Occlusion-Aware Cost Constructor for Light Field Depth Estimation (Supplemental Material)

Yingqian Wang[1], Longguang Wang[1], Zhengyu Liang[1], Jungang Yang[1]✉, Wei An[1], Yulan Guo[1]
[1]National University of Defense Technology

https://github.com/YingqianWang/OACC-Net

Section I introduces the padding strategy of the proposed occlusion-aware cost constructor (OACC). Section II describes details of our OACC-Net. Section III presents additional comparative results on the 4D light field (LF) benchmark. Section IV shows additional visual results achieved by different methods on other LF datasets [5, 9, 14, 17]. Section V discusses the broader impact of our method.

I. Padding Strategy of our OACC

As described in Sec 3.2.1 in the main body of our paper, our OACC can achieve cost construction by performing convolutions on sub-aperture image (SAI) arrays. However, when handling pixels near the boundary of SAIs, some ambiguities can be introduced to the resulting matching costs. Without loss of generality, we take the top-left corner of the SAI as an example to analyze this boundary issue and introduce our padding strategy.

As shown in Figs. I (a)-(c), we apply our OACC to a densely-tiled 5×5 SAI array. Each SAI has a spatial size of $H \times W$. According to Eq. 4 in the main body of our paper, the dilation rate of our OACC is correlated to the predefined disparity d. Specifically, when d=0, as shown in Fig. I (a), the vertical and horizontal dilations equal to the height and width of the SAI, respectively. In this situation, there is no boundary issue and the resulting cost tensor has a spatial size of $H \times W$. When d>0, as shown in Fig. I (b), the vertical/horizontal dilation is smaller than the height/width of the SAI. In this situation, some sampling points of our OACC move across the boundary of their corresponding SAIs and locate on the adjacent SAIs (marked by red boxes). Similarly, when d<0, as shown in Fig. I (c), the vertical/horizontal dilation is larger than the height/width of the SAI, and some sampling points locate on the adjacent SAIs (marked by red boxes) or outside the SAI arrays (marked by black boxes). Note that, pixels marked by the red and black boxes do not provide any correspondence information, while pixels marked by the red boxes can even introduce ambiguities to the resulting matching costs.

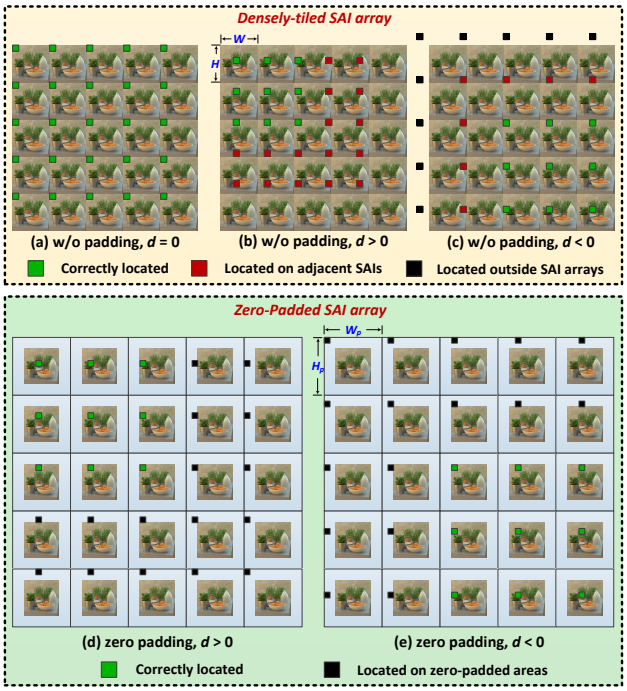

Figure I. An illustration of the boundary issue and our padding strategy. Here, a 5×5 SAI array is used as an example. By using our proposed padding strategy, pixels outside the boundary of SAIs can be assigned as zero values and thus reduce the matching ambiguity of our OACC.

In this paper, we propose a padding strategy for our OACC to reduce matching ambiguities. The core idea of our padding strategy is to assign zero values to all the "out-of-boundary" pixels (marked by both red and black boxes). To achieve this goal, we perform zero-padding to each SAI separately before organizing them into an SAI array, as shown in Figs. I (d) and (e). The vertical and horizontal padding values η_h and η_w can be calculated according to

$$\eta_h = \frac{U-1}{2} \cdot \tilde{d}, \quad \eta_w = \frac{V-1}{2} \cdot \tilde{d}, \tag{I}$$

where U and V denote the angular resolution of the LF (e.g., U=V=5 for a 5×5 LF), $\tilde{d}$=$max\{|d_{max}|, |d_{min}|\}$ de-

Table I. The detailed architecture of our OACC-Net. "ResBlock2D" and "ResBlock3D" represent 2D and 3D residual block, respectively. M denotes the number of SAIs (i.e., $M=U\times V$), and D denotes the number of disparity candidates.

Layers	*Setting*	*Input size*	*Output size*
Feature Extraction			
Conv2D_a	k=3×3	$M\times(H\times W\times 1)$	$M\times(H\times W\times 16)$
ResBlock2D ×8	$k=\begin{bmatrix}3\times3\\3\times3\end{bmatrix}$	$M\times(H\times W\times 16)$	$M\times(H\times W\times 16)$
Conv2D_b	k=3×3	$M\times(H\times W\times 16)$	$M\times(H\times W\times 16)$
Conv2D_c	k=3×3	$M\times(H\times W\times 16)$	$M\times(H\times W\times 8)$
Conv2D_d	k=3×3	$M\times(H\times W\times 8)$	$M\times(H\times W\times 8)$
Cost Construction			
Pad & Reshape	–	$M\times(H\times W\times 8)$	$UH_p\times VW_p\times 8$
OACC & Crop	$k=U\times V$	$UH_p\times VW_p\times 8$ $U\times V\times M$ (mask)	$D\times H\times W\times 512$
Cost Aggregation			
Conv3D_a	k=1×1×1	$D\times H\times W\times 512$	$D\times H\times W\times 160$
Conv3D_b	k=3×3×3	$D\times H\times W\times 160$	$D\times H\times W\times 160$
Conv3D_c	k=3×3×3	$D\times H\times W\times 160$	$D\times H\times W\times 160$
ResBlock3D ×2	$k=\begin{bmatrix}3\times3\times3\\3\times3\times3\end{bmatrix}$ Channel_Att	$D\times H\times W\times 160$	$D\times H\times W\times 160$
Conv3D_d	k=3×3×3	$D\times H\times W\times 160$	$D\times H\times W\times 160$
Conv3D_e	k=3×3×3	$D\times H\times W\times 160$	$D\times H\times W\times 1$
Depth Regression			
Softmax	–	$D\times H\times W\times 1$	$D\times H\times W\times 1$
Regress	–	$D\times H\times W\times 1$	$H\times W\times 1$

notes the maximum absolute value of the predefined disparity (equals to 4 in this paper). After zero-padding, each SAI has a height of $H_p=H+2\eta_h$ and a width of $W_p=W+2\eta_w$. The padded SAIs are then organized into an SAI array for cost construction, and the dilation rates of our OACC are recalculated according to H_p and W_p. It can be proved that η_h and η_w are large enough to make all the sampling points not locate on other views under each candidate disparity. The output of our OACC under disparity d has a height of $(H+(U-1)(d+\tilde{d}))$ and a width of $(W+(V-1)(d+\tilde{d}))$. Finally, cropping is performed to the resulting cost tensor to ensure it has a resolution of $H\times W$. The cropping values can be calculated according to

$$c_h(d)=\frac{U-1}{2}\cdot(d+\tilde{d}),\quad c_w(d)=\frac{V-1}{2}\cdot(d+\tilde{d}),\quad \text{(II)}$$

where $c_h(d)$ and $c_w(d)$ denote the vertical (i.e., top and bottom) and horizontal (i.e., left and right) cropping values, respectively.

