Image Inpainting via Conditional Texture and Structure Dual Generation Supplementary Material

Xiefan Guo[1,2] Hongyu Yang[2*] Di Huang[1,2]
[1]State Key Laboratory of Software Development Environment, Beihang University, Beijing, China
[2]School of Computer Science and Engineering, Beihang University, Beijing, China
{xfguo,hongyuyang,dhuang}@buaa.edu.cn

Abstract

Supplementary material to the main paper.

1. Detailed Network Architecture

The detailed architecture of the generator is shown in Table 1. *In_feat* means the input feature map, *K_size* indicates the convolution kernel size, *Out_chs* tells the channel number of the output feature map, and *S_Uf* represents the stride of convolution or scale factor of upsampling. In the decoder, nearest neighbor upsampling is applied before each partial convolution and skip connections are utilized to combine low-level features with the high-level ones at multiple scales. *BN* indicates Batch Normalization, *Act_Func* represents the type of nonlinearity layer, *ReLU* denotes ReLU non-linear activation, LReLU indicates Leaky ReLU with the slope of 0.2, and $\boldsymbol{F}_{Module}$ represents the output of the corresponding module.

2. Visualization of Feature Maps

We visualize the feature maps learned by the Bi-GFF and CFA modules to give more insights into them.

Bi-directional Gated Feature Fusion (Bi-GFF). Bi-GFF is used to integrate the structure and texture information, enhancing their consistency. In particular, $\boldsymbol{G}_t$ and $\boldsymbol{G}_s$ are learned to control the integration degree. Dot product between them and two different kinds of feature maps are calculated, thus $\boldsymbol{G}_t$ and $\boldsymbol{G}_s$ have texture- and structure-specific responses, respectively, as shown in Figure 1 (c, d).

Contextual Feature Aggregation (CFA). The CFA module is developed to refine the generated contents by region affinity learning and multi-scale feature aggregation. In Figure 1 (e), we visualize the attention maps learned by CFA, which reveal that the module is aware of contextual semantics and it is able to model long-term spatial dependency. We further visualize the multi-scale weight maps, *i.e.*, $\boldsymbol{W}^1$, $\boldsymbol{W}^2$, $\boldsymbol{W}^4$ and $\boldsymbol{W}^8$, which can be adaptively adjusted to make the method suitable for different inpainting cases. Specifically, for most face images, the weight maps with dilation rate = 1 have a stronger intensity, making the model aggregate more details; while for facades, the weight maps with dilation rate = 2, 4 play a dominant role, making the model focus more on the intermediate features.

3. Additional Quantitative Comparion

In Table 2 and Table 3, we report the quantitative comparion of the proposed method and the current state-of-the-art on the CelebA and Paris StreetView datasets, respectively. Our method performs favorably against the others.

4. Additional Qualitative Comparion

More comparison examples on the CelebA, Paris StreetView and Places2 datasets are shown in Figure 2. It can be seen that our method generates more semantically plausible and photo-realistic results than its counterparts.

We also show more comparison with EdgeConnect [6] and PRVS [2] in Figure 3, mainly because these methods claim to improve results by reconstructing image structures as ours. The proposed model recovers more meaningful structures, leading to better results.

5. Additional Visual Results

Figure 4, 5, and 6 show more visual results of our approach achieved on the CelebA, Paris StreetView and Places2 datasets, respectively.

6. Object Removal Results

Figure 7 shows the results of our approach on the object removal task, which demonstrates its effectiveness, practicality and generalization ability.

[*] Corresponding author.

References

[1] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM TOG*, 28(3):24, 2009. 4, 5

[2] Jingyuan Li, Fengxiang He, Lefei Zhang, Bo Du, and Dacheng Tao. Progressive reconstruction of visual structure for image inpainting. In *ICCV*, 2019. 1, 4, 6

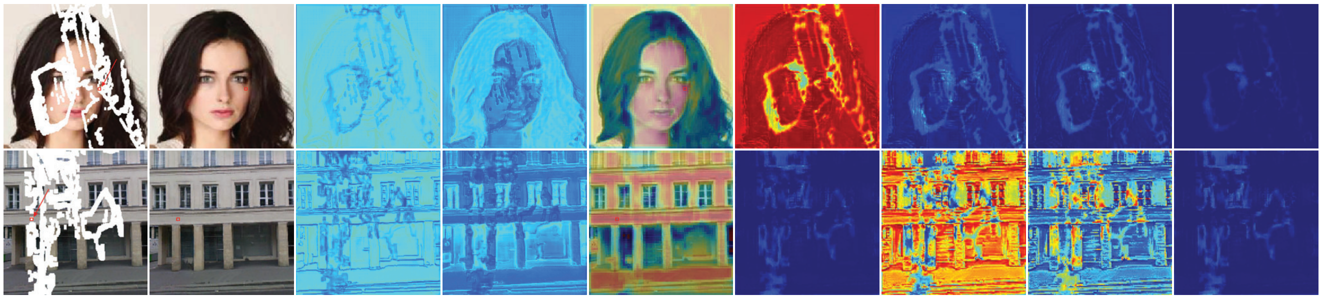
[3] Jingyuan Li, Ning Wang, Lefei Zhang, Bo Du, and Dacheng Tao. Recurrent feature reasoning for image inpainting. In *CVPR*, 2020. 4, 5

