Tech Science Press

Overlapping Shadow Rendering for Outdoor Augmented Reality

Naira Elazab[1], Shaker El-Sappagh[2,3], Ahmed Atwan[4], Hassan Soliman[1], Mohammed Elmogy[1], Louai Alarabi[5,*] and Nagham Mekky[1]

[1]Faculty of Computers and Information, Mansoura University, Mansoura, 35516, Egypt
[2]Universidade de Santiago de Compostela, Santiago de Compostela, 15705, Spain
[3]Faculty of Computers and Artificial Intelligence, Benha University, Benha, 13512, Egypt
[4]Faculty of Computer Sciences and Information Technology, Northern Border University, Arar, Saudi Arabia
[5]College of Computer and Information Systems, Umm Al-Qura University, Mecca, Saudi Arabia
*Corresponding Author: Louai Alarabi. Email: lmarabi@uqu.edu.sa

Abstract: Realism rendering methods of outdoor augmented reality (AR) is an interesting topic. Realism items in outdoor AR need advanced impacts like shadows, sunshine, and relations between unreal items. A few realistic rendering approaches were built to overcome this issue. Several of these approaches are not dealt with real-time rendering. However, the issue remains an active research topic, especially in outdoor rendering. This paper introduces a new approach to accomplish reality real-time outdoor rendering by considering the relation between items in AR regarding shadows in any place during daylight. The proposed method includes three principal stages that cover various outdoor AR rendering challenges. First, real shadow recognition was generated considering the sun's location and the intensity of the shadow. The second step involves real shadow protection. Finally, we introduced a shadow production algorithm technique and shades through its impacts on unreal items in the AR. The selected approach's target is providing a fast shadow recognition technique without affecting the system's accuracy. It achieved an average accuracy of 95.1% and an area under the curve (AUC) of 92.5%. The outputs demonstrated that the proposed approach had enhanced the reality of outside AR rendering. The results of the proposed method outperformed other state-of-the-art rendering shadow techniques' outcomes.

Keywords: Augmented reality; outdoor rendering; virtual shadow; shadow overlapping; hybrid shadow map

1 Introduction

In just a few years, augmented reality (AR) would inevitably become a popular technology. Computer and smartphone developments make it very desirable to improve AR techniques. These techniques facilitate the augmentation of images with probable renderings of virtual geometry, entertainment, education, and information purposes. Consistent conduct of both unreal and actual items in AR is necessary. To produce unreal items that seem like real-life components, two kinds

of consistency need to be achieved: Spatial consistency and visual consistency. Visual consistency is matched by appropriate rendering and illumination effects [1].

Most of the past studies in AR concentrated mostly on motion tracking and spatial consistency. The appropriate registration of unreal items with the actual world is required to place unreal items properly. However, visual consistency acquired less interest in the past. Rendering and illumination effects have now been utilized. Many AR systems discard that illumination effects and thus have problems with lacking oblique lighting. Visual consistency among unreal and actual views needs the appropriate global illumination condition to be simulated. Eventually, visual consistency is partly due to aliasing items and light bleeding [2]. This paper handles the stated visual consequences by introducing an efficient interacting technique for quality shadow tracking rendering, which can be used in AR.

In a comparison of indoor rendering, outdoor rendering contains more components, such as the location of sunlight, shades, rainbows, trees, and lawn. This paper seeks to present a functional explanation for many essential variables for outside rendering situations. The sun's location and relations between an unreal item and different real items are more substantial parts regarding outdoor scenes. These operators are essential because they're the distinguished aspects of the outdoor scenes [3]. Recently, AR was becoming a more exciting subject in computer graphics (CG) [4], which inspires researchers to get better techniques. In AR, reality can be performed by adding shadows and producing relations between unreal and actual items [5].

Generally, the realism of AR is an essential phase in CG, especially in the 21st century. Here, to generate a realistic unreal item in actual outdoor scenes, the location of sunlight, shades, and relations between unreal items are considered. Shadows are essential resources for outdoor parts. Rendering outside shadow is searched to visualize actual environments in various subjects, such as artists, environment simulation, gaming, and design architectures [6]. Shadows are the most distinguished variables concerning the improvement of realism in outdoor senses by recognizing the depth of the view and utilizing the range between items present. Without shades, it is intense to assimilate and recognize the actual size of items compared with the others, which is positioned further away [7].

Smooth shadows are designed to be utilized outside scenes, where their range from the sun is cosmic. Larger parts in outdoor scenes need a modified and specific shadow generation method. This method should show the difference between shades of the items positioned nearer to the camera's viewpoint and these positioned further away. AR can be utilized in various fields. It is now significantly popular, which is considered an impressive interdisciplinary technology. Specifically, smartphones contain effective processors, larger quality displays, and more highly developed cameras. They are authorized with characteristics, such as GPS, digital compasses, and accelerometers, which can be added to AR to get more widespread applications [8]. A suitable approach can combine all stated factors in AR to produce more realistic applications. Finally, it can give an extra benefit by combining the unreal with actual items all through a day in an AR environment [9].

This paper involves a new suggestion to produce a realistic real-time virtual shadow, considering shadow overlapping in outdoor environments. A smooth shadow production method with good quality and less expensive rendering is introduced for large outdoor scenes. Applying the proposed shadow method in AR is an additional contribution of this paper to generate unreal shadows on other unreal and actual items. This paper presents an enhanced technique for realistic rendering of virtual objects that use lighting information from real-world environments.

This method includes three main phases covering various criteria for outdoor AR rendering. The first mechanism is real shadow recognition that mange the detection of real shadow in the scene to measure its intensity. The second mechanism is the real shadow protection, which blocks real shadow parts from more rendering to reduce rendering time and handle shadow overlap. Finally, the third mechanism is the virtual shadow generation that manages virtual shadow generation based on a hybrid shadow map and rendering it in the outdoor scenes.

The remainder of this paper is organized as follows. In Section 2, a brief overview of the most appropriate techniques for the research topic is given, especially illumination estimation and shadow generation. Section 3 introduces the proposed method of shadow generation and casting based on a hybrid shadow map. In Section 4, the used dataset and the experimental results are described. Finally, Section 5 discusses the conclusion and future work directions.

2 Related Work

Several criteria must be fulfilled to guarantee realistic AR and Virtual Reality (VR) environments, such as an appropriate and stabilized pose of the objects regarding the user's view, an axiomatic hologram interaction, and a high frame rate to ensure fluidity of the stream. Most recent studies have been done to develop tracking methods for AR applications. Besides, the calculating capacity of AR devices has also increased. As a result of these accomplishments, the augmented objects' visual quality began to gain importance in AR simulations' realism. It can be revealed by extreme precision and the high frame rate of the augmented 3D models. However, it is affected by bad and slow environmental lighting in the simulation environment. Till lately, few studies were interested in shadowing in AR. Nonetheless, significant research is currently being made to enhance the knowledge base, which can be employed in outdoor AR environments.

