Supplementary Material of Flow-Fill

Cairong Wang[1], Yiming Zhu[1], and Chun Yuan[1,2]

[1] Tsinghua Shenzhen International Graduate School, China
[2] Peng Cheng National Laboratory, China

1 Network Architecture

Feature extraction network. We use the reimplemented generator proposed in [3] as the masked images' feature extraction network E_θ. It is a coarse-to-fine inpainting generator with gated convolution. Different from [3], we do not employ the contextual attention but a self-attention layer [1] instead in the refine network, and we add batch norm to each layer. Details are shown in Tab. 1. All selected feature maps are scaled to 128×128 by convolution and then concatenate together as the ft.

Guided texture generator. Given masked image I_m and upscaled structural prior $I_s \uparrow$ as input, guided texture generator G_ϕ generates the final texture-rich result in the second stage. Following [2], G_ϕ is composed of an encoder, decoder, and several residual blocks. The difference is that we replace all vanilla convolutions with gated convolution [3]. Fig. 1 shows the details.

2 More Qulitative Results on Benchmark Datasets

In order to demonstrate the inpainting performance and diversity of Flow-Fill, we present more qualitative results on the benchmark dataset in this section.

As shown in Fig. 2 (Places2 dataset), the inpainting result of our method can generate more photorealistic and diverse texture details. On the other hand, results from ICT and PIC tend to be ambiguous and inconsistent with boundaries.

Fig. 3 shows the inpainting results on Celeb-HQ datasets. It is obvious that Flow-Fill can produce various visual-pleasing inpainting results if the input image is badly damaged.

We designed a GUI operation interface to perform real-time image matting and restoration functions. As shown in Fig. 4, Flow-Fill can also apply in the specified object erasure for an image. It is worth noting that the speed of Flow-Fill is significantly faster (0.192s per image) than the current diverse inpainting methods. Therefore Flow-Fill offers a good prospect for real-time object erasure for video.

Table 1. Details of the feature extraction network E_θ. A Batch Normalization follows each layer. The first layer has four channels of input, including three channels of masked image and one channel of mask. We choose the output of the layer indicated in orange as the features ft of masked images.

