Image Inpainting via Conditional Texture and Structure Dual Generation

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

*Deep generative approaches have recently made considerable progress in image inpainting by introducing structure priors. Due to the lack of proper interaction with image texture during structure reconstruction, however, current solutions are incompetent in handling the cases with large corruptions, and they generally suffer from distorted results. In this paper, we propose a novel two-stream network for image inpainting, which models the **structure-constrained texture synthesis** and **texture-guided structure reconstruction** in a coupled manner so that they better leverage each other for more plausible generation. Furthermore, to enhance the global consistency, a Bi-directional Gated Feature Fusion (Bi-GFF) module is designed to exchange and combine the structure and texture information and a Contextual Feature Aggregation (CFA) module is developed to refine the generated contents by region affinity learning and multi-scale feature aggregation. Qualitative and quantitative experiments on the CelebA, Paris StreetView and Places2 datasets demonstrate the superiority of the proposed method. Our code is available at* https://github.com/Xiefan-Guo/CTSDG.

1. Introduction

Image inpainting [3] refers to the process of reconstructing damaged regions of an image while simultaneously maintaining its overall consistency, which is a typical low-level visual task with many practical applications, such as photo editing, distracting object removal, and restoring corrupted parts.

As with most computer vision problems, image inpainting has been largely advanced by the widespread use of deep learning during the past decade. Different from the traditional methods [2, 5] that gradually fill in missing areas by searching for the most similar patches from known regions, the deep generative ones [19, 7, 31, 33] capture more high-level semantics and do a better job for images with
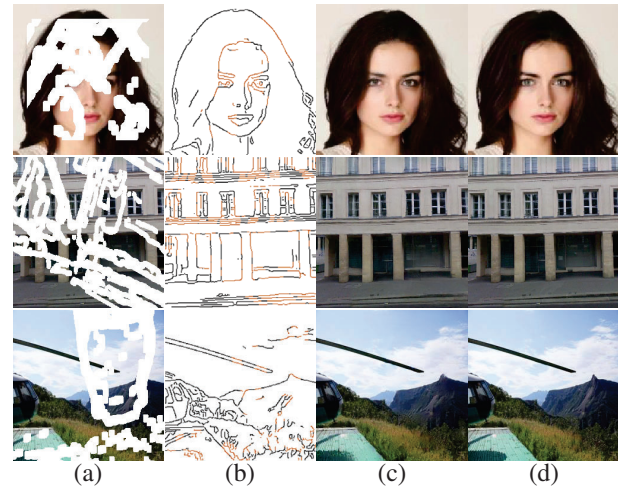
---

*Corresponding author.

Figure 1: High-quality inpainting results. From left to right: (a) input corrupted images, (b) our reconstructed structures, (c) our filled results, and (d) ground-truth images.

non-repetitive patterns. There also exists another trend to combine the advantages of deep generative and traditional patch-based methods for image inpainting [35, 30, 24, 15], delivering inpainting contents with both realistic textures and plausible semantics. Moreover, updated versions of vanilla convolution are investigated [13, 27, 36], where operations are masked and normalized to be conditioned only on valid pixels, achieving promising performance for irregular corruptions. Nevertheless, the methods above expose a common drawback in recovering the global structure of the image, as a generative network is not as powerful as expected for this issue.

To deal with this problem, a number of multi-stage methods are proposed to explicitly incorporate structure modeling, which hallucinate structures of missing regions in the first stage and use them to guide pixel generation in the second stage. For instance, EdgeConnect [18] encodes such structures by edges, while [20] and [28] adopt intermediate edge-preserved smooth images and foreground contours. These alternatives show the results with improved structures and textures. Unfortunately, acquiring reasonable edges from corrupted images is itself a very challeng-

ing task, and taking unstable structural priors tends to incur large errors in those series-coupled frameworks.

More recently, a few attempts mix the modeling processes of structures and textures. PRVS (Progressive Reconstruction of Visual Structure) [10] and MED (Mutual Encoder-Decoder) [14] are the representatives, and they generally exploit a shared generator for both textures and structures. Despite some performance gains reported, the relationship between structures and textures is not fully considered in this single entangling architecture. In particular, since image structures and textures correlate throughout the network, it is difficult for them to convey holistic complementary information to assist the other side. Such a fact indicates that there is still much space for improvement.

In this paper, we propose a novel two-stream network which casts image inpainting into two collaborative subtasks, *i.e.*, structure-constrained texture synthesis and texture-guided structure reconstruction. In this way, the two parallel-coupled streams are individually modeled and combined to complement each other. Correspondingly, a two-branch discriminator is developed to estimate the performance of this generation, which supervises the model to synthesize realistic pixels and sharp edges simultaneously for global optimization. In addition, we introduce a novel Bi-directional Gated Feature Fusion (Bi-GFF) module to integrate the rebuilt structure and texture feature maps to enhance their consistency, along with a Contextual Feature Aggregation (CFA) module to highlight the clues from distant spatial locations to render finer details. Due to the dual generation network as well as the specifically designed modules, our approach is able to achieve more visually convincing structures and textures (see Figure 1, zoom in for a better view).