Stumpfel [10] worked in the lighting of sunlight to create the realism of scenes. Sunshine is a mix of all lightings that come from sunlight and the diffuse of other items. In other words, sunshine involves sunlight, dissipate sky rays, and rays reflected from the earth. The power of the skylight isn't regular and depends on the purity of the sky. Hosek et al. [11] made crucial work on sky shade production and depended on the Perez model suffering from turbidity. Practical sky shade remains dependent on [12,13] methods. To attain real combined reality, shadows perform a significant position and are the essential operator for the view's 3D effects [14]. AR imitation of shadows for unreal items in actual scenes is challenging due to reconstructing actual scenes, particularly when estimating the actual view geometry and identifying lightings [15]. Jacobs et al. [16] organized the categorization of the lighting techniques into two organizations, which are popular lighting [17] and relighting [18] in AR environments. The reliability of shade structure with the proper approximation of lighting location is discussed in detail [19].

Casting unreal shadows on different unreal and actual items are one of the current problems in AR. Haller et al. [20] altered shadow volume to create a shadow in AR. This technique depended on the unreal items that seem like the actual ones, but with less precision in the simulated environment, which was named phantoms. The silhouette of the unreal and the phantom items was discovered. Phantoms shades night be cast on unreal items, and unreal shades might be cast on phantoms items. This technique needs several phantoms to cover the actual view. Silhouette recognition, the costly section of shadow volumes, is the critical drawback of this method, mainly when it is a complex view. Identifying an actual item and creating the silhouettes is another issue in this technique.

Jacobs et al. [21] presented a method to produce the unreal shade of actual items regarding unreal lighting, while the actual items and the unreal lighting were built with 3D sensing elements. They argued that projection shades are useful for simpler items, while shadow mapping is complex [22]. They introduced a real-time rendering approach to emulate the regular shadow of unreal items in AR. Aittala [23] used a convolution shadow map to create a smooth shadow in AR that applied a mip-map filter and summed-area tables [24] to improve cloudiness with inconstant radius. The technique was appropriate to view outside and indoor views.

Castro et al. [25] suggested a technique to create smooth shadows with less aliasing that woked on the fixed range per the sign, but with just one camera. The technique also performed one sphere mapping, like work in [26], but chose the most representative light source for the scene. This is essential due to the equipment constraints of mobile devices. The technique favored self-shadow and smooth shade. They applied filter techniques, such as percentage closer filter (PCF) [27] and variance shade map (VSM) [28], to create smooth shades.

Knecht et al. [29] used an approach in radiosity for mixing the unreal items with the actual scenes. Some challenges led to mixing the immediate radiosity and differential rendering, such as light bleeding and double shading. The last function eliminated conflicting bleeding. Non-real time rendering is triggered by collecting background pictures at differing times, which can be an essential difference in their research [30,31]. Xing et al. [32] proposed a stable method that estimated the outside lighting by using an essential set of each frame's lighting variables and constant features. Like prior work, this research was a point of view-dependent.

One of AR's most considerable problems is the precise lighting regarding the scenes to make a process seems more real [33]. In the case of internal rendering, light and impact of different items on unreal items and vice versa are essential that it may be taken into consideration to create items more reality. In the case of outside rendering concerning the sun, the impact of sunlight represents an essential part.

Kolivand et al. [34] presented an approach for using the sunlight's impact on unreal items in AR in any position and daytime. The technique's critical problem was casting shade just in regular areas because of the utilization of projection shadow for shadow production. This research attempted to overcome the prior problem related to casting unreal and actual items augmented items, such as what is visible on actual items throughout the daytime. In addition, Kolivand et al. [35] utilized hybrid shadow maps (HSMs) to cast delicate shadows on other virtual and actual items.

Barreira et al. [36] proposed approaching daylight environmental illumination and utilizing this data in outdoor AR to add unreal objects with shadows. The lighting variables were obtained in real-time from sensor data on recent mobile. Even though their approach produced visually AR images, the technique was not suitable in large and complicated environments. Kolivand et al. [37] presented unreal heritage taking into consideration virtual shadow. 3D lightwave created a historic building. This approach used a simple marker-less setting camera. They applied a semi-soft shadow map technique to generate and render an unreal shadow. This method was applied only on their Revitage device, which was not easy for portable usage.

Wei et al. [38] presented a shadow volume technology method based on the recognized edges of shadows in environments. The unreal shadow of items is equipped as curves, and a sampling method was presented. The more complicated shade interaction should be performed. The changes in lighting and different intensity of shadows were not taking into account in this approach. Osti et al. [39] proposed adding holograms into AR systems' actual views based on image lighting

techniques. They applied a negative shadow drawing technique that plays a role in the last photo-realistic affectation of a hologram in the actual environment. The principal limit in this method was the determination for the intensity of shadow was manual. Most of the mentioned techniques are interested in indoor shadow because it simple, unlike outdoor. Most of them disregard shadow overlapping between different objects in the scene. The bleeding light problem appeared when applied previous shadow map techniques in the scene with shadow overlapping, as shown in Fig. 1. This work handled this problem and reduced the rendering time while maintaining the shadow's high quality.
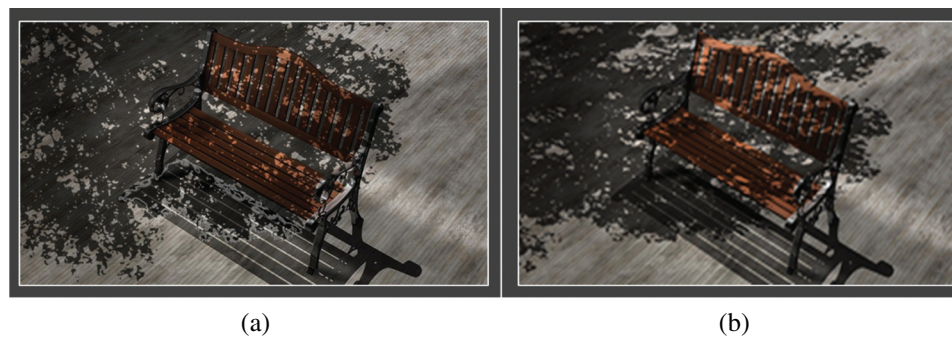

(a)   (b)

Figure 1: Light bleeding problem: (a) Penumbra from the tree shadow will be inappropriately bleeding via the seat, and (b) Light bleeding is handled with the proposed approach

3 The Proposed Shadow Technique

This paper introduces the shadow generation technique in AR in which the overlap between actual and unreal shadows is solved. When these unreal shadows are added using shadow maps, the individual's pixels that locate in the unreal shadow get the scaled intensity using a suitable rate variable. Casting unreal shadows on other unreal and actual items should be supported in realistic outside scenes. The proposed approach is presented to attain this issue. The proposed shadow creation approach is facilely applied not only in any unreal views but also in all AR systems. This method provides three contributions for virtual shadow generation in the AR system. The main contributions of the paper are summarized in the following points:

- Enhance the quality of generated shadows in the high-density scene.
- Create outdoor realism rendering without worry about shadow overlapping.
- Generate real-time shadow by increasing frame per second (FPS) compared with the different shadow map techniques.

