

Image Fine-grained Inpainting (arXiv:2002.02609)

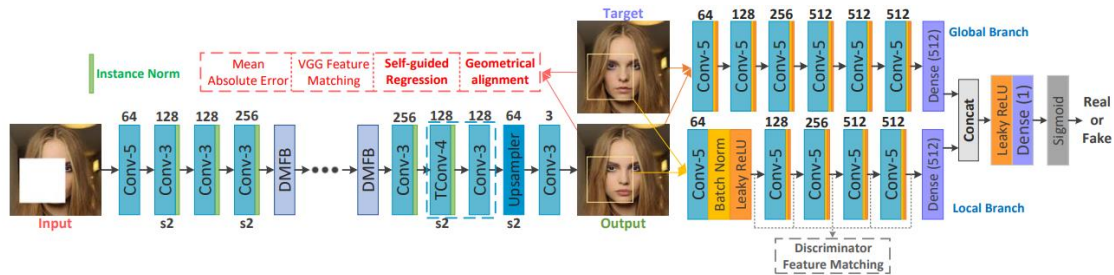
Zheng Hui, Jie Li, Xiumei Wang, and Xinbo Gao* School of Electronic Engineering, Xidian University

Xi'an, China

<https://arxiv.org/abs/2002.02609>

Abstract

Image inpainting techniques have shown promising improvement with the assistance of generative adversarial networks (GANs) recently. However, most of them often suffered from completed results with unreasonable structure or blurriness. To mitigate this problem, in this paper, we present a one-stage model that utilizes dense combinations of dilated convolutions to obtain larger and more effective receptive fields. Benefited from the property of this network, we can more easily recover large regions in an incomplete image. To better train this efficient generator, except for frequently-used VGG feature matching loss, we design a novel self-guided regression loss for concentrating on uncertain areas and enhancing the semantic details. Besides, we devise a geometrical alignment constraint item to compensate for the pixel-based distance between prediction features and ground-truth ones. We also employ a discriminator with local and global branches to ensure local-global contents consistency.



One-stage : DMFB, Global(local) Branch

Loss : Self-guided Regression, Geometrical alignment

Contributions

a novel self-guided regression loss to explicitly correct the low-level features

according to the normalized error map computed by the output and ground-truth images.

a geometrical alignment constraint to supplement the shortage of pixel-based VGG features matching loss.

a dense multi-scale fusion generator, which has the merit of strong representation ability to extract useful features.



Ground-truth Image

Input Image

CA [30]

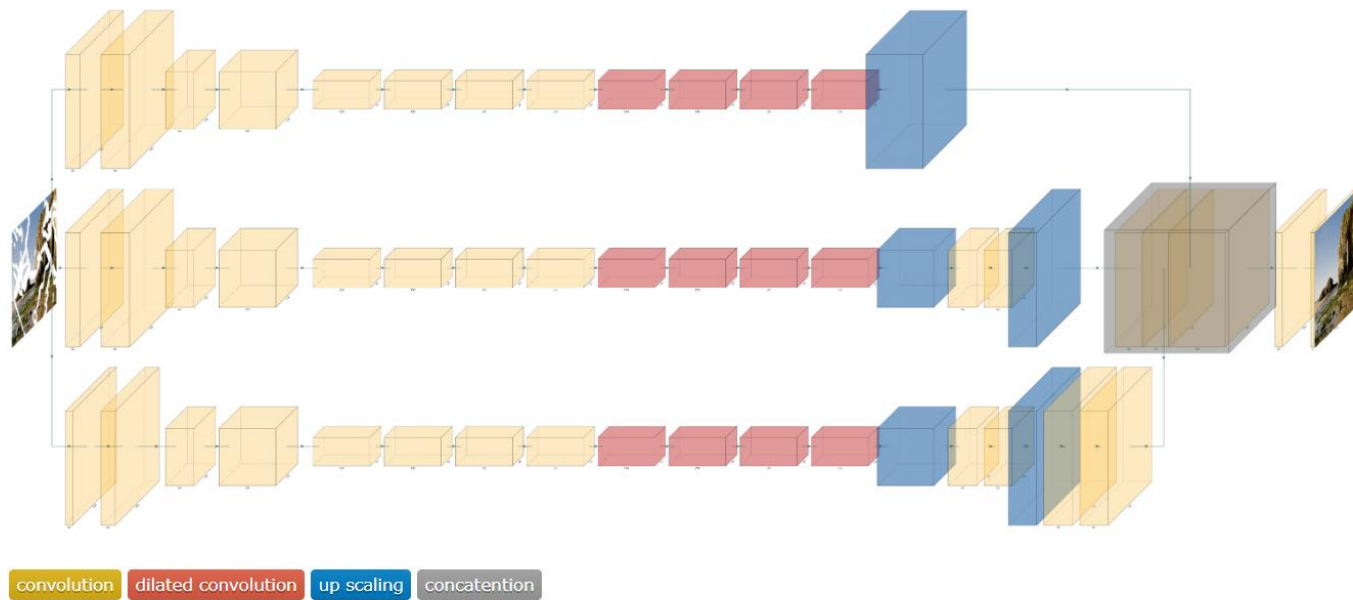
GMCNN [26]