Experiments are extensively conducted on the CelebA [16], Paris StreetView [4] and Places2 [39] datasets for evaluation. Qualitative and quantitative results demonstrate that our model significantly outperforms the state-of-the-art.

The main novelties and contributions are as follows:

- We propose a novel two-stream network for image inpainting, which models structure-constrained texture synthesis and texture-guided structure reconstruction in a coupled manner so that the dual generation tasks better facilitate each other for more accurate results.

- We design a Bi-directional Gated Feature Fusion (Bi-GFF) module to share and combine information between the structure and texture features for consistency enhancement and a Contextual Feature Aggregation (CFA) module to yield more vivid details by modeling long-term spatial dependency.

- We achieve the new state-of-the-art performance on multiple public benchmarks both qualitatively and quantitatively.

2. Related Work

2.1. Traditional Methods

The traditional methods can be mainly summarized into two categories, *i.e.*, diffusion-based and patch-based. Diffusion-based methods [3, 1] render missing regions referring to the appearance information of the neighboring ones. Their results are not so good due to this preliminary searching mechanism. In patch-based methods [2, 29], pixel completion is conducted by searching and pasting the most similar patches from undamaged regions of source images, which takes advantage of long-distance information. These methods achieve better performance, but they are computationally expensive when calculating patch similarities between missing and available regions and struggle to reconstruct patterns with rich semantics.

2.2. Deep Generative Methods

The deep generative methods [35, 36, 8, 34, 38, 40, 37, 26, 12] are currently dominating, which effectively extract meaningful semantics from damaged images and recover reasonable contents with high visual fidelity, owing to their powerful feature learning ability.

Recently, Wang *et al.* [25] significantly improve the quality of image synthesis with sharper edges by involving structural information. Subsequently, a number of multi-stage methods that serially incorporate additional structural priors are proposed, producing more impressive results. EdgeConnect [18] extracts image structures by edges, based on which the holes are filled. Xiong *et al.* [28] show a similar model while it employs foreground object contours as structure priors instead of edges. Ren *et al.* [20] point out that edge-preserved smooth images convey better global structure since more semantics are captured. But these methods are sensitive to the accuracy of structures (*e.g.* edges and contours) which is not easy to guarantee. To overcome this weaknesses, several methods attempt to exploit the correlation of textures and structures. Li *et al.* [10] design a visual structure reconstruction layer to progressively entangle the generation of image contents and structures. Yang *et al.* [32] introduce a multi-task framework to generate sharp edges by adding structural constraints. Liu *et al.* [14] present a mutual encoder-decoder network to simultaneously learn the CNN features that correspond to structures and textures with different layers. However, it is rather difficult to model both textures and structures and make them sufficiently complement each other in a single shared architecture.

Our study also makes use of image structural information and figures out a different but more effective two-stream network, where structure-constrained texture synthesis and texture-guided structure reconstruction are jointly considered. The two subtasks better facilitate each other, leading
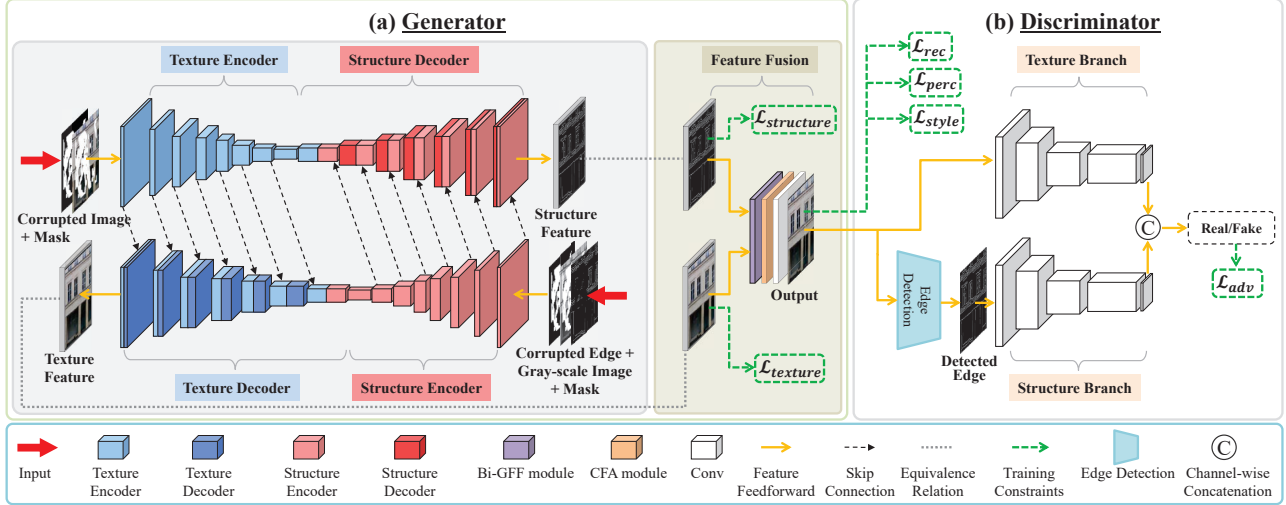
Figure 2: Overview of the proposed method (best viewed in color). **Generator**: Image inpainting is cast into two subtasks, *i.e.*, *structure-constrained texture synthesis* (left, blue) and *texture-guided structure reconstruction* (right, red), and the two parallel-coupled streams borrow encoded deep features from each other. The Bi-directional Gated Feature Fusion (Bi-GFF) module and Contextual Feature Aggregation (CFA) module are stacked at the end of the generator to further refine the results. **Discriminator**: The image branch estimates the generated texture, while the edge branch guides structure reconstruction.