[4] Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *ECCV*, 2018. 4, 5

[5] Hongyu Liu, Bin Jiang, Yibing Song, Wei Huang, and Chao Yang. Rethinking image inpainting via a mutual encoder-decoder with feature equalizations. In *ECCV*, 2020. 4, 5

[6] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Qureshi, and Mehran Ebrahimi. Edgeconnect: Structure guided image inpainting using edge prediction. In *ICCVW*, 2019. 1, 4, 6

[7] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. In *ICCV*, 2019. 4, 5



(a) Input (b) Ours (c) $\boldsymbol{G}_t$ (d) $\boldsymbol{G}_s$ (e) attention map (f) $\boldsymbol{W}^1$ (g) $\boldsymbol{W}^2$ (h) $\boldsymbol{W}^4$ (i) $\boldsymbol{W}^8$

Figure 1: Visualization of the feature maps learned by the network.

Image Encoder, Edge Decoder							**Edge Encoder, Image Decoder**						
Module	*In_Feat*	*K_Size*	*Out_chs*	*S_Uf*	*BN*	*Act_Func*	*Module*	*In_Feat*	*K_Size*	*Out_chs*	*S_Uf*	*BN*	*Act_Func*
I_ec_1	$\boldsymbol{I}_{in}$	7	64	2	N	ReLU	E_ec_1	$\boldsymbol{E}_{in}, \boldsymbol{I}_{in}$	7	64	2	N	ReLU
I_ec_2	$\boldsymbol{F}_{I\_ec\_1}$	5	128	2	Y	ReLU	E_ec_2	$\boldsymbol{F}_{E\_ec\_1}$	5	128	2	Y	ReLU
I_ec_3	$\boldsymbol{F}_{I\_ec\_2}$	5	256	2	Y	ReLU	E_ec_3	$\boldsymbol{F}_{E\_ec\_2}$	5	256	2	Y	ReLU
I_ec_4	$\boldsymbol{F}_{I\_ec\_3}$	3	512	2	Y	ReLU	E_ec_4	$\boldsymbol{F}_{E\_ec\_3}$	3	512	2	Y	ReLU
I_ec_5	$\boldsymbol{F}_{I\_ec\_4}$	3	512	2	Y	ReLU	E_ec_5	$\boldsymbol{F}_{E\_ec\_4}$	3	512	2	Y	ReLU
I_ec_6	$\boldsymbol{F}_{I\_ec\_5}$	3	512	2	Y	ReLU	E_ec_6	$\boldsymbol{F}_{E\_ec\_5}$	3	512	2	Y	ReLU
I_ec_7	$\boldsymbol{F}_{I\_ec\_6}$	3	512	2	Y	ReLU	E_ec_7	$\boldsymbol{F}_{E\_ec\_6}$	3	512	2	Y	ReLU
E_dc_1	$\boldsymbol{F}_{I\_ec\_7}$		512	2			I_dc_1	$\boldsymbol{F}_{E\_ec\_7}$		512	2		
	, $\boldsymbol{F}_{E\_ec\_6}$	3	512	1	Y	LReLU		, $\boldsymbol{F}_{I\_ec\_6}$	3	512	1	Y	LReLU
E_dc_2	$\boldsymbol{F}_{E\_dc\_1}$		512	2			I_dc_2	$\boldsymbol{F}_{I\_dc\_1}$		512	2		
	, $\boldsymbol{F}_{E\_ec\_5}$	3	512	1	Y	LReLU		, $\boldsymbol{F}_{I\_ec\_5}$	3	512	1	Y	LReLU
E_dc_3	$\boldsymbol{F}_{E\_dc\_2}$		512	2			I_dc_3	$\boldsymbol{F}_{I\_dc\_2}$		512	2		
	, $\boldsymbol{F}_{E\_ec\_4}$	3	512	1	Y	LReLU		, $\boldsymbol{F}_{I\_ec\_4}$	3	512	1	Y	LReLU
E_dc_4	$\boldsymbol{F}_{E\_dc\_3}$		512	2			I_dc_4	$\boldsymbol{F}_{I\_dc\_3}$		512	2		
	, $\boldsymbol{F}_{E\_ec\_3}$	3	256	1	Y	LReLU		, $\boldsymbol{F}_{I\_ec\_3}$	3	256	1	Y	LReLU
E_dc_5	$\boldsymbol{F}_{E\_dc\_4}$		256	2			I_dc_5	$\boldsymbol{F}_{I\_dc\_4}$		256	2		
	, $\boldsymbol{F}_{E\_ec\_2}$	3	128	1	Y	LReLU		, $\boldsymbol{F}_{I\_ec\_2}$	3	128	1	Y	LReLU
E_dc_6	$\boldsymbol{F}_{E\_dc\_5}$		128	2			I_dc_6	$\boldsymbol{F}_{I\_dc\_5}$		128	2		
	, $\boldsymbol{F}_{E\_ec\_1}$	3	64	1	Y	LReLU		, $\boldsymbol{F}_{I\_ec\_1}$	3	64	1	Y	LReLU
E_dc_7	$\boldsymbol{F}_{E\_dc\_6}$		64	2			I_dc_7	$\boldsymbol{F}_{I\_dc\_7}$		64	2		
	, $\boldsymbol{E}_{in}, \boldsymbol{I}_{in}$	3	64	1	Y	LReLU		, $\boldsymbol{I}_{in}$	3	64	1	Y	LReLU
Bi-GFF module													
CFA module													
Output: Conv.(1, 1, 3); Tanh													

