EdgeConnect: Structure Guided Image Inpainting using Edge Prediction (Supplementary Material)

Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Qureshi, and Mehran Ebrahimi
University of Ontario Institute of Technology, Canada
{kamyar.nazeri, eric.ng, tony.joseph, faisal.qureshi, mehran.ebrahimi}@uoit.ca
http://www.ImagingLab.ca http://www.VCLab.ca

1. Network Architectures

1.1. Generators

We follow a similar naming convention as those presented in [7]. Let `c7s1-k` denote a 7×7 Convolution-SpectralNorm-InstanceNorm-ReLU layer with k filters and stride 1 with reflection padding. Let `dk` denote a 4×4 Convolution-SpectralNorm-InstanceNorm-ReLU layer with k filters and stride 2 for down-sampling. Let `uk` be defined in the same manner as `dk` with transpose convolution for up-sampling. Let `Rk` denote a residual block of channel size k across both layers. We use dilated convolution in the first layer of `Rk` with dilation factor of 2, followed by spectral normalization and instance normalization.

The architecture of our generators is adopted from the model proposed by Johnson *et al.* [3]:
`c7s1-64, d128, d256, R256, R256, R256, R256, R256, R256, R256, R256, u128, u64, c7s1-*`.

The final layer `c7s1-*` varies depending on the generator. In the edge generator G_1, `c7s1-*` has channel size of 1 with sigmoid activation for edge prediction. In the image completion network G_2, `c7s1-*` has channel size of 3 with $\tanh$ (scaled) activation for the prediction of RGB pixel intensities. In addition, we remove spectral normalization from all layers of G_2.

1.2. Discriminators

The discriminators D_1 and D_2 follow the same architecture based on the 70×70 PatchGAN [2, 7]. Let `Ck-s` denote a 4×4 Convolution-SpectralNorm-LeakyReLU layer with k filters of stride s. The discriminators have the architecture `C64-2, C128-2, C256-2, C512-1, C1-1`. The final convolution layer produces scores predicting whether 70×70 overlapping image patches are real or fake. LeakyReLU [5] is employed with slope 0.2.

2. Experimental Results

	Mask	Precision	Recall
CelebA	0-10%	51.38	48.64
	10-20%	46.05	42.28
	20-30%	40.98	36.97
	30-40%	35.96	30.57
	40-50%	32.34	25.48
	50-60%	30.17	20.26
Places2	0-10%	48.68	46.70
	10-20%	43.55	41.22
	20-30%	38.71	36.20
	30-40%	34.51	31.36
	40-50%	31.85	27.04
	50-60%	30.53	22.42
PSV	0-10%	56.57	53.95
	10-20%	52.03	48.71
	20-30%	47.56	43.35
	30-40%	43.63	38.07
	40-50%	41.19	32.93
	50-60%	39.44	27.48

Table 1: Quantitative performance (256×256) of our edge generator G_1 trained on Canny edges.

We provide additional results produced by our model over the following datasets:

- CelebA ($202,599$ images)
- CelebHQ ($30,000$ images)
- Places2 (10 million+ images)
- Paris StreetView ($14,900$ images)

For CelebA, we crop the center of the image and resize it to the appropriate resolution. For Paris StreetView, since the images in the dataset are elongated (936×537), we separate each image into three: 1) Left 537×537, 2) middle 537×537, 3) right 537×537, of the image for a total of $44,700$ images. All images are rescaled to 256×256 for quantitative results, and 512×512 for qualitative results.

