[POSTER] Realtime Generation of Caustic Images Using a Deep Neural Network

Takurou Okamoto* Osaka University
Yuki Uranishi Osaka University
Tomohiro Mashita Osaka University
Photchara Ratsamee Osaka University
Kiyoshi Kiyokawa Nara Institute of Science and Technology
Haruo Takemura Osaka University

Abstract

We propose a method for generating caustic images in real time using a deep/convolutional neural network (CNN). To do so, training images are first rendered using photon mapping, and the CNN learns the correspondences between the depth images and caustic images. After learning, the CNN generates a caustic image from a depth image within 55 milliseconds. In addition, the similarity between the generated caustic images and the ground truth shows that our method is very promising for generation of the caustic images for a number of known objects. While the method does not handle objects in which ground truth is not already known.This method can play an important role in scenes used for stage video production and interactive art in the future.

Index Terms: H.5.1 [Information interfaces and presentation]: Multimedia Information Systems—Artificial, Augmented, Virtual Realities; I.3.7 [Computer graphics]: Three-Dimensional Graphics and Realism—Color, shading, shadowing, and texture

1 Introduction

Appearance of an object is affected not only by color, shading, and gloss, but also by its transparency. There are a lot of interior designs and works of art that utilize reflected or refracted light by transparent objects, such as sun catchers, crystal balls, glass works and so on. By applying the texture of a transparent object, it is expected to give a strong impression and realism to the audience. Amano [1] made it possible to control color, contrast, transparency and gloss of an object surface in real time using a coaxial optical projector-camera system. On the other hand, for real transparent objects, caustics are projected on their background by reflected and refracted light. Amano did not mention about optical phenomena other than object surface properties. At present, rendering of caustics for transparent objects by existing methods such as ray tracking was too processing intensive. For example, general methods to calculate global illumination, such as Path-tracing [3] and Photon mapping [5], demanded a large amount of calculation in order to remove noise or to calculate photons.

One existing method to output data in response to specific image patterns is that of a neural network. In this research, we propose a real time caustics rendering method using deep learning. Our method is suitable for interactive systems that require a fast response such as participatory media arts.

2 Proposed System

2.1 System Overview

In this research, we propose a system to generate caustics copied onto the surface of an opaque object to enable transparency in real time. The basic system form is shown to Figure 2 (a), and a prototype is shown to Figure 2 (b). At first, an opaque object is set on the screen in order to project caustics by a light source. The position and shape of the object is first obtained from the camera, then the output caustics image is generated by inputting the learned model. The resulting caustic is then rendered and projected onto corresponding position on the screen, making the image's shadow appear to be that of its transparent counterpart.

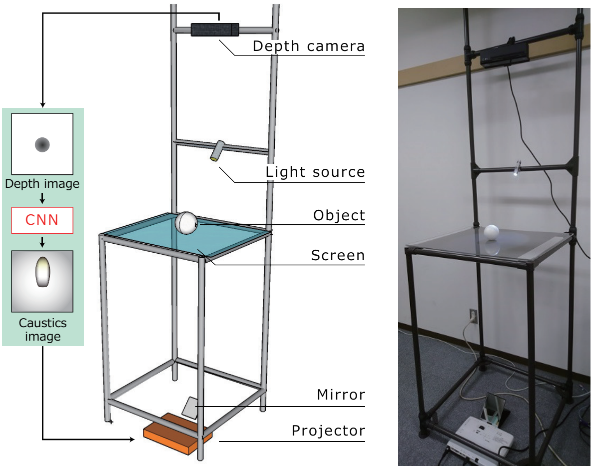
Figure 1: An example of a caustic generated by proposed system

2.2 Caustics Generation with Deep Neural Network

The caustics projected on background is determined by parameters about the object, refractive index, transmittance and colors, and positional relative of the object, light source and the background. When parameters about the object and the position of the light source and the background is fixed, the caustics is determined by the position and shape of the object. In proposed method, caustics images are rendered by reproducing the environment in CG space. Virtual depth camera set on the same position as real environment, then the position and shape are got as depth images from it. At the same time, when it is assumed that the object is a transparent having a specific material and characteristic, caustics images on the background are rendered. These processes are executed while changing the position and posture with objects of various shapes, and a large number of combinations of the depth image and the focused pattern image are prepared. By letting the neural network learn the image pairs (hereinafter called dataset) with the depth images as the input and the caustics images as the target output, the shape and position of the object which can be read from the depth image are associated to the characteristics of caustics. In this way, learning model for rendering caustics from depth images is created.

In order to exploit the correlation of each pixel value, CNN is used for this network. With reference to CNN used in the method of Simo-serra [6] et al., We adopted the network that compresses images into small feature map and outputs by restring images to its the same resolution as input images. We denote with Figure 3 the adopted construction of network. Outputs from the last layer

*e-mail: okamoto.takurou@lab.ime.cmc.osaka-u.ac.jp



(a) System outline drawing (b) Prototype

Figure 2: Proposed system

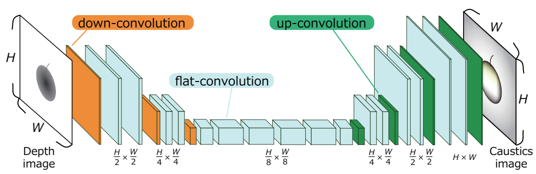


Figure 3: Network construction

were applied to sigmoid function, and outputs from the other layers were applied to Rectified Liner Unit (ReLU) function. Mean square error (MSE) was adopted as the loss function in this model. Parameters of each layer were calculated so that MSE becomes smaller. Adagrad [2] was adopted as the method to update gradient.

