanan Penanda Untuk Augmentasi Realiti Mudah Alih Dalam Persekitaran Nyata

Siok Yee Tan & Haslina Arshad*

Faculty of Technology & Information Science, Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor

**Corresponding author: esther@ukm.edu.my*

Received 31 August 2023
Accepted 6 November 2023, Available online 1 December 2023

ABSTRACT

The ultimate goals of Augmented Reality (AR) applications are to provide a better management and ubiquitous access to information using seamless techniques in which the interactive real world is combined with an interactive computer-generated world, creating one coherent environment. The performance of the AR algorithm in terms of brightness, rotation, and scale in the real environment had been addressed as an issue due to the controlled environment. Hence, a performance evaluation of the AR algorithm in a real controlled environment is proposed in this paper. BRISK detector, FREAK descriptor, and Hamming distance matcher algorithms are implemented in a mobile AR application in order to evaluate the AR algorithm. The mobile AR application is run on the Samsung Note 9 smartphone. The image used in the evaluation is the Graffiti image from the Mikolajczyk data set. Graffiti image is fully printed in an A4 size paper and is attached to a wall with a height of 1.5m. This performance evaluation is able to evaluate the robustness of the AR algorithm in terms of brightness value from 0 Watts per square meter up to 70 Watts per square meter. The robustness of the AR algorithm in terms of scale invariance was evaluated from the distance of 5cm up to 50cm from the input image. The AR algorithm can obtain an accuracy of 83.49%, 70.89%, and 72.78% in terms of brightness changes, scale changes, and rotation changes respectively. This work introduces a more suitable performance evaluation for an AR application in a real controlled environment.

Keywords**:** Accuracy evaluation, mobile augmented reality, illumination, scale and rotation

ABSTRAK

Matlamat utama aplikasi Augmentasi Realiti (AR) adalah untuk menyediakan pengurusan yang lebih baik dan akses maklumat yang meluas melalui teknik tanpa batas di mana dunia nyata interaktif digabungkan dengan dunia maya interaktif, mencipta satu persekitaran yang koheren. Prestasi algoritma AR dari segi kecerahan, putaran, dan skala dalam persekitaran nyata telah ditangani sebagai isu kerana persekitaran yang terkawal. Oleh itu, penilaian prestasi algoritma AR dalam persekitaran terkawal nyata dicadangkan dalam manuskrip ini. Algoritma pengesan BRISK, pemerihalan FREAK, dan algoritma perpadanan jarak Hamming dilaksanakan dalam aplikasi AR bergerak untuk menilai algoritma AR. Aplikasi AR mudah alih dijalankan pada telefon pintar Samsung Note 9. Imej yang digunakan dalam penilaian adalah imej Graffiti dari set data Mikolajczyk. Imej Graffiti dicetak sepenuhnya dalam kertas saiz A4 dan dilekatkan pada dinding dengan ketinggian 1.5m. Penilaian prestasi ini mampu menilai keteguhan algoritma AR dari segi nilai kecerahan dari 0 Watt per meter persegi hingga 70 Watt per meter persegi. Keteguhan algoritma AR dari segi tidak keseragaman skala dinilai dari jarak 5cm hingga 50cm dari imej input. Algoritma AR dapat mencapai ketepatan 83.49%, 70.89%, dan 72.78% dari segi perubahan kecerahan, perubahan skala, dan perubahan putaran masing-masing. Manuskrip ini memperkenalkan penilaian prestasi yang lebih sesuai untuk aplikasi AR dalam persekitaran nyata terkawal.

Kata Kunci: penilaian keteguhan, augmentasi realiti mudah alih, pencahayaan, skala dan putaran

INTRODUCTION

Augmented Reality (AR) is the combination of the real world and the virtual content where it allows users to interact with virtual objects in real-time (Azuma 1997). Smartphones have been identified as the most promising devices for AR (Arshad et al. 2016). In order to achieve optimal performance in mobile AR applications, choosing a suitable AR algorithm is an important issue as the smartphone has its limited processing capabilities and memories compared to the PC platform (Shang et al. 2016). Tracking in AR consists of three important components; detector, descriptor, and matcher. The performance of the AR algorithm in terms of brightness, rotation, and scale in the real environment had been addressed as an issue due to the controlled environment (Liu et al. 2010 & Cheng et al. 2020). Evaluation performance of the mobile AR algorithm in a real environment is an issue that has not been fully resolved yet. The evaluation of the AR algorithm had been proposed by previous researchers using a standard dataset to evaluate the robustness of the algorithm in terms of brightness value, rotation, and scale (Alahi et al. 2012; Van et al. 2010; Khan et al. 2018; Tan et al. 2019; Alshazly et al. 2017). Testing the algorithm using a standard dataset is only able to test the performance of the algorithm without being influenced by a controlled environment. Evaluation of algorithms in a real controlled environment to test the performance of an algorithm is also important. Therefore, a testing algorithm in a real environment should be done to test the performance of an algorithm when it is being influenced by a controlled environment such as a very bright environment or very far recognition distance. This paper will discuss how the performance evaluation of the AR algorithm in the real environment is carried out to handle the controlled environment problem. Among the tests carried out in the real environment is the robustness of the AR algorithm in terms of a brightness value, rotation, and scale changes.