		Hybrid		Canny	
	Mask	G_1	GT	G_1	GT
ℓ_1 (%)†	0-10%	0.31	0.23	0.29	0.25
	10-20%	0.79	0.55	0.76	0.59
	20-30%	1.42	0.93	1.38	1.00
	30-40%	2.19	1.35	2.13	1.45
	40-50%	3.10	1.82	3.03	1.97
	50-60%	4.95	2.61	4.89	2.88
SSIM*	0-10%	0.985	0.990	0.985	0.988
	10-20%	0.959	0.978	0.961	0.972
	20-30%	0.926	0.959	0.928	0.951
	30-40%	0.886	0.940	0.890	0.930
	40-50%	0.841	0.920	0.846	0.906
	50-60%	0.767	0.891	0.771	0.872
PSNR*	0-10%	39.24	42.43	39.60	41.77
	10-20%	33.26	37.48	33.51	36.81
	20-30%	29.80	34.65	30.02	34.00
	30-40%	27.21	32.59	27.39	31.92
	40-50%	25.12	30.87	25.28	30.21
	50-60%	22.03	28.49	22.11	27.68
FID†	0-10%	0.22	0.11	0.20	0.13
	10-20%	0.56	0.24	0.53	0.31
	20-30%	1.13	0.41	1.08	0.57
	30-40%	1.90	0.61	1.80	0.88
	40-50%	2.99	0.83	2.82	1.25
	50-60%	5.67	1.14	5.30	1.79

Table 2: Comparison of quantitative results (256×256) between Hybrid (HED$\odot$Canny) and Canny edges over CelebA. Statistics are shown for generated edges (G_1) and ground truth edges (GT). †Lower is better. *Higher is better.

Accuracy of Edge Generator Table 1 shows the accuracy of our edge generator G_1 across all three datasets. We measure precision and recall for various mask sizes.

Comprehensive Results Tables 3 and 4 shows the quantitative performance of our model compared to existing methods over the datasets CelebA and Paris StreetView. Figures 2, 3 and 4 display these results graphically. Additional inpainting results of our proposed model are shown in figures 5 and 6.

3. Alternative Edge Generating Systems

We compare the quantitative results between Canny and a combination of HED and Canny edges (*i.e.* HED$\odot$Canny). Generated images based on the combined edges gave the best performance. However, our generator G_1 is unable to generate these type of edges accurately during training. Table 2 shows G_1 trained on HED$\odot$Canny had the poorest performance out of all methods despite its ground truth counterpart achieving the best performance. Figure 1 shows the results of G_1 trained using hybrid edges.

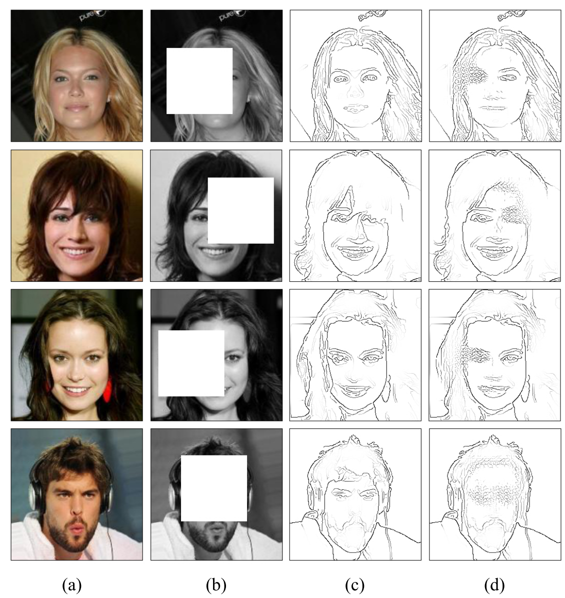

Figure 1: Generated edges by G_1 trained using hybrid (HED$\odot$Canny) edges (512×512). Images are best viewed in color. (a) Original Image. (b) Image with Masked Region. (c) Ground Truth Edges. (d) Generated Edges.