to more convincing textures and structures in dual generation.

3. Approach

As illustrated in Figure 2, the proposed method is implemented as a generative adversarial network, where the two-stream generator jointly synthesizes image textures and structures, and the discriminator judges their quality and consistency. In this section, we detailedly describe the generator, the discriminator, and the loss functions.

3.1. Generator

The generator is a two-stream architecture, modeled by a U-Net variant, as shown in Figure 2 (a). At the encoding stage, the corrupted image and its corresponding edge map are individually projected into the latent space, where the left branch focuses on texture features and the right branch targets structure features. At the decoding stage, the texture decoder synthesizes structure-constrained textures by borrowing structure features from the structure encoder, while the structure decoder recovers texture-guided structures by taking texture features from the texture encoder. With such a dual generation architecture, structures and textures well complement each other, leading to improved results.

In this encoder-decoder based backbone, we replace all the vanilla convolutions with the partial convolution layers to better capture information from irregular boundaries, since partial convolutions are conditioned only on uncorrupted pixels. Besides, skip connections are utilized to produce more sophisticated predictions by combining low-level and high-level features at multiple scales. To enhance the consistency of the rebuilt structures and textures, the

feature maps output by the two branches are further fused to render the final result through a specially designed Bi-GFF module followed by a CFA module. Refer to the supplementary material for more details of the backbone.

Bi-directional Gated Feature Fusion (Bi-GFF). This module is proposed to further combine the decoded texture and structure features. It exchanges messages between the two kinds of information, where soft gating is exploited to control the rate. Due to this integration operation, the feature is refined and simultaneously texture- and structure-aware. Figure 3 illustrates the Bi-GFF module.

Specifically, the texture feature map output by the decoder is denoted as F_t and the structure feature map is denoted as F_s. To build texture-aware structure features, a soft gating G_t, which controls to what extent the texture information is integrated, is formulated as:

$$G_t = \sigma\left(g\left(\text{Concat}\left(F_t, F_s\right)\right)\right), \quad (1)$$

where $\text{Concat}(\cdot)$ is channel-wise concatenation, $g(\cdot)$ is the mapping function implemented by a convolution layer with the kernel size of 3, and $\sigma(\cdot)$ is Sigmoid activation. With G_t, we adaptively merge F_t into F_s as:

$$F_s^{'} = \alpha(G_t \odot F_t) \oplus F_s, \quad (2)$$

where α is a training parameter initialized to zero, and $\odot$ and $\oplus$ denote element-wise multiplication and element-wise addition, respectively.

Symmetrically, we calculate the structure-aware texture feature $F_t^{'}$ as follows:

$$G_s = \sigma\left(h\left(\text{Concat}\left(F_t, F_s\right)\right)\right), \quad (3)$$

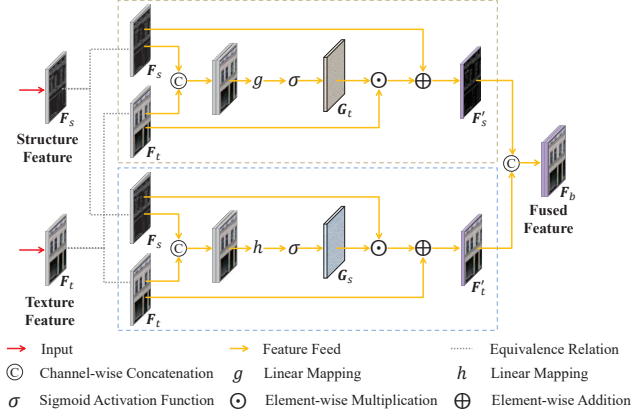$$F_t^{'} = \beta(G_s \odot F_s) \oplus F_t, \quad (4)$$

Figure 3: Illustration of the Bi-directional Gated Feature Fusion (Bi-GFF) module, which entangles the decoded structure and texture features to refine the results.

where h follows the same pattern as g and β is a training parameter initialized to zero as α.

