A New Raw Holoscopic Image Simulator and Data Generation

Bodor Almatrouk [1], Hongying Meng [1], Akuha Aondoakaa[2], and Rafiq Swash[1]
[1]Dept. of Electronic and Electrical Engineering, Brunel University London, Uxbridge, United Kingdom
{bodor.almatrouk, hongying.meng, rafiq.swash}@brunel.ac.uk
[2] sa.akuha@gmail.com

***Abstract*—A holoscopic imaging system is a method for capturing three-dimensional information with a single camera and aperture, using a micro-lens array between the camera's main lens and image sensor. Due to the lack of holoscopic cameras on the market, researchers need a holoscopic simulator and a dataset to work with. Most of the available holoscopic simulators and datasets generate up-sampled viewpoint images, which are extracted from the holoscopic raw image. This paper proposes a new holoscopic simulator that generates a raw holoscopic image with variable resolution more efficiently. The proposed holoscopic simulator captures a 3D model, generates viewpoint images with a proper baseline separation using a Blender add-on function, and uses a multi-view stereo algorithm to generate ground-truth depth maps. To generate a raw holoscopic image, viewpoint image pixels are remapped to their associated micro-lenses, producing two forms of holoscopic images, a grid of viewpoint images, and a grid of elemental images. The holoscopic simulator is fast and efficient compared to existing holoscopic simulators. A comprehensive holoscopic dataset is also presented, including raw holoscopic 1.0 images and associated viewpoint images at multiple resolutions, as well as a holoscopic toolbox for viewpoint images and elemental images. The proposed holoscopic simulator and dataset offer a valuable resource for researchers interested in holoscopic imaging and provide a new approach to generating holoscopic images and associated data, potentially facilitating research in related areas and advancing the field of holoscopic imaging.**

***Index Terms*—Holoscopic, elemental image, viewpoint image, dataset**

I. Introduction

Holoscopic imaging, also known as integral imaging, uses a single camera and aperture to capture 3D data. A micro-lens array (MLA) between the camera's main lens and photo sensor creates a two-plane camera model. As shown in Fig. 1 (a), conventional 2D cameras use a single-plane camera model to converge all viewpoint images of the scene at the camera sensor (image plane). Holoscopic cameras place the MLA on the image plane and are focused on the sensor. The first plane, the MLA, would take viewpoint images (VPIs). Instead of converging them into one image, the micro-lenses are used to break down the viewpoint images (VPIs) light rays into their respective directions in the second plane. Fig. 1 (b) shows the output image as an array of micro-images formed in the second plane behind the MLA. These micro-images are known as elemental images (EIs). VPIs are orthographic images where EIs are perspective images. Since it stores 2D EIs and 2D VPIs, a holoscopic camera can record a 4D light field. Recording 4D data. With the emergence of holoscopic cameras, lenses, and displays, as well as technologies like virtual reality (VR), and augmented reality (AR), which leverage holoscopic technology for a more realistic experience, holoscopic has acquired a lot of attention. Holoscopic technology, unlike Stereoscopic, is underdeveloped. Researchers face a shortage of holoscopic cameras, only Lytro, which is no longer available, and Raytrix, which is designed for industrial use and is expensive are available. Thus, researchers need an inexpensive, accessible holoscopic image simulator and dataset to experiment with different configurations to build prototypes and use ground truth data for exploration, benchmarking, and evaluation.

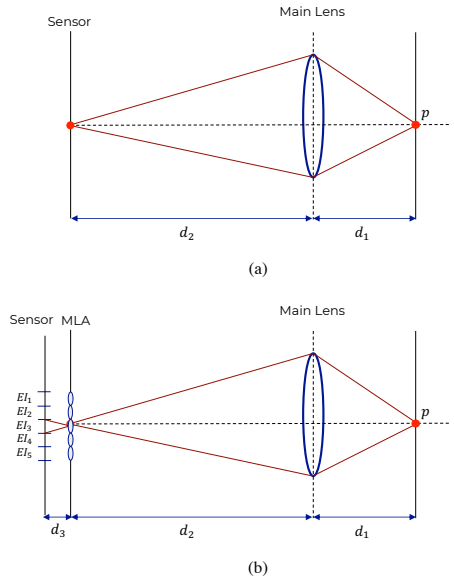

Fig. 1. (a) Conventional camera model where the sensor is placed on the image plane. (b) Holoscopic camera model where the MLA is placed on the image plane and focused on the image sensor