	Mask	CA	GLCIC	PConv	Ours
ℓ_1 (%)†	0-10%	1.33	0.91	**0.29**	**0.29**
	10-20%	2.48	2.53	0.78	**0.76**
	20-30%	3.98	4.67	1.42	**1.38**
	30-40%	5.64	6.95	2.19	**2.13**
	40-50%	7.35	9.18	3.08	**3.03**
	50-60%	9.21	11.21	4.96	**4.89**
	Fixed	2.80	3.83	**2.35**	2.39
SSIM*	0-10%	0.947	0.947	**0.985**	**0.985**
	10-20%	0.888	0.865	0.956	**0.961**
	20-30%	0.819	0.773	0.924	**0.928**
	30-40%	0.750	0.689	0.884	**0.890**
	40-50%	0.678	0.609	0.840	**0.846**
	50-60%	0.614	0.560	0.768	**0.771**
	Fixed	0.882	0.847	**0.891**	**0.891**
PSNR*	0-10%	31.16	30.24	**39.65**	39.60
	10-20%	25.32	24.09	33.19	**33.51**
	20-30%	22.09	20.71	29.68	**30.02**
	30-40%	19.94	18.50	27.15	**27.39**
	40-50%	18.41	17.09	25.15	**25.28**
	50-60%	17.18	16.24	22.00	**22.11**
	Fixed	25.34	22.13	**25.63**	25.49
FID†	0-10%	3.24	16.84	**0.20**	**0.20**
	10-20%	13.12	58.74	**0.53**	**0.53**
	20-30%	29.47	102.97	**1.08**	**1.08**
	30-40%	47.55	136.47	1.81	**1.80**
	40-50%	68.40	163.95	**2.81**	2.82
	50-60%	76.70	167.07	5.46	**5.30**
	Fixed	1.90	25.21	1.92	**1.90**

Table 3: Comparison of quantitative results (256×256) over CelebA with CA [6], GLCIC [1], PConv [4], Ours (end-to-end). The best result of each row is boldfaced. †Lower is better. *Higher is better.

	Mask	CA	GLCIC	PConv	Ours
ℓ_1 (%)†	0-10%	0.75	0.86	**0.43**	**0.43**
	10-20%	2.10	2.20	1.14	**1.09**
	20-30%	3.80	3.86	2.04	**1.91**
	30-40%	5.53	5.58	3.02	**2.82**
	40-50%	7.23	7.34	4.17	**3.94**
	50-60%	9.06	9.02	6.12	**5.87**
	Fixed	3.22	3.23	2.92	**2.77**
SSIM*	0-10%	0.964	0.949	**0.975**	**0.975**
	10-20%	0.905	0.878	0.933	**0.938**
	20-30%	0.835	0.800	0.881	**0.892**
	30-40%	0.766	0.724	0.826	**0.842**
	40-50%	0.695	0.648	0.765	**0.784**
	50-60%	0.625	0.588	0.678	**0.700**
	Fixed	0.847	0.840	0.847	**0.860**
PSNR*	0-10%	32.45	30.46	**36.39**	36.31
	10-20%	26.09	25.72	30.71	**31.23**
	20-30%	22.80	22.90	27.57	**28.26**
	30-40%	20.74	21.02	25.43	**26.05**
	40-50%	19.35	19.66	23.66	**24.20**
	50-60%	18.17	18.71	21.34	**21.73**
	Fixed	23.68	24.07	24.78	**25.23**
FID†	0-10%	2.26	6.50	**0.43**	0.44
	10-20%	9.10	18.77	1.32	**1.20**
	20-30%	20.62	35.66	2.97	**2.49**
	30-40%	34.31	53.53	5.65	**4.35**
	40-50%	49.80	70.36	10.00	**7.20**
	50-60%	55.78	69.95	21.10	**13.98**
	Fixed	7.26	7.18	6.44	**4.57**

Table 4: Comparison of quantitative results (256×256) over Paris StreetView with CA [6], GLCIC [1], PConv [4], Ours (end-to-end). The best result of each row is boldfaced. †Lower is better. *Higher is better.

