Supplementary Material for "Single Image Reflection Separation via Component Synergy"

Qiming Hu Xiaojie Guo*
College of Intelligence and Computing, Tianjin University, Tianjin, China
huqiming@tju.edu.cn, xj.max.guo@gmail.com

A. Quantitative Comparison of Reflection

We present a quantitative comparison of reflection layers predicted by previous methods and ours on four real-world benchmark datasets in Table 1. It tells that our design achieves state-of-the-art overall performance over other competitors, which further demonstrates the effectiveness of DSRNet for handling the SIRS problem. Note that, we conduct a gamma correction for the results of all the methods with respect to their corresponding ground-truth reflection layers, in order to evaluate the content-related fidelity, rather than their magnitude. For a fair comparison, we leverage $\tilde{\mathbf{R}} = \mathbf{I} - \mathbf{T}$ as the reference in this comparison.

B. Additional Evaluation Metrics

The comparative evaluation results regarding the reconstruction quality of the transmission layer are presented in Table 2. This table showcases the outcomes based on three additional assessment metrics. Among them, DeltaE [7] reflects the accuracy of color restoration, DeepFeatures [11] displays the precision of semantic information restoration, while FID [3] conveys the reality of the reconstructed content. The outstanding performance exhibited across these additional indicators reaffirms the superiority of our approach in contrast to alternative methods.

C. Network Structure

The detailed designs of the two stages of our proposed DSRNet are listed in Table 4 and Table 5, respectively. We assume that the size of an input image is 224×224. The "Number" columns show the number of blocks that correspond to the "Block Name" in the architecture and the "Order" indicates their feedforward order in the network.

D. Additional Ablation Study

In order to better demonstrate the efficacy of the special designs proposed in the main body, an additional ablation study over a pure baseline (UNet) is conducted, as illustrated in Table 3, including the usage of the learnable residue module (LRM), the feature encoder without interaction (FE), and DSFNet (Base+DSFNet). As shown, the introduction of the LRM significantly improves the performance of the baseline. Introducing a feature encoder can also considerably promote its metrics, while the performance will be further improved by interacting the dual-stream features in the feature encoder.

E. Advantage over YTMT

We provide the curves of the training error of DSRNet with YTMT and our proposed MuGI feature interaction mechanisms in Fig. 1. The other settings are kept the same. It shows that the one with the MuGI mechanism obtains a faster speed of the training error reduction, which further demonstrates the information efficacy of MuGI.

F. More Visual Comparisons

More visual comparisons are provided in Figs. 2-11, including the predicted transmission layers with their corresponding reflection layers and residue terms. The real-world samples are drawn from the SIR^2 dataset [8] and the Real45 dataset provided by [2]. As can be seen, there are fewer residual reflection components in our separated transmission layers compared with other methods. Moreover, benefiting from the improved reflection model, our approach can simultaneously enhance the separated reflection layers, making them clearer for recognition, segmentation, etc. The reflection layers estimated by the previous approaches are either too weak (methods with the additive model, including Zhang *et al.* [12], RAGNet [6], and YTMT [4]) or mixed with unwanted transmission components (methods with the coefficient based and alpha blending map based models, including BDN [10], IBCLN [5] and Dong *et al.* [1]), while ours are much more attractive.

References

[1] Zheng Dong, Ke Xu, Yin Yang, Hujun Bao, Weiwei Xu, and Rynson W. H. Lau. Location-aware single image reflection

*Corresponding author.