II. Related Work

Holoscopic images can be rendered using direct rendering or orthogonal VPI rendering followed by reverse engineering, with the latter being more common due to its simplicity. Holoscopic 1.0 and Holoscopic 2.0 are the two main Holoscopic image designs. Early Holoscopic rendering techniques generate a 2D array of viewpoint images. Levoy et al. [1]

This article has been accepted for publication in a future proceedings of this conference, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/ICIVC58118.2023.10270247, 2023 8th International Conference on Image, Vision and Computing (ICIVC)

introduced this method, which uses a tool that works with SGI's GL library and is written in C and C++. Swash et al. [2] introduced a more camera-realistic simulator that allows for the simulation of various camera parameters such as sensor size, focal length, and more. Two different ways to render the scene were produced. One camera module uses Rhino 3D models and the POV-Ray system clock to render all viewpoint images with a single click, while the other constructs a virtual lens array for a fly's eye in the virtual environment using constructive solid geometry and is rendered using an orthographic projection camera. Recent work uses Blender to configure and render high-resolution VPIs with depth maps [3, 4]. These simulators were designed to provide results that mimic high-resolution VPIs from Lytro cameras, but they were not developed for generating raw Holoscopic images due to the lengthy rendering time. There are many Holoscopic datasets available for various applications. Most of these datasets were captured using Lytro cameras, including the multi-view Holoscopic dataset captured with multiple Lytro Illum cameras for investigating multi-view Holoscopic processing [5]. Other Lytro datasets include focal stacks, such as the dataset presented in [6], and the multi-view object recognition benchmark dataset designed for Holoscopic image analysis [7]. There are also datasets captured with Raytrix cameras, such as the dataset introduced in [8] for blur-aware calibration and the dataset presented in [9] for researching Holoscopic properties and comparing Lytro and Raytrix cameras. Synthetic Holoscopic images can be generated using platforms such as PovRay and Blender [10, 11, 12].

This paper describes a synthetic holoscopic image simulator. The 4D plenoptic function underpins this simulator. Instead of a hexagonal grid, a holoscopic camera had a rectangular grid of micro-lenses for simplicity. This simulator also serves to lessen the simulation duration. Holoscopic technology has many challenges, including a trade-off between micro-images EIs and viewpoint image resolution. For high-resolution VPIs and ELs, the ideal holoscopic camera would have a large array of micro-lenses (MLA) and large MLs. Camera sensor sizes prevent such a design. The camera would be too bulky such as Stanford University's [13]. Here is a dataset with multiple EIs and VPIs values of the same images for trade-off experiments and identifying the best research topic solution.

III. Raw Holoscopic Image Simulator

This section outlines the procedure for creating the simulator to generate the holoscopic image as shown in Fig. 2. A 3D model is loaded as the first step, and viewpoint images are generated with a proper baseline separation (camera shift) using the simulator, which we created as a Blender add-on function. Ground-truth depth maps are then generated using the viewpoint images. Finally, the holoscopic image and its associated 5D matrix are generated. To generate the raw holoscopic image, pixels of viewpoint images are re-sampled into elemental images based on the holoscopic 5D plenoptic function and holoscopic design. Details of each step are presented in the following subsections.