II. Details of our OACC-Net

The detailed structure of our OACC-Net is shown in Table I. In the feature extraction stage, a 3×3 convolution (i.e., Conv2D_a) is used to extract initial feature with a channel depth of 16. Then, eight residual blocks (i.e., ResBlock2D) are applied for deep feature extraction. Finally, three 3×3 convolutions are used to integrate the extracted features for

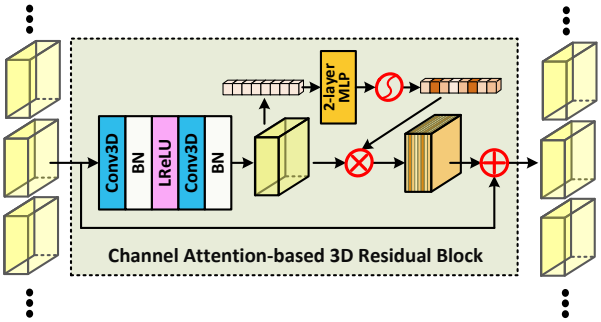

Figure II. The architecture of our channel attention-based 3D residual block (i.e., ResBlock3D).

cost construction. We use LeakyReLU with a leaky factor of 0.1 for activation, and perform batch normalization after each convolution except the last one (i.e., Conv2D_d).

After feature extraction, we obtained an LF feature of size $M\times H\times W\times 8$, where $M=U\times V$ denotes the number of views. Then, we perform zero-padding (as described in Sec. I) to each SAI and organize the padded SAIs into an array of size $UH_p\times VW_p\times 8$. The proposed OACC (with a kernel size of $U\times V$) takes the padded SAI array and an occlusion mask (of size $U\times V\times M$) as its input for cost construction. The generated cost tensor has a channel depth of 512 to fully incorporate the correspondence information from all the views.

In the cost aggregation stage, a 3D convolution (with a kernel size of 1×1×1) is first used to reduce the channel depth from 512 to 160. Then, eight 3D convolutions (with a kernel size of 3×3×3) are used for deep cost aggregation. The middle four convolutions are organized into two residual blocks, and channel attention mechanism is adopted at the end of each residual block to highlight contributive channels, as illustrated in Fig. II. Similar to the feature extraction stage, we use LeakyReLU with a leaky factor of 0.1 for activation and perform batch normalization after each 3D convolution except the last one (i.e., Conv3D_e).

III. Results on the 4D LF Benchmark

Table II reports the quantitative results (i.e., BadPix0.07, BadPix0.03, BadPix0.01, and MSE) of our method and the compared methods. Figures III and IV show the estimated disparity maps and the corresponding error maps on the eight validation scenes. Figure V shows the estimated disparity maps on the four test scenes.

IV. Results on different LF datasets

Figures VI, VII, and VIII show the comparative visual results achieved by SPO [19], EPINET [12] and our method on different kinds LF datasets [5, 9, 14, 17].