Methods	Real20 (20)		Objects (200)		Postcard (199)		Wild (55)		Average	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Zhang *et al.* [12]	23.93	0.490	26.94	0.515	23.88	0.427	30.20	0.655	25.91	0.493
BDN [10]	22.29	0.427	27.78	**0.560**	23.72	0.414	30.12	0.627	26.12	0.501
IBCLN [5]	23.03	0.420	27.11	0.501	24.14	0.423	29.92	0.605	26.02	0.477
RAGNet [6]	23.05	0.367	26.98	0.455	24.05	0.364	29.66	0.586	25.90	0.428
YTMT [4]	23.37	0.475	25.47	0.406	22.97	0.442	29.12	0.629	24.76	0.450
Ours	**25.03**	**0.507**	**27.68**	0.543	**24.59**	**0.443**	**30.63**	**0.656**	**26.61**	**0.513**
Dong *et al.*† [1]	23.91	0.498	27.67	**0.561**	24.27	0.435	30.23	0.631	26.38	0.514
Ours†	**24.94**	**0.509**	**28.44**	0.560	**25.06**	**0.443**	**30.87**	**0.641**	**27.16**	**0.518**

Table 1: Quantitative results of reflections on four real-world benchmark datasets of methods. The best results are indicated in **bold**. † indicates extra training data are involved as in [1].

Metrics	Zhang *et al.*	ERRNet	IBCLN	YTMT	Dong *et al.*†	Ours	Ours†
DeltaE ↓	12.709	6.995	6.817	6.735	6.458	6.008	**5.708**
DeepFeatures ↓	1.104	0.083	0.086	0.081	0.066	0.066	**0.060**
FID ↓	2.683	0.354	0.516	0.327	0.360	0.325	**0.279**

Table 2: Quantitative results of the transmission restoration on three additional metrics, which is averaged on Real20 and SIR^2 datasets. The best results are indicated in **bold**. † indicates extra training data are involved as in [1].

Real20 + SIR^2	Base	Base+LRM	Base+FE	Base+DSFNet
PSNR	18.81	20.09	22.39	23.99
SSIM	0.818	0.838	0.845	0.887

Table 3: Additional ablation study with a simple UNet (Base) as the baseline.

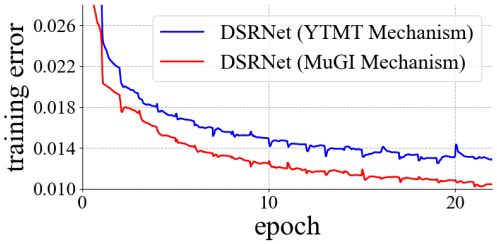


Figure 1: Training error of DSRNet with YTMT [4] and our proposed MuGI mechanisms, respectively. Our interaction strategy yields an accelerated rate of error reduction.

removal. In *ICCV*, pages 4997–5006, 2021. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

[2] Qingnan Fan, Jiaolong Yang, Gang Hua, Baoquan Chen, and David P. Wipf. A generic deep architecture for single image reflection removal and image smoothing. In *ICCV*, pages 3258–3267, 2017. 1, 9, 10, 11, 12

[3] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NeurIPS*, 30, 2017. 1

[4] Qiming Hu and Xiaojie Guo. Trash or treasure? an interactive dual-stream strategy for single image reflection separation. In *NeurIPS*, pages 24683–24694, 2021. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

[5] Chao Li, Yixiao Yang, Kun He, Stephen Lin, and John E. Hopcroft. Single image reflection removal through cascaded refinement. In *CVPR*, pages 3562–3571, 2020. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

[6] Yu Li, Ming Liu, Yaling Yi, Qince Li, Dongwei Ren, and Wangmeng Zuo. Two-stage single image reflection removal with reflection-aware guidance. *CoRR*, abs/2012.00945, 2020. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

[7] Gaurav Sharma, Wencheng Wu, and Edul N. Dalal. The ciede2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. *Color Research and Application*, 30(2):21–30, 2005. 1