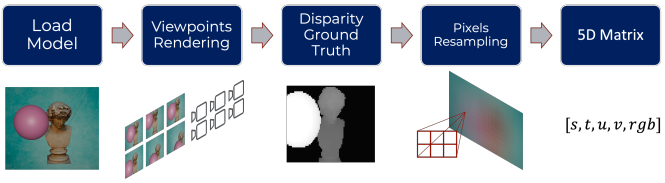

Fig. 2. Overall structure of the holoscopic simulator: First, the model is loaded into the simulator, then the viewpoint images and the ground truth depth map are generated using a plenoptic 4D function, and then the pixels are re-sampled. Produced in the five-dimensional matrix

A. Viewpoint Generation

When capturing a holoscopic image from a camera, a grid of EIs is formed, and VPs, which are the images from multiple perspectives captured by the main lens (first plane), are then extracted. In this simulator, a number of two-dimensional viewpoint images must be rendered first to create a holoscopic image by reverse engineering. The resolution of viewpoint images VPI is the function of the final holoscopic image size, the number of micro-lenses, and the sensor resolution s:

$$I_x = \frac{s_w}{M_x},\ I_y = \frac{s_h}{M_y} \tag{1}$$

where I_x is the VPI width and I_h is the VPI height, M is the number of micro-lenses in x and y directions. s is the sensor resolution where its height is h and width is w.The generation process starts with centring the camera on the intended central viewpoint image view of the 3D model as seen in Fig. 3. Further, it involves selecting the number of elemental images, dimensions of the viewpoint image, focal length and the baseline between each viewpoint image.

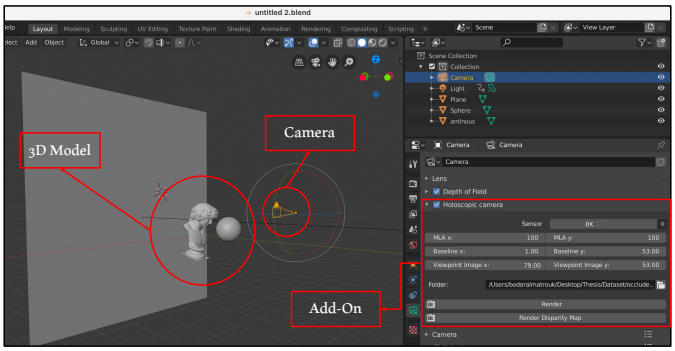

Fig. 3. Place the object in the frame and DOF. Users can choose a predefined camera sensor size (HD, 4K, 8K) and viewpoint image dimensions. Automatically generate elemental images and baseline. User-adjustable parameters change with sensor size. The values can be changed freely if the sensor is "Other".

The process generates multiple images from different angles. Simulator rendering time is crucial. This simulator generates a large number of low-resolution viewpoint images to reconstruct a raw holoscopic image using a large array of cameras, unlike other holoscopic simulators that generate a small number of high-resolution VPIs, which is unrealistic due to the sensor size of holoscopic cameras. A modal operator

This article has been accepted for publication in a future proceedings of this conference, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/ICIVC58118.2023.10270247, 2023 8th International Conference on Image, Vision and Computing (ICIVC)

(background operator) checks process completion at regular intervals over a few seconds. The modal operator is useful but slows to process. Thus, a custom simulator without modal operators reduced processing time. A non-modal while loop with a simple check was used to verify process completion. Instead of attaching pre and post-functions after rendering each frame, we continuously check process completion. These functions will stack up and cause exponential inefficiency if the operator is not completed or cancelled while rendering. Instead of attaching and detaching, the loop calls these functions per frame. We have tested both our simulator and the simulator by Honauer, K., et al. [3] to render the same 3D model of 10,000 viewpoint images, each is 79 x 53 in resolution to generate almost an 8K holoscopic image, and as can be seen in Fig. 4, our simulator is 5 times faster with linear speed. A sample of 3x3 of the VPIs can be seen in Fig. 5.