Finally, we fuse $\boldsymbol{F}'_s$ and $\boldsymbol{F}'_t$ to obtain the integrated feature map $\boldsymbol{F}_b$ by channel-wise concatenation:

$$\boldsymbol{F}_b = \text{Concat}(\boldsymbol{F}'_s, \boldsymbol{F}'_t). \tag{5}$$

Contextual Feature Aggregation (CFA). To better learn which existing regions contribute to filling holes, this module is designed, which enhances the correlation between local features of an image and maintains the overall image consistency. It is inspired by [35], but unlike its fixed-scale patch matching scheme, in this study, multi-scale feature aggregation is adopted to encode rich semantic features at multiple scales so that it well balances the accuracy and complexity to handle more challenging cases, in particular, scale changes. The detailed process is depicted in Figure 4.

To be specific, given a feature map $\boldsymbol{F}$, we first extract the patches of 3×3 pixels and calculate their cosine similarities as:

$$\boldsymbol{S}^{i,j}_{contextual} = \left\langle \frac{\boldsymbol{f}_i}{\|\boldsymbol{f}_i\|_2}, \frac{\boldsymbol{f}_j}{\|\boldsymbol{f}_j\|_2} \right\rangle, \tag{6}$$

where $\boldsymbol{f}_i$ and $\boldsymbol{f}_j$ correspond to the i-th and j-th patch of the feature map, respectively.

We then apply softmax to the similarities to obtain the attention score of each patch:

$$\hat{\boldsymbol{S}}^{i,j}_{contextual} = \frac{\exp\left(\boldsymbol{S}^{i,j}_{contextual}\right)}{\sum_{j=1}^{N} \exp\left(\boldsymbol{S}^{i,j}_{contextual}\right)}. \tag{7}$$

Next, the extracted patches are reused to reconstruct the feature map based on the attention map:

$$\tilde{\boldsymbol{f}}_i = \sum_{j=1}^{N} \boldsymbol{f}_j \cdot \hat{\boldsymbol{S}}^{i,j}_{contextual}, \tag{8}$$
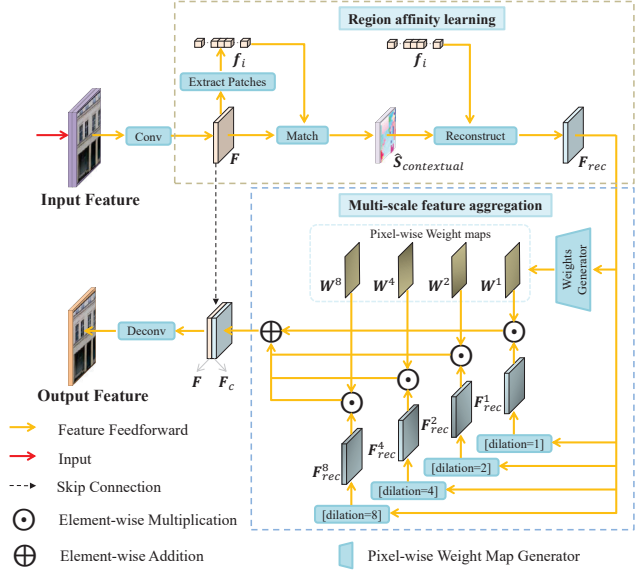

Figure 4: Illustration of the Contextual Feature Aggregation (CFA) module, which models long-term spatial dependency by capturing features at diverse semantic levels.

where $\tilde{\boldsymbol{f}}_i$ is the i-th patch of the reconstructed feature map $\boldsymbol{F}_{rec}$. The operations above are implemented as convolution, channel-wise softmax, and deconvolution, respectively.

When the feature map is reconstructed, four sets of dilated convolution layers with different dilation rates are used to capture multi-scale semantic features:

$$\boldsymbol{F}^k_{rec} = \text{Conv}_k\left(\boldsymbol{F}_{rec}\right), \tag{9}$$

where $\text{Conv}_k(\cdot)$ denotes dilated convolution layers with dilation rate of k, $k \in \{1, 2, 4, 8\}$.

To better aggregate the multi-scale semantic features, we further design a pixel-level weight map generator G_w, which aims to predict the pixel-wise weight maps. In our implementation, G_w consists of two convolution layers with the kernel size of 3 and 1, respectively, each of which is followed by ReLU non-linear activation, and the number of the output channels for G_w is set to 4. The pixel-wise weight maps are calculated as:

$$\boldsymbol{W} = \text{Softmax}\left(G_w\left(\boldsymbol{F}_{rec}\right)\right), \tag{10}$$

$$\boldsymbol{W}^1, \boldsymbol{W}^2, \boldsymbol{W}^4, \boldsymbol{W}^8 = \text{Slice}(\boldsymbol{W}), \tag{11}$$