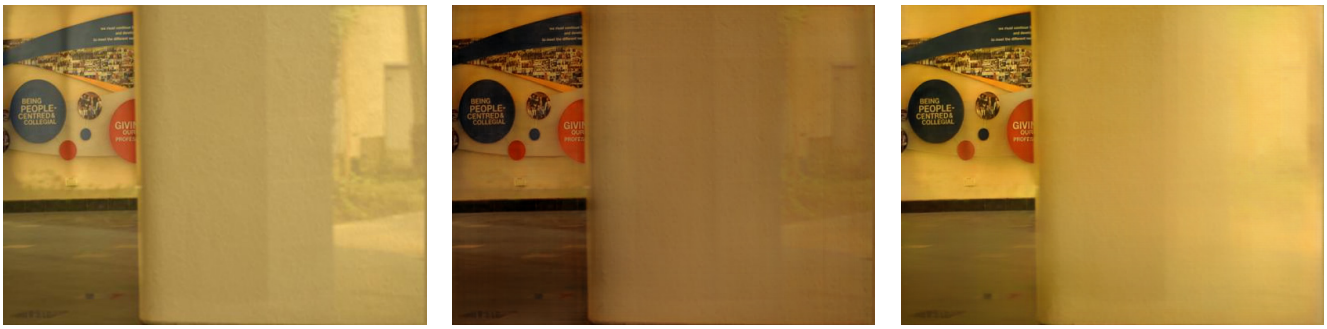
[8] Renjie Wan, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, and Alex C. Kot. Benchmarking single-image reflection removal algorithms. In *ICCV*, pages 3942–3950, 2017. 1, 3, 4, 5, 6, 7, 8

[9] Kaixuan Wei, Jiaolong Yang, Ying Fu, David P. Wipf, and Hua Huang. Single image reflection removal exploiting misaligned training data and network enhancements. In *CVPR*, pages 8178–8187, 2019. 3, 5, 7, 9, 11

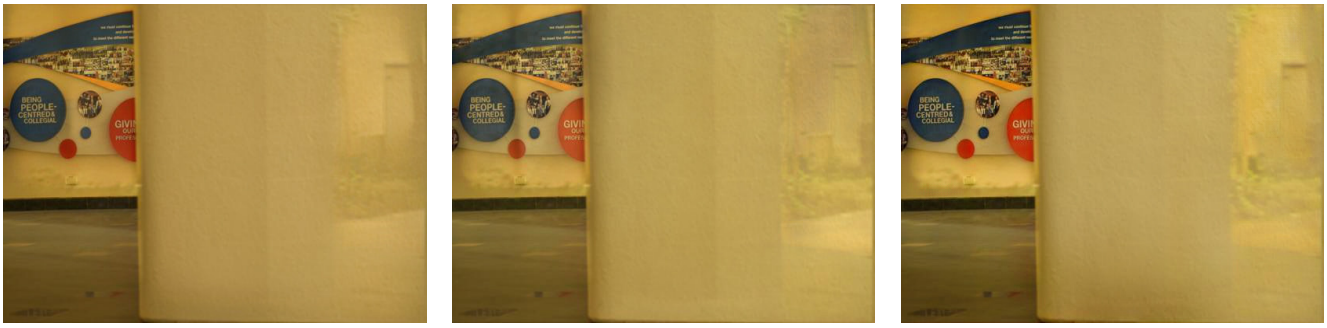
[10] Jie Yang, Dong Gong, Lingqiao Liu, and Qinfeng Shi. Seeing deeply and bidirectionally: A deep learning approach for single image reflection removal. In *ECCV*, pages 675–691, 2018. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

[11] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pages 586–595, 2018. 1

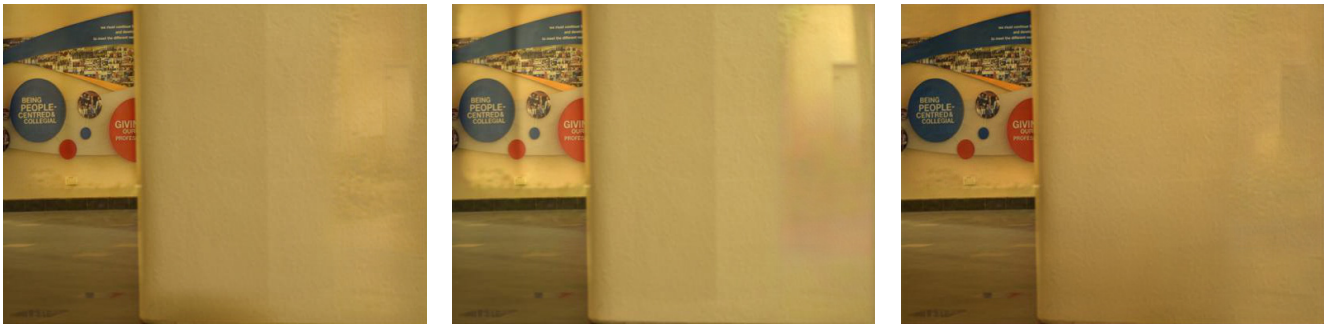
[12] Xuaner Cecilia Zhang, Ren Ng, and Qifeng Chen. Single image reflection separation with perceptual losses. In *CVPR*, pages 4786–4794, 2018. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12