Module	layer Name	Type	Filter Size	#Channels (input, output)	Stride	Spatial Size (input, output)	Dilation/Factor	Non-linearity
Coarse Network	Coarse encoder layer1	gated conv.	5	(4, 32)	1	(256, 256)	1/-	LeakyReLU(0.2)
	Coarse encoder layer2	gated conv.	4	(32, 64)	2	(256, 256)	1/-	LeakyReLU(0.2)
	Coarse encoder layer3	gated conv.	3	(64, 64)	1	(256, 128)	1/-	LeakyReLU(0.2)
	Coarse encoder layer4	gated conv.	4	(64, 128)	2	(128, 128)	1/-	LeakyReLU(0.2)
	Coarse encoder layer5	gated conv.	3	(128, 128)	1	(128, 64)	1/-	LeakyReLU(0.2)
	Coarse encoder layer6	gated conv.	3	(128, 128)	1	(64, 64)	1/-	LeakyReLU(0.2)
	Coarse encoder layer7	gated atrous conv.	3	(128, 128)	1	(64, 64)	2/-	LeakyReLU(0.2)
	Coarse encoder layer8	gated atrous conv.	3	(128, 128)	1	(64, 64)	4/-	LeakyReLU(0.2)
	Coarse encoder layer9	gated atrous conv.	3	(128, 128)	1	(64, 64)	8/-	LeakyReLU(0.2)
	Coarse encoder layer10	gated atrous conv.	3	(128, 128)	1	(64, 64)	16/-	LeakyReLU(0.2)
	Coarse encoder layer11	gated conv.	3	(128, 128)	1	(64, 64)	1/-	LeakyReLU(0.2)
	Coarse encoder layer12	gated conv.	3	(128, 128)	1	(64, 64)	1/-	LeakyReLU(0.2)
	Coarse decoder layer1	gated deconv.	3	(128, 64)	1	(64, 128)	1/2	LeakyReLU(0.2)
	Coarse decoder layer2	gated conv.	3	(64, 64)	1	(128, 128)	1/-	LeakyReLU(0.2)
	Coarse decoder layer3	gated deconv.	3	(64, 32)	1	(128, 256)	1/2	LeakyReLU(0.2)
	Coarse decoder layer4	gated conv.	3	(32, 16)	1	(256, 256)	1/-	LeakyReLU(0.2)
	Coarse decoder layer5	gated conv.	3	(16, 3)	1	(256, 256)	1/-	-
Refine Network	Refine encoder layer1	gated conv.	5	(4, 32)	1	(256, 256)	1/-	LeakyReLU(0.2)
	Refine encoder layer2	gated conv.	4	(32, 32)	2	(256, 256)	1/-	LeakyReLU(0.2)
	Refine encoder layer3	gated conv.	3	(32, 64)	1	(256, 128)	1/-	LeakyReLU(0.2)
	Refine encoder layer4	gated conv.	4	(64, 64)	2	(128, 128)	1/-	LeakyReLU(0.2)
	Refine encoder layer5	gated conv.	3	(64, 128)	1	(128, 64)	1/-	LeakyReLU(0.2)
	Refine encoder layer6	gated conv.	3	(128, 128)	1	(64, 64)	1/-	LeakyReLU(0.2)
	Refine encoder layer7	gated conv.	3	(128, 128)	1	(64, 64)	1/-	LeakyReLU(0.2)
	Refine encoder layer8	gated atrous conv.	3	(128, 128)	1	(64, 64)	2/-	LeakyReLU(0.2)
	Refine encoder layer9	gated atrous conv.	3	(128, 128)	1	(64, 64)	4/-	LeakyReLU(0.2)
	Refine encoder layer10	gated atrous conv.	3	(128, 128)	1	(64, 64)	8/-	LeakyReLU(0.2)
	Refine encoder layer11	gated atrous conv.	3	(128, 128)	1	(64, 64)	16/-	LeakyReLU(0.2)
	Self attention layer	self attention	-	(128, 128)	-	(64, 64)	-/-	-
	Refine decoder layer1	gated conv.	3	(128, 128)	1	(64, 64)	1/-	LeakyReLU(0.2)
	Refine decoder layer2	gated conv.	3	(128, 128)	1	(64, 64)	1/-	LeakyReLU(0.2)
	Refine decoder layer3	gated deconv.	3	(128, 64)	1	(64, 128)	1/2	LeakyReLU(0.2)
	Refine decoder layer4	gated conv.	3	(64, 64)	1	(128, 128)	1/-	LeakyReLU(0.2)
	Refine decoder layer5	gated deconv.	3	(64, 32)	1	(128, 256)	1/2	LeakyReLU(0.2)
	Refine decoder layer6	gated conv.	3	(32, 16)	1	(256, 256)	1/-	LeakyReLU(0.2)
	Refine decoder layer7	gated conv.	3	(16, 3)	1	(256, 256)	1/-	-

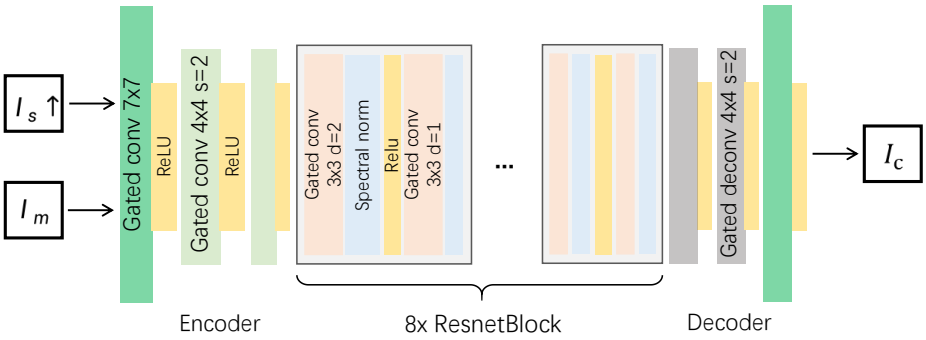

Fig. 1. Details of the guided texture generation network G_ϕ. 's' means stride, 'd' means dilation.