DMFN (Ours)

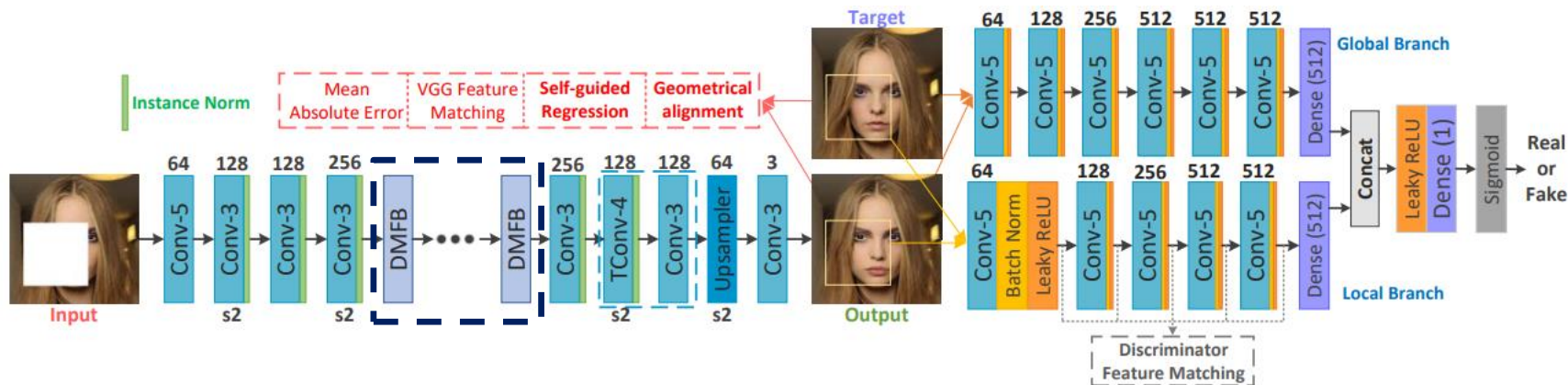
Figure 1. Visual comparisons on CelebA-HQ. **Best viewed with zoom-in.**