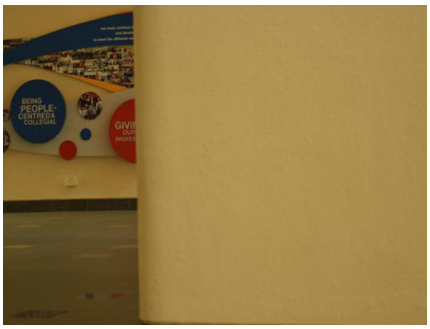
Input | Zhang *et al.* [12] | BDN [10]



ERRNet [9] | IBCLN [5] | RAGNet [6]



YTMT [4] | Dong *et al.* [1] | Ours



GT

Figure 2: Visual comparison of $\hat{T}$ between state-of-the-arts and ours on a sample from the SIR^2 dataset [8] ("055.jpg" in the Wild subset).

Figure 3: Visual comparison of $\hat{R}$ between state-of-the-arts and ours on a sample from the SIR^2 dataset [8] ("055.jpg" in the Wild subset).

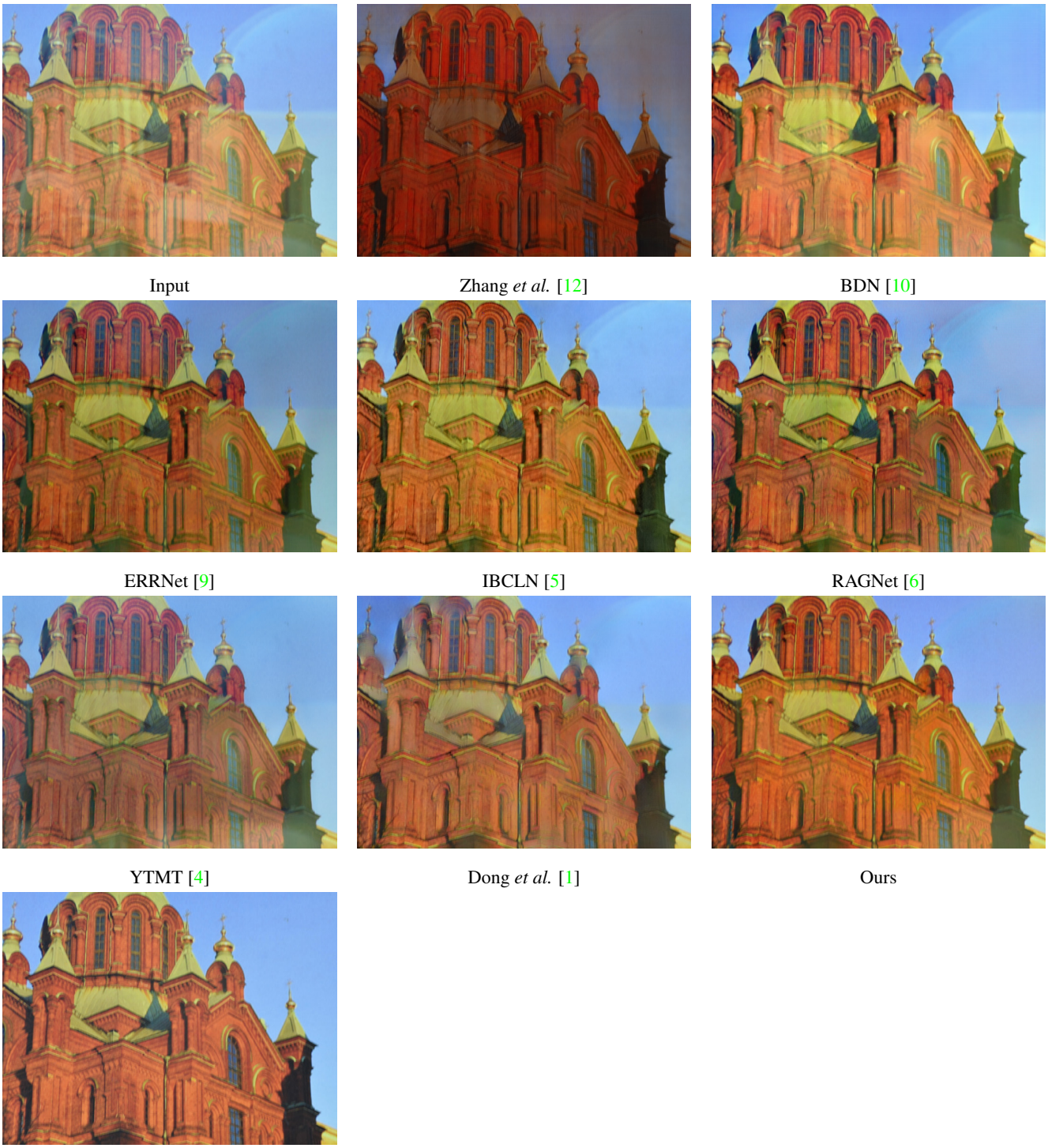


Figure 4: Visual comparison of $\hat{T}$ between state-of-the-arts and ours on a sample from the SIR^2 dataset [8] ("085.png" in the Postcard subset).

Input

Zhang *et al.* [12]