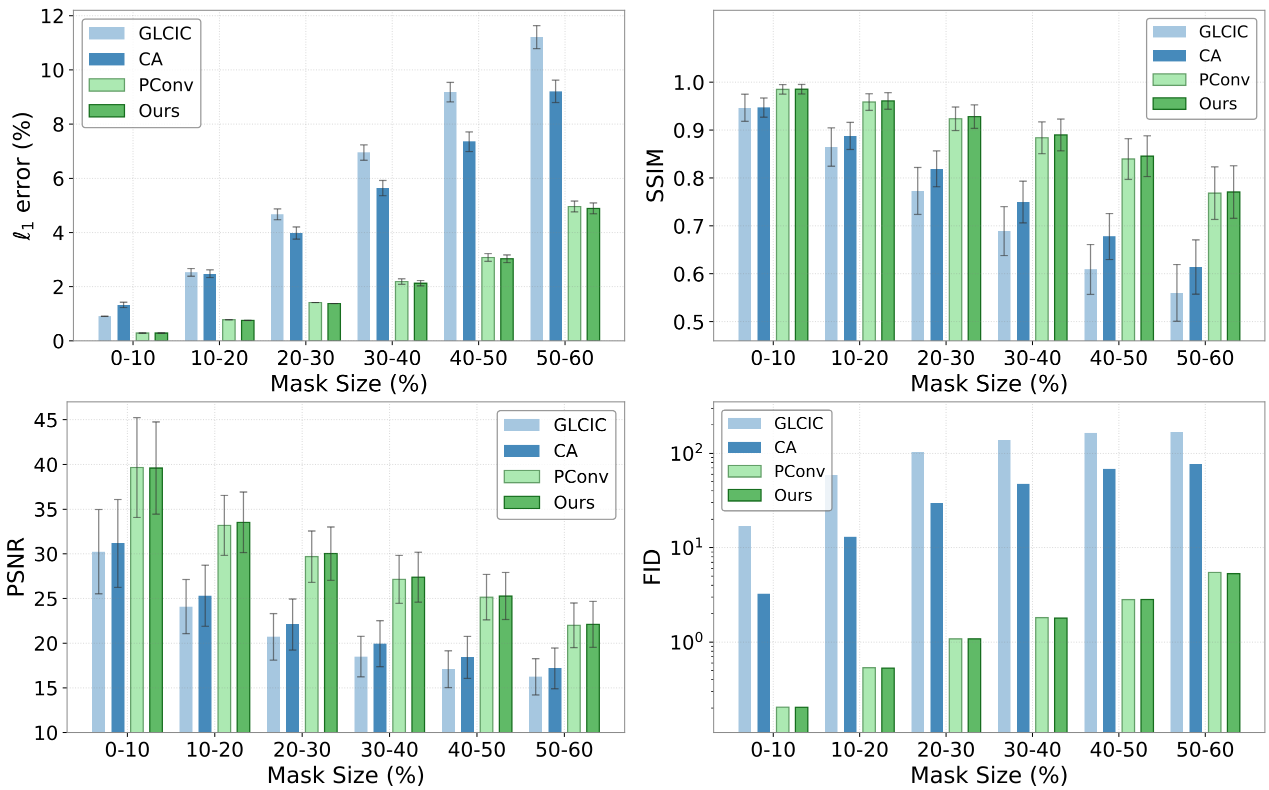

Figure 2: Effect of relative mask sizes on ℓ_1, SSIM, PSNR, and FID on the CelebA dataset.