GMCNN (Generative Multi-column Convolutional Neural Networks), NIPS 2018

Model architecture



DMFB - dense multi-scale fusion block



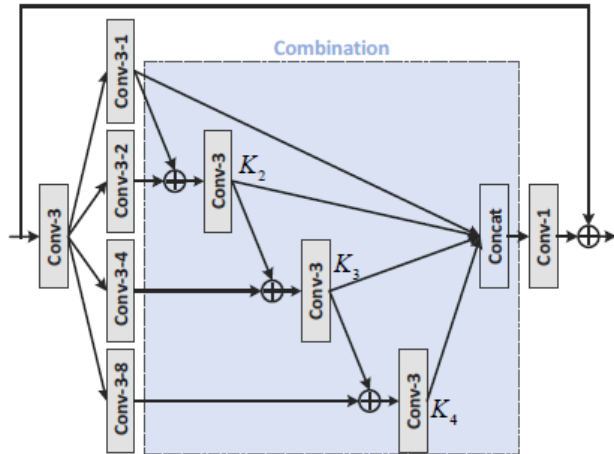
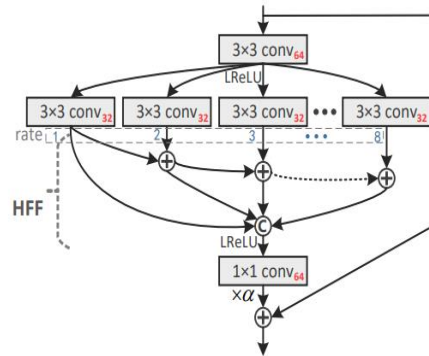
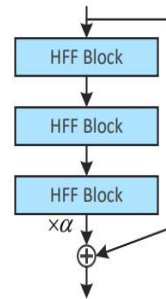


Figure 2. The architecture of the proposed dense multi-scale fusion block (DMFB). Here, “Conv-3-8” indicates 3×3 convolution layer with the dilation rate of 8 and \oplus is element-wise summation. Instance normalization (IN) and ReLU activation layers followed by the first convolution, second column convolutions and concatenation layer are omitted for brevity. The last convolutional layer only connects an IN layer. The number of output channels for each convolution is set to 64 except for the last 1×1 convolution (256 channels) in DMFB.



(a) Hierarchical Feature Fusion Block (HFFB)



(b) Residual-in-Residual Fusion Block (RRFB)

Fig. 3. The basic blocks proposed in this work. (a) We employ 8 dilated convolutions, each of them has 32 output channels for reducing block parameters. (b) RRFB is used in our primary and perception-oriented models and α is the residual scaling parameter [9], [20].

Progressive Perception-Oriented Network for Single Image Super-Resolution (*arXiv:1907.10399*)

Self-guided regression loss

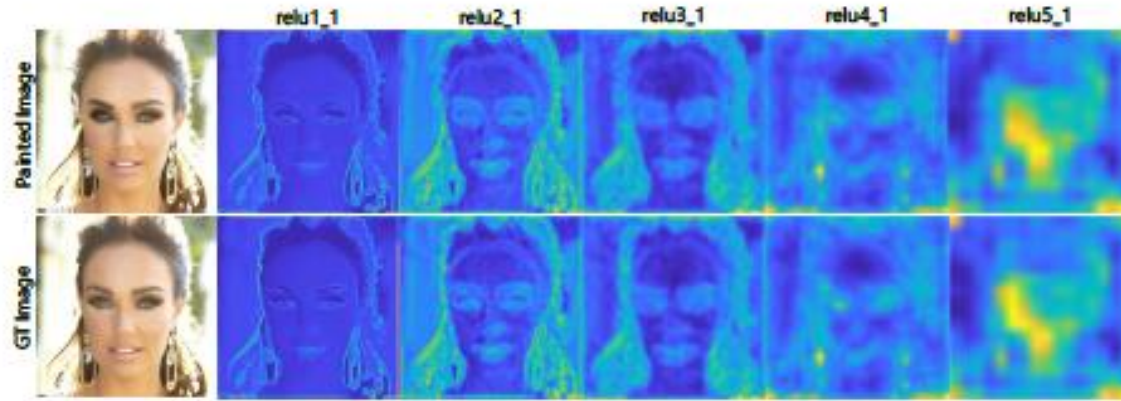


Figure 4. Visualization of average VGG feature maps.

$$\text{average feature maps} \quad ; F_{A_{avg}}^l = \frac{1}{M} \sum_{m=1}^M F_{A_m}^l$$