BDN [10]

IBCLN [5]

RAGNet [6]

YTMT [4]

Dong *et al.* [1]

Ours

Ours (Residue)

GT

Figure 5: Visual comparison of $\hat{R}$ between state-of-the-arts and ours on a sample from the SIR^2 dataset [8] ("085.png" in the Postcard subset).

Input

Zhang *et al.* [12]

BDN [10]

ERRNet [9]

IBCLN [5]

RAGNet [6]

YTMT [4]

Dong *et al.* [1]

Ours

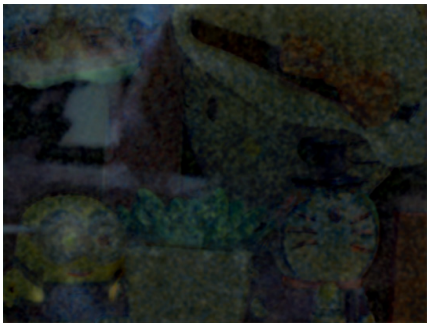
GT

Figure 6: Visual comparison of $\hat{T}$ between state-of-the-arts and ours on a sample from the SIR^2 dataset [8] ("053.jpg" in the SolidObject subset).

Input

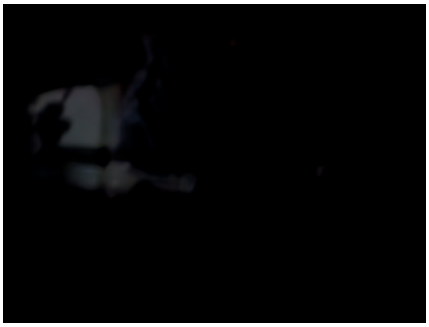
Zhang *et al.* [12]

BDN [10]



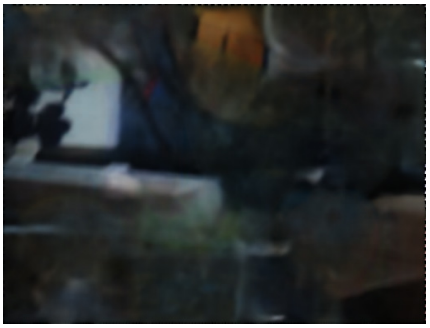
IBCLN [5]