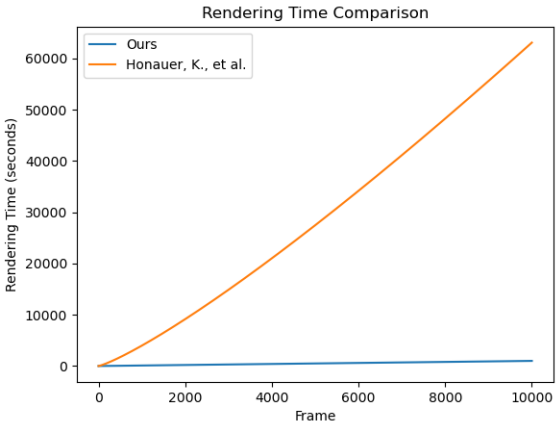

Fig. 4. Comparison between the performance of our simulator and the simulator by Honauer, K., et al. [3]. We are plotting the rendering time it takes to render the frames and as can be seen, our simulator is 5 times faster with linear speed

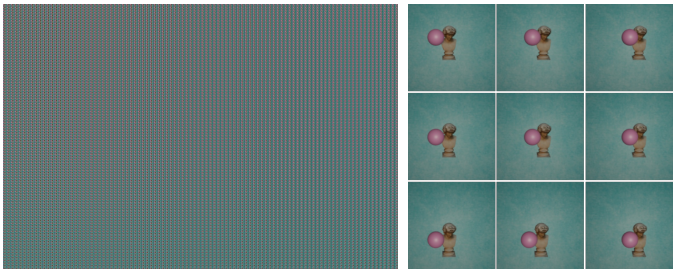

Fig. 5. The resulting image is a grid of VPIs (Left) that can be rearranged later to a raw holoscopic image (The images are not visible since there is a large grid of low resolution (79x53) viewpoint images). Right: sample of nine viewpoint images captured from different perspectives that will be re-sampled into holoscopic image

B. Ground Truth Depth Map

Since Blender is being used to generate the holoscopic images, the ground truth disparity map is computed using Blender's internal simulator. For each pixel of the VPI, Blender's internal engine is able to compute the distance between the camera pixel and the colliding object using orthographic projection. This distance is the true depth of the image. Thus, creating a true depth map. Since depth Z, baseline b, and focal length f are known when using Blender, disparity d can be computed using triangulation:

$$d = \frac{bf}{Z} \tag{2}$$

A disparity map, however, must be calculated relative to the resolution of the image sensor, as detailed in [14]:

$$d = bfr\left(\frac{1}{Zs} - \frac{1}{f_t s}\right) \tag{3}$$

where s is the sensor size, d is the focus distance and r is the resolution of the image in x, f_t is the focal distance and y direction. The ground truth disparity can be seen in Fig. 6.

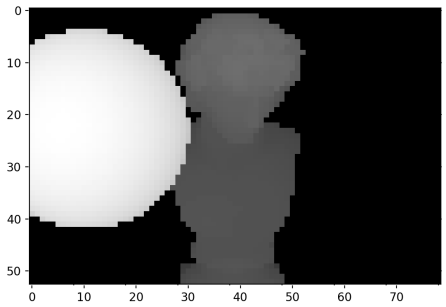

Fig. 6. Simulated ground truth disparity. The viewpoint image generates a low-resolution disparity. Brighter pixels indicate closer objects.

This step is crucial to generate a ground truth depth map for evaluation.

C. Resampling to Create Holoscopic Image

While shooting in low light, cameras are less likely to capture accurate colours and clear details because they capture light reflected off objects. A hypothetical function called the plenoptic function [15] captures each light through space at any time, recording every possible perspective from every possible location at every possible instant.

$P(\theta, \phi, x, y, z, \lambda, t)$ captures the light ray's angular direction, position, time, and wavelength when it enters the camera. Since captures are statics and RGB colour channels replace wavelength, this function can be reduced to 4D by ignoring t and λ. The 5D function's redundant information in an unobstructed space can also be ignored since the beam's radiance does not change unless it is obstructed.

