

Article

Test and Validation of a Multi-Block Solution for Improved Tracking in Outdoor Scenarios: A Case Study in the Pinocchio Park

Massimo Magrini , Fabrizio Matarese and Davide Moroni 

Istituto di Scienza e Tecnologie dell'Informazione, Area della Ricerca di Pisa, 56124 Pisa, Italy

* Correspondence: massimo.magrini@isti.cnr.it

Abstract: Augmented reality techniques have recently found many applications in the field of cultural heritage. When used in outdoor scenarios, however, this technology can face several issues, mainly due to unstable light conditions, jeopardizing the users' experience. Various solutions to this problem have been proposed in the literature; however, none of them are fully effective. This paper introduces a solution based on a multi-block image target segmentation and a dedicated add-on for the Unity3D game engine. After tests in the lab, the solution was validated in a real scenario at Pinocchio Park (Collodi, Italy), using two different Augmented Reality (AR) libraries and comparing it to a standard methodology. Quantitative results show that the proposed approach provides superior performance and usability. Although the proposed solution is still open to improvement, it combines effectiveness and ease of implementation without any drawbacks.

Keywords: augmented reality; Unity3D; image tracking; image recognition



Citation: Magrini, M.; Matarese, F.; Moroni, D. Test and Validation of a Multi-Block Solution for Improved Tracking in Outdoor Scenarios: A Case Study in the Pinocchio Park. *Information* **2022**, *13*, 449. <https://doi.org/10.3390/info13100449>

Academic Editors: Ramon Fabregat, Jorge Bacca-Acosta and N.D. Duque-Mendez

Received: 16 July 2022

Accepted: 20 September 2022

Published: 25 September 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The Pinocchio Park was established in Collodi (Pistoia, Italy), the birthplace of the author of the famous novel, in 1956. It is made up of a set of monuments inspired by the novel, created by artists of the time, each invited to use their expressive sensitivity. The Institute of Information Sciences and Technologies of the CNR "A. Faedo" in collaboration with the Carlo Collodi National Foundation, developed an application to enhance one of the most important monuments that the theme park holds: the Piazzetta dei Mosaici by Venturino Venturi. This square is made up of a set of 21 mosaics depicting scenes from the book Pinocchio [1]. Following the original dream of an interactive theme park that requires the active participation of its guest with the monuments and works of art, the application gives life to the episodes described by Venturi on the mosaics of the square, bringing them to life in Augmented Reality [2], with multimedia animations faithful to Collodi's story. The application has different modes of use, including a kind of game: the visitor must frame the various mosaics (arranged in random order) in the chronological order from the novel with his smartphone. During the development, some tracking problems emerged due to the instability of the ambient light and the complexity of outdoor environments. To overcome these issues, a new original solution was introduced. The basic idea consists of using a multi-block approach to deal with local variations and improving the overall localization capabilities. The approach is general and can be used in several scenarios where natural feature tracking is prone to fail. In our work, we designed a strategy and developed a scalable method, which was then tested in a relevant environment, namely Pinocchio Park and its Piazzetta dei Mosaici.

The main contributions of the paper can be summarized as follows. First, a new multi-block system for improving recognition and tracking in AR applications based on natural feature tracking is proposed. The solution is accompanied by an open-source software released by the authors, which makes the results reproducible and might be

beneficial to the community working in AR. Secondly, thanks to the developed software, an optimization procedure can be used to select the best size in the multi-block approach. The method is also validated in a real environment by testing its outcome in the presence of different quantitatively and analytically described situations. This work validates the method proposed in PERCOM 2021 [3].