The rest of the paper is structured as follows. Section 2 discussed the algorithm used in AR. Section 3 introduced the configuration proposed to evaluate the AR algorithm. Section 4 then discussed the results of each test. Finally, Section 5 presented the conclusion.

AUGMENTED REALITY ALGORITHM

The image recognition process in the AR application can be performed using three main algorithms; detector, descriptor, and matcher. The detection process is the first step in the image recognition process to detect the important feature in both input image (image to be detected) and reference image (image stored in the database). Many detector algorithms have been introduced; SIFT (Lowe 2004), FAST (Rosten et al. 2006) and BRISK (Leutenegger et al. 2011). SIFT detector is a floating-point detector which is not suitable for a mobile AR application due to the low efficiency of the SIFT detector. BRISK is a binary detector and it is the enhancement of the FAST detector. Once the important features are detected by the detector, the descriptor algorithm is used to extract and describe the important features (keypoints). Previous researchers have also proposed many descriptor algorithms such as SIFT (Lowe, 2004), SURF (Bay et al. 2008), BRIEF (Calonder et al. 2012), ORB (Rublee et al., 2011), BRISK (Leutenegger et al. 2011) and FREAK (Alahi et al. 2012). SIFT and SURF are floating-point descriptions and BRIEF, ORB, BRISK and FREAK are binary subtitles in the image processing field. FREAK descriptor has been recognized as the most distinctive descriptor in a mobile AR application compared to other descriptors (Tan et al. 2019). Matcher algorithms need to be used to match all the keypoints extracted from the input image and reference image. There are two types of distance; Euclidean distance and Hamming distance which are commonly used in image recognition (Bay et al. 2008). Euclidean distance matched the keypoints between two images by comparing the distance between keypoints using the metric (Danielsson 1980). Both keypoints are considered as the same keypoints if the distance between keypoints is small (certain threshold). Euclidean distance is usually used by floating-point descriptors like SIFT and SURF. Hamming distance is used by binary descriptors like BRIEF, ORB, and FREAK. Hamming distance is defined as the number of different bits between strings. Hamming distance can be calculated quickly because it only needs to perform XOR bitwise operation on both strings. Logical operation like XOR is fast and simple, it is directly supported by the computer processor architecture. An example of Hamming distance is given for a better understanding; given two 8-bits strings “11011011” and “11011100”, the Hamming distance between these two strings is three. This matching algorithm for binary descriptor can perform in a very short computation time compared to Euclidean distance. The last step in the image recognition process is position estimation. This is to determine the position of the input image and the virtual object in order to accurately superimpose the virtual object on top of the input image. After all these processes are executed, a 3D object will then be augmented over the input image with the correct position (Uchiyama et al. 2012). Figure 1 shows the image recognition process in the AR application.

AR researchers in recent years have been working hard to develop an efficient and robust mobile AR application (Tan et al. 2018, Nafea et al. 2022). Effective and robustness are the standard performance measures for an AR application. Efficiency is defined as the ability to match the key points between input and reference image in the shortest time possible. Robustness is generally be defined as accurately matching the key points between input and reference image in the presence of brightness, rotation, and scale changes (Tan et al. 2016). The term “accuracy” can also be used to describe the robustness of the image recognition algorithm. Hence, all the AR algorithms should be evaluated carefully before applying them to an AR application in order to create an efficient and robust mobile AR application. Testing algorithm using a standard dataset is important however the performance of the algorithm couldn’t be tested under a controlled environment such as a very bright environment or a very far recognition system (Zeng et al. 2014). Hence, the performance of the AR algorithm should also be tested in the real environment in order to accurately evaluate the performance of the AR algorithm under a controlled environment.