As shown in Fig. 7 (b), a holoscopic camera uses a two-plane system to capture light rays from all angles. The camera's main lens, the first plane, captures light rays' VPIs. The holoscopic image's VPIs at (s, t) are formed on this plane. MLA divides convergence points into rays and records their

This article has been accepted for publication in a future proceedings of this conference, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/ICIVC58118.2023.10270247, 2023 8th International Conference on Image, Vision and Computing (ICIVC)

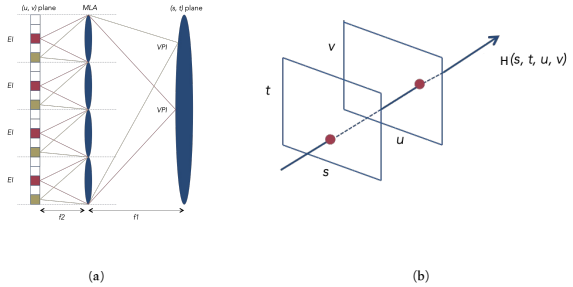

Fig. 7. Plenoptic light rays. (a) The plenoptic function should give the radiant intensity at (x,y,z) and direction (θ,ϕ). (b) Holoscopic two-plane model with elemental image plane (u, v) and viewpoint image plane (s, t).

angular information. Thus, elemental images EIs [16] form behind each micro-lens at coordinates (u, v).

Fig. 7 (a) shows that each MLA convergence point has multiple rays of light, each of which represents a unique VPI. Moreover, all the micro-lenses receive light from a single VPI at precisely the same point. Therefore, using a raw holoscopic image, pixels from the same location in an EI can be grouped together to generate a VPI. This allows for resampling the pixels in the EIs so that they are represented as VPIs. This process can be reversed. As shown in equation 4, this holoscopic image simulator re-samples VPI pixels to EIs. The raw holoscopic image rendered using this simulator can be seen in Fig. 8.

$$EI(u, v) = \sum_{s=1}^{s} \sum_{t=1}^{t} VPI(s_u, t_v) \tag{4}$$

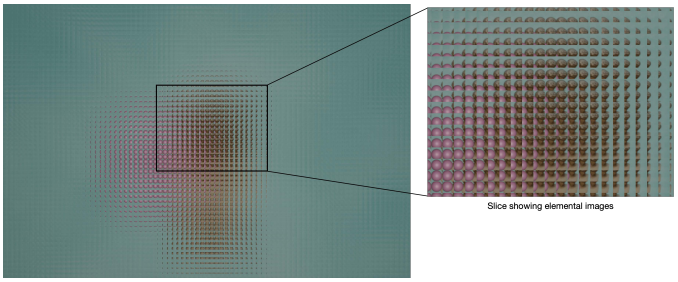

Fig. 8. The generated holoscopic image consisting of 74x53 elemental images that are 100x100 each

where EI at coordinate (u, v) is equal to the pixel at the location equal to EI coordinate (s_u, t_v) of VPI at coordinate (s, t). Since each EI comprises a single pixel of each VPI, for a given VPI, the number of its pixels is equal to the amount EIs in the holoscopic image. The higher the spatial resolution of the VPIs, the greater the number of EIs. Nevertheless, the image resolution will decrease as the number of EIs increases. Thus, there is a compromise between angular and spatial resolution when recording an holoscopic image [17].

D. Holoscopic 5D Data Structure

Holoscopic images comprise a large number of EIs and VPIs. To easily access and manipulate specific EIs and VPIs, given a calibrated holoscopic image, an equation is produced that loops through the entire holoscopic image pixels to generate a 5D parameterisation holoscopic function matrix such as:

$$\begin{aligned} H &= (u, v, s_1 : s_n, t_1 : t_m) \\ s &= (u - 1) * n + 1 \\ s_n &= (u * n) \\ t &= (v - 1) * m + 1 \\ t_m &= v * m \end{aligned} \tag{5}$$

where (u, v) are the positions of EIs in the u, v plane, $n$and m are the number of pixels in each EI and (s, t) are the position of VPIs in the s, t. An extra dimension has been added to the function, consisting of RGB colour channels generating a final 5D function.