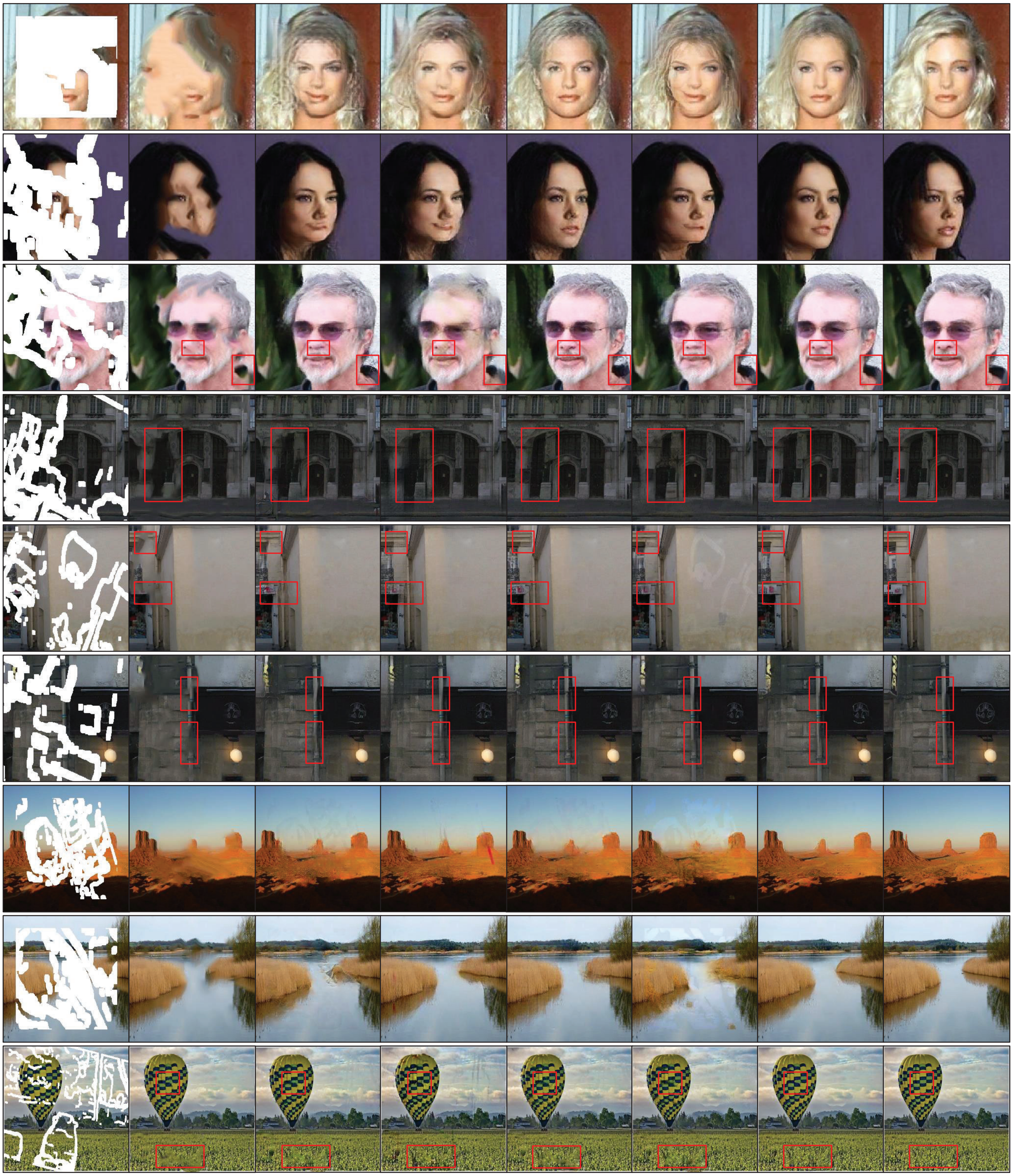
Table 1: The detailed architecture of the generator.