RAGNet [6]

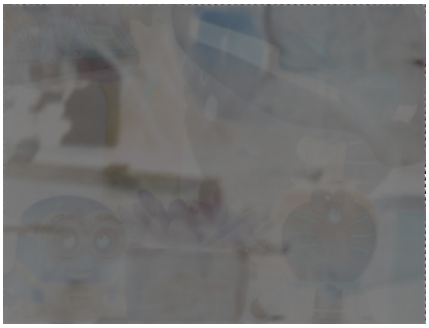


YTMT [4]

Dong *et al.* [1]



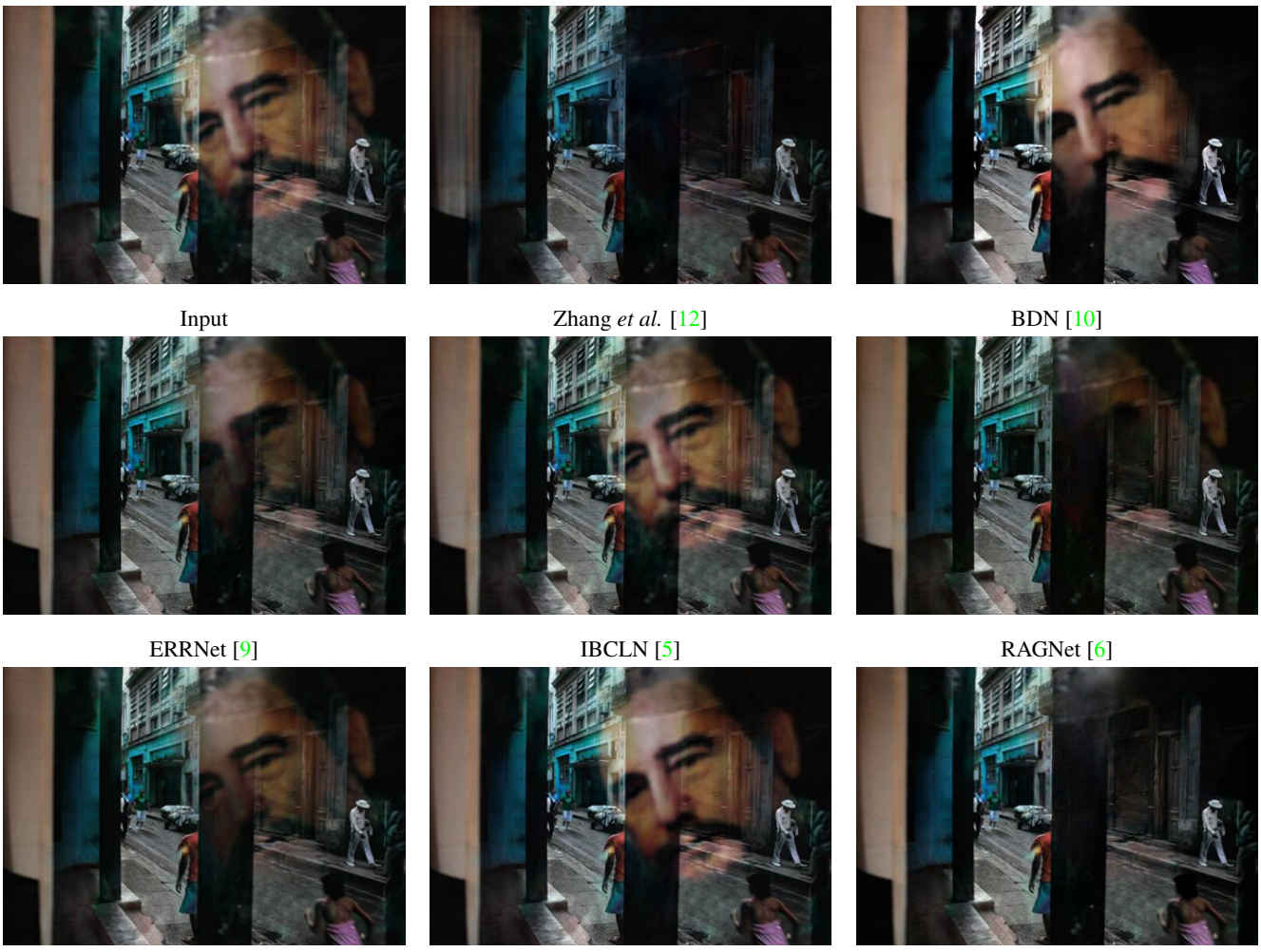
Ours



Ours (Residue)

GT

Figure 7: Visual comparison of $\hat{R}$ between state-of-the-arts and ours on a sample from the SIR^2 dataset [8] ("053.jpg" in the SolidObject subset).



Input | Zhang *et al.* [12] | BDN [10]

ERRNet [9] | IBCLN [5] | RAGNet [6]

YTMT [4] | Dong *et al.* [1] | Ours

Figure 8: Visual comparison of $\hat{T}$ between state-of-the-arts and ours on a sample from the real45 dataset [2] ("qingnan-new2-1-input.jpg").

Input | Zhang *et al.* [12] | BDN [10]

IBCLN [5] | RAGNet [6] | YTMT [4]

Dong *et al.* [1] | Ours