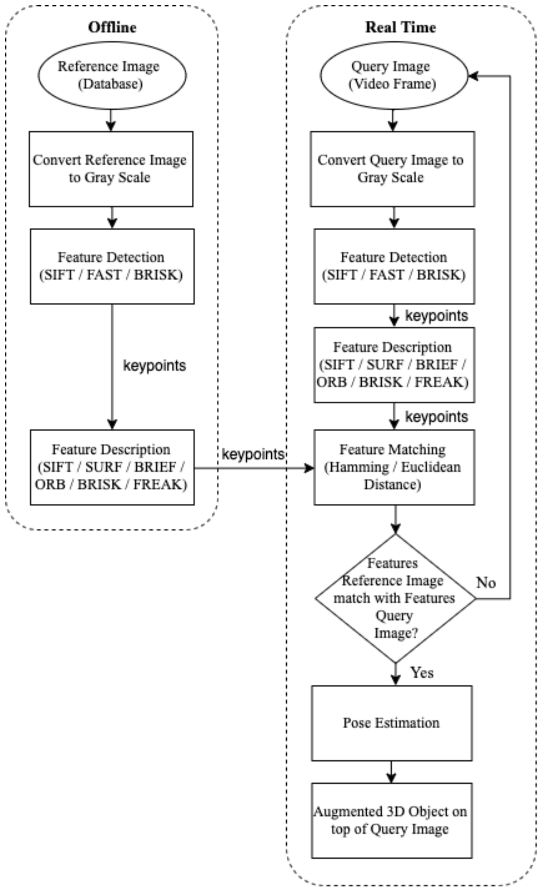

FIGURE 1. Image recognition process in the AR application

EVALUATION CONFIGURATION IN REAL ENVIRONMENT

Testing the AR algorithm in the real environmental should be able to test the performance of the algorithm in a controlled environment. The testing incorporates the BRISK detector as the detection algorithm, the FREAK descriptor as the descriptor algorithm, and the Hamming distance serving as the matching algorithm, in alignment with the methodology outlined in the prior research by Tan et al. in 2019. The performance evaluation carried out in this test are brightness, rotation, and scale invariance test which will determine the accuracy or robustness of an image recognition algorithm. All the algorithms involved in this testing are from OpenCV 2.4.9 and are implemented in a mobile AR application. The application is implemented in an Android phone (Samsung Note 9) which fulfils the minimum requirement to run a mobile AR application. This smartphone is powered by a Quad-core 1.7 GHz CPU processor and has one built-in camera that can take video with 1080P at 28fps or 720P at 30f. This smartphone operates with Android OS 4.2 and Nvidia Tegra 3 chipset. The reference image used in this test is the Graffiti image (210mm x 297mm) from the Mikolajczyk data set (Mikolajczyk et al., 2003). Graffiti images are printed in an A4 size paper and attached to the wall at a height of 1.5m. The room to carry out the testing is 3.0m width and 3.5m length with two Philips fluorescent lamps (each lamp has a total energy of 36 Watts) as shown in Figure 2.

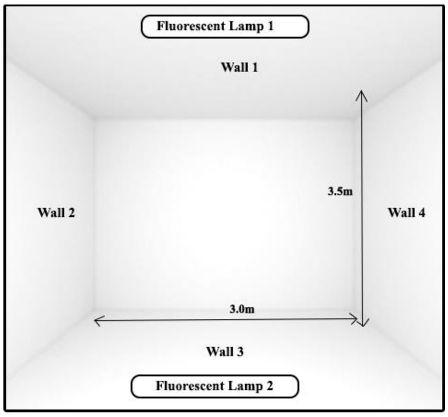

FIGURE 2. Room conditions to carry out testing in real environment

BRIGHTNESS VALUE

Evaluation of the robustness of brightness value in AR application in a real environment had been done by other researchers without disclosing the environment conditions involved in the study (Liu et al. 2009, Bleser et al. 2009). Therefore, this paper will discuss in detail the test configurations used to obtain different brightness values in the real environment.

Evaluation of brightness value is done in a room condition as shown in Figure 2. Fluorescent lamp 2 is not turned on while fluorescent lamp 1 is on. According to inverse square law, the value of the brightness is inversely proportional to the square distance of the lighting source as shown in Figure 3 (Richard 2006). Based on the inverse square law, the image which is closer to the lighting source will obtain a higher brightness value compared to the image which is far from the lighting source. Therefore, the evaluation of brightness value in mobile AR applications will be carried out based on the reverse square law. Brightness value in the real environment is calculated based on the inverse square law equation:

$$\text{Brightness Value} = \frac{F}{4\pi D^2} \quad (1)$$

Where F is the amount of energy emitted by the fluorescent lamp in the Watt unit. D is the distance between the fluorescent lamp and the input image. Therefore, the unit for brightness value is watt/m^2. The fluorescent lamp used in this study has a total energy of 36 Watts. The distance between the fluorescent lamp and the input image is calculated for each test. The percentage of accuracy obtained in each brightness value is calculated by using Equation 2.

$$\text{Percentage of Accuracy} = \frac{\text{No.of correct matches (n)}}{\text{No.of matches (N)}} * 100\ \% \quad (2)$$