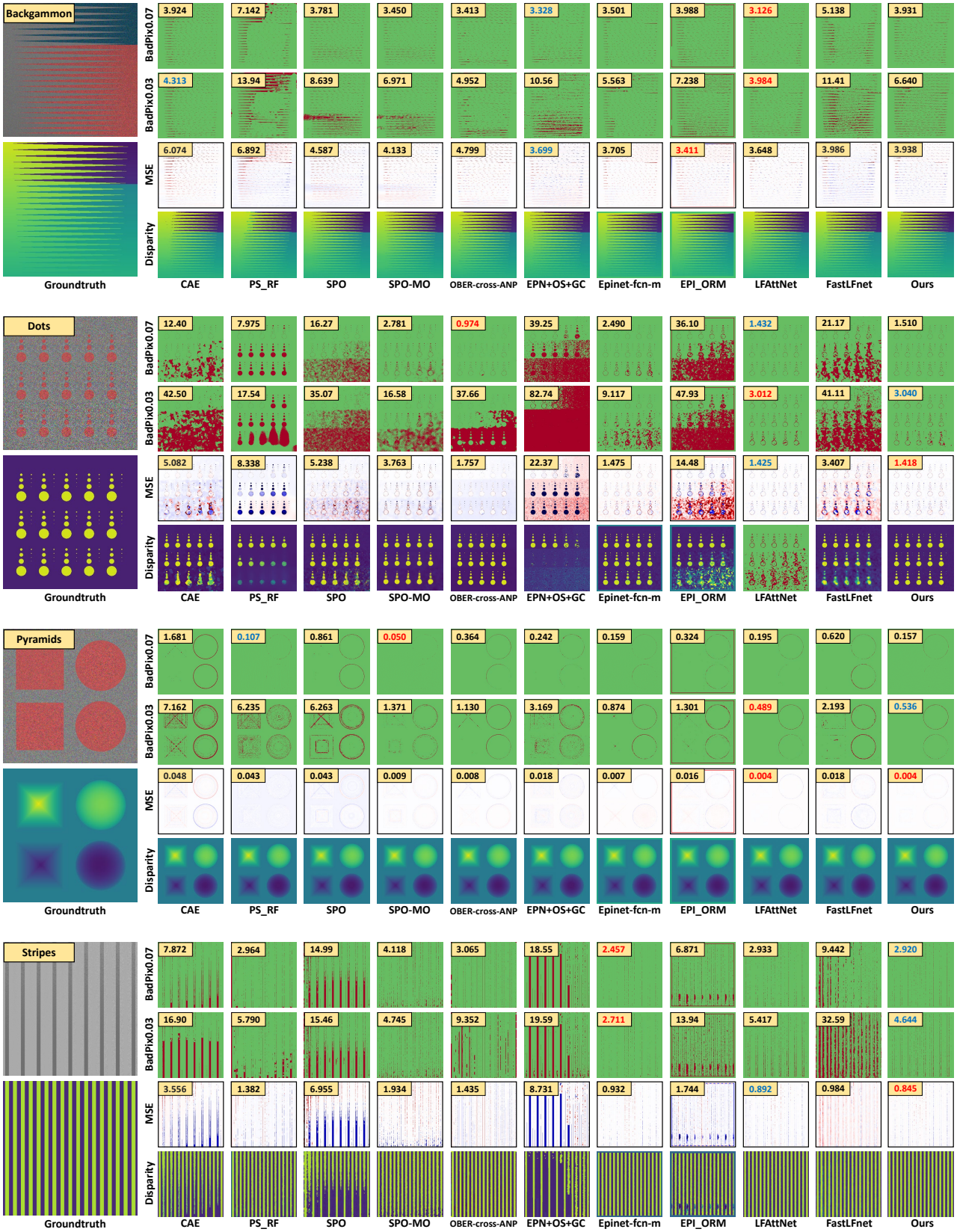

Figure III. Visual comparisons of disparity and error maps on validation scenes "backgammon", "dots", "pyramids", and "stripes" [2]. Corresponding quantitative scores (BadPix0.07, BadPix0.03, and MSE) are reported on the top-left corner of each error map.

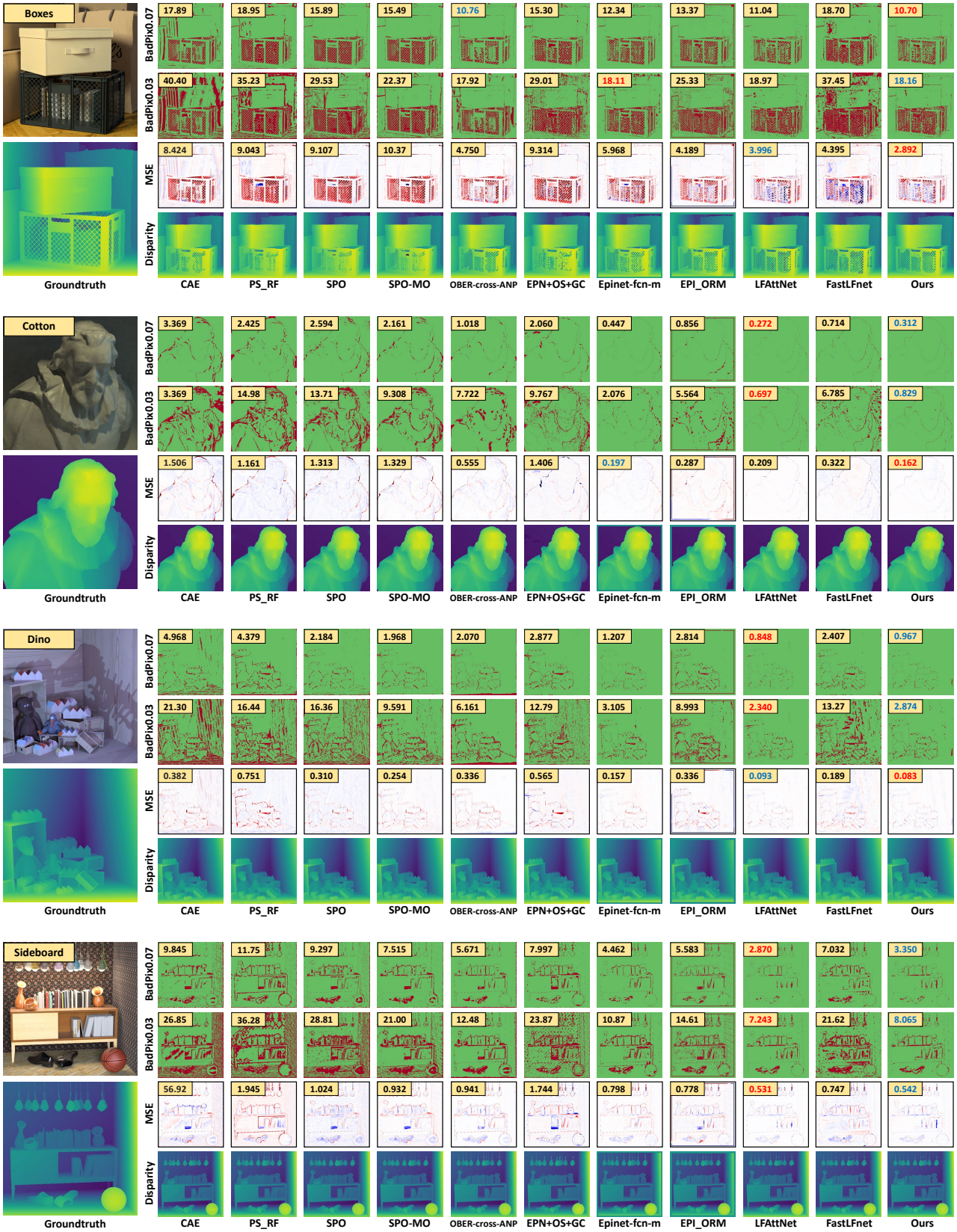

Figure IV. Visual comparisons of disparity and error maps on validation scenes "boxes", "cotton", "dino", and "sideboard" [2]. Corresponding quantitative scores (BadPix0.07, BadPix0.03, and MSE) are reported on the top-left corner of each error map.

Table II. Quantitative results (i.e., BadPix0.07 (BP07), BadPix0.03 (BP03), BadPix0.01 (BP01), and MSE×100 (MSE)) achieved by different LF depth estimation methods on the 4D LF benchmark [2]. The best results are in red and the second best results are in blue.