IV. Generated Raw Holoscopic Image Dataset

Since holoscopic imaging has been getting a lot of attention lately because of its capability to capture full angular information and its simple design, the availability of sufficient holoscopic image datasets is vital for researchers. The resolution of each VPI versus the number of EIs poses a trade-off in holoscopic technology. Using the holoscopic simulator, a characterised and accessible holoscopic image dataset is presented as seen in Fig. 10, which consists of image scenes rendered with varying EIs and VPIs resolutions, allowing researchers to experiment with the trade-off and determine the optimal approach for their specific areas of research.

The dataset consists of 15 holoscopic images as seen in Fig. 10, all of which are produced with an 8k sensor and an f5 lens. Five different EIs resolutions are used to construct each scene: 20×20, 40×40, 60×60, 80×80, and 100×100, as seen in Fig. 9. The table I lists the configurations of the dataset.

TABLE I
EACH HOLOSCOPIC IMAGE THAT IS 8K IN SIZE IS GENERATED WITH THESE VPIS AND EIS SIZE SPLIT. THE NUMBER OF THE ELEMENTAL IMAGES IN THE X AND Y DIRECTIONS IS EQUAL TO THE RESOLUTION OF THE VIEWPOINT IMAGES

Elemental Images Resolution	Viewpoint Image Resolution
20x20	397x265
40x40	198x132
60x60	132x88
80x80	99x66
100x100	79x53

The generated scenes are simplistic, yet they contain a broad variety of challenging features. These features include but are not limited to, textured backgrounds, reflections, dense settings, glass, shadows, and other similar elements. Therefore, they can be applied to a wide variety of problems and algorithms.

Within the dataset folder is a ”src” folder, which contains Python scripts that carry out activities such as clipping VPIs and ELs, constructing a holoscopic image from VPIs, and

This article has been accepted for publication in a future proceedings of this conference, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/ICIVC58118.2023.10270247, 2023 8th International Conference on Image, Vision and Computing (ICIVC)

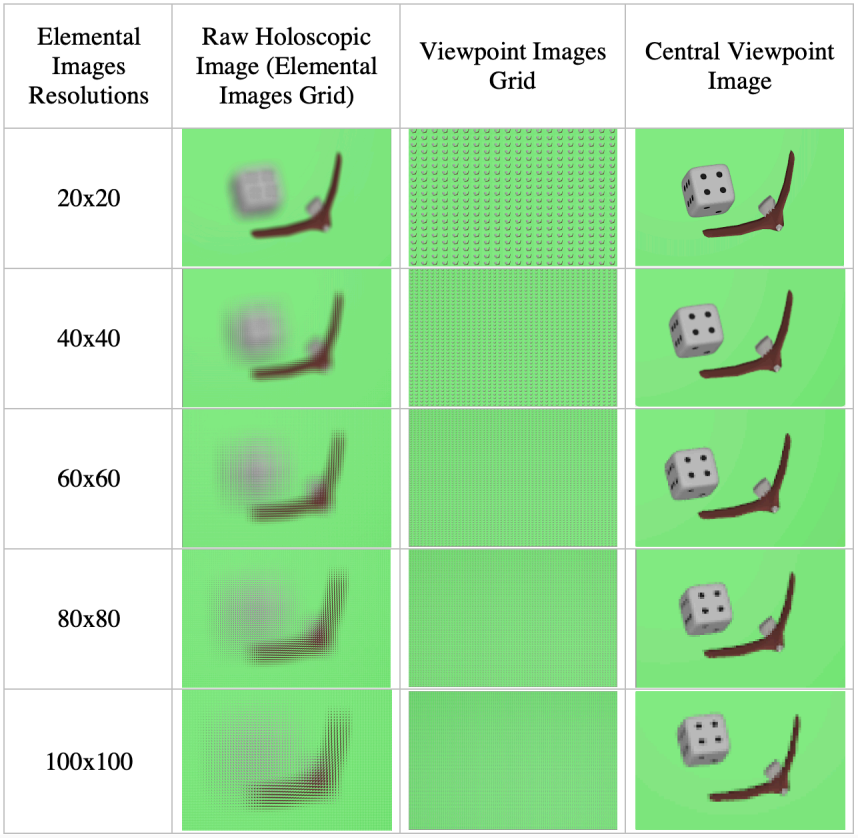

Fig. 9. Holoscopic images of different scales with the associated viewpoints images grid and central viewpoint image