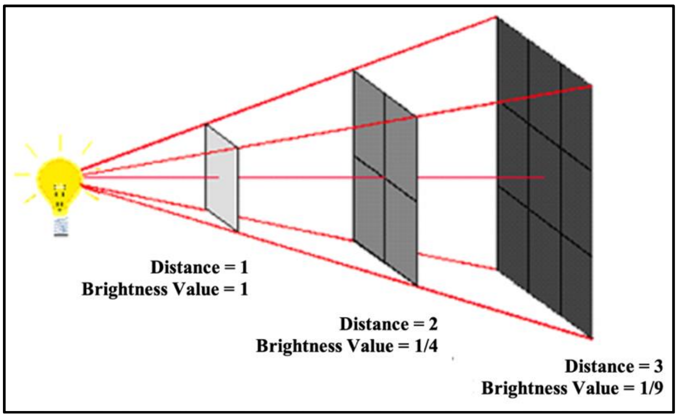

FIGURE 3. Inverse square law of brightness

At the beginning of the test, the smartphone's camera is placed at the farthest distance from the lighting source. The height of the smartphone's camera is set to be parallel to the input image. The distance between the input image and the smartphone's camera is set at 20cm. Then the distance of the input image and the smartphone's camera is gradually moved towards the lighting source and the percentage of accuracy is recorded. The brightness value can be calculated in each condition by determining the distance between the lighting source and the input image. This test is repeated 50 times at each condition by using a mobile AR application that uses a BRISK detector, FREAK descriptor, and Hamming distance matcher. Figure 4 shows the configuration test performed to evaluate the AR algorithm in the real environment with the change in brightness value.

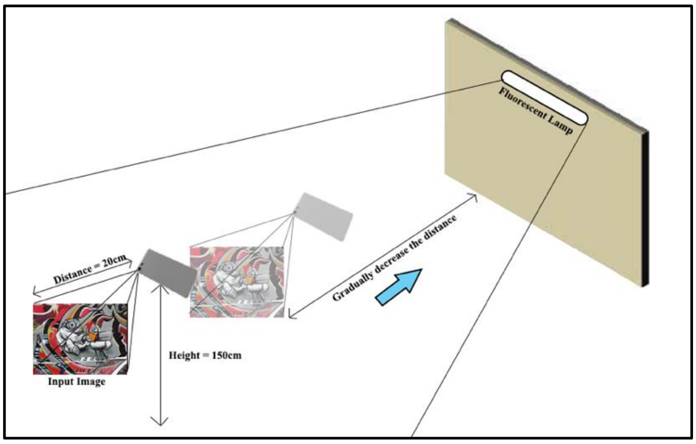

FIGURE 4. Configuration to evaluate the ar algorithm with a change in brightness value

SCALE

Configuration test performed to evaluate the performance of the AR algorithm towards scale invariance in the real environment is based on previous research where they test the accuracy of the descriptor in PC-based AR (Greg 2011). The input image is in a small-scale when the distance between the smartphone's camera and the input image is far. The input image will gradually increase in size when the smartphone's camera is moved towards the input image. The Graffiti input image is attached to the Wall 4 shown in Figure 2. Both fluorescent lamps are turned on throughout the test.

The smartphone's camera is placed closest to the input image (5cm from the input image) at the beginning of the test. Then the distance of the smartphone's camera and the input image is gradually increased by moving away from the smartphone's camera from the input image. The height of the smartphone's camera is set to be parallel to the input image. The percentage of accuracy obtain at each distance is recorded and the accuracy is calculated based on Equation 2. This test is repeated 50 times at each distance by using a mobile AR application that uses a BRISK detector, FREAK descriptor, and Hamming distance matcher. Figure 5 shows the configuration test performed to evaluate the AR algorithm in the real environment with a change in scale. value.

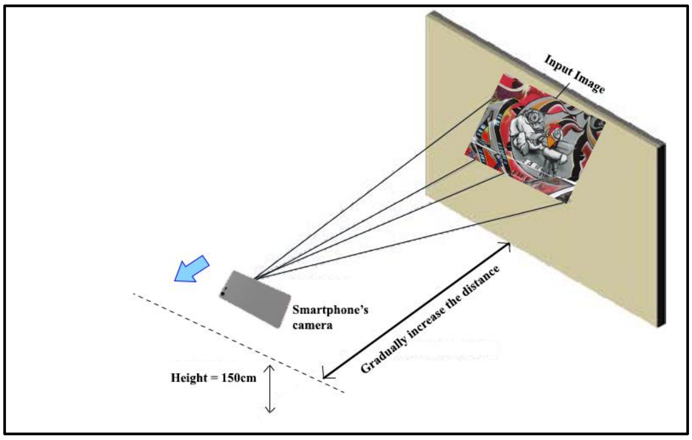

FIGURE 5. Configuration to Evaluate the AR Algorithm with a Change in Scale

Rotation

Evaluation test carried out to evaluate the performance of the AR algorithm towards rotation invariance in the real environment is referred to previous research that tests the accuracy of detector and descriptor in PC-based AR. The Graffiti input image is attached to Wall 4 (Figure 2). Both fluorescent lamps are turned on throughout the testing. The distance between the smartphone's camera and the input image is set at 20cm. The smartphone is rotated 30 degrees clockwise up to 360 degrees and the percentage of accuracy is recorded using Equation 2. The test is conducted 50 times at each degree. Figure 6 shows the configuration test performed to evaluate the AR algorithm in the real environment with a change in rotation.

