Real-time Underwater Caustics for Mixed Reality 360° Videos

Stephen Thompson Andrew Chalmers Taehyun Rhee*

Computational Media Innovation Centre, Victoria University of Wellington, NZ

Abstract

We present a novel mixed reality (MR) rendering solution that illuminates and blends virtual objects into underwater 360° video with real-time underwater caustic effects. Image-based lighting is used in conjunction with underwater caustics to provide automatic ambient and high frequency underwater lighting. This ensures that the caustics and virtual objects are lit and blend into each frame of the video semi-automatically and in real-time. We provide an interactive interface with intuitive parameter controls to fine tune caustics to match with the background video.

Index Terms: Computing methodologies—Graphics systems and interfaces—Mixed / augmented reality

1 Introduction

360° cameras have advanced such that it is possible to film in underwater environments. While high-fidelity blending of virtual objects with 360° video (360-video) has improved in recent research [4], compositing into underwater footage needs to address the additional challenges posed by the complex lighting that occurs in water. One such lighting effect is caustics, the patterns of light created when light refracts through the water surface. Underwater caustics has not been addressed in recent solutions to 360° MR rendering [4], resulting in unrealistic blending of virtual objects into underwater 360-video. Furthermore, rendering caustics in computer graphics is expensive, posing the additional challenge of producing a high frame rate suitable for modern HMDs (e.g., 90FPS per eye).

We present a novel method to enhance the visual quality of underwater MR in 360-video with blended underwater caustics. Our underwater caustics are derived from real-time water surface simulation. This setup provides intuitive parameter controls to create a real-time caustic map for each frame. Our interface is easy to use, semi-automatic, allows for iterative parameter refinement, and does not require pre-computation. Once setup, the underwater lighting effects will be automatically updated by the captured light in every frame of the 360-video. Our solution is integrated into commercial game engines (e.g., Unity and Unreal Engine 4 (UE4)) to provide a convenient tool for creating underwater MR contents using 360-videos.

Our main contributions are summarized as follows:

- We developed a novel method for real-time caustics suitable for underwater MR rendering with 360-video. Our method provides believable caustics at high frame rates (over 120FPS per eye), suitable for modern HMDs.
- We provide a tool to blend virtual objects and water caustics into underwater 360-video semi-automatically. The interface provides intuitive parameter controls integrated into commercial game engines.

*e-mail: taehyun.rhee@ecs.vuw.ac.nz

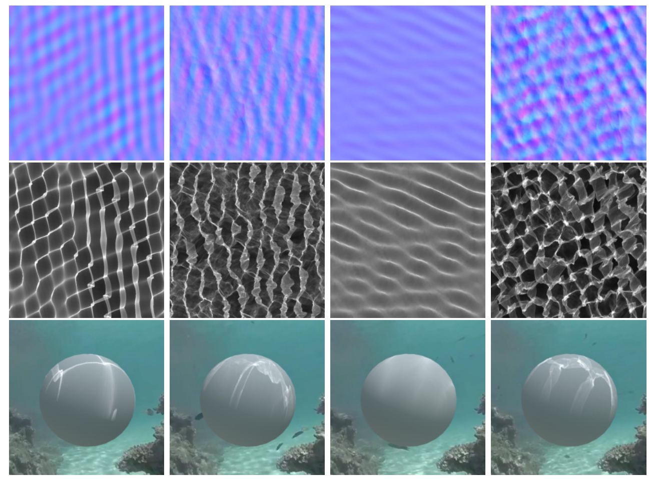

Figure 1: Water surface normal maps with corresponding caustic maps and virtual objects rendered with caustics.

2 Lighting and Composition

Previous work by Rhee et al. [4] achieved seamless blending of virtual objects into 360-video by simulating ambient lighting with image based lighting (IBL), detecting the light sources in the video to provide high frequency lighting, and composition with shadows using differential rendering. We adapt this setup for the general lighting and composition, and enhance it for underwater MR with caustics. We assume the main light source underwater is the sun and is located in the top hemisphere of the video. We use the work of Iorns et al. [3] to apply inverse tone mapping to convert low dynamic range (LDR) to high dynamic range (HDR) videos for IBL. To reduce flickering in the IBL caused by caustics on the ocean floor becoming over exposed, we apply the inverse tone mapping to the top half of the video, where we assumed the main light source is located.

