Probeless and Realistic Mixed Reality Application in Presence of Dynamic Light Sources

Salma Jiddi[*] Philippe Robert[†] Eric Marchand[‡] Anthony Laurent[†] Matthieu Fradet[†]
Pierrick Jouet[†] Caroline Baillard[†]

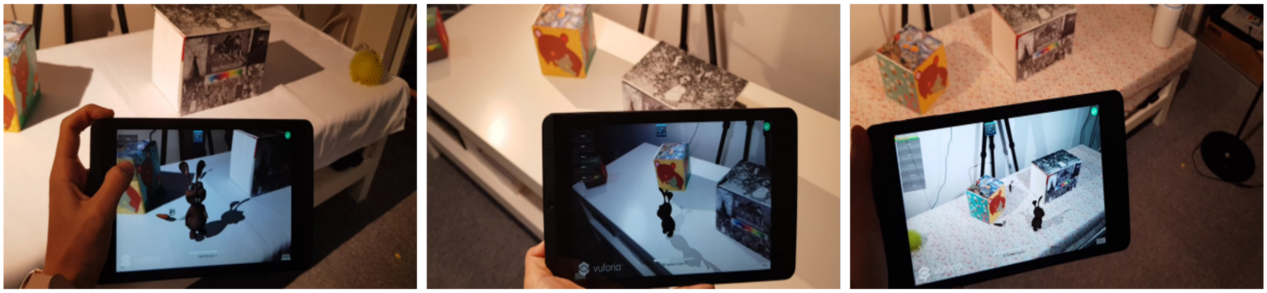

Figure 1: Captures of our Mixed Reality (MR) application for real scenes with a uniform and Lambertian surface (left), a uniform and specular surface (center) and a challenging textured surface (right). The lighting condition can be changed within the MR experience.

Abstract

In this work, we consider the challenge of achieving a coherent blending between real and virtual worlds in the context of a Mixed Reality (MR) scenario. Specifically, we have designed and implemented an interactive demonstrator that shows a realistic MR application without using any light probe. The proposed system takes as input the RGB stream of the real scene, and uses these data to recover both the position and intensity of light sources. The lighting can be static and/or dynamic and the geometry of the scene can be partially altered. Our system is robust in presence of specular effects and handles both uniform and/or textured surfaces.

Index Terms: Scene Analysis, Photometry, Modeling, Shadows, Textures, Lighting, Reflectance, Mixed Reality;

1 Introduction

Mixed Reality applications consist in seamlessly blending the real and the virtual worlds. In fact, MR scenarios aim at creating one single mixed world where both real and computer-generated scenes coexist and interact in a visually coherent way. Subsequently, a user should not be able to distinguish between real and virtual objects.

Although MR technology is already present in the consumer-product market for a number of industrial applications such as training and entertainment, it still lacks *realism* [2]. For instance, in most cases, virtual objects are rendered using an arbitrary and static lighting that does not correspond to the real-world lighting conditions. In order to address this problem, proposed algorithms and systems must recover real-world lighting and surface reflectance properties in order to achieve realistic computer-graphics renderings (e.g., realistic shadows in terms of orientation and intensity, specular reflections, etc.). Furthermore, if the real lighting changes, the system must be able to detect these changes and update, in real-time, the virtual lighting characteristics. In our work, we have designed and implemented a system which analyzes the real scene in order to recover its geometric and photometric properties. Specifically, our main contributions are:

- Without using any light probe, we recover both the 3D position and intensity of light sources present in the scene. The lighting can be static and/or dynamic.
- Our system handles both uniform and challenging textured surfaces.
- The MR application runs in real-time on a tablet and delivers interactive and visually coherent augmentations.

The remainder of this paper is structured as follows: in Section (2.1), we specify the main requirements of our demonstrator, then in Section (2.2) we describe the overall architecture of our system and briefly present its components. Finally, in Section (2.3), we describe the experience which an end-user has as well as the features that our demonstrator offers.

2 Demonstrator Description

2.1 Requirements

The main goal of our demonstrator is to achieve probeless and realistic augmentations of the real scene. The considered scenario is the following (Fig. 2):

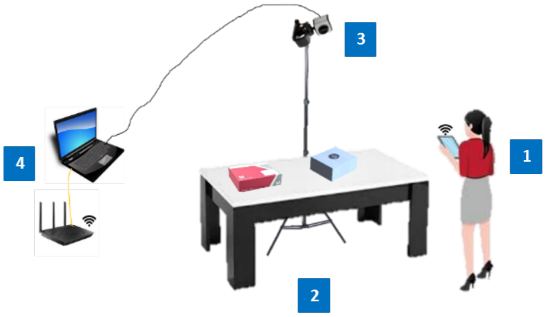

Figure 2: The overall setup of our Mixed Reality demonstrator.

[*]Technicolor / IRISA
[†]Technicolor
[‡]Université de Rennes 1 / IRISA

- (1) An end-user stands in front of a real scene, holding a tablet through which he sees the augmentations.
- (2) The scene is mainly composed of a principal planar surface (table, desk, etc.) and two objects. One of the considered objects can be freely moved within the user's experience whereas the remaining one is fixed.
- (3) In addition to the tablet, a static camera observes the scene from the top. The RGB stream of this camera is used for scene analysis.
- (4) Finally, expensive computations are launched on a laptop that communicates the results using wireless network , in real-time, to the tablet.