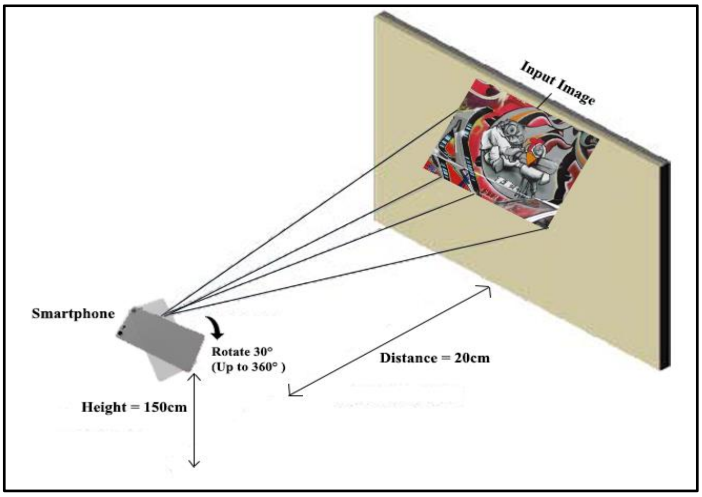

FIGURE 6. Configuration to Evaluate the AR Algorithm with a Change in Rotation

RESULT

Evaluation of AR algorithm in terms of brightness value was carried out using a mobile AR application and the test configuration was discussed in Section 3.1. Figure 7 shows the accuracy percentage obtained by the AR algorithm in different brightness values. A mean has also been conducted to determine the percentage of mean accuracy of the AR algorithm in terms of brightness value in the real environment. Table 1 shows the mean accuracy and standard deviation obtained by the AR algorithm and the accuracy obtained using the standard dataset in terms of brightness value based on the previous researcher (Tan et al. 2019).

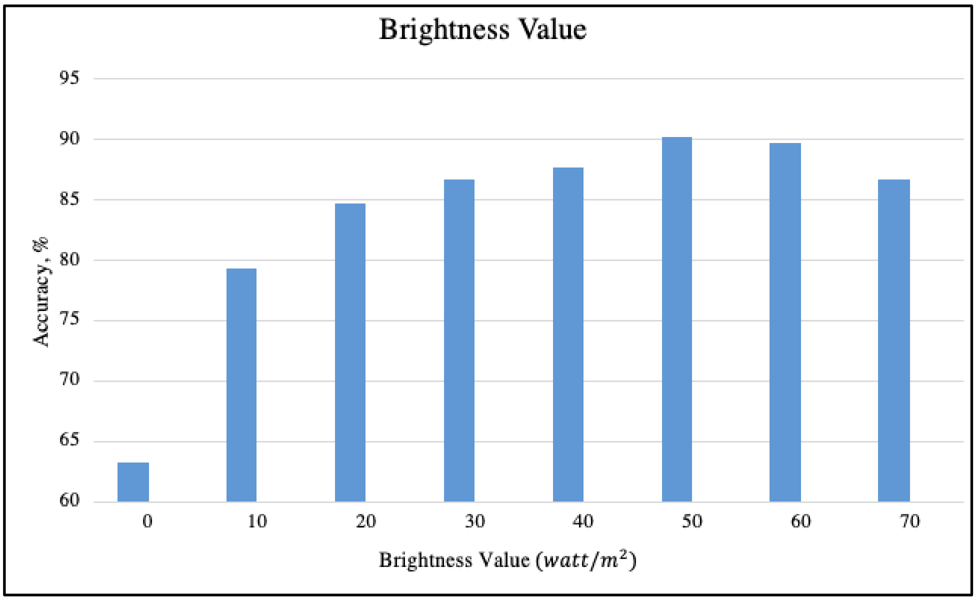

FIGURE 7. Accuracy Percentage Obtained by the AR Algorithm in Different Brightness Value

TABLE 1. Mean Accuracy and Standard Deviation Obtained by the AR Algorithm in Terms of Brightness Value

Condition	Environment	Mean (%)	Standard Deviation
Brightness Value	Real Environment	83.490	5.527
	Dataset (Tan et al., 2019)	84.041	-

The results showed that the highest accuracy percentage is obtained when the brightness condition is at 50 watt/m^2. This is because of this brightness condition had undergone a minimum change in brightness value compared to the reference image. The normal brightness condition allows the AR algorithm to function robustly. When the brightness value decreases (0 to 40 watt/m^2) or increase (70 watt/m^2), the percentage accuracy obtained by the AR algorithm decreases. Although the accuracy percentage decreased, the image recognition process has not failed despite the decreasing or increasing of the brightness value.