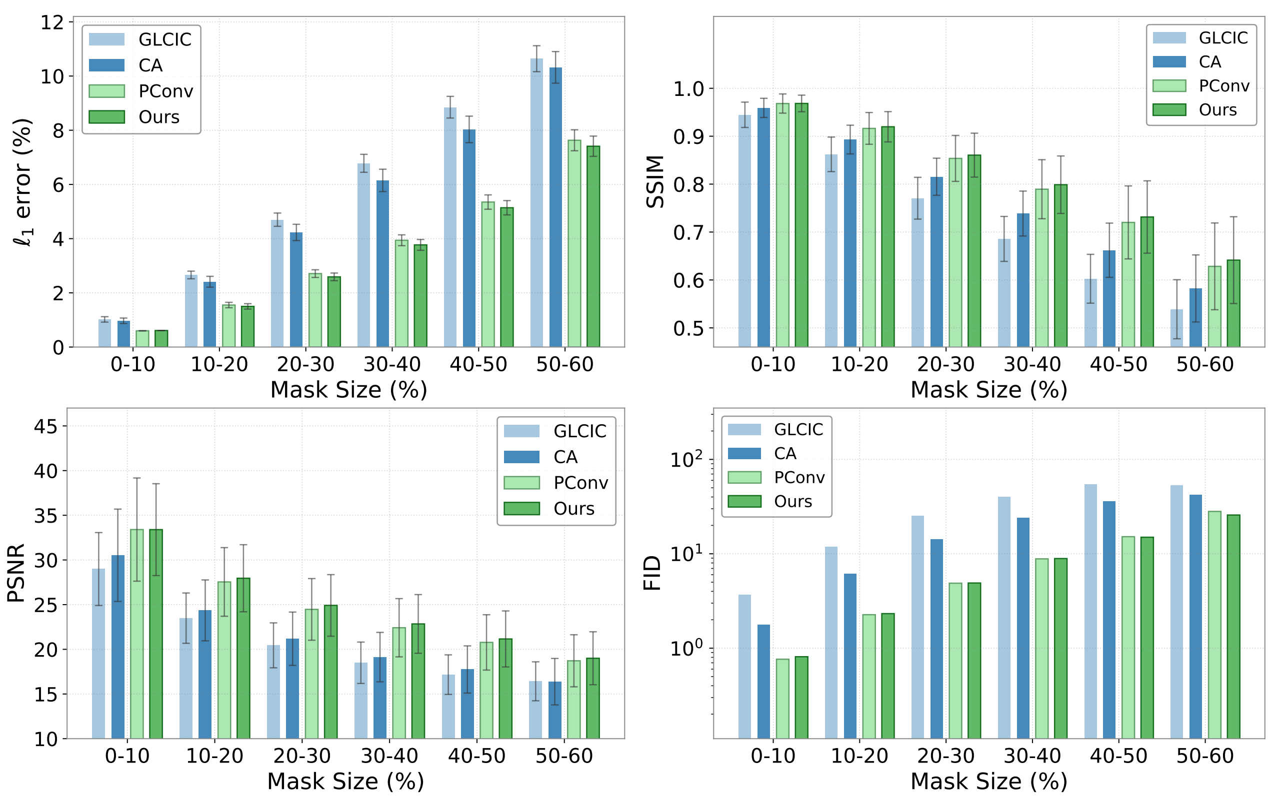

Figure 3: Effect of relative mask sizes on ℓ_1, SSIM, PSNR, and FID on the Places dataset.