	Backgammon				***Dots***				***Pyramids***				***Stripes***			
	BP07	***BP03***	***BP01***	***MSE***	***BP07***	***BP03***	***BP01***	***MSE***	***BP07***	***BP03***	***BP01***	***MSE***	***BP07***	***BP03***	***BP01***	***MSE***
LF_OCC [15]	13.52	44.90	91.40	22.78	9.695	31.09	76.02	3.185	1.450	25.57	92.86	0.077	18.33	54.69	98.63	7.942
CAE [18]	3.924	4.313	17.32	6.074	12.40	42.50	83.70	5.082	1.681	7.162	27.54	0.048	7.872	16.90	39.95	3.556
PS-RF [4]	7.142	13.94	74.66	6.892	7.975	17.54	78.80	8.338	0.107	6.235	83.23	0.043	2.964	5.790	41.65	1.382
SPO [19]	3.781	8.639	49.94	4.587	16.27	35.06	58.07	5.238	0.861	6.263	79.20	0.043	14.99	15.46	21.87	6.955
SPO-MO [11]	3.450	6.971	28.27	4.133	2.781	16.58	41.02	3.763	0.050	1.371	13.50	0.009	4.118	4.745	27.57	1.934
OBER-cross-ANP [10]	3.413	4.952	13.66	4.700	0.974	37.66	73.13	1.757	0.364	1.130	8.171	0.008	3.065	9.352	44.72	1.435
OAVC [1]	3.121	5.117	49.05	3.835	69.11	75.38	92.33	16.58	0.831	9.027	33.66	0.040	2.903	19.88	28.14	1.316
EPN+OS+GC [8]	3.328	10.56	55.98	3.699	39.25	82.74	84.91	22.37	0.242	3.169	28.56	0.018	18.54	19.60	28.17	8.731
Epinet-fcn [12]	3.580	6.289	20.89	3.629	3.183	12.73	41.05	1.635	0.192	0.913	11.87	0.008	2.462	3.115	15.67	0.950
Epinet-fcn-m [12]	3.501	5.563	19.43	3.705	2.490	9.117	35.61	1.475	0.159	0.874	11.42	0.007	2.457	2.711	11.77	0.932
Epinet-fcn-9×9 [12]	3.287	4.482	15.39	3.909	4.030	18.70	44.64	1.980	0.147	0.604	8.913	0.007	2.413	2.876	14.75	0.915
EPI-Shift [6]	22.89	40.53	70.58	12.79	43.92	53.18	74.55	13.15	1.242	7.315	40.48	0.037	22.72	47.70	78.95	1.686
EPI_ORM [7]	3.988	7.238	34.32	3.411	36.10	47.93	65.71	14.48	0.324	1.301	19.06	0.016	6.871	13.94	55.14	1.744
LFAttNet [13]	3.126	3.985	11.58	3.648	1.432	3.012	15.06	1.425	0.195	0.488	2.063	0.004	2.933	5.417	18.21	0.892
FastLFnet [3]	5.138	11.41	39.84	3.986	21.17	41.11	68.15	3.407	0.620	2.193	22.19	0.018	9.442	32.60	63.04	0.984
DistgDisp [16]	5.824	10.54	26.17	4.712	1.826	4.464	25.37	1.367	0.108	0.539	4.953	0.004	3.913	6.885	19.25	0.917
OACC-Net (ours)	3.931	6.640	21.61	3.938	1.510	3.040	21.02	1.418	0.157	0.536	3.852	0.004	2.920	4.644	15.24	0.845

	Boxes				***Cotton***				***Dino***				***Sideboard***			
	BP07	***BP03***	***BP01***	***MSE***	***BP07***	***BP03***	***BP01***	***MSE***	***BP07***	***BP03***	***BP01***	***MSE***	***BP07***	***BP03***	***BP01***	***MSE***
LF_OCC [15]	26.03	60.70	91.48	9.593	4.743	38.11	88.70	1.074	15.37	50.17	88.81	0.944	17.91	50.55	84.65	2.073
CAE [18]	17.89	40.40	72.69	8.424	3.369	15.50	59.22	1.506	4.968	21.30	61.06	0.382	9.845	26.85	56.92	0.876
PS_RF [4]	18.95	35.23	76.39	9.043	2.425	14.98	70.41	1.161	4.379	16.44	75.97	0.751	11.75	36.28	79.98	1.945
SPO [19]	15.89	29.52	73.23	9.107	2.594	13.71	69.05	1.313	2.184	16.36	69.87	0.310	9.297	28.81	73.36	1.024
SPO-MO [11]	15.49	22.37	49.77	10.37	2.161	9.038	32.08	1.329	1.968	9.591	42.64	0.254	7.515	21.00	52.90	0.932
OBER-cross-ANP [10]	10.76	17.92	44.96	4.750	1.108	7.722	36.79	0.555	2.070	6.161	22.76	0.336	5.671	12.48	32.79	0.941
OAVC [1]	16.14	33.68	71.91	6.988	2.550	20.79	61.35	0.598	3.936	19.03	61.82	0.267	12.42	37.83	73.85	1.047
EPN+OS+GC [8]	15.30	29.01	67.35	9.314	2.060	9.767	54.85	1.406	2.877	12.79	58.79	0.565	7.997	23.87	66.35	1.744
Epinet-fcn [12]	12.84	19.76	49.04	6.240	0.508	2.310	28.06	0.191	1.286	3.452	22.40	0.167	4.801	12.08	41.88	0.827
Epinet-fcn-m [12]	12.34	18.11	46.09	5.968	0.447	2.076	25.72	0.197	1.207	3.105	19.39	0.157	4.462	10.86	36.49	0.798
Epinet-fcn-9×9 [12]	12.25	18.66	45.73	6.036	0.464	2.217	25.27	0.223	1.263	3.221	23.44	0.151	4.783	11.82	40.49	0.806
EPI-Shift [6]	25.95	44.14	74.36	9.790	2.176	10.68	46.86	0.475	5.964	22.14	64.16	0.392	11.80	36.64	73.42	1.261
EPI_ORM [7]	13.37	25.33	59.68	4.189	0.856	5.564	42.94	0.287	2.814	8.993	41.04	0.336	5.583	14.61	52.59	0.778
LFAttNet [13]	11.04	18.97	37.04	3.996	0.271	0.697	3.644	0.209	0.848	2.339	12.22	0.093	2.869	7.243	20.73	0.530
FastLFnet [3]	18.70	37.45	71.82	4.395	0.714	6.785	49.34	0.322	2.407	13.27	56.24	0.189	7.032	21.62	61.96	0.747
DistgDisp [16]	13.31	21.13	41.62	3.325	0.489	1.478	7.594	0.184	1.414	4.018	20.46	0.099	4.051	9.816	28.28	0.713
OACC-Net (ours)	10.70	18.16	43.48	2.892	0.312	0.829	10.45	0.162	0.967	2.874	22.11	0.083	3.350	8.065	28.64	0.542