Performance of AR algorithm in terms of scale invariance is carried out using a mobile AR application and the test configuration was discussed in Section 3.2. Figure 8 shows the accuracy percentage obtained by the AR algorithm in different scales (distance between the smartphone's camera and the input image). A mean and standard deviation tests have also been conducted to determine the percentage mean accuracy of the AR algorithm in terms of scale invariance in the real environment. Table 2 shows the mean accuracy and standard deviation obtained by the AR algorithm and the mean accuracy obtained using the standard dataset in terms of scale invariance based on the previous researcher (Tan et al. 2019).

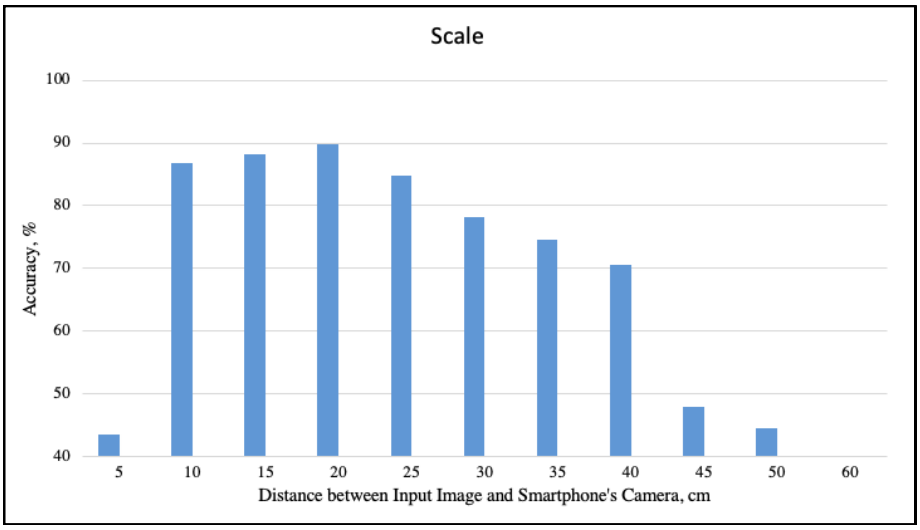

FIGURE 8. Accuracy percentage obtained by the AR algorithm in different scale

TABLE 2. Mean accuracy and standard deviation obtained by the AR algorithm in terms of scale

Condition	Environment	Mean (%)	Standard Deviation
Scale Invariance	Real Environment	70.890	8.471
	Dataset (Tan et al, 2019)	90.156	-

The results clearly showed that the highest accuracy is achieved when the distance between the input image and the smartphone's camera is at 20cm as the size of the image is almost similar to the size of the reference image. When the distance between the input image and the smartphone's camera is at 5cm, the image is too large to be detected. The accuracy percentage is high throughout the sequence of scale images (from 10cm to 40cm), but the accuracy percentage starts to drop dramatically at 45cm onwards. The accuracy percentage of the AR algorithm drops to 0% when the distance between the input image and the smartphone's camera reach 60cm because the image is too small to be detected.

Evaluation of AR algorithm in terms of rotation invariance is carried out using a mobile AR application and the test configuration was discussed in Section 3.3. Figure 9 shows the accuracy percentage obtained by the AR algorithm in different rotation. A mean and standard deviation test has also been conducted to determine the percentage mean accuracy of the AR algorithm in terms of rotation invariance in the real environment. Table 3 shows the mean accuracy and standard deviation obtained by the AR algorithm and the mean accuracy obtained using the standard dataset in terms of rotation invariance based on the previous researcher (Tan et al. 2019).