Metrics	LPIPS†			PSNR¶			SSIM¶		
Mask Ratio	0-20%	20-40%	40-60%	0-20%	20-40%	40-60%	0-20%	20-40%	40-60%
PatchMatch [1]	0.059	0.202	0.371	29.81	23.49	18.77	0.878	0.704	0.516
PConv [4]	0.046	0.122	0.221	31.89	26.48	21.32	0.899	0.750	0.558
DeepFillv2 [7]	0.040	0.107	0.214	32.48	26.93	21.70	0.906	0.757	0.569
RFR [3]	0.031	0.090	0.185	33.50	27.63	22.69	0.916	0.780	0.603
EdgeConnect [6]	0.042	0.117	0.215	32.12	26.79	21.66	0.904	0.758	0.566
PRVS [2]	0.039	0.112	0.209	32.34	26.89	21.78	0.908	0.762	0.573
MED [5]	0.037	0.106	0.203	32.68	27.01	21.86	0.907	0.763	0.575
Ours	**0.028**	**0.081**	**0.179**	**33.91**	**27.73**	**22.70**	**0.920**	**0.788**	**0.609**

Table 2: Objective quantitative comparison on CelebA (†Lower is better; ¶Higher is better).

Metrics	LPIPS†			PSNR¶			SSIM¶		
Mask Ratio	0-20%	20-40%	40-60%	0-20%	20-40%	40-60%	0-20%	20-40%	40-60%
PatchMatch [1]	0.078	0.195	0.362	30.70	25.31	20.59	0.881	0.689	0.499
PConv [4]	0.058	0.133	0.273	32.05	26.66	22.17	0.898	0.741	0.538
DeepFillv2 [7]	0.050	0.128	0.269	32.31	26.92	22.48	0.905	0.752	0.551
RFR [3]	0.041	0.112	0.234	32.69	27.33	22.76	0.919	0.772	0.568
EdgeConnect [6]	0.053	0.129	0.262	31.98	26.70	22.39	0.903	0.757	0.554
PRVS [2]	0.051	0.125	0.254	32.23	26.89	22.50	0.910	0.762	0.563
MED [5]	0.050	0.122	0.248	32.36	26.97	22.44	0.915	0.760	0.559
Ours	**0.039**	**0.107**	**0.226**	**32.93**	**27.48**	**22.89**	**0.923**	**0.777**	**0.573**

Table 3: Objective quantitative comparison on Paris StreetView (†Lower is better; ¶Higher is better).



(a) Input (b) PatchMatch (c) PConv (d) DeepFillv2 (e) RFR (f) MED (g) Ours (h) Ground-truth

Figure 2: Qualitative comparison on CelebA, Paris StreetView and Places2 (zoom in for a better view): (a) input corrupted images, (b) PatchMatch [1], (c) PConv [4], (d) DeepFillv2 [7], (e) RFR [3], (f) MED [5], (g) Ours, and (h) ground-truth images.