	Bedroom				***Bicycle***				***Herbs***				***Origami***			
	BP07	***BP03***	***BP01***	***MSE***	***BP07***	***BP03***	***BP01***	***MSE***	***BP07***	***BP03***	***BP01***	***MSE***	***BP07***	***BP03***	***BP01***	***MSE***
LF_OCC [15]	18.34	54.13	88.88	0.530	19.00	54.25	90.06	7.673	17.72	47.36	87.39	22.96	18.78	52.47	88.40	2.223
CAE [18]	5.788	25.36	68.59	0.234	11.22	23.62	59.64	5.135	9.550	23.16	59.24	11.67	10.03	28.35	64.16	1.778
PS_RF [4]	6.015	22.45	80.68	0.288	17.17	32.32	79.80	7.926	10.48	21.90	66.47	15.25	13.57	36.45	80.32	2.393
SPO [19]	4.864	23.53	72.37	0.209	10.91	26.90	71.13	5.570	8.260	30.62	86.62	11.23	11.69	32.71	75.58	2.032
SPO-MO [11]	3.228	13.91	43.80	0.152	10.05	22.47	50.47	5.617	8.269	19.71	46.08	12.05	9.411	23.07	53.99	1.667
OBER-cross-ANP [10]	3.329	9.558	28.91	0.185	8.683	16.17	37.83	4.314	7.120	14.06	36.83	10.44	8.665	20.03	42.16	1.493
OAVC [1]	4.915	19.09	64.76	0.212	12.22	25.46	64.74	4.886	8.733	29.65	74.76	10.36	12.56	30.59	69.35	1.478
EPN+OS+GC [8]	7.543	16.76	58.93	1.188	11.60	24.86	64.10	6.411	9.190	25.72	67.13	11.58	10.75	27.09	67.35	10.09
Epinet-fcn [12]	2.403	6.921	33.99	0.213	9.896	18.05	46.37	4.682	12.10	28.95	62.67	9.700	5.918	14.37	45.93	1.466
Epinet-fcn-m [12]	2.299	6.345	31.82	0.204	9.614	16.83	42.83	4.603	10.96	25.85	59.93	9.491	5.807	13.00	42.21	1.478
Epinet-fcn-9×9 [12]	2.287	6.291	31.23	0.231	9.853	17.19	43.85	4.929	17.75	34.54	59.86	9.423	6.339	13.92	42.17	1.646
EPI-Shift [6]	8.297	21.51	55.45	0.284	20.79	39.59	68.48	6.920	14.19	26.66	56.98	17.01	11.52	33.75	73.45	1.690
EPI_ORM [7]	5.492	14.66	51.02	0.298	11.12	21.20	51.22	3.489	8.515	24.60	68.79	4.468	8.661	22.95	56.57	1.826
LFAttNet [13]	2.792	5.318	13.33	0.366	9.511	15.99	31.35	3.350	5.219	9.483	19.27	6.605	4.824	8.925	22.19	1.733
FastLFnet [3]	4.903	15.92	52.88	0.202	15.38	28.45	59.24	4.715	10.72	23.39	59.98	8.285	12.64	33.65	72.36	2.228
DistgDisp [16]	2.349	5.925	17.66	0.111	9.856	17.58	35.72	3.419	6.846	12.44	24.44	6.846	4.270	9.816	28.42	1.053
OACC-Net (ours)	2.308	5.707	21.97	0.148	8.078	14.40	32.74	2.907	6.616	46.78	86.41	6.561	4.065	9.717	32.25	0.878

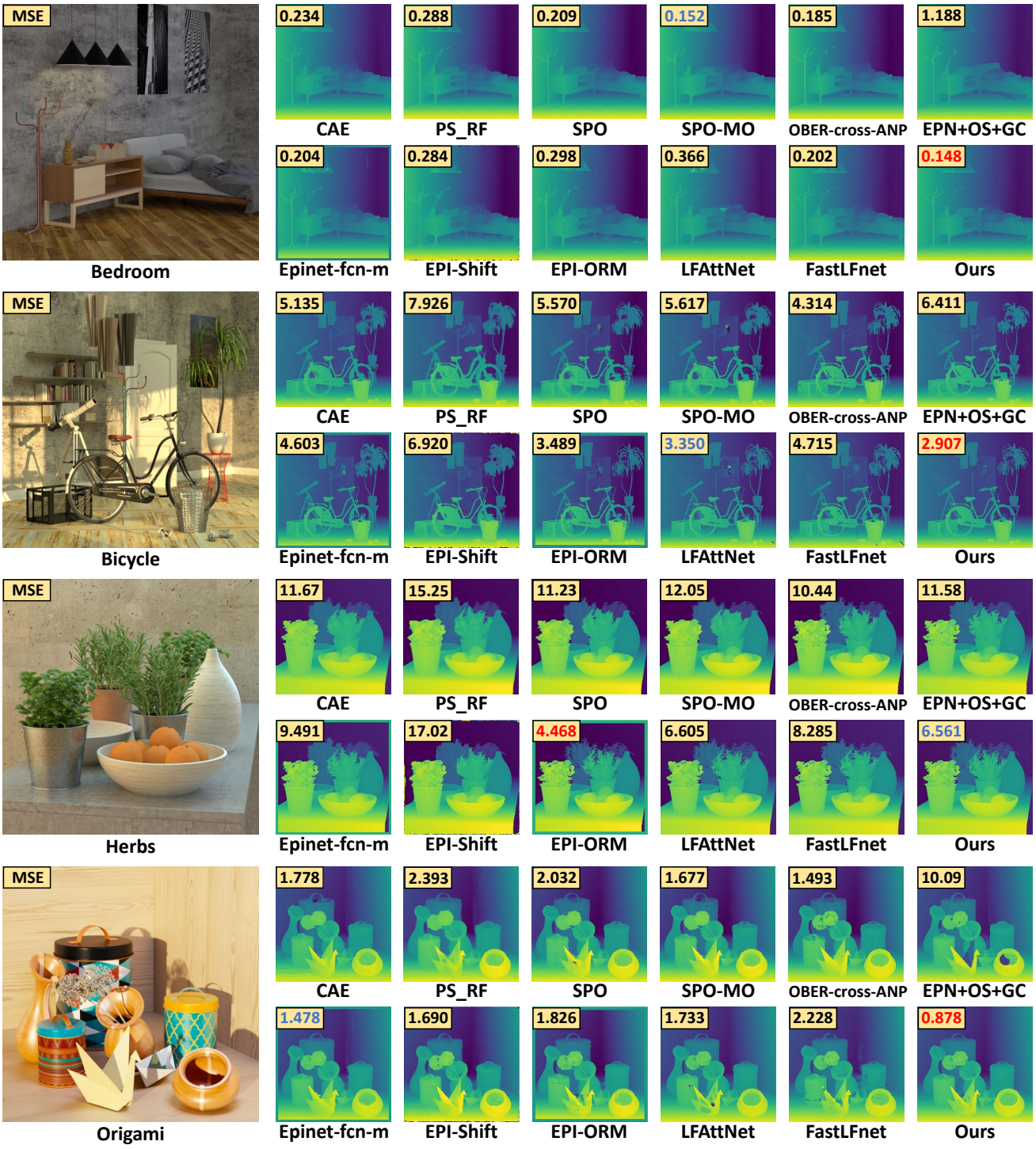

Figure V. Visual comparisons of disparity maps on test scenes "bedroom", "bicycle", "herbs", and "origami" [2]. The groundtruth disparity of these scenes are not released. The MSE of each method (copied from the benchmark site) is reported on the left-top corner.