Self-guided regression loss

P : position

AP : average pooling with kernel size of 2, stride of 2

Guidance mask : 0~1

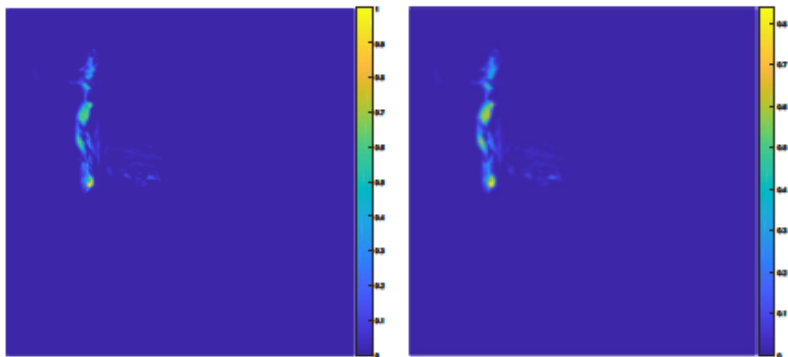


Figure 5. Visualization of guidance maps. (Left) Guidance map $M^1_{guidance}$ for “relu1_1” layer. (Right) Guidance map $M^2_{guidance}$ for “relu2_1” layer. These are corresponding to Figure 4.

$$M_{error} = \frac{1}{3} \sum_{c \in \mathcal{C}} (I_{out,c} - I_{gt,c})^2, \quad (2)$$

$$M_{guidance,p} = \frac{M_{error,p} - \min(M_{error})}{\max(M_{error}) - \min(M_{error})}, \quad (3)$$

$$\mathcal{L}_{self-guided} = \sum_{l=1}^2 w^l \frac{\|M^l_{guidance} \odot (\Psi^l_{I_{gt}} - \Psi^l_{I_{output}})\|_1}{N_{\Psi^l_{I_{gt}}}}, \quad (5)$$

relu1_1 / relu2_1

Geometrical alignment constraint

The geometrical center for the k -th feature map along axis u is calculated as

$$c_u^k = \sum_{u,v} u \cdot \mathbf{R}(k, u, v) / \sum_{u,v} \mathbf{R}(k, u, v), \quad (6)$$

where response maps $\mathbf{R} = \text{VGG}(\mathbf{I}; \theta_{\text{vgg}}) \in \mathbb{R}^{K \times H \times W}$.

$\mathbf{R}(k, u, v) / \sum_{u,v} \mathbf{R}(k, u, v)$ represents a spatial probability distribution function. c_u^k denotes coordinate expectation along axis u . Then, we pass both the completed image $\mathbf{I}_{\text{output}}$ and ground-truth image \mathbf{I}_{gt} through the VGG network and obtain the corresponding response maps \mathbf{R}' and \mathbf{R} . Given these response maps, we compute the centers $\langle c_u^{k'}, c_v^{k'} \rangle$ and $\langle c_u^k, c_v^k \rangle$ using Equation 6. Then, we formulate the geometrical alignment constraint as

$$\mathcal{L}_{\text{align}} = \sum_k \sum_{u,v} \left\| \langle c_u^{k'}, c_v^{k'} \rangle - \langle c_u^k, c_v^k \rangle \right\|_2^2. \quad (7)$$

Feature matching losses

compares the activation maps in the intermediate layers of VGG19

$$\mathcal{L}_{fm_vgg} = \sum_{l=1}^5 w^l \frac{\|\Psi_{\mathbf{I}_{gt}}^l - \Psi_{\mathbf{I}_{output}}^l\|_1}{N_{\Psi_{\mathbf{I}_{gt}}^l}}, \quad (8)$$

$$\mathcal{L}_{fm_dis} = \sum_{l=1}^5 w^l \frac{\|D_{local}^l(\mathbf{I}_{gt}) - D_{local}^l(\mathbf{I}_{output})\|_1}{N_{D_{local}^l(\mathbf{I}_{gt})}}, \quad (9)$$

$$\begin{aligned} \mathcal{L}_{total} = & \mathcal{L}_{mae} + \lambda (\mathcal{L}_{self-guided} + \mathcal{L}_{fm_vgg}) \\ & + \eta \mathcal{L}_{fm_dis} + \mu \mathcal{L}_{adv} + \gamma \mathcal{L}_{align}, \end{aligned}$$