Figure 9: Visual comparison of $\hat{R}$ between state-of-the-arts and ours on a sample from the real45 dataset [2] ("qingnan-new2-1-input.jpg").

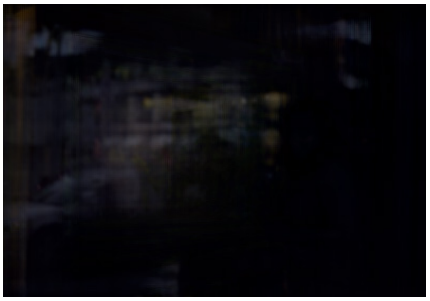
Input | Zhang *et al.* [12] | BDN [10]

ERRNet [9] | IBCLN [5] | RAGNet [6]

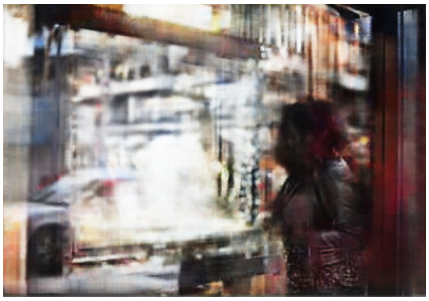
YTMT [4] | Dong *et al.* [1] | Ours

Figure 10: Visual comparison of $\hat{T}$ between state-of-the-arts and ours on a sample from the real45 dataset [2] ("qingnan-new2-31-input.jpg").

Input



Zhang *et al.* [12]



BDN [10]

IBCLN [5]

RAGNet [6]

YTMT [4]

Dong *et al.* [1]

Ours

Figure 11: Visual comparison of $\hat{R}$ between state-of-the-arts and ours on a sample from the real45 dataset [2] ("qingnan-new2-31-input.jpg").