The proposed approach is illustrated in Fig. 2. It comprises of three major phases: (1) Recognition of real shadow, (2) Real shadow protection, and (3) Virtual shadow rendering. The recognition of the real shadow step manages the location of the position and form of the actual shadows in the view of actual objects. Once the real shadow contour is known, it is conceivable to compute a scaling factor for each material in shadow, representing the shadow region's intensity. A binary shadow mask is generated in the actual shadow protection phase to prevent these pixels in an actual shadow from any scaling. The scaling factor has matched the shadow of the non-overlapping areas with the pixels in the actual shadow. In the shadow rendering phase, a consistent shadow technique, a hybrid shadow map (HSM), is utilized to create the unreal shadows.

The shadow intensity is identified with the proper scaling factor computed previously. Overlap amongst real and unreal shadows is blocked by utilizing the previous phase's mask.
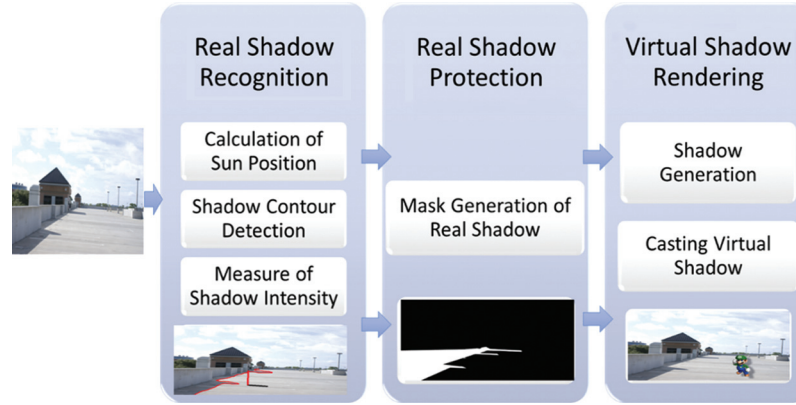

Figure 2: The proposed framework for outdoor overlapping shadow rendering

3.1 Recognition of Real Shadow

The recognition shadow step manages the location and contours of the actual shadows of the actual items in sight. Lalonde et al. [40] approach is applied for the shadow recognition step in our proposed method. The difference is that false positives are discard using object positions rather than integrating the scene structure to erase false positives. Thus, the search area in the shadow inference step is reduced by the proposed approach.

Once the actual shadow contour is apparent, a shadow scaling variable per material can be determined, representing the shadow area's color intensity. This variable is measured to match the color of the not overlapping areas with the points in the actual shadow. Extra clarification of the shadow recognition step will be introduced in the following.

3.1.1 Calculation of Sun Position

Julian's mathematical modeling is adopted because it is simple to manage in a real-time environment. Julian's date is a precise technique to calculate the sun's position [41]. The position could be calculated for a specific longitude, latitude, date, and time using Julian date. The time of day is calculated using Eq. (1).

$$t = t_s + 0.17 \sin\left(\frac{4\pi(J - 180)}{373}\right) - 0.129 \sin\left(\frac{2\pi(J - 8)}{355}\right) + 12\frac{SM - L}{\pi} \qquad (1)$$

where t is solar time, t_s is standard time, J is Julian date, SM is a standard meridian, and L is longitude. The sun-oriented declination is ascertained using Eq. (2). The time is computed in decimal hours and degrees in radians. Finally, zenith and azimuth can be ascertained by Eqs. (3) and (4).

$$\delta = 0.4093 \sin\left(\frac{2\pi(J - 8)}{368}\right) \qquad (2)$$

$$\theta_s = \frac{\pi}{2} - \sin^{-1}\left(\sin l \sin \delta - \cos l \cos \delta \cos \frac{\pi t}{12}\right) \tag{3}$$

$$\varphi_s = \tan^{-1}\left(\frac{-\cos \delta \sin \frac{\pi t}{12}}{\cos l \sin \delta - \sin l \cos \delta \cos \frac{\pi t}{12}}\right) \tag{4}$$

where θ_s is solar zenith, φ_s is solar azimuth, and l is latitude. With a count of zenith and azimuth sun's position will end up plainly self-evident. Algorithm 1 lists the pseudo-code for determining sun location.

Algorithm 1: The pseudo-code for determining sun location

Input: longitude l, latitude L, Julian date J, time zone tz and standard time t_s
Output: Solar zenith θs, solar azimuth φ_s
Step 1: Calculate solar time
  $T = t_s + 0.17^* \sin(4^*\pi^*(J - 180)/373) - 0.129^* \sin(2^*\pi^*(J - 8)/355) + (12^*(tz - L)/\pi)$
Step 2: Calculate sun-oriented declination
  $\delta = 0.4093^* \sin(2^*\pi^*(J - 8)/368)$
Step 3: Calculate solar zenith
  $\theta_S = (\pi^*2) - \sin^{-1}(\sin l^* \sin \delta - \cos l^* \cos \delta^* \cos(\pi^*t/12))$
Step 4: Calculates solar azimuth
  $\varphi_S = \tan^{-1}(-(\cos \delta^* \sin \pi^*t/12)/(\cos l^* \sin \delta - (\sin l^* \cos \delta^* cos(\pi^*t/12))))$
Return θs, φs
End

3.1.2 Shadow Shape Detection

The shadow pixels in sight must be identified to avoid any post-processing of the scene's actual shadows. Two kinds of shadows exist soft shadows and hard shadows. The purpose of this paper is not to establish a new shadow recognition technique but rather that the three-phase process is open in this first step for the option of shadow recognition method.