where $\text{Softmax}(\cdot)$ is channel-wise softmax and $\text{Slice}(\cdot)$ is channel-wise slice. Finally, the multi-scale semantic features are aggregated to produce the refined feature map $\boldsymbol{F}_c$ by element-wise weighted sum:

$$\boldsymbol{F}_c = \left(\boldsymbol{F}^1_{rec} \odot \boldsymbol{W}^1\right) \oplus \left(\boldsymbol{F}^2_{rec} \odot \boldsymbol{W}^2\right) \oplus \\ \left(\boldsymbol{F}^4_{rec} \odot \boldsymbol{W}^4\right) \oplus \left(\boldsymbol{F}^8_{rec} \odot \boldsymbol{W}^8\right). \tag{12}$$

Note, as the mask update mechanism of partial convolution layers is exploited, there is no need to distinguish

the foreground and background pixels of the image as [35] does. Skip connection [21] is adopted to prevent semantic damage caused by patch-shift operations and a pair of convolution and deconvolution layers are seamlessly embedded into our architecture to improve computational efficiency.

3.2. Discriminator

Motivated by global and local GANs [7], Gated Convolution [36] and Markovian GANs [9], we develop a two-stream discriminator to distinguish genuine images from the generated ones by estimating the feature statistics of both texture and structure. The discriminator is shown in Figure 2 (b). The texture branch includes three convolution layers with the kernel size of 4 and stride of 2, tailed by two convolution layers with the kernel size of 4 and stride of 1. We use the Sigmoid non-linear activation function at the last layer and the Leaky ReLU with the slope of 0.2 for other layers. The structure branch shares the same pattern as the upper stream, where the input edge map is detected by a residual block [6] followed by a convolution layer with the kernel size of 1. Finally, the outputs of the two branches are concatenated in the channel dimension, based on which we calculate the adversarial loss.

Different from the case in the texture branch, it is intractable to optimize the adversarial loss of the structure branch only with the detected edge map, mainly due to the sparse nature of the edge. We therefore adopt the gray-scale image as an additional condition and feed the paired data as the input in the structure branch, as several previous studies do [28, 18]. As such, the structure discriminator not only estimates the authenticity of the generated structure, but also guarantees its consistency with the ground-truth image. Besides, spectral normalization [17] is used, as it proves effective in solving the well-known training instability problem of generative adversarial networks.

3.3. Loss Functions

The model is trained with a joint loss, containing the reconstruction loss, perceptual loss, style loss and adversarial loss, to render visually realistic and semantically reasonable results.

Formally, let G be the generator and D be the discriminator. Denote by $\boldsymbol{I}_{gt}$ the ground-truth image, $\boldsymbol{E}_{gt}$ the complete edge map, $\boldsymbol{Y}_{gt}$ the gray-scale image, $\boldsymbol{M}_{in}$ the initial binary mask (with value 1 for existing region, 0 otherwise), $\boldsymbol{I}_{in} = \boldsymbol{I}_{gt} \odot \boldsymbol{M}_{in}$ the damaged image, $\boldsymbol{E}_{in} = \boldsymbol{E}_{gt} \odot \boldsymbol{M}_{in}$ the damaged edge map, and $\boldsymbol{Y}_{in} = \boldsymbol{Y}_{gt} \odot \boldsymbol{M}_{in}$ the damaged gray-scale image. The output of our generator is defined as $\boldsymbol{I}_{out}, \boldsymbol{E}_{out} = G(\boldsymbol{I}_{in}, \boldsymbol{E}_{in}, \boldsymbol{Y}_{in}, \boldsymbol{M}_{in})$.

Reconstruction Loss. We adopt the ℓ_1 distance between $\boldsymbol{I}_{out}$ and $\boldsymbol{I}_{gt}$ as the reconstruction loss, formulated as:

$$\mathcal{L}_{rec} = \mathbb{E}\left[\|\boldsymbol{I}_{out} - \boldsymbol{I}_{gt}\|_1\right]. \qquad (13)$$

Perceptual Loss. Since the reconstruction loss struggles to capture high-level semantics, we introduce the perceptual loss $\mathcal{L}_{perc}$ to evaluate the global structure of an image. It measures the ℓ_1 distance of $\boldsymbol{I}_{out}$ to $\boldsymbol{I}_{gt}$ in the feature space defined by the VGG-16 network [23] pre-trained on ImageNet [22]:

$$\mathcal{L}_{perc} = \mathbb{E}\left[\sum_i \|\phi_i\left(\boldsymbol{I}_{out}\right) - \phi_i\left(\boldsymbol{I}_{gt}\right)\|_1\right], \qquad (14)$$

where $\phi_i(\cdot)$ denotes the activation map of the i-th pooling layer from VGG-16 given the input image $\boldsymbol{I}_*$. In our implementation, *pool*-1, *pool*-2 and *pool*-3 are used.

