MPIB: An MPI-Based Bokeh Rendering Framework for Realistic Partial Occlusion Effects

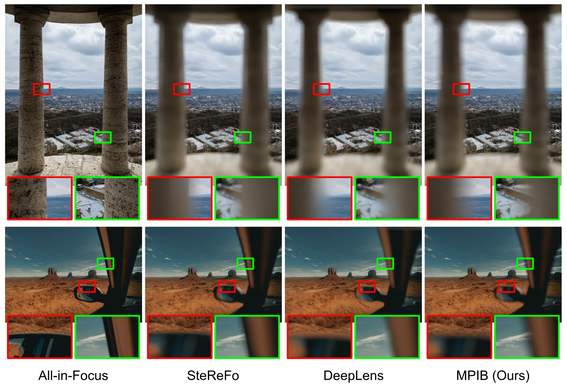
Juewen Peng[1], Jianming Zhang[2], Xianrui Luo[1], Hao Lu[1], Ke Xian[1*], and Zhiguo Cao[1]

[1]School of AIA, Huazhong University of Science and Technology [2]Adobe Research

ECCV TEL AVIV 2022

Problem Statement

Our goal is to render realistic partial occlusion effects, a phenomenon that blurry objects near a camera are semi-transparent, resulting in partial appearance of occluded background.



MPI Representation

Multiplane Image: A set of fronto-parallel planes, each of which encodes an RGB color image and an alpha map.

$$\{(c_i, \alpha_i) \,|\, i = 1, 2, ..., N\}$$

Layer Composition for Novel View Synthesis:

$$I = \sum_{i=1}^{N} \left(c_i \alpha_i \prod_{j=i+1}^{N} (1 - \alpha_j) \right)$$

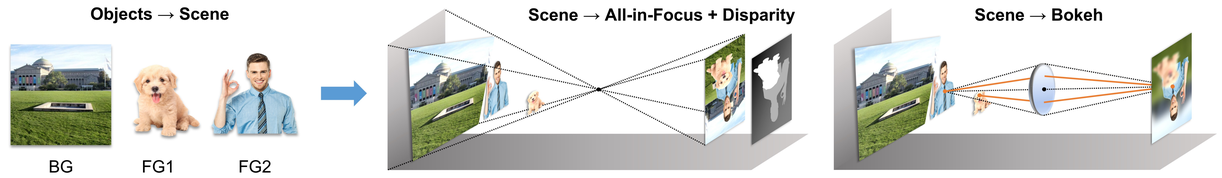
Layer Composition for Bokeh Rendering (Ours):

$$B = \frac{\sum_{i=1}^{N} \left((c_i \alpha_i * K_i) \prod_{j=i+1}^{N} (1 - \alpha_j * K_j) \right)}{\sum_{i=1}^{N} \left((\alpha_i * K_i) \prod_{j=i+1}^{N} (1 - \alpha_j * K_j) \right)}$$

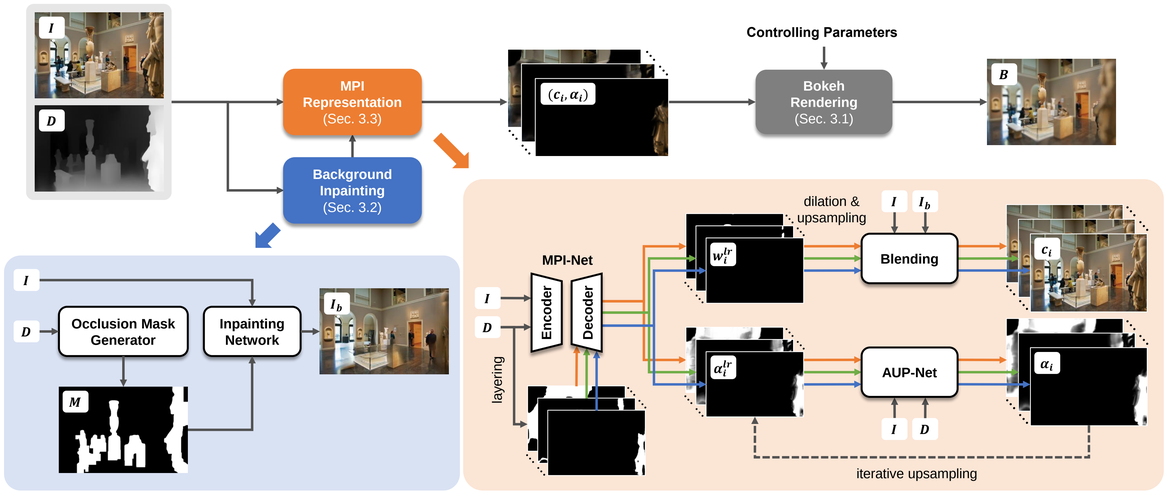
K represents blur kernels and $*$ denotes convolution.