2. Background

For the creation and development of AR applications, two crucial issues are represented by the accuracy of user tracking and the registration of 2D/3D content to real-world features. Tracking in the literature can be based on different and complementary approaches. The most common is based on the so-called marker-based tracking, which requires the introduction in the physical scene of particular artefacts, mainly consisting of coded targets, such as ArUco markers [4]. Other approaches are markerless and are based on tracking natural visual features visible in the scene without introducing any foreign change. Markerless strategies usually deploy additional non-visual information to ease the recognition of the scene; for instance, they can take advantage of the knowledge of the GPS position, usually available outdoors, or other specific data, such as beacons in radiofrequency [5]. In cultural heritage applications, both the marker-based and the markerless approaches to tracking have found suitable use. Clearly, the use of standardized markers makes easier the tracking task and it is not site specific. That is, an AR application resorting to markers can be replicated in different scenarios and it is expected to work under most of the variable conditions that might be encountered outdoors, such as varying light, shadows, casted shadows of trebling elements (such as trees and foliage) and partial occlusions. However, introducing such foreign objects might be impossible or not practical in other scenarios, such as hard-to-reach walls or sculptures. Again, markers can visually disturb the sight of the physical artwork, negatively impacting the fruition of cultural heritage. Therefore, markerless approaches are to be preferred for their more immediate and possibly more engaging access to the augmented content.

However, user perception and smoothness of use are key factors that have been analyzed in several studies. For instance, in [6], the authors analyze the feedback of users of a mobile AR experience deployed in several locations and countries through interviews and surveys. In [7], the authors consider the analysis of a free interaction metaphor between users and heritage landmarks, allowing for its exploration throughout different periods. Metaio SDK is employed to transform this concept into an Android application. Specific examples consisting of the Leaning Tower of Pisa, the Cathedral and the Baptistery are used to validate the proposed solution and collect the users' evaluation. It is shown that AR applications may attract visitors and can enrich the sighting.

It might be argued that target loss during tracking or the failure in recognizing the target are significant shortcomings during the fruition of AR content and might jeopardize the user's acceptance. Therefore, it is necessary to take care of the accuracy and robustness of tracking since the registration of the 2D/3D content strongly depends on that. For further examples, we refer the reader to a complete survey of AR in cultural heritage [8].

In general, the recognition and tracking of 2D targets outdoors are always very critical and several attempts, more or less effective, to solve the problem can be found in the literature. In [9], a workflow is exposed to solve outdoor tracking issues. Starting from assumptions similar to ours (markerless multi-image approach in a real environment and difficulty in robust outdoor tracking), the research proposes a method to overcome some of these problems using Vuforia's image recognition approach. The researchers analyzed the lighting dynamics of the site (Parliament Buildings National Historic Site of Canada, in Ottawa), thus, having the ability to prepare a set of images useful for recognition for each time of day during the various seasons. In the system devised by the researchers, the user had to stand in the same spot where the target images had been taken, thus, using standard locations for the experience.

Other researchers [10], intending to recognize a series of real-world locations under different illumination conditions, developed an image recognition method based on natural

features and a user location system, then processing part of the data on a server. Using location information to limit irrelevant data was critical to the system's performance presented by the researchers. To do so, they quantized the user's location and considered only location data from nearby location cells. They developed a method for incrementally updating the local feature database of features on the handset when the user changes location.

Another approach adopts a system called Indirect AR [11], which replaces the live camera feed with a previously captured panoramic image. One of the most significant benefits of this approach in place of traditional AR is the greater registration accuracy. In traditional AR, any registration error is visible directly between the physical object and virtual annotation. In Indirect AR, the same registration error is only visible between the device and surroundings. That means that the registration between virtual annotations and the panorama representing the real world is always perfect, even if the registration between the device and the real world is not. One of the major drawbacks of this implementation of indirect AR is the dependence on pre-captured panoramas. To have an ideal experience, it would be necessary to have a panorama located exactly where the user was. Implementing indirect AR would typically mean having panoramas everywhere. While this is not possible, the authors approximated, using panoramas collected by Navteq as they travel most of the roads. There are two possible problems with this, however, exemplified by two questions: would users be able to look at the panorama and find the nearby point of interest in their view of the real world? Is the experience really similar to standard AR? However, if the Indirect AR solution presents an interesting alternative approach to solving outdoor image tracking problems, it currently remains a solution that does not allow the level of immersion that characterizes a standard AR experience.