3 Real-time Underwater Caustics

Water Surface Simulation: Rendering caustics requires a surface to scatter the light and generate the caustic patterns. We simulate a virtual water surface to avoid the difficulty of trying to detect the water surface with environmental factors (i.e. refraction, reflections, over-exposure) present in the video. Two common methods for real-time water surface simulation are Gerstner [2] or FFT [6] waves. We chose Gerstner waves because it is efficient to calculate and easy to control individual waves. Each Gerstner wave is calculated as a summation of several individual waves. This calculation is defined as:

$$P(x,y,t) = \begin{bmatrix} x + \sum(Q_i A_i \times D_i.x \times \cos(w_i D_i \cdot (x,y) + \varphi_i t)) \\ y + \sum(Q_i A_i \times D_i.y \times \cos(w_i D_i \cdot (x,y) + \varphi_i t)) \\ \sum(A_i \sin(w_i D_i \cdot (x,y) + \varphi_i t)) \end{bmatrix} \quad (1)$$

where Q, D, w, A and φ represents the individual wave sharpness, direction, wavelength, amplitude and period respectively. The inputs are the horizontal position on the plane (x and y) and time (t). We define medium to large waves using Gerstner waves and apply two scrolling normal maps to add small wave detail, such as those caused by wind or surface tension.

Ideally, the water surface should tile seamlessly in order for the corresponding caustics to repeat correctly. Unless the wave parameters are carefully chosen, Gerstner waves do not tile seamlessly. We seamlessly tile the water surface by up-scaling the texture to only use 90% of the pixels. The unused pixels are wrapped to the opposite side of the texture and blended using linear interpolation. This produces seamless results as long as the wavelengths are not too large compared to the size of the texture.

Caustic Map: We base our caustics rendering on the method detailed by Shah et. al. [5]. Using a 100x100 subdivided plane, we treat each vertex as a ray, projecting and refracting it through the water surface onto a plane underneath. The intensity of the resulting triangles can be calculated as the ratio between the original triangle area and the projected triangle area, additively blending overlapping triangles together to create the final caustics. Calculating the intensity per triangle will produce a blocky appearance, so we generate a triangle fan around each vertex of the projected mesh [1], treating each triangle in the triangle fan as a caustic triangle and averaging the intensities to get the total intensity for the vertex. We calculate the light intensity per pixel instead of per vertex to produce smooth caustics while maintaining some detail from the normal maps. The resulting caustics are rendered to a texture (caustic map) at the start of each frame.

We also take the light direction within the 360-video into account when generating the caustics. We consider the water surface as a plane and refract the detected sun light direction through it to approximate the direction of the sun above the water surface. The light direction above the water surface is used for projecting light rays through the water surface to create a closer match between our caustics and the caustics in the 360-video.

Caustics: Caustics are rendered on the virtual objects by projecting the caustic map onto the objects using the detected light direction to determine the uv coordinates. The uv position is calculated as:

$$uv = (W.x, W.z) \times S - W.y \times (L.x, L.z) \tag{2}$$

with W, S and L representing the world position, world to uv scaler and light direction respectively. The caustic map is multiplied with the detected light color so that the caustic pattern is used rather than a solid light color.

4 Results

4.1 MR Caustics and Interface

The results of real-time underwater caustics after each stage of the pipeline described in section 3 is shown in Figure 1. For MR content creation, we developed a plugin for game engines that provides an interface to semi-automatically blend virtual objects into underwater 360-videos (Figure 2). The parameters of our interface are separated in two: The general lighting and composition, and caustics. Because the lighting is computed in real-time, users can interactively adjust parameters while observing the changes in the game engine's scene view. The water surface used for generating caustics is made from multiple waves. For easy user control, we provide a set of global parameters rather than individual parameters. During video playback, the user can rapidly adjust the underwater lighting to match the background video. The water surface and corresponding caustic map is viewed through the game engine's scene view, making it easier to match with the 360-video. Once the parameters are set, the lighting conditions will automatically update with each frame.

