AR360: *Dynamic Illumination for Augmented Reality with Real-Time Interaction*

A'aeshah Alhakamy
Computer and Information Science Department.
Indiana University - Purdue University (IUPUI)
Indianapolis, USA
e-mail: aalhakam@iupui.edu

Mihran Tuceryan
Computer and Information Science Department.
Indiana University - Purdue University (IUPUI)
Indianapolis, USA
e-mail: tuceryan@iu.edu

Abstract—Current augmented and mixed reality systems suffer a lack of correct illumination modeling where the virtual objects render the same lighting condition as the real environment. While we are experiencing astonishing results from the entertainment industry in multiple media forms, the procedure is mostly accomplished offline. The illumination information extracted from the physical scene is used to interactively render the virtual objects which results in a more realistic output in real-time. In this paper, we present a method that detects the physical illumination with dynamic scene, then uses the extracted illumination to render the virtual objects added to the scene. The method has three steps that are assumed to be working concurrently in real-time. The first is the estimation of the direct illumination (incident light) from the physical scene using computer vision techniques through a 360° live-feed camera connected to AR device. The second is the simulation of indirect illumination (reflected light) from the real-world surfaces to virtual objects rendering using region capture of 2D texture from the AR camera view. The third is defining the virtual objects with proper lighting and shadowing characteristics using shader language through multiple passes. Finally, we tested our work with multiple lighting conditions to evaluate the accuracy of results based on the shadow falling from the virtual objects which should be consistent with the shadow falling from the real objects with a reduced performance cost.

Keywords-illumination; augmented reality; image-based lighting; incident light; reflected light.

I. INTRODUCTION

A realistic rendering of an augmented object in the real physical scene would not be achievable without correct illumination data. There are some challenges in obtaining an accurate illumination model which depicts the reality under limited assumptions. The achievement of a photorealistic, interactive and dynamic augmented reality scene acquired multiple factors to integrate the illumination effects seamlessly that present a coherent outcome. In this paper, we introduce a method that uses algorithms to produce a visually cohesive rendering utilizing extracted information about the light from the real scene and apply it to the virtual scene.

The method includes three main parts that should work at the same time in real-time for a result that looks as realistic as the current environment. Our experiment can be used on any AR devices that have an instrumented 360° camera with live-feed feature (e.g., a head-mounted display, handheld mobile device, or webcam camera). The first part of the method is the direction and angle estimation of direct illumination (i.e., incident light) which is the light falling on the objects from the light source directly and then mirroring onto the observer's eyes. In this part, we used a 360° camera with live-feed for capturing the entire physical environment images which are going to be used in light direction estimation.

The second part involves the reflected light also known as indirect illumination where the light is reflected from different surfaces after hitting other object surfaces. This part used 2D texture to capture specific regions from the main AR camera. The regions are captured where the virtual objects are located, then the color and light falling on that region can be depicted on the virtual object using image-based lighting mode.

The third part used shading language and CG techniques on the augmented objects to express the estimated and captured lighting condition and shadowing effects from the previous parts in multiple passes. The main aim of this study is to explore the existing computer vision methods and computer graphics algorithms to accomplish a realistic augmented reality scene.

The final step of this paper is to evaluate our experiment through a comparison with our previous work and estimate the level of progress and accuracy of the final scenes.

II. RELATED WORK

The realism in augmented, virtual, and mixed reality is not a novel subject; the believable perception has been covered in multiple research through different techniques. In this paper, we focus mainly on the augmented reality where some virtual objects are added into the physical scene. Hence, this section will summarise some of the previous and associated work in this area. Although each study has some required assumptions where the system could not be applied without them, there are many other limitations that need to be addressed and developed in the future.

First, we are going to discuss the techniques used to estimate the direction and angle of **direct illumination** (incident light). The dynamic sampling of incident radiance is one of these techniques where the samples are created at the center point of the object. However, the local variation of the lighting on the object can be handled with more accurate and optimized outcomes. The Iterated Closest Point (ICP) algorithm was used at the run-time by Sloan et al. [1] which distributes a set of sample points uniformly over the light source. A hardware solution known as "light probe" which is a mirror sphere placed in the real scene was proposed by

Unger et al. [2] to capture the direct illumination through a sequence of HDR panoramic images using an optical system that included both light reflection and refraction.

Even though finding a perfect reflector surface is not ideal, Rahman et al. [3] applied Fresnel term and Schlick approximation to determine the fraction of light source reflected from a surface. The constant albedo of the surface represented the fraction of the source light in the Lambertian model [4].