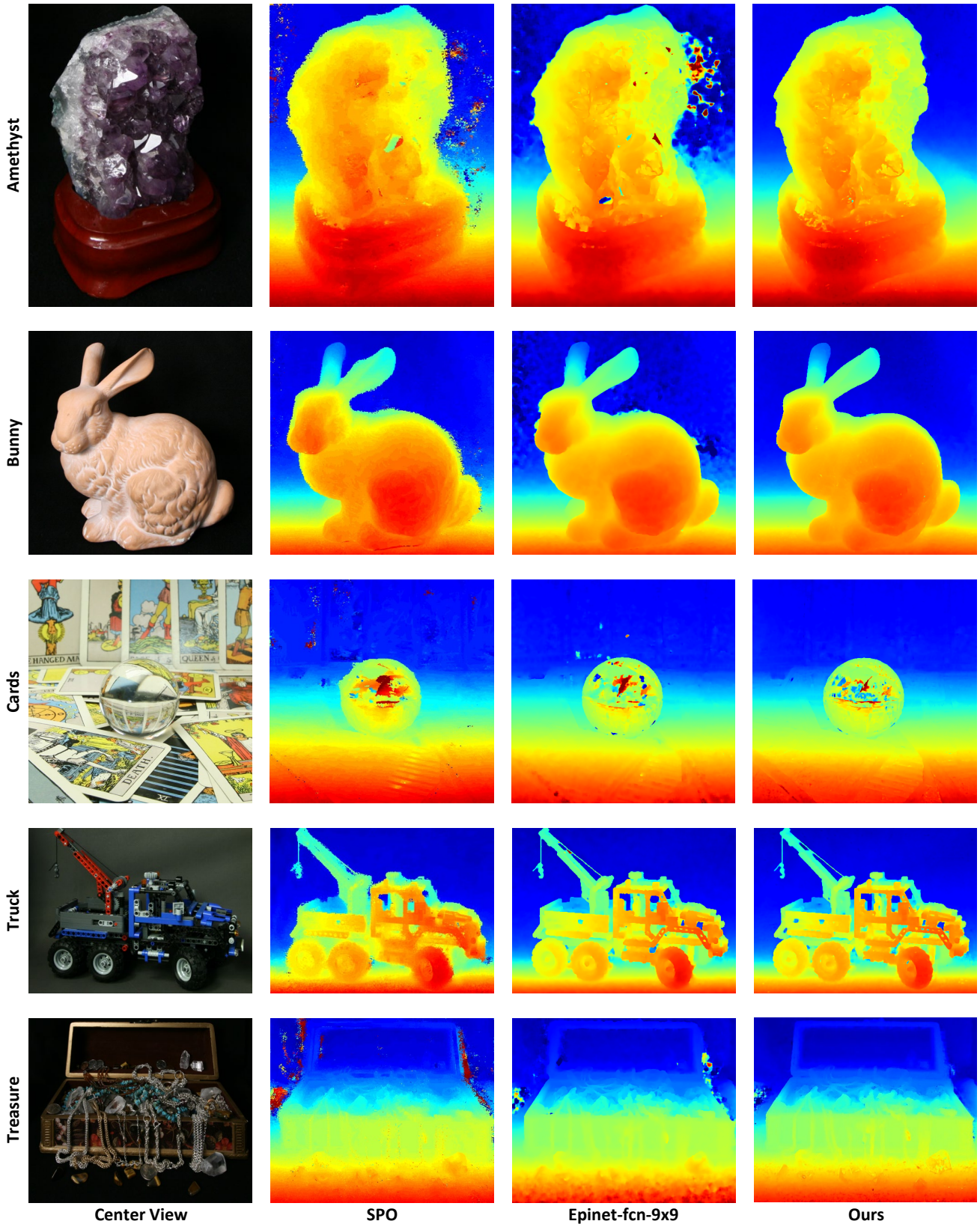

Figure VI. Visual results achieved by SPO [19], EPINET [12], and our method on the Stanford Gantry LF dataset [14]. Groundtruth disparity of these real-world LFs are unavailable.

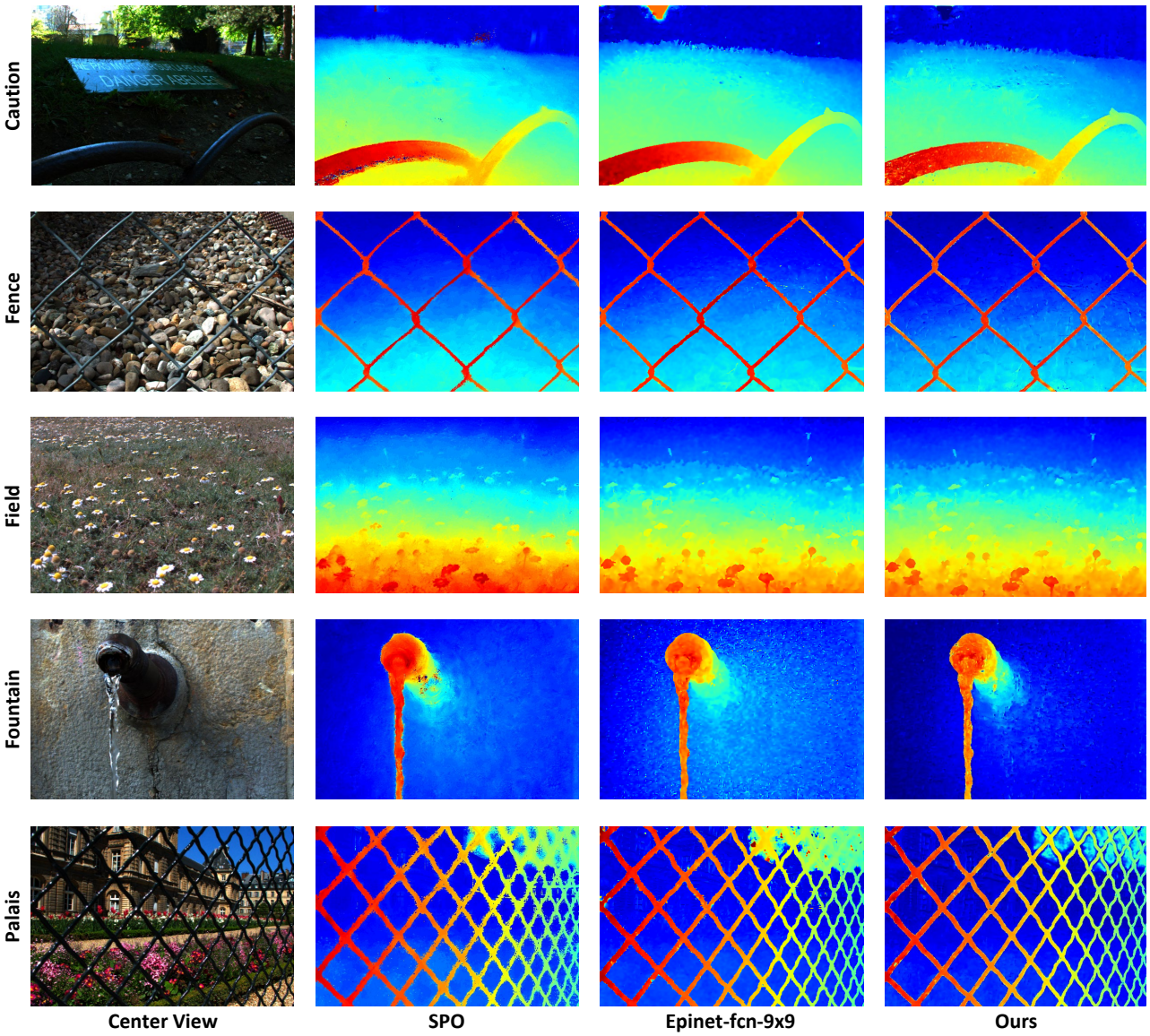

Figure VII. Visual results achieved by SPO [19], EPINET [12], and our method on LFs captured by Lytro cameras [5, 9]. Groundtruth disparity of these real-world LFs are unavailable.