Style Loss. We further include the style loss to ensure style consistency. Similarly, the style loss calculates the ℓ_1 distance between feature maps:

$$\mathcal{L}_{style} = \mathbb{E}\left[\sum_i \|(\psi_i\left(\boldsymbol{I}_{out}\right) - \psi_i\left(\boldsymbol{I}_{gt}\right))\|_1\right], \qquad (15)$$

where $\psi_i(\cdot) = \phi_i(\cdot)^T \phi_i(\cdot)$ denotes the Gram matrix constructed from the activation map ϕ_i.

Adversarial Loss. The adversarial loss is to guarantee the visual authenticity of the reconstructed image as well as the consistency of textures and structures, defined as:

$$\mathcal{L}_{adv} = \min_G \max_D \mathbb{E}_{\boldsymbol{I}_{gt}, \boldsymbol{E}_{gt}}\left[\log D\left(\boldsymbol{I}_{gt}, \boldsymbol{E}_{gt}\right)\right]$$
$$+ \mathbb{E}_{\boldsymbol{I}_{out}, \boldsymbol{E}_{out}} \log\left[1 - D\left(\boldsymbol{I}_{out}, \boldsymbol{E}_{out}\right)\right]. \qquad (16)$$

Intermediate Loss. To encourage the structure and texture features to be accurately captured by the two decoders, respectively, we introduce intermediate supervisions on $\boldsymbol{F}_s$ and $\boldsymbol{F}_t$:

$$\mathcal{L}_{inter} = \mathcal{L}_{structure} + \mathcal{L}_{texture}$$
$$= \text{BCE}(\boldsymbol{E}_{gt}, \mathcal{P}_s(\boldsymbol{F}_s)) + \ell_1(\boldsymbol{I}_{gt}, \mathcal{P}_t(\boldsymbol{F}_t)), \qquad (17)$$

where $\mathcal{P}_s$ and $\mathcal{P}_t$ denote the projection functions implemented by a residual block followed by a convolution layer, which map $\boldsymbol{F}_s$ and $\boldsymbol{F}_t$ to edge map and RGB image, respectively.

In summary, the joint loss is written as:

$$\mathcal{L}_{joint} = \lambda_{rec}\mathcal{L}_{rec} + \lambda_{perc}\mathcal{L}_{perc} + \lambda_{style}\mathcal{L}_{style}$$
$$+ \lambda_{adv}\mathcal{L}_{adv} + \lambda_{inter}\mathcal{L}_{inter}, \qquad (18)$$

where λ_{rec}, λ_{perc}, λ_{style}, λ_{adv} and λ_{inter} are the tradeoff parameters, and we empirically set $\lambda_{rec} = 10$, $\lambda_{perc} = 0.1$, $\lambda_{style} = 250$, $\lambda_{adv} = 0.1$, and $\lambda_{inter} = 1$.

4. Experiments

Extensive experiments are conducted on three public datasets for both subjective and objective evaluation. Ablation studies are also performed to validate the specifically designed architecture and modules.
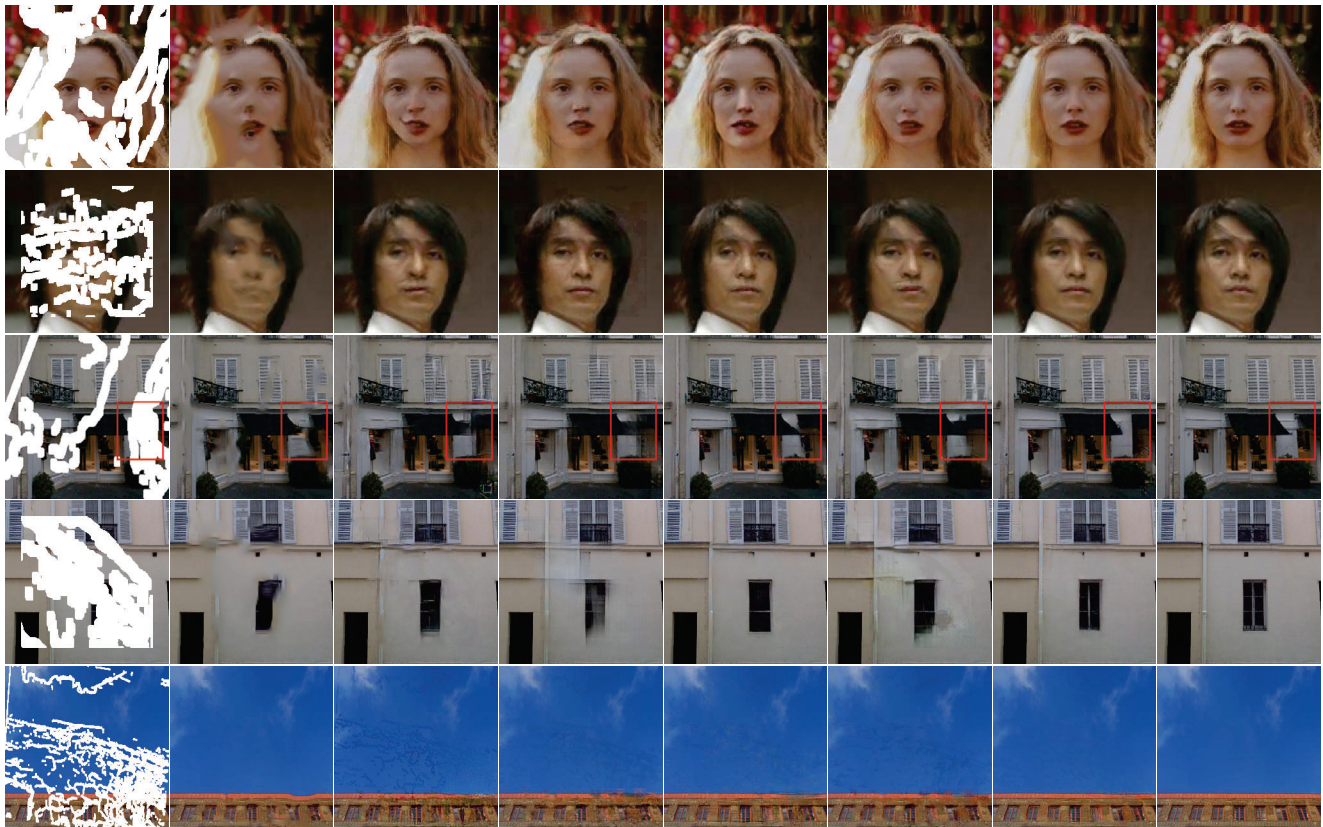
| (a) Input | (b) PatchMatch | (c) PConv | (d) DeepFillv2 | (e) RFR | (f) MED | (g) Ours | (h) Ground-truth |