Bokeh Generator

Given a background RGB image and several foreground RGBA images, we first construct a 3D scene through randomly setting the positions and disparities of different objects. Then, we can easily synthesize an all-in-focus image and a corresponding disparity map, and produce a bokeh image by our ray tracing-based bokeh generator.



Framework



Background Inpainting Module: Using an occlusion mask generator together with an off-the-shelf inpainting model to generate a background image.

MPI Representation Module: Combining an MPI-Net and an AUP-Net to obtain a high-resolution scene representation of multiple RGBA planes.

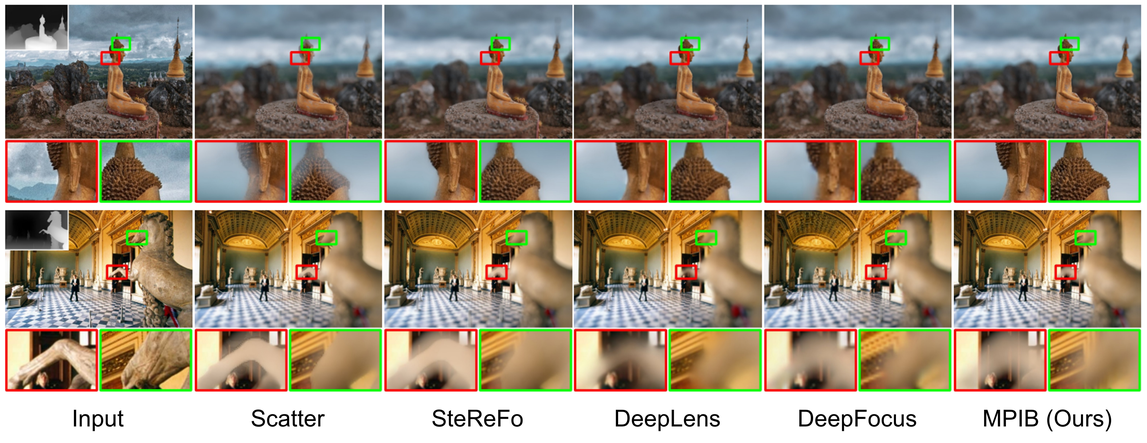
Bokeh Rendering Module: The constructed scene representation can be reused to produce multiple bokeh images according to different controlling parameters.

Paper and Code: `https://juewenpeng.github.io/MPIB/`

Experiments

Quantitative Results on Synthetic Dataset:

Method	Constant disparity for each object					Varying disparity for each object				
	LPIPS↓	PSNR↑	PSNR_{ob}↑	SSIM↑	SSIM_{ob}↑	LPIPS↓	PSNR↑	PSNR_{ob}↑	SSIM↑	SSIM_{ob}↑
Gather	0.042	32.6	25.4	0.978	0.896	0.044	33.0	25.9	0.979	0.903
Scatter	0.029	33.9	26.7	0.983	0.921	0.027	34.8	27.8	0.985	0.934
RVR	0.103	28.8	21.9	0.951	0.783	0.113	28.9	22.0	0.951	0.787
SteReFo	0.038	32.8	26.4	0.976	0.923	0.040	32.5	26.3	0.973	0.928
DeepLens	0.068	29.5	24.9	0.945	0.897	0.055	29.8	24.9	0.957	0.911
DeepFocus	0.083	33.4	30.3	0.938	0.923	0.063	35.2	31.9	0.968	0.952
Ours ($N_{bg} = 1$)	0.011	36.7	30.0	0.989	0.951	0.019	36.8	30.5	0.986	0.956
Ours ($N_{bg} = 3$)	**0.008**	**39.3**	**33.0**	**0.991**	**0.963**	**0.017**	**38.8**	**33.0**	**0.988**	**0.966**

User Study Results on Real-World Images:

Method	Refocusing on the foreground				Refocusing on the background			
	Scatter	SteReFo	DeepLens	DeepFocus	Scatter	SteReFo	DeepLens	DeepFocus
Ours	**89.4%**	**72.7%**	**83.4%**	**86.1%**	**82.9%**	**76.7%**	**77.2%**	**82.8%**