TABLE 3. Mean accuracy and standard deviation obtained by the AR algorithm in terms of rotation

Condition	Environment	Mean (%)	Standard Deviation
Brightness Value	Real Environment	72.771	8.174
	Dataset (Tan et al. 2019)	88.135	-

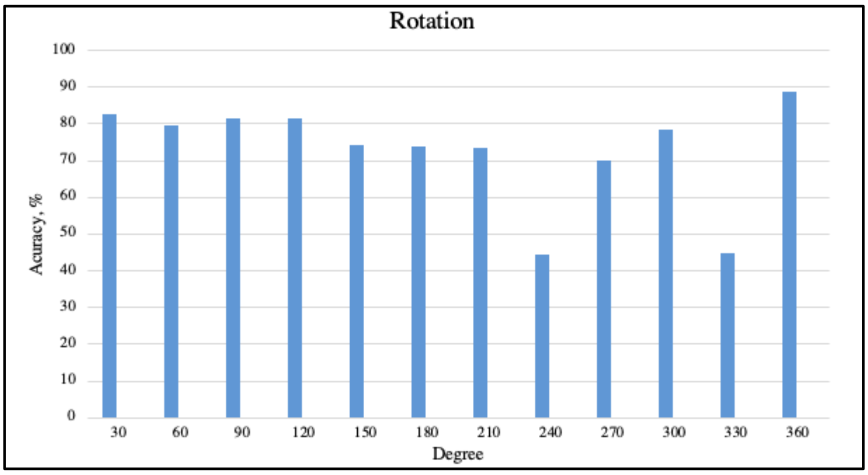

FIGURE 9. Accuracy percentage obtained by the AR algorithm in different rotation

The results showed that the highest accuracy percentage is obtained at rotation condition 360 degrees followed by rotation condition 30 degrees which are 88.59% and 82.50% respectively. This is because the two input images had undergone a minimum rotation change in the real environment and most of the AR algorithms can function accurately. As the smartphone starts to rotate gradually from the input image, the accuracy percentage of the AR algorithm began to decline especially when the scale condition reached 240 degrees and 330 degrees. However, the accuracy percentage of the AR algorithm did not change abruptly throughout the complete sequence of rotated images from 30 degrees to 360 degrees because all the AR algorithms used in this work are designed to extract features in high rotation variation.

Based on the result shown above, all the evaluations have been carried out smoothly and an optimum accuracy percentage up to 90.15% in the brightness value test has been achieved. The means of accuracy obtained in all the three tests are above 70% and the standard deviation value is low. The low standard deviation shows that all the data are clustered closely around the mean which is more reliable.

DISCUSSION

In this paper, a performance evaluation of the mobile AR algorithm in a real environment is presented. The goal was to compare the performance of the mobile AR algorithm in real environment and standard dataset. The steps to evaluate the performance accuracy of the mobile AR algorithm in a real environment is proposed in this work. The performance accuracy of the mobile AR algorithm using a standard dataset is obtained from the previous researcher which also tested the algorithm in terms of scale invariance, rotation invariance, and brightness value (Tan et al. 2019). The previous researcher had also tested the performance of the mobile AR algorithm proposed in this paper; BRISK detector, FREAK descriptor and Hamming Distance matcher. Hence, this work able to compare the mean accuracy of each invariance obtained from this work with the mean accuracy obtain from the previous paper. Based on the results performance accuracy of the mobile AR algorithm in the real environment are always lower compared to the standard dataset. The performance of the mobile AR algorithm in the real environment with different brightness able to obtain the mean accuracy up to 83.490% but able to obtain up to 84.041% by using standard datasets. While the performance of the mobile AR algorithm in the real environment and by using standard datasets in terms of scale invariance able to obtain the mean accuracy up to 70.890% and 90.156% respectively. The performance of the mobile AR algorithm in a real environment in terms of rotation invariance able to obtain the mean accuracy up to 72.771% but able to obtain up to 88.135% by using standard datasets. These results show that the performance accuracy of mobile AR algorithm in the real environment are always affected due to the huge range of controlled environment. Compared with evaluation using standard datasets through the mobile processor, images from real environments contain more variation in lighting, background, image quality, etc. One of the controlled conditions is the light conditions which ranging from dim lighting or darkness to bright sunshine. Therefore, the AR algorithm in real environments is much more challenging than by using a standard dataset. Moreover, many real-world applications require high accuracy performance in a controlled environment. Therefore, the evaluation of the AR algorithm in a real environment is needed to test the performance of the AR algorithm which might affect by the controlled environment especially outdoor environments. In summary, the performance gap often arises from the mismatch between the training data and the complexity of real-world scenarios, leading to a lack of robustness and adaptability in the algorithm's application to diverse and dynamic environments.

CONCLUSION

This paper aims to introduce a performance evaluation of the AR algorithm in the real environment. The AR algorithm tested in this research are BRISK detector, FREAK descriptor, and Hamming distance matcher. The performance evaluation proposed in this research can evaluate the robustness of the AR algorithm in terms of brightness value from 0 Watts per square meter up to 70 Watts per square meter. The robustness of the AR algorithm in terms of scale invariance has been successfully evaluated from the distance of 5cm up to 50cm from the input image. The performance of the AR algorithm in terms of rotation invariance in the real environment has been evaluated from 30 degrees angle up to 360 degrees angle. All the evaluations can achieve at least 70% of mean accuracy in the real environment. This paper has successfully presented a reliable performance evaluation for a mobile AR algorithm in a real environment.