Figure 5: Qualitative comparison on CelebA, Paris StreetView and Places2 (zoom in for a better view): (a) input corrupted images, (b) PatchMatch [2], (c) PConv [13], (d) DeepFillv2 [36], (e) RFR [11], (f) MED [14], (g) Ours, and (h) ground-truth images.

4.1. Experimental Settings

We evaluate the proposed method on the CelebA [16], Paris StreetView [4] and Places2 [39] datasets, which are widely adopted in the literature, and we follow their original training, testing, and validation splits. Irregular masks are obtained from [13] and classified based on their hole sizes relative to the entire image with an increment of 10%. All the images and corresponding masks are resized to 256×256 pixels.

The model is implemented in PyTorch. Training is launched on a single NVIDIA 1080TI GPU (11GB) with the batch size of 6, optimized with the Adam optimizer. Analogous to [13], we first use a learning rate of 2×10^{-4} for initial training, then finetune the model with a learning rate of 5×10^{-5}, and freeze the Batch Normalization (BN) parameters of the generator. The discriminator is trained with a learning rate of 1/10 of the generator. It takes around 4 days to train the models on CelebA and Paris StreetView and 10 days on Places2. The fine-tuning is completed within one day. The detailed architectures of the networks are shown in the supplementary material.

4.2. Qualitative Comparison

Figure 5 compares our results with the ones of the representative methods including the current state-of-the-arts on the three benchmarks. It can be seen, as a classical patch-based method, PatchMatch [2] fails in handling large holes. PConv [13] is suitable for irregular corruptions, but obvious artifacts can be observed in Figure 5 (c). DeepFilllv2 [36] suffers from over-smoothing predictions and distorted structures. With the Recurrent Feature Reasoning module, RFR [11] yields competitive results; however, the details are still not so elegant as ours (the face and sky in Figure 5 serve as examples). MED [14] attempts to correlate structure and texture generation, while the shared generator is inadequate for generating sharp edges and clear textures (*e.g.*, facades in Figure 5).

We also show additional comparison to EdgeConnect [18] and PRVS [10] in Figure 6 as these methods all claim to improve results by reconstructing image structures. Comparatively, the proposed model recovers more reasonable and sharper structures, leading to better results.

To sum up, our model is able to hallucinate more meaningful structures and vivid textures through dual generation.