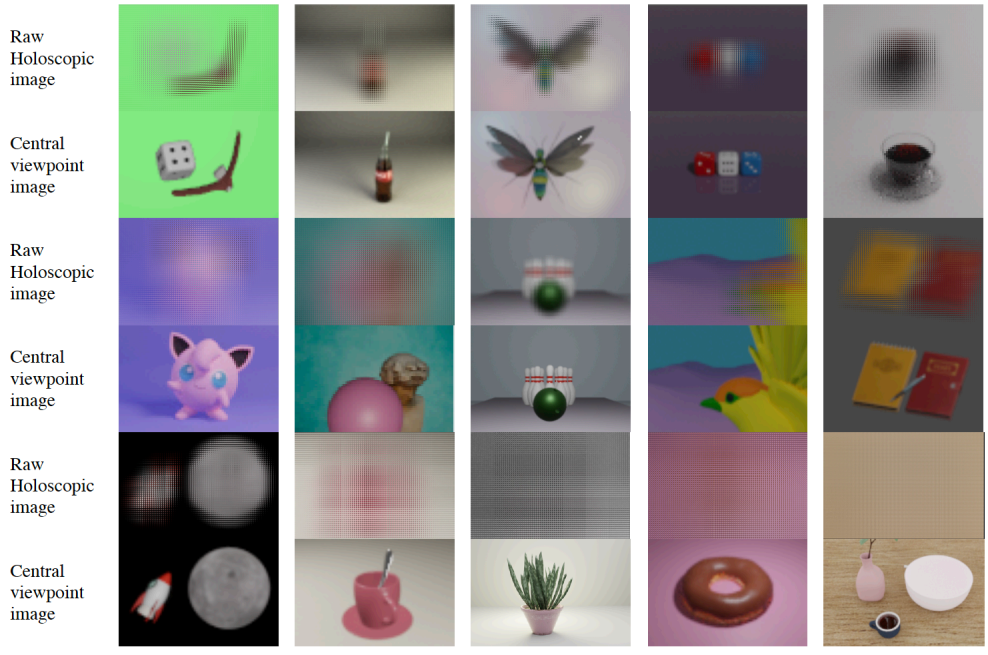

Fig. 10. Our holoscopic image dataset includes textured environments, reflections, dense settings, glass, shadows, and so on. Here is a sample of 15 images of the raw holoscopic images and their central VPI of the datasets.

This article has been accepted for publication in a future proceedings of this conference, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/ICIVC58118.2023.10270247, 2023 8th International Conference on Image, Vision and Computing (ICIVC)

extracting VPIs from a holoscopic image [1]. For each scene of the dataset, the structure of the folder is as follows:

Scene folder
- 20x20
  - scene_vpi.png - a grid of viewpoint images of size 8k
  - "scene_el.png" - A Lenslet holoscopic image with a resolution of 8k and with multiple elemental images
  - A 5D array of the holoscopic image in .mat format
  - Ground-truth disparity map of central viewpoint image
- 40x40
- 60x60
- 80x80
- 100x100

V. Conclusion

The few holoscopic image simulators that exist simulate a small number of high-resolution images, such as Lytro camera images. Our work aimed to reduce processing time, simplify the solution, and provide a wide range of adjustable settings. After creating VPIs, the simulator reverse engineers to create an EIs grid based on the plenoptic 4D function. The simulator generates a ground truth disparity map, VPI and EI grids, and holoscopic images. A 5D holoscopic matrix stores the entire holoscopic image structure alongside the images for convenience. The simulator's linear rendering time was its main contribution.