3 Experiment and Discussion

3.1 Learning Model

We selected objects of eight shapes as learning objects and assumed an environment where these objects were placed on the table. A set of dataset (hereinafter called learning dataset) of 5,000 pairs per one shape, a total of 40,000 pairs were rendered while randomly changing the position in the horizontal direction and the rotation about the vertical direction with respect to the table surface. POV-Ray [4] was used for rendering them, and the width and the height of the images were 256 and 256 pixels, respectively. Of these learning dataset, 10% of 4,000 pairs were randomly removed as test dataset used only for evaluation, and the remaining 36,000 pairs were used as training dataset. The model was learned using the training dataset in 100 epoch. We define a learned model of n epochs as model n. In addition, in order to investigate the generalized performance of the learned model, five types of objects were rendered separately from the learning dataset, and 20 sets of data per one shape, 100 sets of dataset (hereinafter called unknown dataset) were prepared. 100 pairs of depth images and caustics images were randomly extracted from the training and test dataset. We used these dataset and unknown dataset in order to measure the processing time.

3.2 Experimental Results

In order to investigate the reproducibility of the caustics, the similarity between caustics images created by proposed method and target output images rendered by POV-Ray was calculated. Each depth images of 100 pairs of training, test and unknown dataset were inputted into model into model 1, model 5, model 20 and model 100, then caustics images were outputted. Caustics images which were set as target images were regarded as Ground Truth (GT). Sum of Squared Difference (SSD) between them and output images. In this case, the equation of SSD is given as

Table 1: Average of standard deviation of $f_{\text{SSD}}(Y, Y^*)$

	training		test		unknown	
model	Average	Standard deviation	Average	Standard deviation	Average	Standard deviation
1	303.38	132.24	304.71	166.39	1039.2	485.00
5	71.436	31.980	83.966	76.131	1122.8	554.19
20	32.687	21.547	39.377	36.292	1145.4	564.73
100	18.053	19.321	19.655	20.224	1171.8	573.37

Table 2: Processing time

item	training	test	unknown	POV-Ray
time (sec)	0.05470	0.05469	0.05469	0.3234
standard deviation	0.007898	0.007805	0.007806	-

$f_{\text{SSD}}(Y, Y^*) = ||Y - Y^*||^2{}_{\text{FRO}}$. Here, Y is the output image and $Y*$ is the GT. For your reference, the output images and the GT when test dataset were inputted to model 100 are in Figure. X. The result of calculation of SSD averages and standard deviation are given in Table 1. In addition to, each processing time at the time of outputting caustics images when each dataset was inputted to the model 100 was recorded. As the comparison object, average processing time per image when randomly selecting 100 pairs from the learning data so that each shape was equally include in POV-Ray was recorded too. The result in each term is given in Table 2, but the standard deviation of POV-Ray was not recorded because there was no function to output processing time for each image.

3.3 Discussion

As shown in Table 1, the value of $f_{\text{SSD}}(Y, Y^*)$ decreased for training and test dataset as the number of learning increased, but changes could not be read for unknown dataset. From this, it was found that caustics images of this method had high reproducibility for shapes included in the learning data. From the viewpoint of processing time, it was found that proposed method was carried out in real time since the processing time fell below 55 milliseconds (Estimation of real-time processing), as shown in Table 2.

4 Conclusion

In this research, we have proposed the method to render caustics images in real time using deep learning. Caustics images with high reproducibility was obtained for an object whose shape is known. The future task is to improve the generalization performance by making ingenious multiple depth images and devising such as changing the network structure.

Acknowledgements

This work was supported by JSPS KAKENHI Grant Number JP16H02858.

References

[1] T. Amano. Projection based real-time material appearance manipulation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2013.

[2] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.*, July 2011.

[3] J. T. Kajiya. The rendering equation. In *Proceedings of the 13th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '86, pages 143–150, New York, NY, USA, 1986. ACM.

[4] P. of Vision Raytracer Pty. Ltd. POV-Ray - the persistence of vision raytracer, 2008. [Online; accessed 19-July-2016].

[5] T. J. Purcell, C. Donner, M. Cammarano, H. W. Jensen, and P. Hanrahan. Photon mapping on programmable graphics hardware. In *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS Conference on Graphics Hardware*, HWWS '03, pages 41–50, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.

[6] E. Simo-Serra, S. Iizuka, K. Sasaki, and H. Ishikawa. Learning to simplify: Fully convolutional networks for rough sketch cleanup. *ACM Trans. Graph.*, 35(4):121:1–121:11, July 2016.