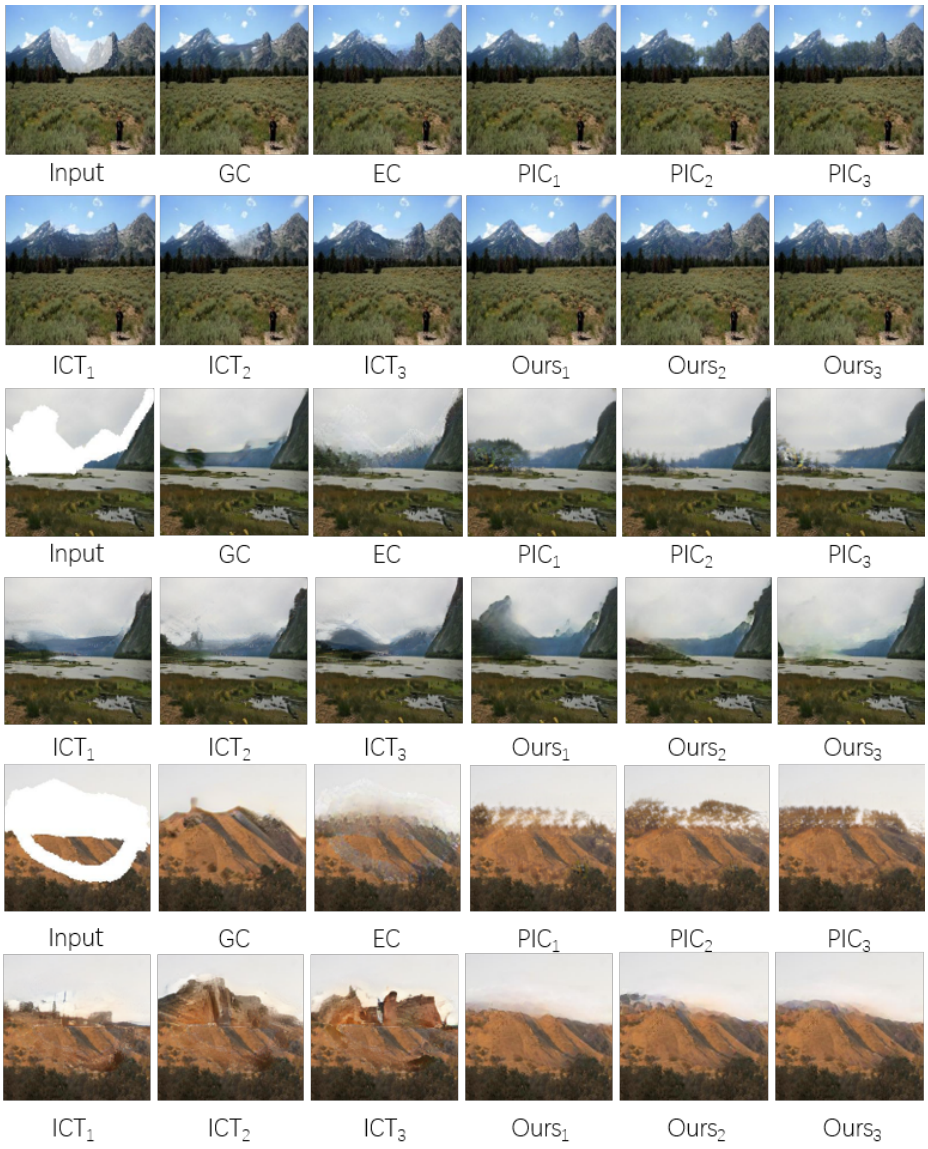

Fig. 2. More qualitative results on Places2. The inpainting result of our method can generate more photorealistic and diverse texture details. On the other hand, results from ICT and PIC tend to be ambiguous and inconsistent with boundaries.