Some authors presented a study [12] that seeks to improve the robustness of outdoor AR applications by mitigating the effect of light sensitivity on marker-based AR. The proposed approach allows—by default—a 'standard' marker-based AR framework to detect recorded physical objects and accurately overlay AR content on them. In case marker tracking fails due to inadequate illumination conditions, they propose the complementary use of a field-of-view (FoV) estimation technique (typically used in sensor-based AR applications). The FoV estimation algorithm detects whether the physical object is actually within the user's FoV and then attempts to project the AR content as accurately as possible. The authors "hybridized" the application by incorporating a sensor-based AR capability; the latter was enhanced by the implementation of geolocation-based raycasting to accurately detect when the user is actually in the line of sight with the sides of the registered polygon.

The above considerations show that it is worthwhile to carry out research towards the realization of more robust and accurate tracking systems for markerless AR applications. Indeed, the impact on the fruition of AR content in cultural heritage is significant and there is a growing demand and general interest. The following section introduces the proposed methods, the results of which are reported in Section 4. Section 5 discusses the results, while Section 6 concludes the paper with directions for future work.

3. Materials and Methods

Identifying a development system capable of efficiently managing 2D and 3D animations with the broad support of augmented reality technology for every Android and iOS mobile device was necessary. The choice, therefore, fell on Unity 3D Engine with the AR Foundation framework, a sort of wrapper of the features of ARKit (iOS) and ARCore (Android). This framework allows one to maintain a single code (C #) for both platforms. Target image recognition and tracking were initially implemented using only one image for each mosaic. To position the 3D content in front of the objects, a custom algorithm was devised that allows for disconnecting the animation from the recognized image: in this way, the system does not continually try to correct the positioning, a method that often introduces visible jitter. For this purpose, two augmented reality features were used, namely plane detection and anchors: the user initially chooses a plane from the AR scene so that each subsequent positioning refers to it. Once a target has been recognized, the system places an "anchor" in the scene, to which the animation hooks. To remove the 3D

models from the scene, a function was implemented in the code that constantly checks if the model is still within the camera view. The method is formalized in Algorithm 1.

Algorithm 1 Description of the image recognition and insertion of the AR experience.

```

while the augmented reality session is live:
  if an image is recognized:
    - Wait for 60 frames
  if the image was tracked for 60 frames:
    - instantiate a new GameObject with an anchor component
    - find the digital content for the recognized mosaic
    - add the content as a child of the anchor

```

The algorithm initially waits for the user to frame one of the mosaics. When the software finds a correlation with a target image present within the recognition library, it waits for a certain number of frames (experimentally fixed at 60) before beginning the instantiation of the animation. If the image is still tracked after this period, the code builds an “anchor” object in the scene to be used as a parent for the animation. In this way, the animations have stable and consistent positioning with the framed environment.

Tracking using the AR Foundation framework proved to be robust during indoor testing with scaled reproduction of three mosaics. However, in the actual outdoor situation, the application often failed to recognize the mosaic and then instantiate the corresponding animation. This issue is caused mainly by the variability in outdoor lighting conditions. Apart from variations due to weather conditions and from different times of the day, the most severe problems depend on shadows cast by plants around monuments, which irregularly invalidate the target images, rendering them unusable.

As explained in Section 2, in general, the recognition and tracking of 2D targets outdoors are always very critical and several attempts, more or less effective, to solve the problem have been investigated.

Some frameworks, such as Vuforia or Wikitude, provide an effective solution by leveraging 3D object and model recognition, thus, extending the ability to track the framed environment. These libraries, therefore, base their operation on the recognition of 3D targets. There are many cases where this type of approach cannot be used: for example, in our case in Pinocchio Park, the targets consist of mosaics, which are basically 2D shapes.