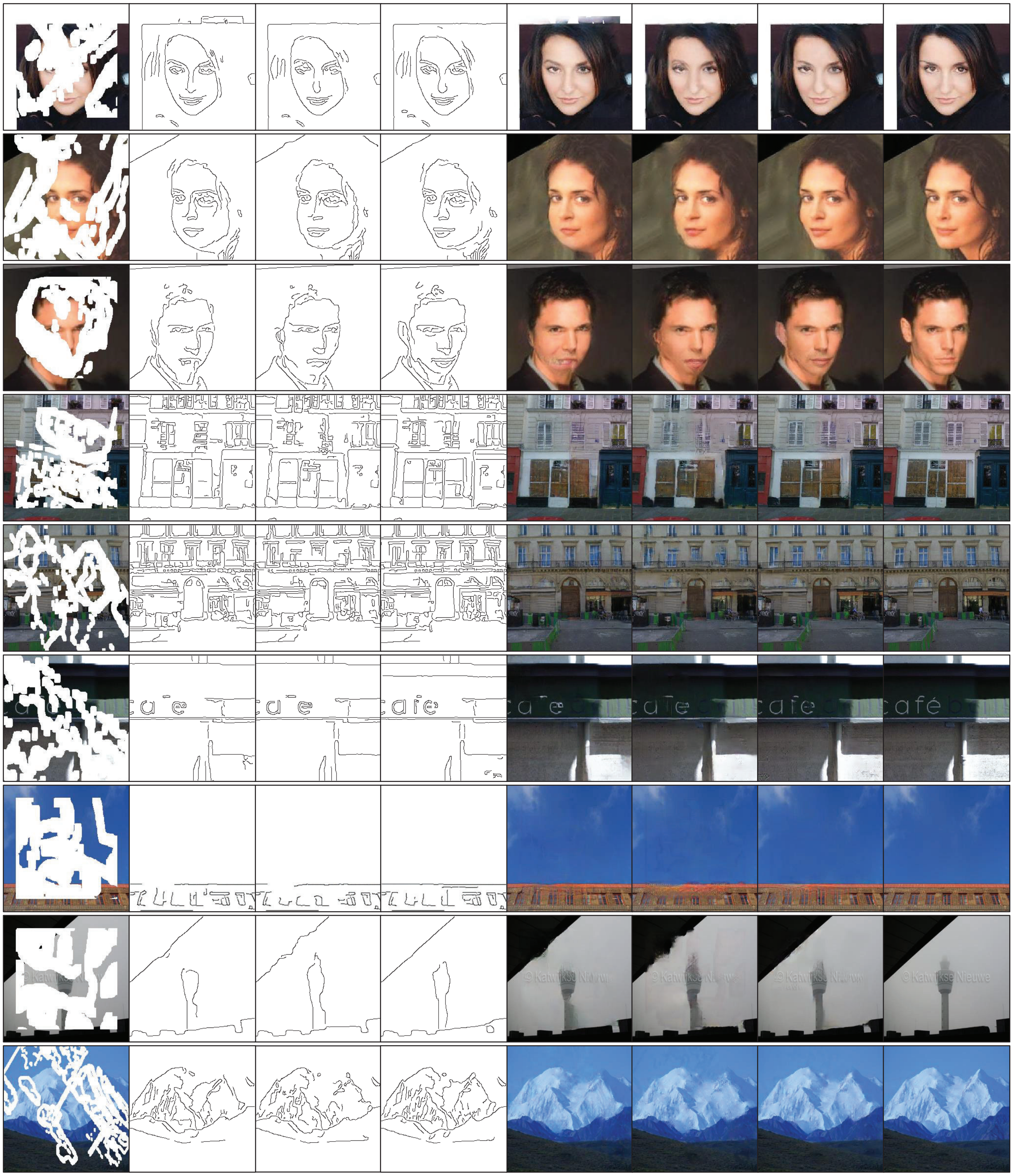
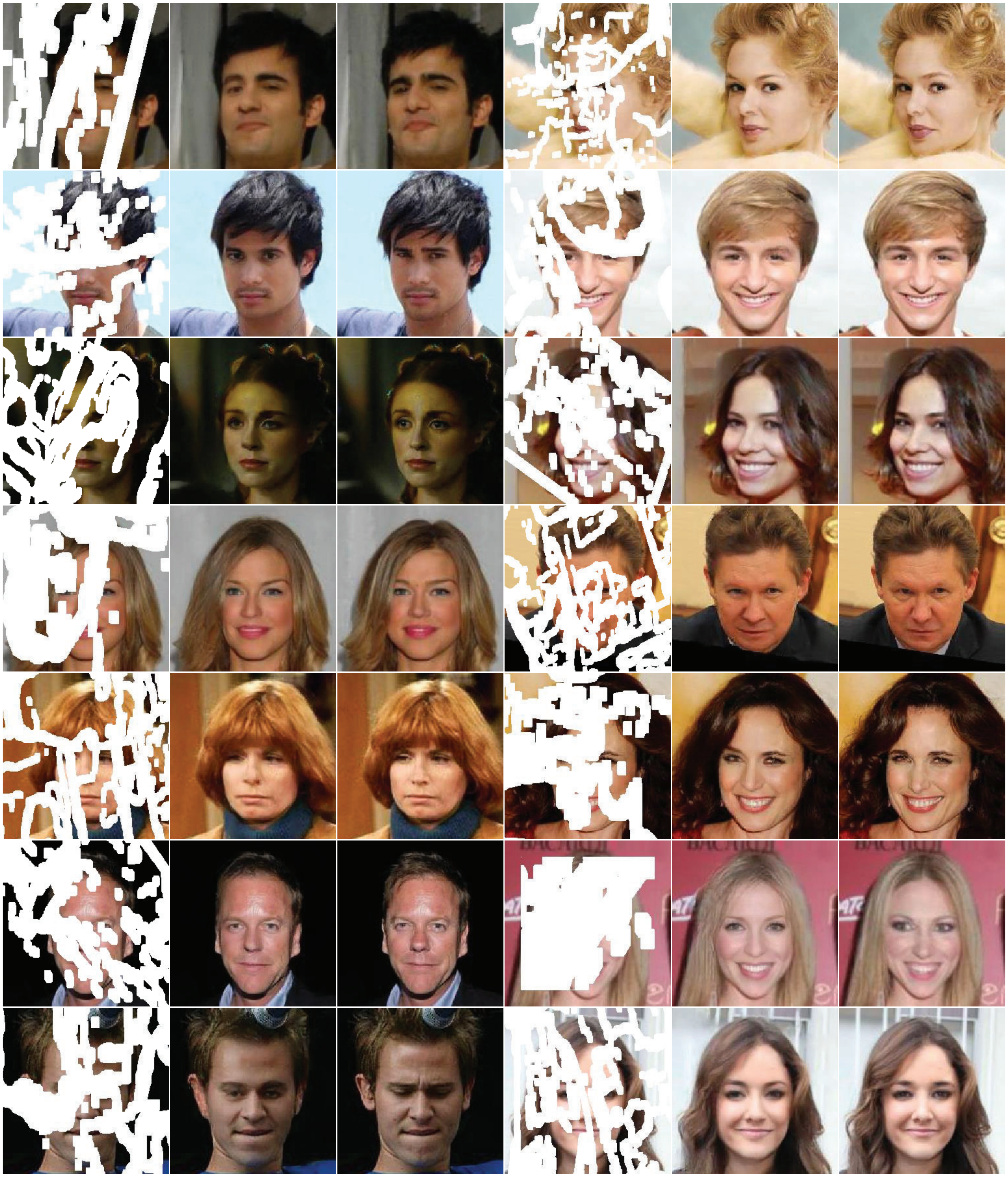