Shadow recognition is designed to identify areas with cast shadows. Recent methods have used primarily invariants of lighting that could fail if image quality is low. Lalonde et al. [40] presented a technique for recognizing shadows on the ground. The approach is based on the observation that traditional outdoor views are restricted mainly in concrete, asphalt, grass, and stone. These shadows can be learned from a sequence of labeled real-world pictures. Indeed, they divide the recognition issue into shadow-non-shadow boundaries. Self-shade and complex geometry are typical phenomena that can confuse the classifier, as illustrated in [40]. They have scene layouts to eliminate false positives. Rather than integrating the scene structure to erase false positives, the proposed approach utilizes areas that are now decided to render virtual objects.

3.1.3 The Measure of Shadow Intensity

After defining the shadow edge, the scaling variable can be determined for each shadow material. The scaling factor is a triplet [ρ_R, ρ_G, ρ_B], which is determined by dividing the shadow's three-color channels with channels of non-shadow pixels. A small area of pixels inside and a small area of pixels outside the shadow was chosen, and a mean for scaling variable was used.

The following is the calculation of the triplet. $C = \{R; G; B\}$ represents the color channel: Red, green or blue. SA represents for shadow area, while NSA represents the non-shadow area; PSA and PNSA are those numbers of pixels in those shadow and non-shadow areas, respectively:

$$\rho_C = \frac{\dfrac{\sum_{\forall_P \in SA} C_P}{P_{SA}}}{\dfrac{\sum_{\forall_{P`} \in NSA} C_{P`}}{P_{NSA}}} \tag{5}$$

This scaling variable 'usually changes across the surface area and relies on the surface's sun position.

3.2 Real Shadow Protection

In this step, a binary shadow mask is generated in the real shadow protection stage, which protects points from any rating within an actual shadow. The rating variable is chosen to balance the non-overlapping areas' color with the actual shadow points. Every point within the actual shadow is successfully blocked from moreover rendering when the result of the real shadow recognition step is utilized to complete the mask of shadow. A binary mask is designed based on the shadow outline. This mask could be utilized to show the points are in shadow or not in the actual model. In a sense, it is regarded as a map of texture that overlays that scene's structures. It is reasonably easy to derive the shadow region: the shadow mask building captures the edge outline as input. It utilizes a region expanding algorithm to full the area of shadow within the shadow outline. Any point within the shadow area that could be extracted from real shadow recognition can be the region's starting.

A gray-level mask will be shaped with values from 0 to 1 rather than a binary one. These shadow points are fully covered (bright). The non-shadow points are not covered (dark). The edge points of the shadow are partially covered (grey). The gradient is quadratic in the soft area to mirror the visibility in visibility concerning the sun's shining. The quadratic gradient's help increases with the degree of softness. For instance, on a sunny day, the sun's solid angle is 0.5 degrees for an outdoor scene. An estimate of the softness level can be determined along with the geometry calculation and the sunlight path. In other instances, the pixel intensities along the edge of shadow recognized can be used. A grey degree could give a chance to be created in shadow masks at every pixel, so it demonstrates what amount of light it gets, without reorganizing the shadow rendering step.

3.3 Virtual Shadow Rendering

A steady procedure shadow map is utilized to create the unreal shadows in the shadow rendering step. The shadow's intensity identifies with the best possible scaling variable handled in advance. Overlapping between actual and unreal shadows is avoided by utilizing a mask created in the previous stage. The individuals' pixels of the unreal shadow that allocate in the actual shadow would be neglected. The pixels in the unreal shadow in the non-covering areas would determine by rating these shading pixels with the variable rate.

The shadows due to the incorporation of unreal objects can be simulated after the actual shadow areas have been identified. Various methods could generate these shadows. Shadow maps are regularly utilized as a part of real-time rendering but suffering from extreme aliasing. The shadows are determined with the sun's estimated location, and the interactions between unreal and actual objects are considered. Digital object shadows are casting on unreal and actual things. Also,

actual objects being able to cast a shadow on unreal objects. The shadow intensity is measured using the scale variable. For each object in which the shadow is cast, this scaling variable varies. The material properties, as well as the ambient illumination in each area, should be reflected.

- **Lighting Position**

A reference system is modified from O(x, y) to unity skybox (O′(x, y)), as shown in Eq. (6) [39]. For this specific reason, a vector subtraction is needed. It is essential to observe that the y-axis is inverted. Hence the newest coordinates contrast with the new origin are represented by Eqs. (7) and (8).

$$O(x, \ y) \rightarrow O'(x, \ y) \tag{6}$$

$$P'_x = P_x - O'_x \tag{7}$$

$$P'y = -(P_y - O'_y) = O'_y - P_y \tag{8}$$

when the point $P_0(x, \ y)$ is univocally identified regarding $O_0(x, \ y)$. The P_{0x} reference is relative to turning around the unity y-axis, while P_{0y} is relative to turning around the unity x-axis. The relations between measures and aspects $\theta \rightarrow P'_x$, $\phi \rightarrow P'_y$ are represented by Eqs. (9) and (10). The picture width fits 360° turning around the unity y-axis, while the picture height fits 180° around the unity x-axis. The application of this technique to an actual scene in the unity editor provides efficient results.

$$P'_x : \theta = \frac{width}{2} : 180 \ \text{deg} \tag{9}$$

$$P'_y : \phi = \frac{height}{2} : 90 \ \text{deg} \tag{10}$$

- **Casting Virtual Shadow in Virtual Item**

Large regions in outdoor views need an improved and precise shade generation approach to show the variation between shadows of the items positioned nearer to the camera viewpoint and positioned remotely of the camera. High quality and low rendering cost soft shadow technique is presented, as needed in large outdoor scenes. Shadow map is suitable for casting shades on different items, but they have problems with aliasing. Using Z splitting on traditional shadow maps and adjusting the parts' resolution can resolve the aliasing out, as numerous studies stated in the literature. Soft shadow is probably the most matched forms of shades that can be viewed for outside rendering. The convolution with Gaussian mask is applied on the enhanced shadow map utilizing Z partitioning to create soft shades.

Even though shades show the real range between items in VR, AR techniques still appear to absent the distance between actual and unreal items. This issue is achieved more in indoor rendering than outside AR approaches because of extended ranges and wide parts in outside scenes. Using a particular fog parameter in a specific part of see frustum that can be separate beforehand makes the unreal items seem not even close to the camera and, therefore, suited to ranges in outside scenes. The approach is illustrated as follows, and the pseudo-code of the algorithm is shown in Algorithm 2.