Input

GT

PICNet [33]

DMFN (Ours)

Figure 7. Visual comparisons on Places2. **Best viewed with zoom-in.**



Input

DMFN (Ours)

GT

Figure 8. Visual results on FFHQ dataset.



Figure 6. Visual comparisons on Paris street view.

Table 1. Quantitative results (center regular mask) on four testing datasets.

Method	Paris street view (100)	Places2 (100)	CelebA-HQ (2,000)	FFHQ (10,000)
	LPIPS / PSNR / SSIM	LPIPS / PSNR / SSIM	LPIPS / PSNR / SSIM	LPIPS / PSNR / SSIM
CA [30]	N/A	0.1524 / 21.32 / 0.8010	0.0724 / 24.13 / 0.8661	N/A
GMCNN [26]	0.1243 / 24.38 / 0.8444	0.1829 / 19.51 / 0.7817	0.0509 / 25.88 / 0.8879	N/A
PICNet [33]	0.1263 / 23.79 / 0.8314	0.1622 / 20.70 / 0.7931	N/A	N/A
DMFN (Ours)	0.1018 / 25.00 / 0.8563	0.1361 / 21.53 / 0.8079	0.0460 / 26.50 / 0.8932	0.0457 / 26.49 / 0.8985

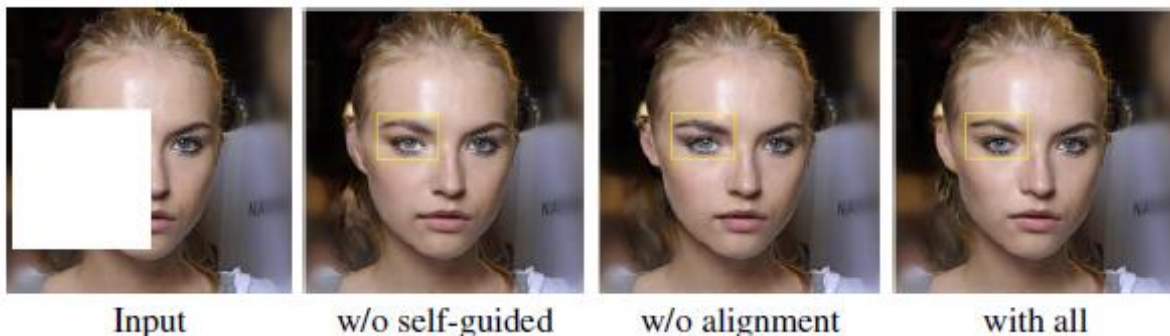


Figure 12. Visual comparison of results using different losses.
Best viewed with zoom-in.

Table 3. Investigation of self-guided regression loss and geometrical alignment constraint.

Metric	w/o self-guided	w/o align	w/o dis_fm	with all
LPIPS↓	0.0537	0.0534	0.0542	0.0530
PSNR↑	25.73	25.63	25.65	25.83
SSIM↑	0.8892	0.8884	0.8870	0.8892

Table 2. Quantitative results of different structures on Paris street view dataset.

Model	rate=2	rate=8	w/o combination	w/o $K_i(\cdot)$	DMFB
Params	803,392	803,392	361,024	361,024	471,808
LPIPS↓	0.1059	0.1067	0.1083	0.1026	0.1018
PSNR↑	24.93	24.91	24.24	24.93	25.00
SSIM↑	0.8530	0.8549	0.8505	0.8561	0.8563



Figure 11. Visual comparison of different structures. **Best viewed with zoom-in.**