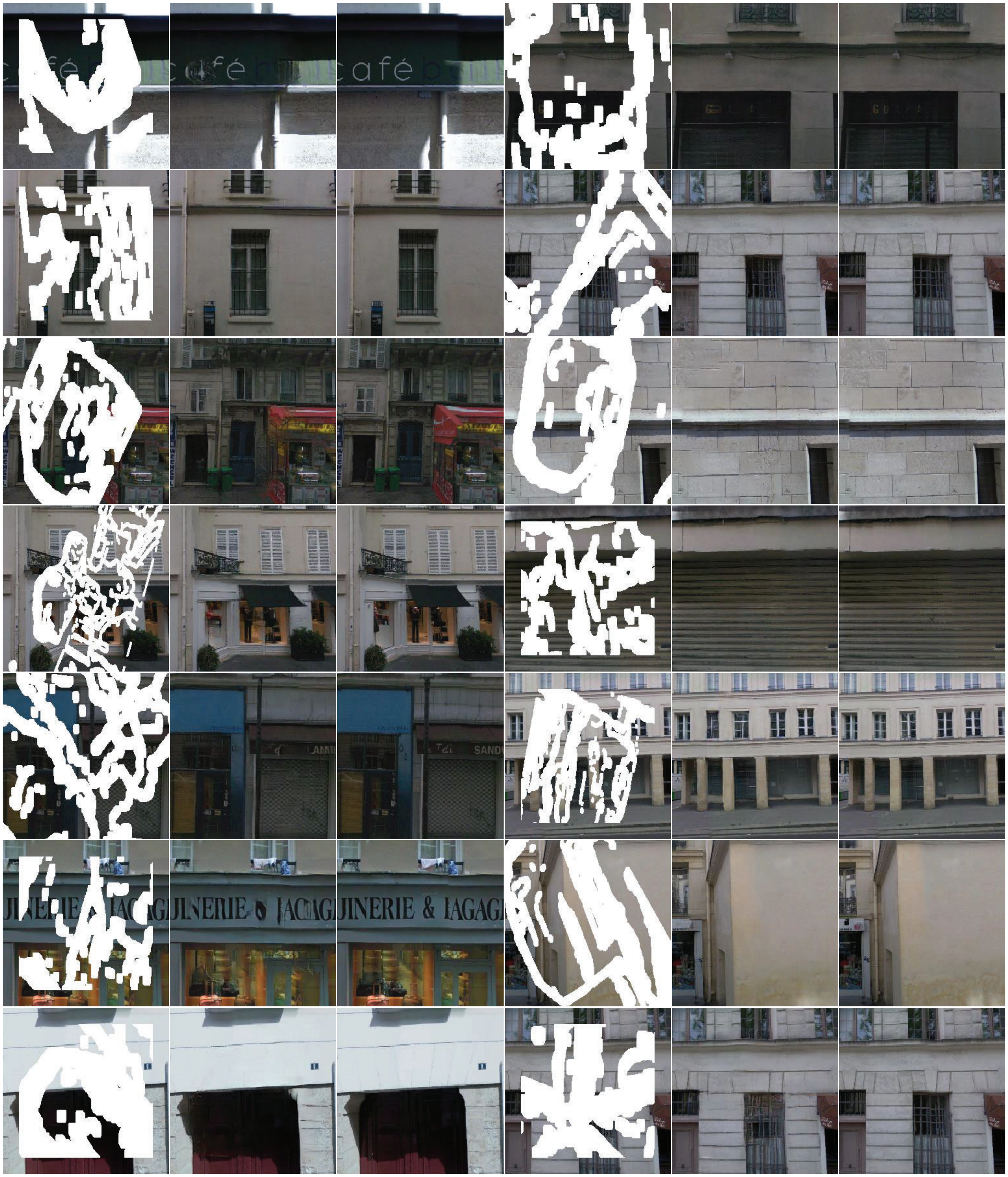