The step result of shadow map rendering is shown in Fig. 3. Additional phases aren't required to create an unreal shadow on unreal items through applying a Z-Gaussian shade map. Because they're based on shade maps, casting the unreal shadows on different items is the key capacity of this type of shadow generating approach.
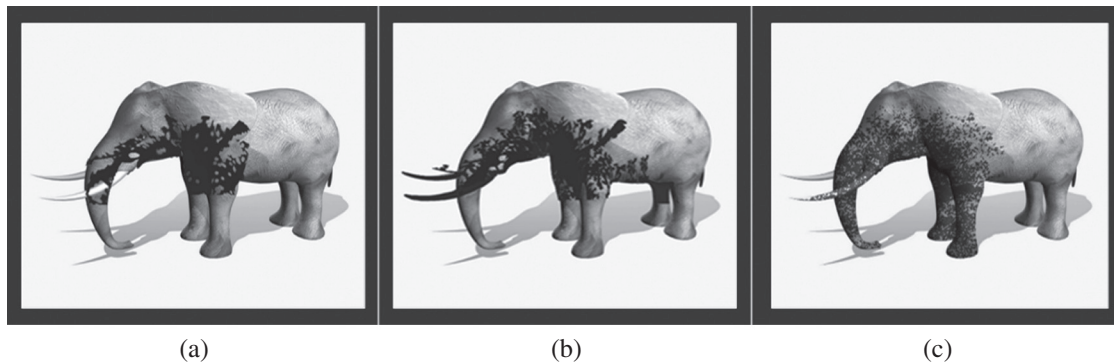
<div align="center">(a)   (b)   (c)</div>

Figure 3: The results of shadow map rendering steps: (a) The first two steps of CSM, (b) applying the Z-partitioning, and (c) Gaussian step result

Using Z partitions and Gaussian on the shade map decreases aliasing through raising high-resolution parts in the view. It can be deeper to the sight level and minimize the resolution for regions of the view far. Z partition is performed by dividing the camera see frustum into sections and stuffing the Z-buffer for every section individually. Assigning suitable resolution to every part is dependent on the part's z value. This strategy is utilized for broad views, for example, large land.

Frustum dividing enables a shade map to be created and to improve the quality of every separate region. The various kinds of dividing have an impact on the ultimate accuracy and rendering time. The logarithmic and specific dividing systems are the most popular forms of dividing. They are dividing full views into numerous regions to handle the resolution in various areas of a view. A significant difference between HSM and this method could be the non-uniform divisions.

Approximating the range distribution applying Gaussian not just creates softer shades also decreases the calculation and storage cost. Dividing the Z depth by two or more depends on the space of items from the camera point of view enables the improvement of every part's quality. The high resolution creates good accuracy while providing reduced FPS. The near parts are placed with large enough quality to improve the reality of items. Reduced resolution decreases the full time of rendering, therefore raising the rate of rendering. Indeed, whenever a broad view such as an outdoor scene is rendered with the exact resolutions, some areas positioned far from the camera might not be shown correctly.

Algorithm 2: The Pseudo Code for the gaussian shadow map rendering algorithm

---

Data: Current camera position, light position, shadow intensity
Result: Shadow map texture
Start
    Rendering the scene from the light source
    Rendering scene from the viewpoint
    Partition z map like layered variance shadow map (LVSM)
    Adjusting resolution
    For All parts in z map
        If Part is close to viewpoint **Then**
            Assign high resolution

<div align="right">(Continued)</div>

          Else
               Assign low resolution
          End If
     End for
     Apply gaussian for all parts
     Apply fog on away parts
End

Higher-quality outputs in a greater shade accuracy but have a problem with rising the rendering time. To solve this issue and keep consistent stability between accuracy and rendering rate, the frustum is separated into various parts. The number of regions could be adjusted. To improve the accuracy of shades, the near regions are adjusted with greater quality. In contrast, in the case of decreasing the full time of rendering, away parts are adjusted with a reduced quality.

4 Results

4.1 Dataset, Hardware, and Software Specifications

We evaluate our approach on 500 consumer photographs downloaded from Label-Me [42]. We randomly selected 355 outdoor images for training and 145 images for testing with a resolution of 640 x 425 pixels. The proposed technique is implemented on the dataset using Unity 3D version 17.2, Vuforia SDK, and on Core (TM) i7, 2.80 GH processor, NVIDIA GTX 1060, and 16 GB RAM, Windows 10, 64-bit operating system, and Graphics APIs DirectX 11.

4.2 Evaluation of Shadow Recognition Methods

In this part, some experiments are conducted to evaluate the selected approach and examine it against different motion casting shadow recognition techniques. To verify the shadow recognition method's capability, the accuracy (ACC) is used by using Eq. (11). The measure of sensitivity (Sen) or recall is used to verify a classifier's ability to identify positive patterns (Eq. (12)). The specificity (Spec) is used to evaluate a method's ability to identify negative patterns by using Eq. (13). The F-measure or Dice similarity coefficient (DSC) evaluates the test's accuracy by using Eq. (14).

$$Acc = \frac{TP + TN}{TP + FP + TN + FN} \tag{11}$$

$$SEN = \frac{TP}{TP + FN} \tag{12}$$

$$SPE = \frac{TN}{TN + FP} \tag{13}$$

$$Precision = \frac{TP}{TP + Fp} \tag{14}$$

$$DEC = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{15}$$

Tab. 1 reveals measurable results. The displayed values are the average on all tested images. The chosen approach aims to provide a quick shadow recognition technique without losing accuracy. Thus, the implemented time of the selected approach is displayed in Fig. 4.