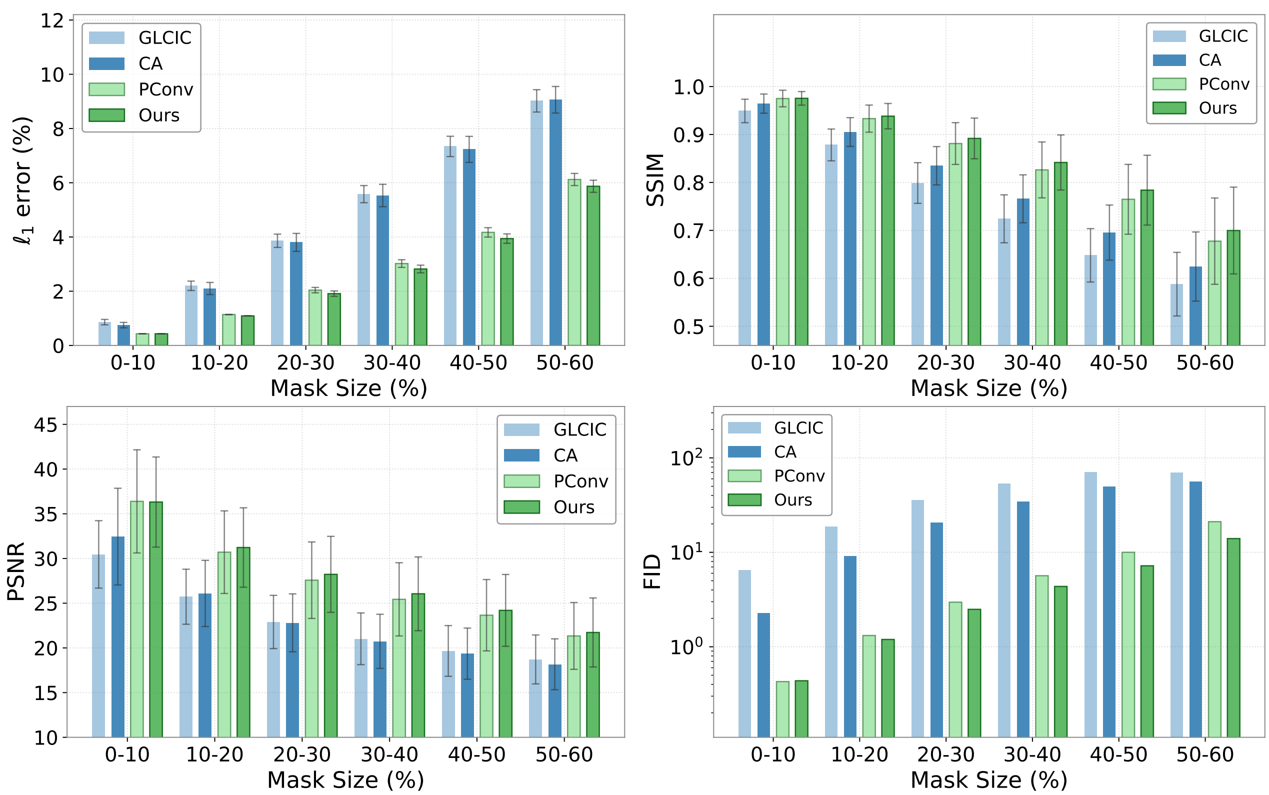

Figure 4: Effect of relative mask sizes on ℓ_1, SSIM, PSNR, and FID on the Paris StreetView dataset.