V. Broader Impact

Our method has many potential applications such as 3D reconstruction, autonomous driving, and robotic systems. With fast and accurate depth estimation, our method can improve both accuracy and real-time performance of these systems.

Although our method achieves improved depth estimation accuracy on different datasets, the performance of our method is less promising in some challenging situations. As shown in Fig. IX, when handling scenes with reflective surfaces (e.g., *bulldozer* [14]), repetitive textures (e.g., *monasRoom* [17]), and illuminance variations (e.g., *bench* [9]), our method generates depth maps with large errors and obvious artifacts. Such failure cases can raise some potential safety issues such as collisions in robotics and accidents in autonomous driving. Consequently, sufficient safety test should be conducted before deploying our method to a specific system.

In the future, we will improve the robustness of our method to various challenging situations such as non-Lambertain surfaces, repetitive textures, textureless regions, illuminance variations, and extreme lighting conditions. We believe our method can benefit both research and industrial communities, and promote the development of LF-based computer vision.

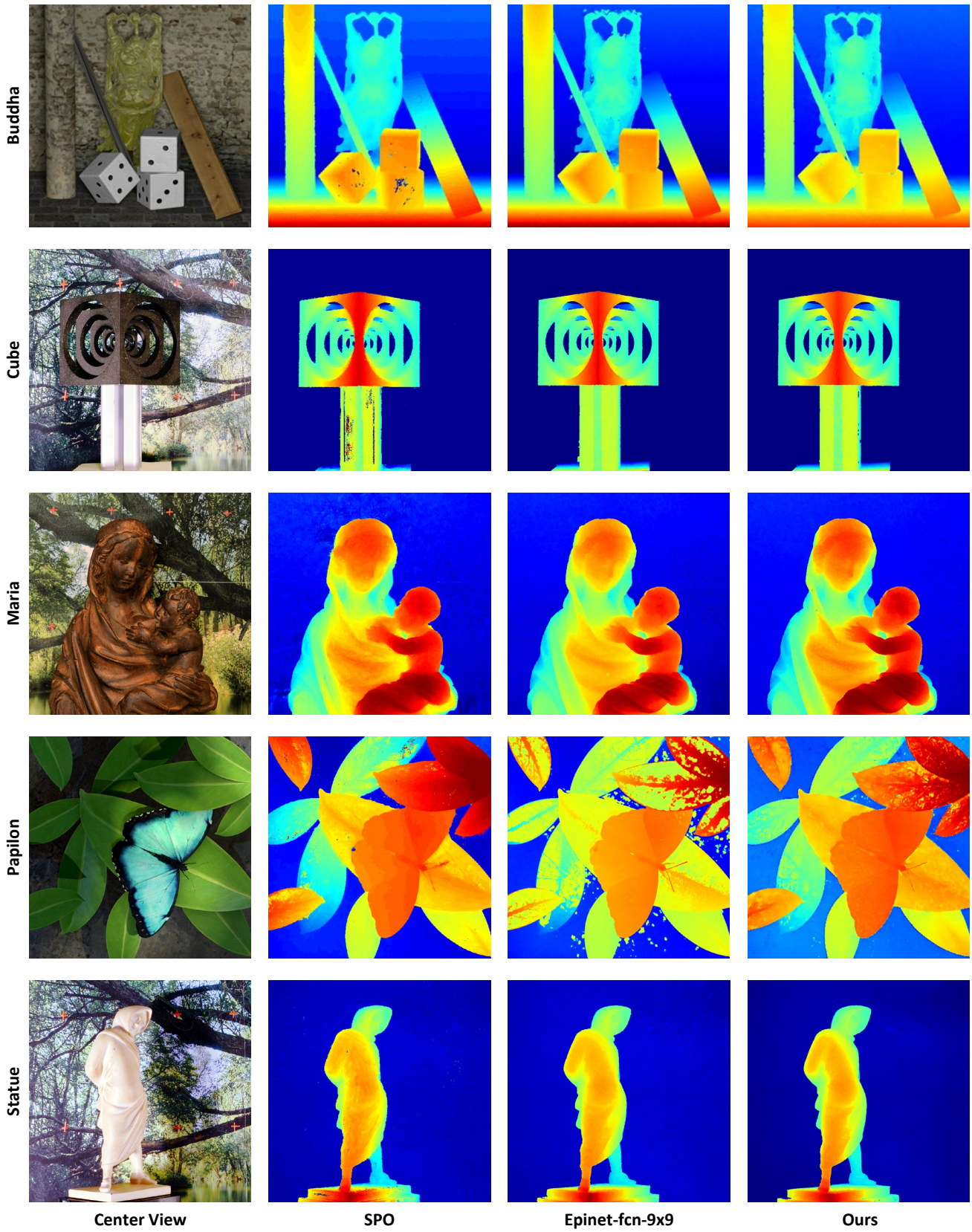

Figure VIII. Visual results achieved by SPO [19], EPINET [12], and our method on the old HCI LF dataset [14].