Table 1: The results of different shadow recognition approaches

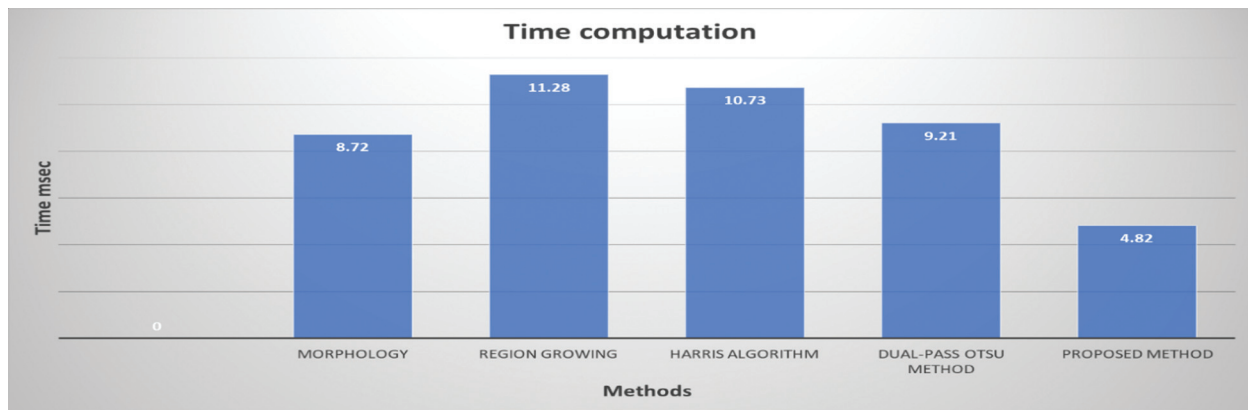| Method	ACC	SEN	SPE	precision	DSC	AUC
Morphology	92.05	79.42	97.20	92.05	85.27	88.31
Region growing	89.39	73.74	96.19	89.39	80.81	84.96
Harris algorithm	88.64	72.22	95.90	88.64	79.59	84.06
Dual-Pass Otsu method	91.35	77.87	96.94	91.35	84.07	87.40
Proposed method	95.14	86.71	98.32	95.14	90.73	92.51

Figure 4: Time computation (in milliseconds) of shadow recognition approaches

4.3 Evaluation of Shadow Rendering for the Proposed Framework

This section shows the results of the tracking shadow and rendering for the presented algorithm. From the view of constant rendering, the proposed strategy is sufficiently precise. The effects presented in this area view those techniques through which those targets and, therefore, the research point are attained. These shadows and the impacts of overlap concerning unreal items in the AR framework are handled gradually. The experiment uses the ARkite to perform real-time tracking of the camera. The proposed method introduces real-time shadow by reducing the time of computation and increasing FPS, as shown in Tabs. 2 and 3. The problem of overlap shadow is solved in this technique, as illustrated in Fig. 6. The enhanced quality of the virtual shadow is presented in this section.

No extra steps are required for unreal shadows to be created by shadow maps on unreal objects. The key capabilities of this type of shadow generation method are to cast unreal shades on other objects. Tab. 2 shows a performance comparison of the proposed algorithm and several methods by using different resolutions. The computation time of the proposed system and some revealed literature techniques is given in Tab. 3.

Table 2: The speed of rendering in different resolutions measured by FPS

Rendering technique	512×512	1024×1024	2048×2048
VSM	109	75	63
LVSM	112	82	76
HSM	123	89	82
Proposed method	132	98	91

Table 3: The time computation (in microseconds) of shadow rendering methods

Rendering technique	512×512	1024×1024	2048×2048
VSM	9.1	13.3	15.8
LVSM	8.9	12.1	13.1
HSM	8.1	11.2	12.1
Proposed method	7.5	10.2	10.9

The proposed solution is ensured to have lower aliasing and more soft edges. The improvements of shadow quality by the proposed method can be shown in Fig. 5. It shows a close-up of object shadow rendered using the proposed method and VSM. A close-up appears some artifacts and aliasing for VSM.

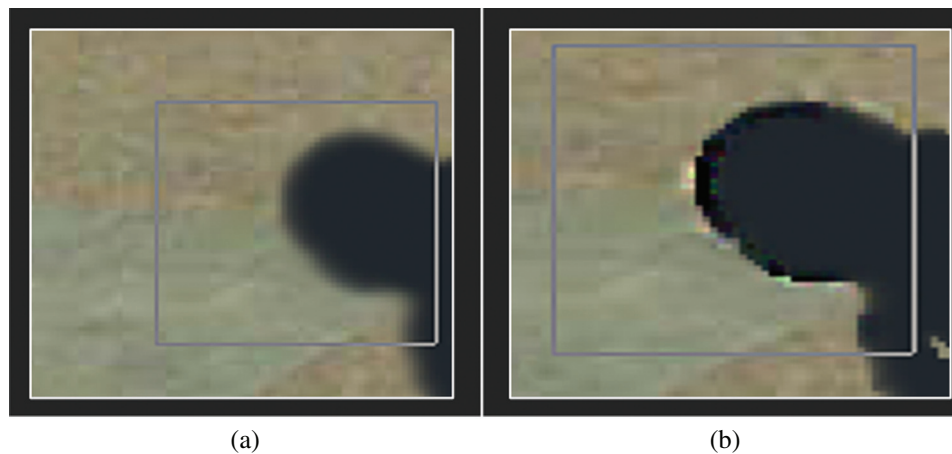(a)   (b)

Figure 5: The close-up of shadow: (a) The proposed method, (b) Using the VSM method

Fig. 6 shows the consequences of calculating an energized virtual symbol strolling around a real tree in the street. Shadows are consequently identified and created utilizing suggested real-time calculation. Execution information truly exhibits that the recommended strategy will be great suiting to utilization into the real-time application.

Fig. 7 illustrates a comparison between prior methods and the proposed approach in the outdoor environment. Fig. 7a is the result of VSM, while 7b is the output of HSM. Fig. 7c shows

the CSM result, while Fig. 7d results from the soft shadow's proposed approach in the outdoor scene. This result shows that our proposed system outperforms other techniques.
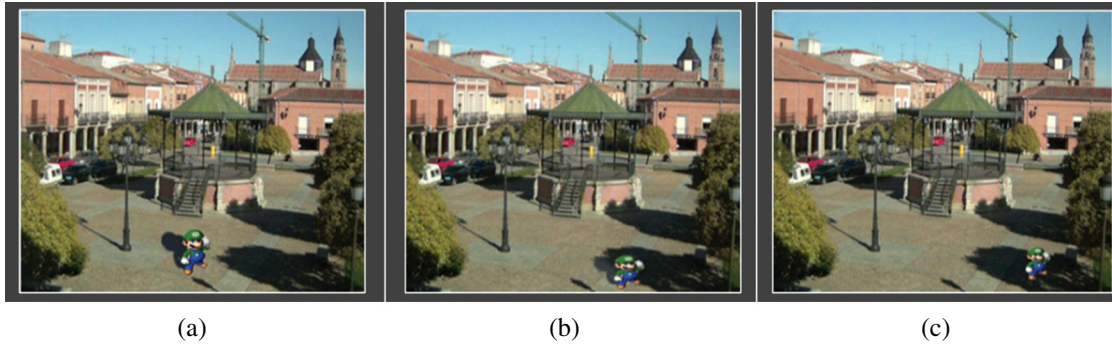

(a)   (b)   (c)

Figure 6: An example of a large scene with overlapping shadows rendering a virtual object using the proposed method: (a) The case of non-overlapping so rendering the complete shadow of an object, (b) The partial overlapping is then rendering shadow of the only shining part, and (c) The completely overlapping, so no rendering shadow