4.2 Performance Evaluation

We tested the performance on a computer using an Nvidia GeForce GTX 1070 graphics card, Intel Xeon Processor E5-1607 v2 with a

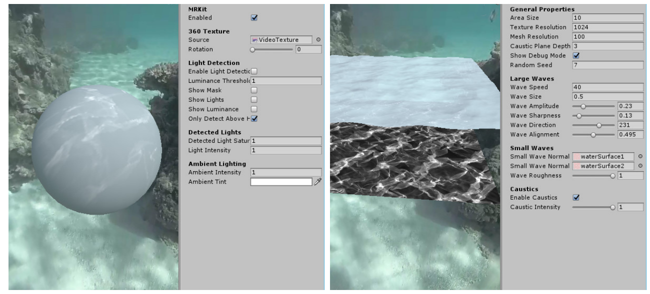

Figure 2: User interface for setting the parameters.

3GHz clock speed and 16GB ram. Generating the 64x32 resolution textures required for IBL (the same as MR360 [4]) took 0.2ms. The caustic map took approximately 0.3ms to render at a resolution of 512x512 pixels, a resolution chosen to be large enough to capture detail while reducing memory usage. Because most of the complex lighting is stored in textures, rendering the final image only requires texture look-ups. We are able to achieve less than 3ms rendering time for scenes using a 4k video and containing four 3D models with 14k triangles each. Increasing from four to twenty models still took less than 5ms to render.

5 Conclusion and Future Work

This paper presents a novel method to illuminate and blend virtual objects into 360-video with real-time underwater caustics. Our underwater caustics are based on caustic maps derived from real-time water surface simulation providing intuitive parameter controls to match the light in underwater 360-videos.

As we know, this is the first solution for real-time mixed reality caustics and blending in underwater 360-video. However, there are a few limitations to be improved in the future work. We assumed the camera is horizontally aligned with the water surface, which could be addressed using image stabilization as a pre-processing step. We also assumed a main directional light (sun) for IBL. Positional lights, such as torches held by nearby divers, will have subtle differences. Future work could address this by including depth and light position detection. Furthermore, experimenting with other underwater effects, such as particles and god rays, could be considered for improving overall blending details.

Acknowledgments

This project was supported by the Entrepreneurial University Programme funded by TEC and in part by the Smart Ideas project funded by MBIE in New Zealand. We thank Boxfish Research for providing 360-videos captured with their underwater 360° camera.

References

[1] M. Ernst, T. Akenine-Möller, and H. W. Jensen. Interactive rendering of caustics using interpolated warped volumes. In *Proceedings of Graphics Interface 2005*, pp. 87–96. Canadian Human-Computer Communications Society, 2005.

[2] A. Fournier and W. T. Reeves. A simple model of ocean waves. *ACM Siggraph Computer Graphics*, 20(4):75–84, 1986.

[3] T. Iorns and T. Rhee. Real-time image based lighting for 360-degree panoramic video. In *Pacific-Rim Symposium on Image and Video Technology*, pp. 139–151. Springer, 2015.

[4] T. Rhee, L. Petikam, B. Allen, and A. Chalmers. Mr360: Mixed reality rendering for 360 panoramic videos. *IEEE Transactions on Visualization & Computer Graphics*, pp. 1379–1388, 2017.

[5] M. A. Shah, J. Konttinen, and S. Pattanaik. Caustics mapping: An image-space technique for real-time caustics. *IEEE Transactions on Visualization and Computer Graphics*, 13(2), 2007.

[6] J. Tessendorf et al. Simulating ocean water. *Simulating nature: realistic and interactive techniques. SIGGRAPH*, 1(2):5, 2001.