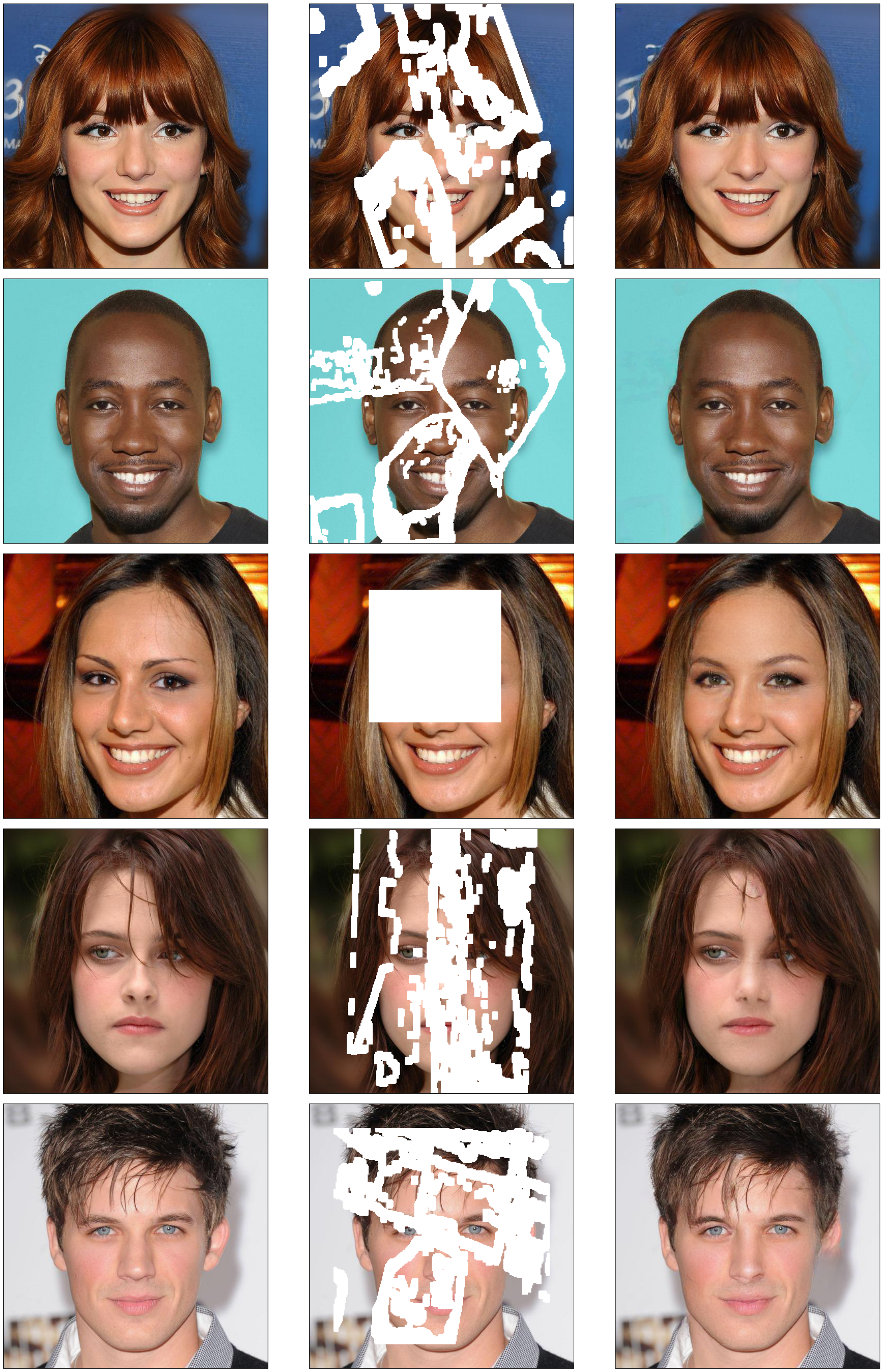

Figure 5: Sample of results with CelebA dataset (512 $\times$ 512). Images are best viewed in color. From left to right: Original Image. Input Image, Generated Result.

Figure 6: Sample of results with Places2 dataset (512 $\times$ 512). Images are best viewed in color. From left to right: Original Image. Input Image, Generated Result.

Figure 7: Sample of results with Places2 dataset (512 $\times$ 512). Images are best viewed in color. From left to right: Original Image. Input Image, Generated Result.