(a)   (b)

(c)   (d)

Figure 7: The soft shadow results in the outdoor scene: (a) VSM, (b) CSM, (c) HSM, (d) Proposed shadow map

5 Discussion

The appearance of scenes is influenced in complex ways by outdoor lighting. In general, solving illumination from surface geometry is a challenging issue. Surprisingly, many outdoor photos provide rich and insightful signs of lighting, such as shadows. A shadow generation system based on the scene's detected shadow is proposed to fix an actual scene's casting shadows onto an unreal entity. Naturally, unresolved problems remain. One deals with preventing double shadows from being generated effectively when making an unreal shadow on top of an actual one, resulting in a much too dark shadow. We have a solution to this. By simply applying the shadow mask developed by the shadow recognition module, it is possible to render unreal shadows across actual shadows without producing double shadows. Unfortunately, there will not be overlapping shadows unless the unreal object casts a shadow on the actual object, causing the same shadow, or vice versa. We have provided a method to evaluate all parameters of a complete model of outdoor lighting based entirely on basic image measurements from shadow images. The technique assumes that time, date, compass heading, and earth position information are available, all of which represent information generated automatically in consumer cameras, which is very likely. Even when the actual shadow caster is entirely invisible or partially visible in the video frame, the experimental results show that our approach generates realistic shadow interactions. It is impossible from conventional 3D-based shadow interaction methods. This is the first step in AR systems to realize practical shadow interactions between virtual and actual objects. More complex shadow interaction simulations can be conducted with more advanced calibration techniques and more detailed depth data. The average time costs of the two phases for shadow recognition and shadow rendering are listed in Tab. 1 and Fig. 4.

The higher the resolution, the greater the time cost, according to Tab. 3. The most time cost is the stage of making shadow recognition. On the time-consumption ratios of detection and rendering, we carried out a step-by-step evaluation. From Tab. 2, the proposed technique outperforms the methods of VSM, LVSM, and HSM in both low and high resolution. Results of assigning different resolutions achieve good results, which gets the highest shadow rendering rate for all resolutions. The proposed framework outperforms VSM, LVSM, and HSM because of faster computation time and is more precise and easily implemented. Another advantage is reducing the amount of overlapping due to non-linear layering.

6 Conclusion

This paper supplies an approach to show the relation between unreal items within AR concerning the shadow. The key contribution and shade enhancement are the looks of realism unreal items in outside rendering. It requires 3D items and shades impacts that improve the reality of the AR systems. The proposed technique comprises a three-stage structure: real shadow recognition, real shadow protection, and virtual shadow rendering. This three-stage calculation created reliable shadows amongst actual and unreal objects continuously. The real shadow recognition phase handles recognizing the positioning and the actual shadows of the actual items in sight; adjust the actual sight's consistency. First, the sun position is calculated. Next, the contour of the real shadow is determined. When the contour of the actual shadow is known, it's probable to estimate the scaling variable in shadow that shows the actual shadow's color intensity. This variable relates the intensity in the shadow with one, not in shadow. In the actual shadow protection phase, a binary shadow mask is applied to guard these pixels in the actual shadow from any rating. The rating variable is selected to relate non-overlap parts' colors with the pixels in the actual shadow. The results of the first phase are utilized to perform the shadow mask. All pixels in the actual

shadow are blocked from rendering, so this step prevents the overlapping between unreal and actual shadow. In the virtual rendering shadow phase, the unreal shadow is generated utilizing the HSM real-time method, considering a predefined rating variable that adapts actual shadow intensity with an unreal one.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Gupta, R. Chaudhary, A. Kaur and A. Mantri, "A survey on tracking techniques in augmented reality based application," in *2019 Fifth Int. Conf. on Image Information Processing*, Shimla, India, pp. 215–220, 2019.

[2] L. Gruber, T. Langlotz, P. Sen, T. Hoherer and D. Schmalstieg, "Efficient and robust radiance transfer for probeless photorealistic augmented reality," in *2014 IEEE Virtual Reality*, Minneapolis, MN, USA, pp. 15–20, 2014.

[3] D. Schmalstieg and T. Höllerer, "Augmented reality? Principles and practice tutorial," in *2016 IEEE Int. Sym. on Mixed and Augmented Reality (ISMAR-Adjunct)*, Los Angeles, CA, USA, 2017.

[4] G. Singh and A. Mantri, "Ubiquitous hybrid tracking techniques for augmented reality applications," in *2015 2nd Int. Conf. on Recent Advances in Engineering & Computational Sciences*, Chandigarh, India, pp. 1–5, 2015.

[5] G. Bhorkar, "A survey of augmented reality navigation," *Foundations and Trends® in Human—Computer Interaction*, vol. 8, no. 2–3, pp. 73–272, 2017. arXiv:1708.05006.

[6] R. Ji, L. Duan, J. Chen, H. Yao, J. Yuan *et al.,* "Location discriminative vocabulary coding for mobile landmark search," *International Journal of Computer Vision*, vol. 96, pp. 290–314, 2011.

[7] B. Jensen, J. S. Laursen, J. Madsen and T. Pedersen, "Simplifying real-time light source tracking and credible shadow generation for augmented reality using ARToolkit," *Medialogy*, Institute for Media Technology, Aalborg University, 2009.

[8] H. Altinpulluk, "Current trends in augmented reality and forecasts about the future," in *10th Annual Int. Conf. of Education, Research and Innovation*, Seville, Spain, 2017.

[9] Y. B. Kim, "Augmented reality of flexible surface with realistic lighting," in *Proc. of the 5th Int. Conf. on Ubiquitous Information Technologies and Applications*, Sanya, China, pp. 1–5, 2010.

[10] J. Stumpfel, "HDR lighting capture of the sky and sun thesis," Master's thesis, *SIGGRAPH Posters*, California Institute of Technology, 2004.

[11] L. Hosek and A. Wilkie, "An analytic model for full spectral sky-dome radiance," *ACM Transactions on Graphics*, vol. 31, pp. 95, 2012.

[12] Y. Liu, X. Qin, S. Xu, E. Nakamae and Q. Peng, "Light source estimation of outdoor scenes for mixed reality," *Visual Computer*, vol. 25, pp. 637–646, 2009.

[13] T. Naemura, T. Nitta, A. Mimura and H. Harashima, "Virtual shadows in mixed reality environment using flashlight-like devices," *Transactions of the Virtual Reality Society of Japan*, vol. 7, pp. 227–237, 2002.