Many holoscopic image databases exist. Most datasets are Lytro camera or high-resolution VPIs, but a few are raw holoscopic 1.0 image (EI-based) datasets. We contributed to a large dataset of raw holoscopic 1.0 images and VPIs at various simulator image resolutions. Fine details, shadows, reflections, etc. Image quality assessment showed that higher-resolution VPIs produced higher-quality EIs, but these EIs have a low scale resolution and a short baseline, making them difficult to use in most stereo-vision algorithms. A larger aperture widens the VPI gap, increasing disparity. The output holoscopic image is also analysed for baseline and DoF. The dataset includes a holoscopic toolbox for VPIs and Els. Our website has the holoscopic simulator, dataset, and toolbox.

References

[1] M. Levoy and P. Hanrahan, "Light field rendering," in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996, pp. 31–42.

[2] M. Swash *et al.*, "Holoscopic 3d imaging and display technology: Camera/processing/display," Ph.D. dissertation, Brunel University London., 2013.

[3] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldluecke, "A dataset and evaluation methodology for depth estimation on 4d light fields," in *Asian Conference on Computer Vision*. Springer, 2016.

[4] A. Isaksen, L. McMillan, and S. J. Gortler, "Dynamically reparameterized light fields," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, 2000, pp. 297–306.

[5] D. G. Dansereau, B. Girod, and G. Wetzstein, "Liff: Light field features in scale and depth," *arXiv preprint arXiv:1901.03916*, 2019.

[6] N. Li, J. Ye, Y. Ji, H. Ling, and J. Yu, "Saliency detection on light field," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[7] K. Fu, Y. Jiang, G.-P. Ji, T. Zhou, Q. Zhao, and D.-P. Fan, "Light field salient object detection: A review and benchmark," *Computational Visual Media*, pp. 1–26, 2022.

[8] M. Labussière, C. Teulière, F. Bernardin, and O. Ait-Aider, "Blur aware calibration of multi-focus plenoptic camera," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 2545–2554.

[9] L. Guillo, X. Jiang, G. Lafruit, and C. Guillemot, "Light field video dataset captured by a r8 raytrix camera (with disparity maps)," Ph.D. dissertation, INTERNATIONAL ORGANISATION FOR STANDARDISATION ISO/IEC JTC1/SC29/WG1 & WG11, 2018.

[10] S. Heber and T. Pock, "Convolutional networks for shape from light field," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 3746–3754.

[11] "Gordon Wetzstein - Synthetic Light Field Archive." [Online]. Available: https://web.media.mit.edu/~gordonw/SyntheticLightFields/

[12] J. Shi, X. Jiang, and C. Guillemot, "A framework for learning depth from a flexible subset of dense and sparse light field views," *IEEE Transactions on Image Processing*, vol. 28, no. 12, pp. 5867–5880, 2019.

[13] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy, "High performance imaging using large camera arrays," in *ACM SIGGRAPH 2005 Papers*, 2005, pp. 765–776.

[14] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldluecke, "A dataset and evaluation methodology for depth estimation on 4d light fields," in *Asian conference on computer vision*. Springer, 2016, pp. 19–34.

[15] E. H. Adelson, J. R. Bergen *et al.*, "The plenoptic function and the elements of early vision," *Computational models of visual processing*, vol. 1, no. 2, pp. 3–20, 1991.

[16] J.-H. Park, Y. Kim, and B. Lee, "Elemental image generation based on integral imaging with enhanced resolution," in *Information Optics and Photonics Technology*, vol. 5642. International Society for Optics and Photonics, 2005, pp. 186–195.

[17] T. G. Georgiev, K. C. Zheng, B. Curless, D. Salesin, S. K. Nayar, and C. Intwala, "Spatio-angular resolution tradeoffs in integral photography." *Rendering Techniques*, vol. 2006, no. 263-272, p. 21, 2006.

[1] https://github.com/BodorAlmatouk/HoloscopicToolbox