A light-based model has a significant impact on several applications utilizing light probes or fish-eye camera. For instance, a full HDR omnidirectional radiance map was acquired close to the location of the virtual object in the real-scene by Debevec et al. [5]. A fish-eye lens which depicted a higher resolution with specular reflections was used in the Supan et al. [6] research. Studiers-tube Tracker was used by Knecht et al. [7] for tracking the lamp and camera location. An environment map with a 180° wide-angle camera was captured by Kamboj et al. [8] to sample the light source using a variance-cut algorithm which finds the brightest area based on minimizing the maximum variance to several divided regions.

The estimation of light direction on a diffuse/glossy object can be accomplished through the Bidirectional Reflection Distribution Function (BRDF) which is a part of the rendering equation.

On the other hand, we also covered the techniques used to obtain **indirect illumination** or simulate the effect of reflected light. The concept of the environment map was also used in this section and achieved a plausible outcome. A region captured from the physical scene was then applied on a virtual object as the main technique of reflected light, where some used cubeMap [1, 9] while others used atlas texels [10].

Franke [11] presented delta Light Propagation Volumes (LPV) a novel volumetric relighting method in mixed reality application that works at real-time which simulate the effect of indirect illumination on the virtual objects onto a real scene and also the reverse.

III. METHOD

We were able to achieve an effective illumination with visual coherence that represented realism in three parts.
For a full overview of the system (see Figure 2).

A. Direct illumination detection

In this part, we focused on finding the light source in the real environment with the assumption that the main light and arbitrary lights could be detected from a 360° live-feed camera. The panoramic view provided a full angle of the physical scene where computer vision tools such as OpenCV could be used to sample the light areas. After using a noise reduction technique, histogram-based thresholding was applied to the captured image to detect the bright saturated areas (see Figure 1). These areas depicted the light sources which represented as contours sorted based on the size, where the greatest area was considered as the main light.

The center of that moment was then assigned to the virtual light location which transferred from Cartesian coordinates (x, y, z) to spherical coordinates (r, θ, φ) using

the inverse spherical projection. The spherical view acted like a skybox in the virtual scene by normalizing each axis which represented the spherical coordinate as follows:

$$Radius(r) = \sqrt{x^2 + y^2} \tag{1}$$

$$\theta = tan^{-1}\left(\frac{y}{x}\right) \tag{2}$$

$$\varphi = cos^{-1}\left(\frac{z}{r}\right) \tag{3}$$

A spherical approximation of a specific radius a modeled to a smaller sphere with radius r is the concept of Mercator projection which cut the sphere into two parallels with latitudes $\pm\varphi$. The panoramic view was also cut into two sides from 0 to 180 west which have the following representation:

$$dw = \frac{360 * x}{w} \tag{4}$$

$$dh = \frac{(360 * 2 - 1) * x}{w + 1} \tag{5}$$

$$x = \frac{sin(\varphi)\cos(\theta)}{\pi * dw} \tag{6}$$

$$y = \frac{cos(\varphi)}{\pi * dh} \tag{7}$$

$$z = \frac{sin(\varphi)\sin(\theta)}{\pi * 90} \tag{8}$$

While the other side had negative values around the x-axis, after finding the angle and location of the light, we required to estimate the light color which is the mean color for each contour. The light intensity is crucial which calculated based on the histogram median to depict the whole environment intensity which reflects the strength of the shadow.
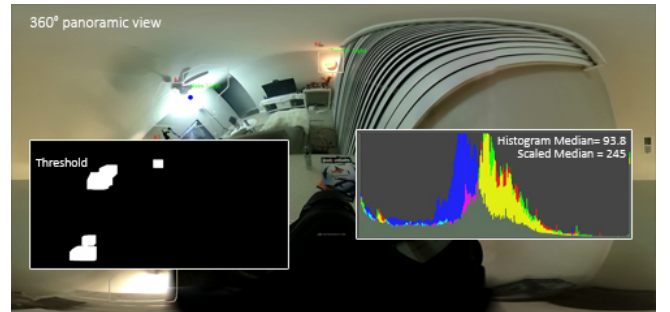

Figure 1. direct illumination detection from a 360° panoramic view using histogram-based thresholding, the line around the light represents the light color.

B. Indirect illumination extraction

This part involved the surrounding environment of the virtual object which is also known as the reflected light. In order to achieve a realistic result towards each object, we captured the regions falling under each virtual element from the main AR camera. Each region had a render camera that captured a 2D texture that would be applied in the next section as part of the image-based lighting model. Coroutine procedure was used to reduce the cost of performance where video background texture was obtained and updated every 0.01f second. Also, the orientation of the AR device was adjusted based on the viewer angle.

Figure 2. system overview which includes the main three part of the whole system: direct illumination estimation, indirect illumination extraction, and shading features, also providing the rendering and interaction procedure.

C. Shading features

The virtual object has numerous features which can be addressed before and during rendering. Shader language with CG/GLSL coding allows identifying the proper material. For complete shading that shows the direct and indirect illumination influence, two types of shader are required.