Figure 3: Visual comparison of different structure-based methods on CelebA, Paris StreetView and Places2 (zoom in for a better view): (a) input corrupted images; (b, c, d) reconstructed structures of EdgeConnect [6], PRVS [2] and Ours; (e, f, g) corresponding filled results of EdgeConnect [6], PRVS [2] and Ours; and (h) ground-truth images.



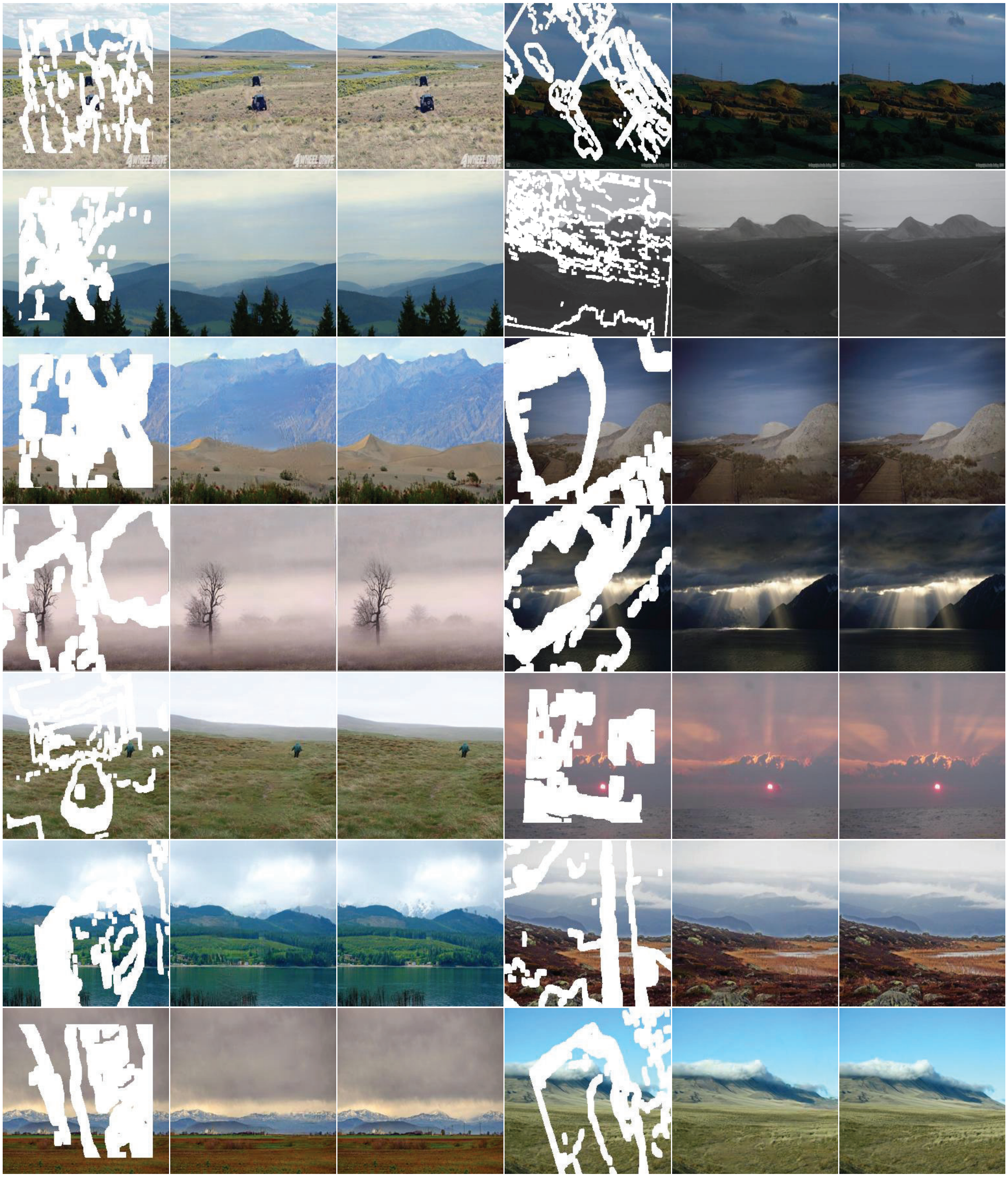
(a) Input (b) Ours (c) Ground-truth (d) Input (e) Ours (f) Ground-truth

Figure 4: Visual results on CelebA (zoom in for a better view): (a, d) input corrupted images, (b, e) our results, and (c, f) ground-truth images.