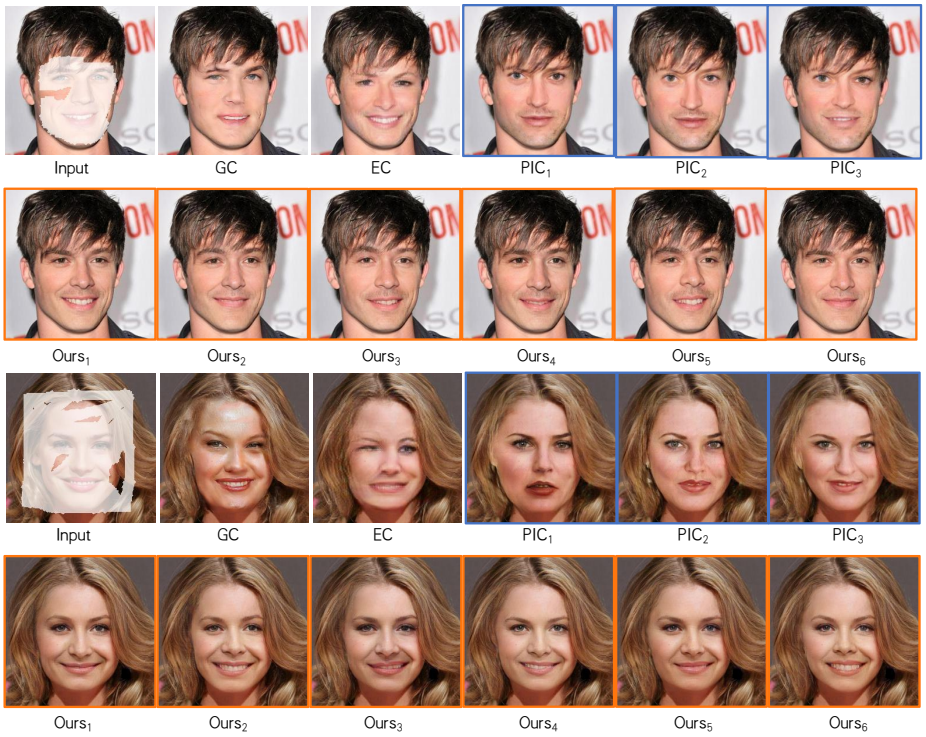

Fig. 3. More qualitative results on CelebA-HQ. It is obvious that Flow-Fill can produce various visual-pleasing inpainting results if the input image is badly damaged.

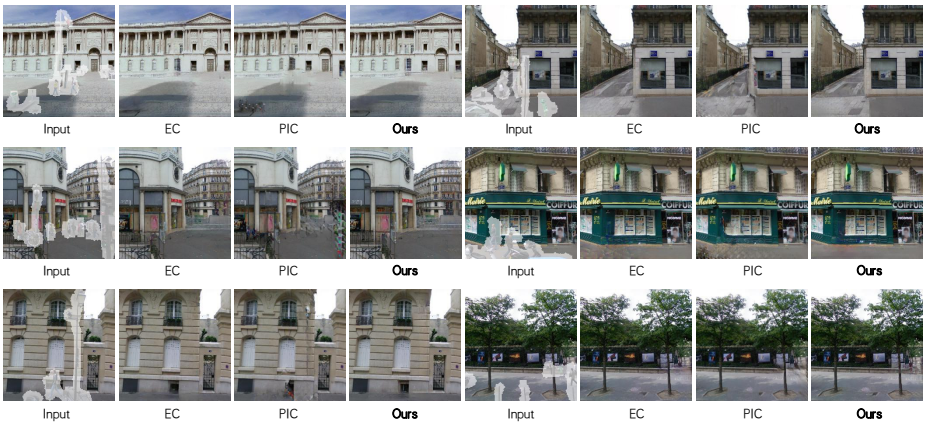

Fig. 4. Qualitative results for objective removal. Users can use our GUI to draw arbitrary erase areas. It can be seen that our method can be better applied to image erasure.

References

1. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017) 1
2. Wan, Z., Zhang, J., Chen, D., Liao, J.: High-fidelity pluralistic image completion with transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4692–4701 (2021) 1
3. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S.: Free-form image inpainting with gated convolution. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4471–4480 (2019) 1, 1