[14] C. Madsen and M. Nielsen, "Towards probe-less augmented reality—A position paper," in *Proc. of the Third Int. Conf. on Computer Graphics Theory and Applications—Volume 1: GRAPP*, Funchal, Madeira, Portugal, pp. 255–261, 2008.

[15] A. Lauritzen and M. McCool, "Layered variance shadow maps," in *Proc. of the Graphics Interface 2008 Conf.*, Windsor, Ontario, Canada, 2008.

[16] K. Jacobs, J. Nahmias, C. Angus, A. R. Martinez, C. Loscos *et al.,* "Automatic generation of consistent shadows for augmented reality," in *Proc. of the Graphics Interface 2005 Conf.*, Victoria, British Columbia, Canada, 2005.

[17] D. Li, B. Xiao, Y. Hu and R. Yang, "Multiple linear parameters algorithm in illumination consistency," in *Proc. of the 11th World Congress on Intelligent Control and Automation*, Shenyang, China, pp. 5384–5388, 2014.

[18] D. Nowrouzezahrai, S. Geiger, K. Mitchell, R. W. Sumner, W. Jarosz *et al.,* "Light factorization for mixed-frequency shadows in augmented reality," in *10th IEEE Int. Sym. on Mixed and Augmented Reality*, Basel, Switzerland, pp. 173–179, 2011.

[19] A. Marinescu, "Achieving real-time soft shadows using layered variance shadow maps (LVSM) in a real-time strategy (RTS) game," *Studia Universitatis Babeș-Bolyai Informatica*, vol. 56, no. 4, pp. 85–94, 2011.

[20] M. Haller, S. Drab and W. Hartmann, "A real-time shadow approach for an augmented reality application using shadow volumes," in *Proc. of the ACM Sym. on Virtual Reality Software and Technology*, pp. 56–65, 2003.

[21] K. Jacobs, J. Nahmias, C. Angus, A. R. Martinez, C. Loscos *et al.,* "Automatic generation of consistent shadows for augmented reality," in *Proc. of the Graphics Interface 2005 Conf.*, Victoria, British Columbia, Canada, 2005.

[22] C. Loscos, G. Drettakis and L. Robert, "Interactive virtual relighting of real scenes," *IEEE Transactions on Visualization and Computer Graphics*, vol. 6, pp. 289–305, 2000.

[23] M. Aittala, "Inverse lighting and photo-realistic rendering for augmented reality," *Visual Computer*, vol. 26, pp. 669–678, 2010.

[24] D. F. Nehab, A. Maximo, R. S. Lima and H. Hoppe, "GPU-efficient recursive filtering and summed-area tables," *ACM Transactions on Graphics*, vol. 30, pp. 176, 2011.

[25] T. K. de Castro , L. Figueiredo and L. Velho, "Realistic shadows for mobile augmented reality," in *14th Sym. on Virtual and Augmented Reality*, Rio Janiero, Brazil, pp. 36–45, 2012.

[26] I. Arief, S. McCallum and J. Hardeberg, "Real-time estimation of illumination direction for augmented reality on mobile devices," in *Color and Imaging Conf., 20th Color and Imaging Conf. Final Program and Proc.*, Society for Imaging Science and Technology, pp. 111–116, 2012.

[27] Y. G. Wang, F. X. Yan, "Research and implement of improved percentage-closer soft shadows," *Computer Engineering and Applications*, 2011.

[28] W. Sterna, "Variance shadow maps light-bleeding reduction tricks," in *GPU Pro 360 Guide to Shadows*, New York: AK Peters/CRC Press, 2011.

[29] M. Knecht, C. Traxler, O. Mattausch and M. Wimmer, "Reciprocal shading for mixed reality," *Computers & Graphics*, vol. 36, pp. 846–856, 2012.

[30] T. Iachini, Y. Coello, F. Frassinetti and G. Ruggiero, "Body space in social interactions: A comparison of reaching and comfort distance in immersive virtual reality," *PLoS One*, vol. 9, no. 11, pp. 25–27, 2014.

[31] P. Kán and H. Kaufmann, "High-quality reflections, refractions, and caustics in augmented reality and their contribution to visual coherence," in *2012 IEEE Int. Sym. on Mixed and Augmented Reality*, Atlanta, GA, USA, pp. 99–108, 2012.

[32] G. Xing, Y. Liu, X. Qin and Q. Peng, "A practical approach for real-time illumination estimation of outdoor videos," *Computers & Graphics*, vol. 36, pp. 857–865, 2012.

[33] C. Herdtweck and C. Wallraven, "Estimation of the horizon in photographed outdoor scenes by human and machine," *PLoS One*, vol. 8, no. 12, e81462, 2013.

[34] H. Kolivand and M. S. Sunar, "Covering photo-realistic properties of outdoor components with the effects of sky color in mixed reality," *Multimedia Tools and Applications*, vol. 72, pp. 2143–2162, 2013.

[35] H. Kolivand and M. S. Sunar, "Realistic real-time outdoor rendering in augmented reality," *PLoS One*, vol. 9, no. 9, e108334, 2014.

[36] J. Barreira, M. Bessa, L. Barbosa and L. Magalhães, "A context-aware method for authentically simulating outdoors shadows for mobile augmented reality," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, pp. 1223–1231, 2018.

[37] H. Kolivand, A. Rhalibi, M. S. Sunar and T. Saba, "ReVitAge: Realistic virtual heritage taking shadows and sky illumination into account," *Journal of Cultural Heritage*, vol. 32, pp. 166–175, 2018.

[38] H. Wei, Y. Liu, G. Xing, Y. Zhang and W. Huang, "Simulating shadow interactions for outdoor augmented reality with RGBD data," *IEEE Access*, vol. 7, pp. 75292–75304, 2019.

[39] F. Osti, G. M. Santi and G. Caligiana, "Real time shadow mapping for augmented reality photo-realistic rendering," *Applied Sciences*, vol. 9, pp. 2225, 2019.

[40] J. F. Lalonde, A. A. Efros and S. Narasimhan, "Detecting ground shadows in outdoor consumer photographs," in *European Conf. on Computer Vision*, Heraklion, Crete, Greece, 2010.

[41] H. Kolivand and Sunar, "Real-time projection shadow with respect to sun's position in virtual environments," *International Journal of Computer Science Issues*, vol. 8, no. 6, pp. 80–84, 2011.

[42] C. A. Brust, S. Sickert, M. Simon, E. Rodner and J. Denzler, "Efficient convolutional patch networks for scene understanding," in *CVPR Workshop on Scene Understanding*, 2015.