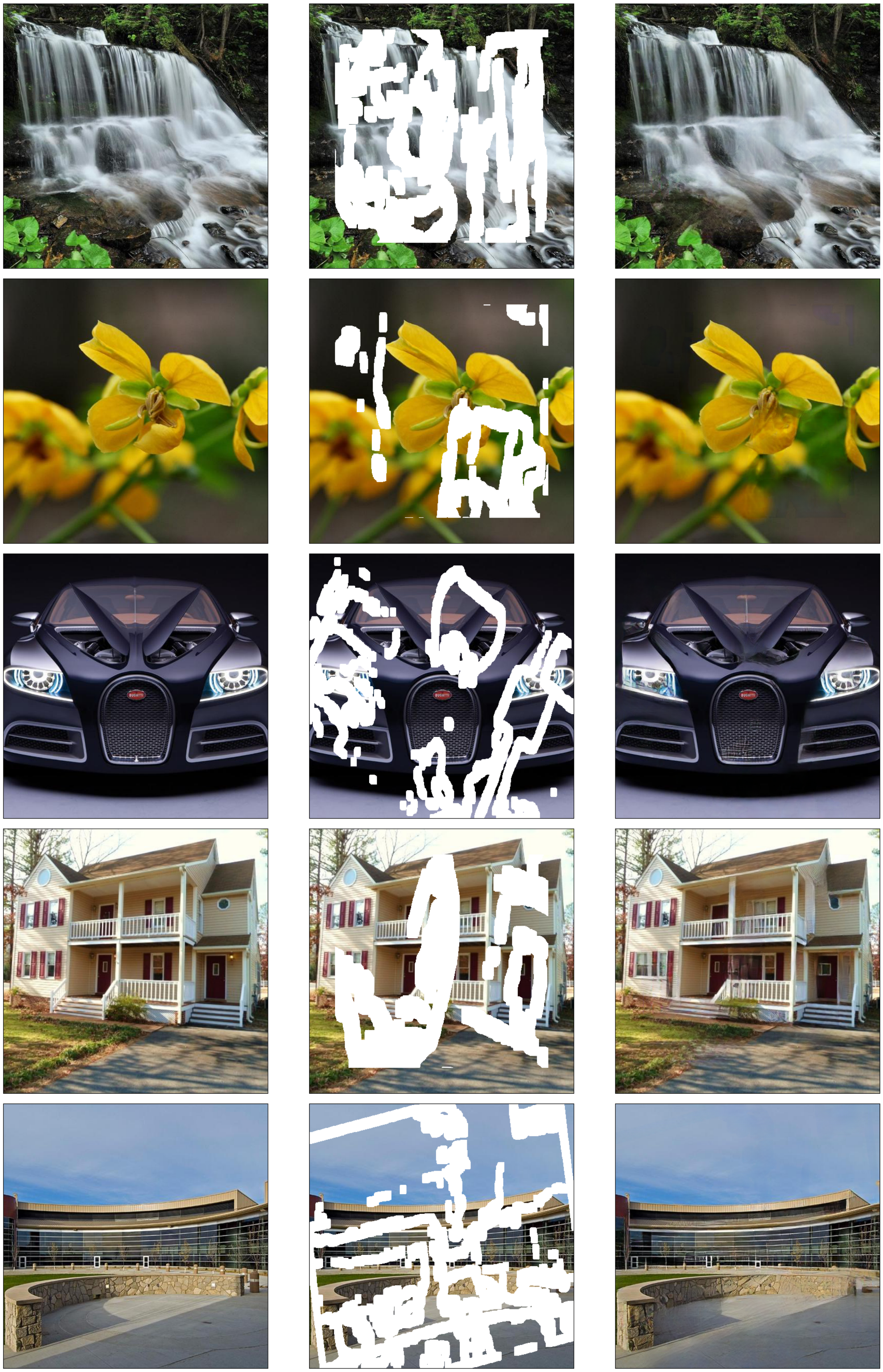

Figure 8: Sample of results with Places2 dataset (512 $\times$ 512). Images are best viewed in color. From left to right: Original Image. Input Image, Generated Result.

References

[1] S. Iizuka, E. Simo-Serra, and H. Ishikawa. Globally and locally consistent image completion. *ACM Transactions on Graphics (TOG)*, 36(4):107, 2017. 3

[2] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[3] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision (ECCV)*, pages 694–711. Springer, 2016. 1

[4] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro. Image inpainting for irregular holes using partial convolutions. In *The European Conference on Computer Vision (ECCV)*, September 2018. 3

[5] A. L. Maas, A. Y. Hannun, and A. Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *ICML Workshop on Deep Learning for Audio, Speech, and Language Processing (WDLASL 2013)*, 2013. 1

[6] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang. Generative image inpainting with contextual attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[7] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *The IEEE International Conference on Computer Vision (ICCV)*, 2017. 1