1) Surface shaders: each virtual object holds three passes: the main light has a forward base pass, any additional

light has forward add pass. These objects are also characterized with several properties which include: main texture, normal map, specular map, vertex/fragment light mode, ambient light, Blinn Phong specular reflection, Fresnel, lambert diffuse reflection, image-based reflection, Ashikhmin, Shirley and Premoze BRDF anisotropy, shadow mode.

2)   Unlit shader: the shadow will fall onto the virtual scene plane, this shader cutout the main color to produce a transparent material which receives the shadows of virtual objects based on the real light source.

D. Rendering

Global illumination is used in the entire AR system to create a more realistic appearance. Forward rendering path technique is the main renderer for each object which used in one or more passes based on how the lights influence the object.

E. Tracking

An image-based marker is employed as an anchor for the initial location using Vuforia Engine. The tool supports several optical and stereo see-through video rendering displays at the installation of VR SDK. However, we write supplementary scripts to support and maintain the different conditions of the light when the virtual object, camera, and marker move.

F. Hardware Description

The method runs on an Intel® Core™ i7-3930k CPU @ 3.20GHz 3201 MHz, six core(s), 64.0 GB RAM, and NVIDIA GeForce GTX 970 GPU. For our result, two cameras were used; for a DSLR Nikon D7200 as main AR camera, and RICOH THETA S 360° with a live feed to capture the environment map.

IV.   EXPERIMENTAL RESULTS

Figure 3. Experiment results, each case has (1) live 360° view, (2) threshold mask of direct illumination detection, (3) final result.

The primary results of our work depicted the direct and indirect illumination instantly. Then, we confirmed the outcome of the method using other lighting conditions such as white light, dim yellow light, sunlight, light coming from a window and others. We also tested our system with different scene setting, i.e., indoor, outdoor, other objects types, or other locations.

By observing the shadow falling from the real objects compared with the shadow of the virtual object, we could promptly evaluate the estimated incident light. The reflected light could observe the influence on the virtual objects when a real object moved, or when real light hit the scene immediately (see Figure 3).

V.   CONCLUSION AND FUTURE WORK

We offered a dynamic illumination method that works in the real-time through estimating the incident light and extracting the reflected light, also, to provide the virtual objects with unique features and properties. Although our system presented a logical interactive outcome, it encountered some challenges that we optimized but that we can improve in future work, such as distinguishing the white area and the light reflected area from the light source using physical-based methods. Also, we can provide a sense of depth and occlusion among the objects utilizing depth map through RGBD sensor.

REFERENCES

[1]   P.-P. Sloan, J. Kautz, and J. Snyder, "Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments." pp. 527-536.

[2]   J. Unger, S. Gustavson, and A. Ynnerman, "Spatially varying image based lighting by light probe sequences," *The Visual Computer,* vol. 23, no. 7, pp. 453-465, 2007.

[3]   S. Rahman, and A. Robles-Kelly, "Estimating Reflectance Parameters, Light Direction, and Shape From a Single Multispectral Image," *IEEE Transactions on Computational Imaging,* vol. 3, no. 4, pp. 837-852, 2017.

[4]   D. Samaras, and D. Metaxas, "Incorporating illumination constraints in deformable models for shape from shading and light direction estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 25, no. 2, pp. 247-264, 2003.

[5]   P. Debevec, "Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography." p. 32.

[6]   P. Supan, I. Stuppacher, and M. Haller, "Image-Based Shadowing in Real-Time Augmented Reality," *IJVR,* vol. 5, no. 3, pp. 1-7, 2006.

[7]   M. Knecht, C. Traxler, O. Mattausch, W. Purgathofer, and M. Wimmer, "Differential instant radiosity for mixed reality." pp. 99-107.

[8]   D. Kamboj, and W. Liu, "Improved Variance Cut Algorithm for Light Source Estimation in Augmented Reality," *International Journal of Computer Applications,* vol. 85, no. 19, 2014.

[9]   L. Gruber, T. Richter-Trummer, and D. Schmalstieg, "Real-time photometric registration from arbitrary geometry." pp. 119-128.

[10]   K. Rohmer, W. Büschel, R. Dachselt, and T. Grosch, "Interactive Near-Field Illumination for Photorealistic Augmented Reality with Varying Materials on Mobile Devices," *IEEE transactions on visualization and computer graphics,* vol. 21, no. 12, pp. 1349-1362, 2015.

[11]   T. A. Franke, "Delta light propagation volumes for mixed reality." pp. 125-132.

Authors' background

Your Name	Position	Research Field	Personal Webpage
A'aeshah Alhakamy	Ph.D. candidate	Computer Science, Imaging & Visualization	https://www.ut.edu.sa/en/web/u12562
Mihran Tuceryan	Professor	Computer Science, Imaging & Visualization	https://www.cs.iupui.edu/~tuceryan/