Experiments have shown that using multiple target images, taken in different light and shadows during the day, slightly improves the recognition ability. However, the reliability is still not sufficient for a publicly available application. Further, to have adequate robustness, it is necessary to acquire many images to cover all possible variations, which is impractical and not efficient. To prevent light variation limited to even a tiny part of the target image from completely invalidating recognition, the strategy of dividing each target photo into smaller blocks was chosen. This results in higher overall reliability; in fact, the probability that at least one block would be recognized is much higher (Figure 1).

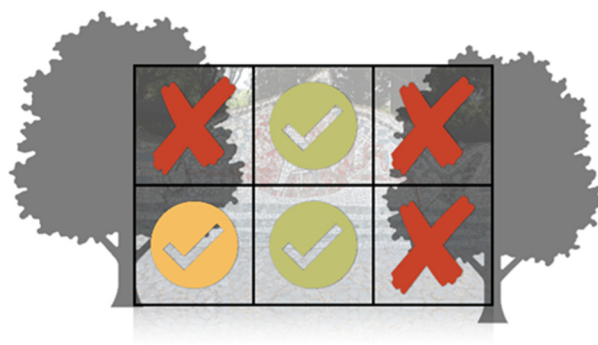


Figure 1. Target blocks recognition and failures.

Our approach can be described in pseudo-code as in Algorithm 2.

Algorithm 2 Steps involved in the creation of content to be attached to each block.

- Find the center of the object using the real measures

For each block:

- Calculate the size of the block using texture size in pixels

- Calculate the real size in meters for the blocks:

- calculate the real distance from the center of the artwork

- create a new texture with all data embedded in its symbolic name

- create a new Unity prefab with its transform shifted to be positioned at the centre of the real object, once instantiated

We consider that p is the probability of a block recognition and $q = (1 - p)$ the probability of a failure. We assume it remains unaltered if the block's dimension is still large enough. Assuming that recognitions are independent events and following the Bernoulli law P_k , the probability of recognizing exactly k blocks is expressed in Equation (1).

$$P_k = \binom{n}{k} p^k q^{n-k} \quad (1)$$

Therefore, the probability that at least one block is recognized can be expressed as in (2).

$$P_{k \geq 1} = \sum_{i=1}^n \binom{n}{i} p^i q^{n-i} \quad (2)$$

For example, if $p = 0.2$, when we divide the target into 2×2 blocks (that is, 4 blocks) the (2) gives 0.59 as a result, which is nearly three-times the original. Anyway, it is easy to extend and confirm such a law, because when the single block becomes too small, some other intrinsic recognition problems arise, mostly related to the device's insufficient resolution or optical quality, etc. The optimal subdivision must be found by experimentation. To facilitate this trial-and-error process, we developed a custom tool for the Unity Editor, to automate the image division task. This tool solves the issue of positioning the actual content in the scene, now that the position of the real object retrieved is no more the same as the entire object and, especially, it is different for each recognized block. This tool, called CropTool, divides a texture into blocks using an arbitrary number of rows and columns and, for each block (using the measures in meters of the entire artwork), creates a Unity empty Game Object with its transform shifted enough to place the digital content at the center of the mosaic as if recognized in the single block solution. The CropTool is freely available at this address: <https://github.com/SolidGorbash/Tool-for-improved-open-air-AR-Image-Recognition> (Accessed on 10 July 2022).

4. Results

The proposed multi-block solution (using a 2×2 block matrix) was then tested in the end-use scenario by comparing it with the original single-block application. This work basically presents and validates a demo showcased at PERCOM2021.

The actual scene consists of 21 mosaics on the walls of the square, each corresponding to a target. A maximum time of 30 s was considered for recognition, a timeout commonly used online with similar applications: in fact, it was noted that a longer timeout tends to discourage audience use of the system. The test was conducted in two different situations: uniform light (e.g., cloudy sky or very low sun) and light with shadows. It was noted that in the case of the presence of shadows, due to the particular shape of the square, the percentage of mosaics affected by the problem remains around 50 percent. The results are summarized in Table 1. Analyzing the 50% shadows column, we see that the probability of recognition is increased from 33% to 90% with the multi-block solution, according to what was expected from (2).