ACKNOWLEDGEMENTS

This work was supported by The National University of Malaysia (UKM-TR2022-03) and Ministry of Higher Education Malaysia (FRGS/1/2018/ICT01/UKM/02/5).

REFERENCE

Alahi A, Ortiz R, Vandergheynst P, editors. FREAK: Fast retina keypoint2012 20122012.

Alshazly HA, Hassaballah M, Ali AA, Wang G, editors. An experimental evaluation of binary feature descriptors2017 2017: Springer Verlag.

Arshad H, Chowdhury SA, Chun LM, Parhizkar B, Obeidy WK. A freeze-object interaction technique for handheld augmented reality systems. Multimedia Tools and Applications. 2016.

Azuma R. A Survey of Augmented Reality. Presence: Teleoperators and Virtual Environments. 1997;6(4):355-85.

Bay H, Ess A, Tuytelaars T, Van Gool L. Speeded-Up Robust Features (SURF). Computer Vision and Image Understanding. 2008;110(3):346-59.

Bleser G. Towards Visual-Inertial Slam for Mobile Augmented Reality 2009.

Calonder M, Lepetit V, Özuysal M, Trzcinski T, Strecha C, Fua P. BRIEF: Computing a local binary descriptor very fast. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2012;34(7):1281-98.

Danielsson PE. Euclidean distance mapping. Computer Graphics and Image Processing. 1980;14(3):227-48.

Greg E. Markerless Affine Region Tracking and Augmentation Using Mser and Sift 2011.

Khan S, Ullah S. Feature-Based Tracking via SURF Detector and BRISK Descriptor. In: Lu H, editor.: Springer; 2018. p. 147-57.

Leutenegger S, Chli M, Siegwart RY, editors. BRISK: Binary Robust invariant scalable keypoints 2011.

Liu M, Scaramuzza D, Pradalier C, Siegwart R, Chen Q, editors. Scene recognition with omnidirectional vision for topological map using lightweight adaptive descriptors2009 2009.

Liu Y, Qin X, Xing G, Peng Q. A new approach to outdoor illumination estimation based on statistical analysis for augmented reality. Computer Animation and Virtual Worlds. 2010;21(3-4):321-30.

Cheng, Mingzhi, Luyue Zhang, and Long Liu. "An augmented reality image registration method based on improved ORB." Journal of Physics: Conference Series. Vol. 1544. No. 1. IOP Publishing, 2020.

Lowe G. SIFT - The Scale Invariant Feature Transform. International Journal. 2004;2:91-110.

Mikolajczyk K, Zisserman A, Schmid C, editors. Shape recognition with edge-based features2003 2003.

Nafea MM, Tan SY, Jubair MA, Abd Mustafa T. A Review of Lightweight Object Detection Algorithms for Mobile Augmented Reality. International Journal of Advanced Computer Science and Applications. 2022;13(11).

Obeidy WK, Arshad H, Tan SY, Rahman H, editors. Developmental Analysis of a Markerless Hybrid Tracking Technique for Mobile Augmented Reality Systems2015 2015.

Richard P. Lecture 7: "Starlight, Starbright" Stellar Brightness. 2006.

Rosten E, Drummond T, editors. Machine learning for high-speed corner detection2006 2006.

Rublee E, Rabaud V, Konolige K, Bradski G, editors. ORB: An efficient alternative to SIFT or SURF2011 2011.

Shang LW, Zakaria MH, Ahmad I. Mobile phone augmented reality postcard. Journal of Telecommunication, Electronic and Computer Engineering. 2016;8(2):135-9.

Tan SY, Arshad H, Abdullah A. A New Illumination Invariant Feature based on FREAK Descriptor in RGB Color Space. Journal of Theoretical and Applied Informat. 2016;93(1):207-14.

Tan SY, Arshad H, Abdullah A. An Efficient and Robust Mobile Augmented Reality Application. International Journal on Advanced Science, Engineering and Information Technology. 2018;8(4-2):1672-.

Tan SY, Arshad H, Abdullah A. Distinctive Accuracy Measurement of Binary Descriptors in Mobile Augmented Reality. PLoS ONE. 2019;14(1).

Uchiyama H, Marchand E. Object Detection and Pose Tracking for Augmented Reality: Recent Approaches. Foundation in Computer Vision. 2012:1-8.

Van De Sande K, Gevers T, Snoek C. Evaluating color descriptors for object and scene recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2010.

Zeng K, She K, Niu X. Feature selection with neighborhood entropy-based cooperative game theory. Computational Intelligence and Neuroscience. 2014.