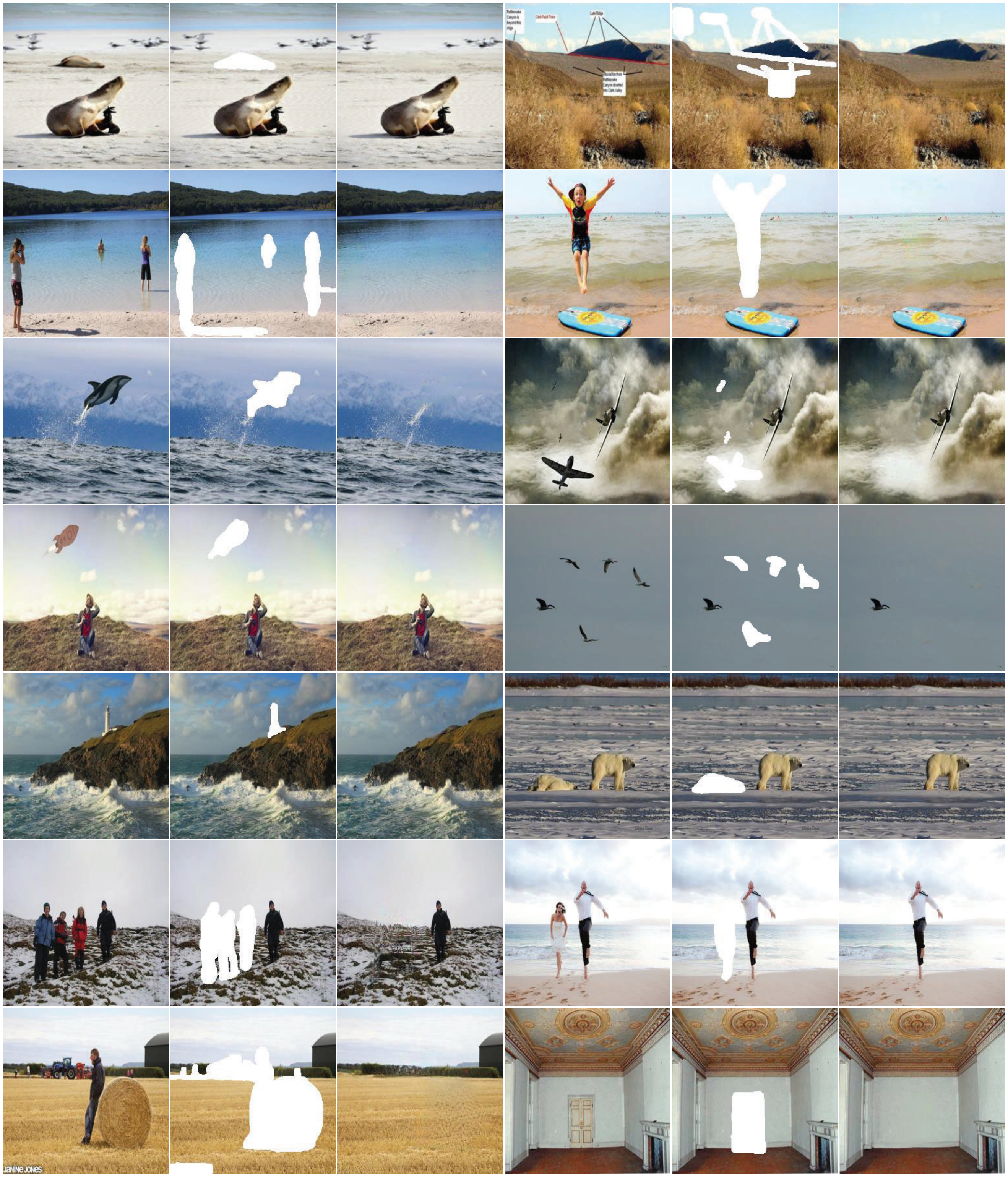
(a) Input (b) Ours (c) Ground-truth (d) Input (e) Ours (f) Ground-truth

Figure 5: Visual results on Paris StreetView (zoom in for a better view): (a, d) input corrupted images, (b, e) our results, and (c, f) ground-truth images.



(a) Input (b) Ours (c) Ground-truth (d) Input (e) Ours (f) Ground-truth

Figure 6: Visual results on Places2 (zoom in for a better view): (a, d) input corrupted images, (b, e) our results, and (c, f) ground-truth images.


Rattlesnake Canyon is beyond this ridge
Clark Fault Trace
Late Ridge
Alluvial fan from Rattlesnake Canyon diverted into Clark Valley
JANINE JONES

(a) Original Image (b) Input (c) Ours (d) Original Image (e) Input (f) Ours

Figure 7: Object removal results (zoom in for a better view): (a, d) original images, (b, e) input images, and (c, f) our results.