Table 1. Total number of recognitions.

Matrix Size	Uniform Light	50% Shadows
1 × 1	16	7
2 × 2	20	19

We conducted a test on four different mosaics, with different amounts of shadow: 50%, 67%, 83% and 100%. For each of these mosaics, we tested four different block matrix sizes. The results of the test are summarized in Table 2.

Table 2. Recognition time (s) for each different block matrix size, over different light conditions.

Shadow	1 × 1	2 × 2	3 × 3	4 × 4
50%	FAIL	1	FAIL	FAIL
67%	15	1	FAIL	FAIL
83%	5	1	FAIL	FAIL
100%	1	1	20	20

We also aimed to compare two different AR libraries in our scenario: AR Foundation and Vuforia. Although the latter generally allows for a recognition of the target at a greater distance, the recognition percentage is lower in the case of partial shadows, as shown in Table 3.

Table 3. Recognition time (s) AR Foundation (AF) vs. Vuforia (VF).

Shadow	AF 2 × 2	VF 2 × 2
50%	1	10
67%	1	5
83%	1	1
100%	1	1

5. Limitations

The optimal dimension of the subdivision matrix must be found according to the type of target and the expected amount and type of shadows. The proposed method should be tested on a broader range of live situations. In this study, we just report the tests related to our testbed, the mosaics square. Anyway, as we previously stated, the CropTool can easily allow experimentation to finetune the algorithm to maximize recognition probability in a specific situation

6. Discussion

The proposed solution yielded positive results, ensuring consistent image recognition on every mosaic in the plaza. This algorithm provides a practical solution that requires fewer images for each target than a complete library of all possible variations in the entire mosaic surface, reducing the number of recognized details the systems need to find a correlation between the provided image and the actual object. Although the optimal size of the sub-block subdivision matrix must be found experimentally with on-site tests, the implemented Crop Tool greatly facilitates the developer in preparing multi-block targets, making these experimental tests easy.

A future version of the tool is currently under development. This new version will automatically exclude irrelevant blocks (e.g., those composed of a uniform color), measured with the ARCore and ARKit tools from the image library.

Notice that, after studying the various solutions presented in the literature, we decided to implement our customized solution because none of the existing ones seemed really

effective. In addition, our project had to be connatural with the spirit of Pinocchio Park and it had to give the user great freedom of movement, implementing a technical solution that would allow visitors to freely explore the Piazzetta and, at the same time, be able to easily activate the various contents in AR. Therefore, we could not follow any path that restricted the user's position and movement at the time of the experience, but we had to grant immediate access to the AR content from most locations and view angles of the mosaics. We also decided to exclude solutions based on calculations performed on servers for several reasons. The first is not to increment the AR applications' traffic load and computational costs; indeed, sustainable use of resources, even computational and band aspects, is a growing concern. Secondly, we had some site-specific constraints, as there is no free and reliable wi-fi connection in our location, so relying on the ability to upload data online for processing could have discouraged or excluded some users from the experience. Further, unlike several works mentioned above, our prototype is based on AR Foundation and not on Vuforia and Wikitude, which offer more control over tracking possibilities, enhancing the basic end-user experience.

7. Conclusions

The capability to have accurate and robust tracking of natural visual features is of paramount importance to delivering AR content in markerless systems. The introduction of approaches in this direction might carve open new and more massive use of AR applications in cultural heritage. For instance, it might permit the realization of applications offering unique and personalized pathways for exploring landmarks without introducing foreign objects to the scene.