Block Name	Output size	Branch 1	Branch 2	Order
VGG-19 Stage 1	224×224	$[3 \rightarrow 64]$	-	1
MuGI Block	224×224	$[64 \rightarrow 64]$	$[64 \rightarrow 64]$	14
VGG-19 Stage 2	112×112	$[64 \rightarrow 128]$	-	2
MuGI Block	112×112	$[128 \rightarrow 128]$	$[128 \rightarrow 128]$	12
VGG-19 Stage 3	56×56	$[128 \rightarrow 256]$	-	3
MuGI Block	56×56	$[256 \rightarrow 256]$	$[256 \rightarrow 256]$	10
VGG-19 Stage 4	28×28	$[256 \rightarrow 512]$	-	4
MuGI Block	28×28	$[512 \rightarrow 512]$	$[512 \rightarrow 512]$	8
VGG-19 Stage 5	14×14	$[512 \rightarrow 512]$	-	5
MuGI Block	14×14	$[512 \rightarrow 512]$	$[512 \rightarrow 512]$	6
BlinearUpsampling2x	28×28	$[512 \rightarrow 512]$	$[512 \rightarrow 512]$	7
DSF Block	56×56	$[1024 \rightarrow 256]$	$[1024 \rightarrow 256]$	9
DSF Block	112×112	$[512 \rightarrow 128]$	$[512 \rightarrow 128]$	11
DSF Block	224×224	$[256 \rightarrow 64]$	$[256 \rightarrow 64]$	13
Conv. 3x3	224×224	$[3 \rightarrow 32]$	$[3 \rightarrow 32]$	15
MuGI Block	224×224	$[32 \rightarrow 32]$	$[32 \rightarrow 32]$	16
DSF Block	224×224	$[128 \rightarrow 64]$	$[128 \rightarrow 64]$	17
Conv. 3x3	224×224	$[64 \rightarrow 64]$	$[64 \rightarrow 48]$	64

Table 4: Architecture of the DSFNet (the first stage of the DSRNet).

Block Name	Output size	Branch 1	Branch 2	Number	Order
MuGI Block	224×224	$[64 \rightarrow 64]$	$[64 \rightarrow 64]$	2	1
Conv. 2x2	112×112	$[64 \rightarrow 128]$	$[64 \rightarrow 128]$	1	2
MuGI Block	112×112	$[128 \rightarrow 128]$	$[128 \rightarrow 128]$	2	3
Conv. 2x2	56×56	$[128 \rightarrow 256]$	$[128 \rightarrow 256]$	1	4
MuGI Block	56×56	$[256 \rightarrow 256]$	$[256 \rightarrow 256]$	4	5
Conv. 2x2	28×28	$[256 \rightarrow 512]$	$[256 \rightarrow 512]$	1	6
MuGI Block	28×28	$[512 \rightarrow 512]$	$[512 \rightarrow 512]$	8	7
Conv. 2x2	14×14	$[512 \rightarrow 1024]$	$[512 \rightarrow 1024]$	1	8
MuGI Block	14×14	$[1024 \rightarrow 1024]$	$[1024 \rightarrow 1024]$	12	9
Conv. 1x1	14×14	$[1024 \rightarrow 2048]$	$[1024 \rightarrow 2048]$	1	10
PixelShuffle	28×28	$[2048 \rightarrow 512]$	$[2048 \rightarrow 512]$	1	11
MuGI Block	28×28	$[512 \rightarrow 512]$	$[512 \rightarrow 512]$	2	12
Conv. 1x1	28×28	$[512 \rightarrow 1024]$	$[512 \rightarrow 1024]$	1	13
PixelShuffle	56×56	$[1024 \rightarrow 256]$	$[1024 \rightarrow 256]$	1	14
MuGI Block	56×56	$[256 \rightarrow 256]$	$[256 \rightarrow 256]$	2	15
Conv. 1x1	56×56	$[256 \rightarrow 512]$	$[256 \rightarrow 512]$	1	16
PixelShuffle	112×112	$[512 \rightarrow 128]$	$[512 \rightarrow 128]$	1	17
MuGI Block	112×112	$[128 \rightarrow 128]$	$[128 \rightarrow 128]$	2	18
Conv. 1x1	112×112	$[128 \rightarrow 256]$	$[256 \rightarrow 512]$	1	19
PixelShuffle	224×224	$[256 \rightarrow 64]$	$[256 \rightarrow 64]$	1	20
MuGI Block	224×224	$[64 \rightarrow 64]$	$[64 \rightarrow 64]$	2	21
Conv. 3x3	224×224	$[64 \rightarrow 3]$	$[64 \rightarrow 3]$	1	22

Table 5: Architecture of the DSDNet (the second stage of the DSRNet).