2.2 Architecture Overview

In our system requirements (Sec 2.1), we mentioned a shared processing between a PC and an Android tablet. The application running on the tablet is developed using Unity whereas the modules running on the laptop are developed in C++. In Fig. 3, a diagram depicts the overall architecture of the proposed system:

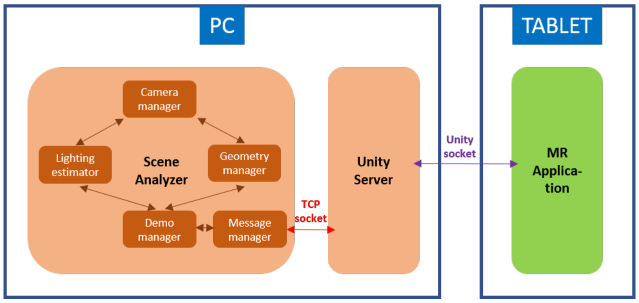

Figure 3: Main components of our MR demonstrator's architecture.

The architecture contains three main components:

- **Scene Analyzer:** it mainly handles lighting and geometry changes in the scene.
- **MR Application:** it is the only module running on the tablet and it handles the rendering of the virtual world using updated lighting and geometry characteristics.
- **Unity Server:** it manages the communication between the *Scene Analyzer* and the *MR application*.

The *Scene Analyzer* contains five different modules:

- **Camera manager** controls the stream of the static camera.
- **Demo manager** updates the changes that might occur in the scene's lighting and/or geometry.
- **Message manager** is the interface between the *Scene Analyzer* and the *Unity Server*. The communication between these modules is achieved via messages delivered over a TCP socket.
- **Geometry manager** aims at detecting and tracking the objects on the principal planar surface. This is achieved using the Vuforia SDK [1]. Changes are detected with regard to the initial position and orientation of the second object and updates are sent to the *Demo manager* module.
- **Lighting estimator** uses the RGB stream provided by the static camera as well as the 3D model recovered using *Geometry manager* in order to estimate both the 3D position and intensity of light sources. The processing is achieved for each captured image as follows: first, we detect cast shadows on the principal planar surface [4]. Then, we correlate the image of detected shadows with a set of precomputed shadow maps. Recovered light sources correspond to shadow maps with high correlation values (Fig. 4). Finally, we recover the intensity of the selected light sources using a Least Squares estimator [3].

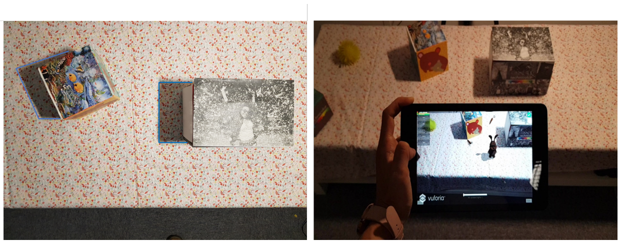

Figure 4: **Left:** Superimposed contours (blue) of the matched shadows (viewed from static camera). **Right:** User view of the augmented scene through the tablet.

2.3 Experience

An end-user stands in front of a real scene holding a tablet through which the MR scenario can be viewed. Our demonstrator runs in real-time and takes into account changes that can occur in lighting (lights switched on/off or moved) and geometry (moving an object on the planar surface). For instance, using a remote control, the user can switch on/off the lights in the scene and see the changes occur on the virtual objects as well. Furthermore, the user can interact with the virtual object in two different ways: (1) by moving its finger on the tablet screen, he changes the position of the virtual object on the planar surface. (2) several animations are accessible via the UI-toggles located on the left side of the tablet screen. The *Amount of realism* achieved by our demonstrator [2] is convincing. For instance, the orientation and intensity of virtual shadows is consistent with real cast shadows. Also, real lighting conditions are automatically detected and used to update the virtual lighting model. Our demonstrator is able to detect up to three light sources and handles, to some extent, windows lighting by recovering it as a distant point light. Further results are available at: *https://youtu.be/sENETegDHnQ*

3 Conclusion

We presented an interactive system that demonstrates realistic augmentations. For future work, we are interested in considering more complex 3D scene models. In fact, the analysis can be achieved using a 3D sensor as in [4] instead of the fixed RGB camera. A continuous RGB-D stream can be integrated in our approach to automatically detect 3D objects on the principal plane without using any markers. Finally, our current lighting estimator recovers light sources as point lights which produce hard shadows. For future work, we aim at extending our light model (e.g., area lights) in order to render virtual shadows that are as coherent as possible with the real shadows.

References

[1] Vuforia, https://www.vuforia.com/.

[2] K. Jacos, C. Loscos, C. Traxler, and M. Wimmer. Classification of illumination methods for mixed reality. In *Computer Graphics Forum (CGF)*, 2006.

[3] S. Jiddi, P. Robert, and E. Marchand. Illumination estimation using cast shadows for realistic augmented reality applications. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2017.

[4] S. Jiddi, P. Robert, and E. Marchand. Estimation of position and intensity of dynamic light sources using cast shadows on textured real surfaces. In *IEEE International Conference on Image Processing (ICIP)*, 2018.