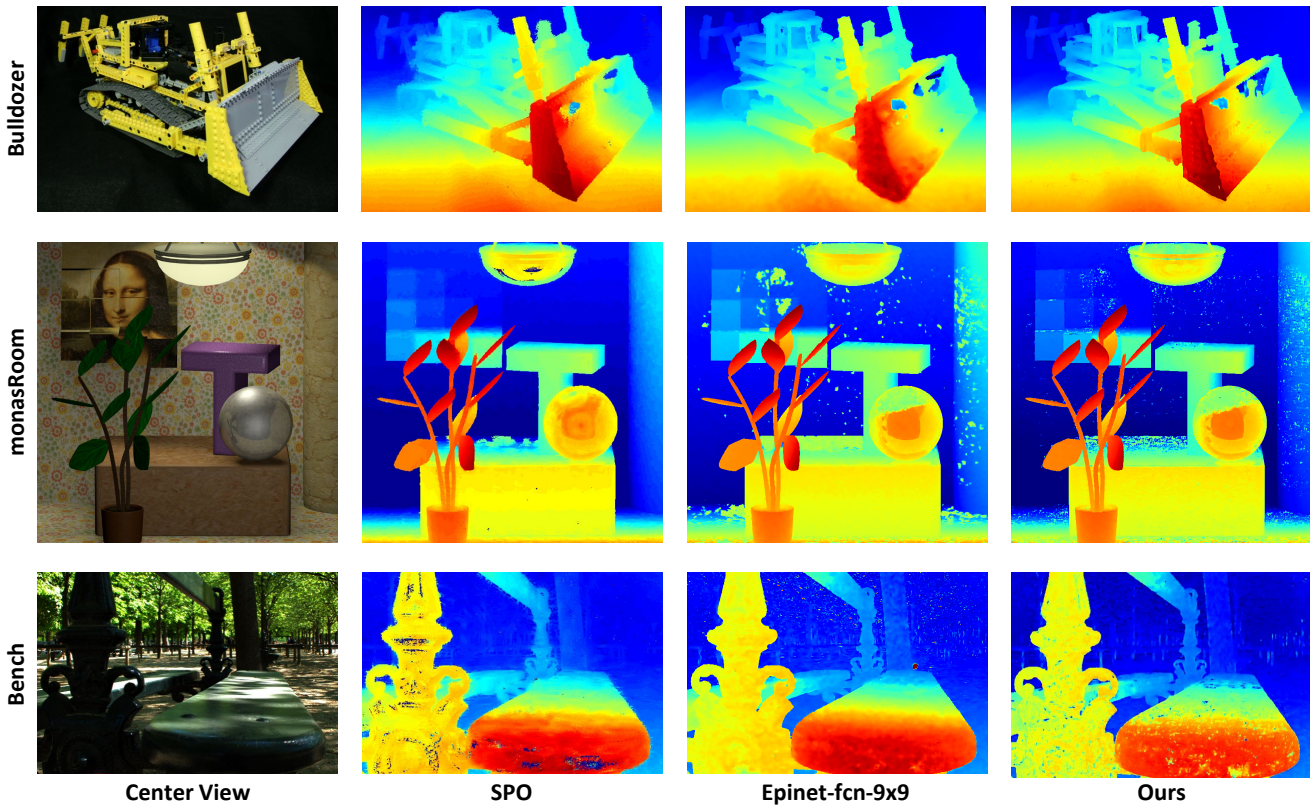

Figure IX. Visual results achieved by SPO [19], EPINET [12], and our method on three challenging scenes (i.e., *bulldozer* [14] with reflective surfaces, *monasRoom* [17] with repetitive textures, and *bench* [9] with illuminance variations).

References

[1] Kang Han, Wei Xiang, Eric Wang, and Tao Huang. A novel occlusion-aware vote cost for light field depth estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 5

[2] Katrin Honauer, Ole Johannsen, Daniel Kondermann, and Bastian Goldluecke. A dataset and evaluation methodology for depth estimation on 4d light fields. In *Asian Conference on Computer Vision (ACCV)*, pages 19–34, 2016. 3, 4, 5, 6

[3] Zhicong Huang, Xuemei Hu, Zhou Xue, Weizhu Xu, and Tao Yue. Fast light-field disparity estimation with multi-disparity-scale cost aggregation. In *International Conference on Computer Vision (ICCV)*, pages 6320–6329, 2021. 5

[4] Hae-Gon Jeon, Jaesik Park, Gyeongmin Choe, Jinsun Park, Yunsu Bok, Yu-Wing Tai, and In So Kweon. Depth from a light field image with learning-based matching costs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):297–310, 2018. 5

[5] Mikael Le Pendu, Xiaoran Jiang, and Christine Guillemot. Light field inpainting propagation via low rank matrix completion. *IEEE Transactions on Image Processing*, 27(4):1981–1993, 2018. 1, 2, 8

[6] Titus Leistner, Hendrik Schilling, Radek Mackowiak, Stefan Gumhold, and Carsten Rother. Learning to think outside the box: Wide-baseline light field depth estimation with epi-shift. In *International Conference on 3D Vision (3DV)*, pages 249–257, 2019. 5

[7] Kunyuan Li, Jun Zhang, Rui Sun, Xudong Zhang, and Jun Gao. Epi-based oriented relation networks for light field depth estimation. In *British Machine Vision Conference (BMVC)*, 2020. 5

[8] Yaoxiang Luo, Wenhui Zhou, Junpeng Fang, Linkai Liang, Hua Zhang, and Guojun Dai. Epi-patch based convolutional neural network for depth estimation on 4d light field. In *International Conference on Neural Information Processing (ICNIP)*, pages 642–652, 2017. 5

[9] Martin Rerabek and Touradj Ebrahimi. New light field image dataset. In *International Conference on Quality of Multimedia Experience (QoMEX)*, 2016. 1, 2, 8, 10

[10] Hendrik Schilling, Maximilian Diebold, Carsten Rother, and Bernd Jähne. Trust your model: Light field depth estimation with inline occlusion handling. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4530–4538, 2018. 5

[11] Hao Sheng, Pan Zhao, Shuo Zhang, Jun Zhang, and Da Yang. Occlusion-aware depth estimation for light field using multi-orientation epis. *Pattern Recognition*, 74:587–599, 2018. 5

[12] Changha Shin, Hae-Gon Jeon, Youngjin Yoon, In So Kweon, and Seon Joo Kim. Epinet: A fully-convolutional neural network using epipolar geometry for depth from light field im-

ages. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4748–4757, 2018. 2, 5, 7, 8, 9, 10

[13] Yu-Ju Tsai, Yu-Lun Liu, Ming Ouhyoung, and Yung-Yu Chuang. Attention-based view selection networks for light-field disparity estimation. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 34, pages 12095–12103, 2020. 5

[14] Vaibhav Vaish and Andrew Adams. The (new) stanford light field archive. *Computer Graphics Laboratory, Stanford University*, 6(7), 2008. 1, 2, 7, 8, 9, 10

[15] Ting-Chun Wang, Alexei A Efros, and Ravi Ramamoorthi. Occlusion-aware depth estimation using light-field cameras. In *IEEE International Conference on Computer Vision (ICCV)*, pages 3487–3495, 2015. 5

[16] Yingqian Wang, Longguang Wang, Gaochang Wu, Jungang Yang, Wei An, Jingyi Yu, and Yulan Guo. Disentangling light fields for super-resolution and disparity estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 5

[17] Sven Wanner, Stephan Meister, and Bastian Goldluecke. Datasets and benchmarks for densely sampled 4d light fields. In *Vision, Modelling and Visualization (VMV)*, volume 13, pages 225–226, 2013. 1, 2, 8, 10

[18] Williem, In Kyu Park, and Kyoung Mu Lee. Robust light field depth estimation using occlusion-noise aware data costs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(10):2484–2497, 2018. 5

[19] Shuo Zhang, Hao Sheng, Chao Li, Jun Zhang, and Zhang Xiong. Robust depth estimation for light field via spinning parallelogram operator. *Computer Vision and Image Understanding*, 145:148–159, 2016. 2, 5, 7, 8, 9, 10