Aiming in this direction, this paper introduced a simple yet effective mechanism to improve tracking in outdoor scenarios. The method is mainly based on the partition of the object to be recognized visually by images. Under some mild assumptions, the theoretical gain in tracking performance was explored through statistical modelling. In addition, the methods were validated in a real scenario, consisting of the Piazzetta dei Mosaici at Pinocchio Park in Collodi. Results showed that the approach is effective and that suitable partitioning schemes might be adopted. A software tool was made available to the community to make the paper's results reproducible and to favor the implementation of markerless AR applications. In addition, the work also offers a comparison of two popular AR libraries, namely AR Foundation and Vuforia. In the future, we will explore further and more refined strategies for partitioning the visual images to be recognized in the scene. Eventually, we plan to study hierarchical multi-scale partitioning and its benefits in larger scenes.

Author Contributions: Conceptualization, methodology M.M.; Software, validation, F.M.; Project administration, funding acquisition, D.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by REGIONE TOSCANA, POR FSE 2014–2020.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Matarese, F.; Magrini, M. *Venturino Venturi e la Piazzetta dei Mosaici del Parco di Pinocchio*; Fondazione Collodi: Collodi, Pistoia, Italy, 2021.
2. Schnalstieg, D.; Höllerer, T. *Augmented Reality: Principles and Practice*; Pearson: Crawfordsville, IN, USA, 2016.
3. Magrini, M.; Magnavacca, J.; Matarese, F. A multi-block method to improve 2D tracking in outdoor Augmented Reality applications. In Proceedings of the IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops), Pisa, Italy, 21–25 March 2022.
4. Garrido-Jurado, S.; Muñoz-Salinas, R.; Madrid-Cuevas, F.J.; Marín-Jiménez, M.J. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognit.* **2014**, *47*, 2280–2292. [[CrossRef](#)]
5. Moroni, D.; Pieri, G.; Tampucci, M.; Masini, D. ARTiCo-AR in Tissue Converting. In Proceedings of the IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops), Pisa, Italy, 21–25 March 2022.

6. Boboc, R.G.; Duguleană, M.; Voinea, G.D.; Postelnicu, C.C.; Popovici, D.M.; Carrozzino, M. Mobile augmented reality for cultural heritage: Following the footsteps of Ovid among different locations in Europe. *Sustainability* **2019**, *11*, 1167. [[CrossRef](#)]
7. Duguleana, M.; Brodi, R.; Girbacia, F.; Postelnicu, C.; Machidon, O.; Carrozzino, M. Time-travelling with mobile augmented reality: A case study on the piazza dei miracoli. In Proceedings of the Euro-Mediterranean Conference, Nicosia, Cyprus, 31 October–5 November 2016; Springer: Cham, Switzerland, 2016; pp. 902–912.
8. Aliprantis, J.; Caridakis, G. A survey of augmented reality applications in cultural heritage. *Int. J. Comput. Methods Herit. Sci. (IJCMHS)* **2019**, *3*, 118–147. [[CrossRef](#)]
9. Blanco-Pons, S.; Carrión-Ruiz, B.; Duong, M.; Chartrand, J.; Fai, S.; Lerma, J.L. Augmented Reality Markerless Multi-Image Outdoor Tracking System for the Historical Buildings on Parliament Hill. *Sustainability* **2019**, *11*, 4268. [[CrossRef](#)]
10. Takacs, G.; Chandrasekhar, V.; Gelfand, N.; Xiong, Y.; Chen, W.C.; Bimpigiannis, T.; Grzeszczuk, R.; Pulli, K.; Girod, B. Outdoors augmented reality on mobile phone using loxel-based visual feature organization. In Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval, Vancouver, BC, Canada, 30–31 October 2008.
11. Wither, J.; Tsai, Y.T.; Azuma, R. Indirect augmented reality. *Comput. Graph.* **2011**, *35*, 810–822. [[CrossRef](#)]
12. Kasapakis, V.; Gavalas, D.; Dzardanova, E. Robust Outdoors Marker-Based Augmented Reality Applications: Mitigating the Effect of Lighting Sensitivity. In *Lecture Notes in Computer Science*; Springer: Cham, Switzerland, 2018.