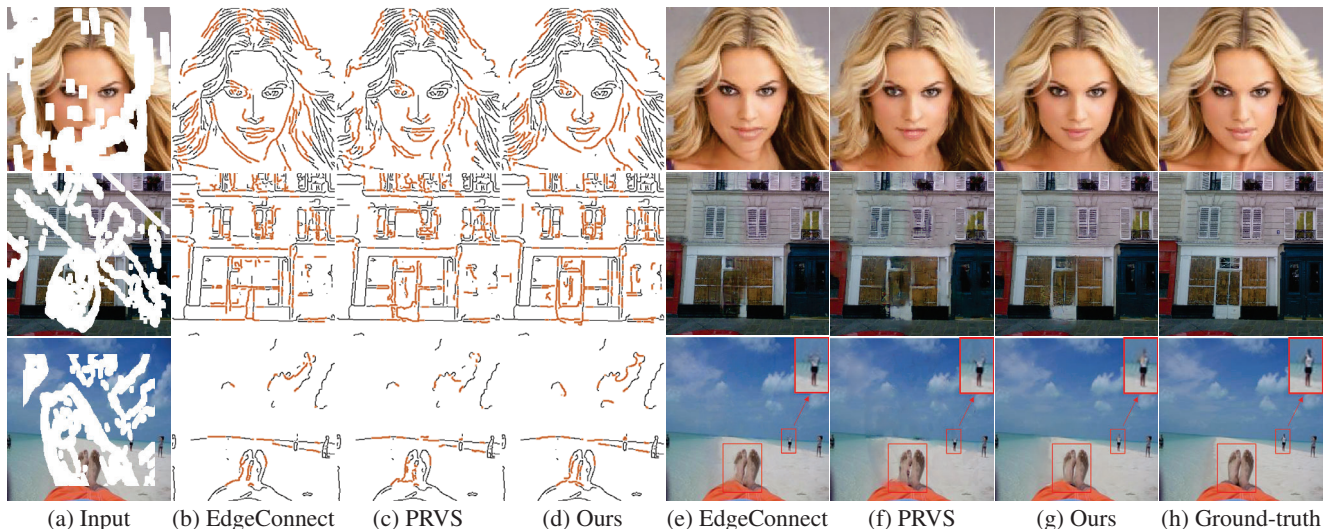| | (a) Input | (b) EdgeConnect | (c) PRVS | (d) Ours | (e) EdgeConnect | (f) PRVS | (g) Ours | (h) Ground-truth |

Figure 6: 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 [18], PRVS [10] and Ours; (e, f, g) corresponding filled results of EdgeConnect [18], PRVS [10] and Ours; and (h) ground-truth images.

Metrics	LPIPS†			PSNR¶			SSIM¶			User Study¶
Mask Ratio	0-20%	20-40%	40-60%	0-20%	20-40%	40-60%	0-20%	20-40%	40-60%	0-60%
PatchMatch [2]	0.074	0.183	0.332	30.02	24.77	20.51	0.864	0.680	0.487	2.7%
PConv [13]	0.065	0.134	0.283	30.19	25.18	21.20	0.885	0.730	0.527	4.0%
DeepFillv2 [36]	0.056	0.123	0.266	30.32	25.34	21.48	0.889	0.735	0.531	14.0%
RFR [11]	0.048	0.101	0.239	30.74	25.80	21.99	0.899	0.750	0.553	23.3%
EdgeConnect [18]	0.061	0.131	0.268	30.28	25.30	21.39	0.886	0.737	0.535	4.7%
PRVS [10]	0.057	0.124	0.257	30.30	25.39	21.50	0.893	0.742	0.541	5.3%
MED [14]	0.053	0.120	0.248	30.41	25.45	21.63	0.895	0.745	0.547	6.0%
Ours	**0.042**	**0.095**	**0.227**	**30.81**	**25.97**	**22.23**	**0.904**	**0.759**	**0.561**	**40.0%**

Table 1: Objective quantitative comparison and user study on Places2 (†Lower is better; ¶Higher is better).

4.3. Quantitative Comparison

Objective evaluation. We quantitatively evaluate the proposed method using three major metrics: LPIPS, PSNR and SSIM, and compare the scores to those of the state-of-the-art counterparts with irregular mask ratios of 0-20%, 20-40% and 40-60%. Table 1 shows the results achieved on the Places2 dataset, where the proposed method outperforms the other approaches, clearly demonstrating its effectiveness. More comparison on the CelebA and Paris StreetView datasets is shown in the supplementary material.

User Study. We further perform subjective user study. 10 volunteers with image processing expertise are involved in this evaluation. They are invited to choose the most realistic image from those inpainted by the proposed method and the representative state-of-the-art approaches. Specifically, each participant has 15 questions, which are randomly sampled from the Places2 dataset. We tally the votes and show the statistics in Table 1. Our method performs more favorably against the other ones by a large margin, clearly validating its effectiveness.

4.4. Analysis on Network Architecture

Our model assumptions include that structure priors are essential to image inpainting and dual generation of textures and structures is beneficial. We therefore experimentally verify the credit of such architecture design on Paris StreetView.

On Structure Priors. To highlight the structure priors, we build a single-stream network as baseline, which fills missing regions by solely modeling texture features, and the discriminator is single-stream accordingly. As shown in Figure 7 (b), the baseline method does not well deal with complex structures and tends to smooth out the detailed textures in case of large corruptions. The quantitative results in Table 2 also indicate that our network with structure priors significantly improves the baseline.

On Two-stream Network Architecture. To further highlight the two-stream dual generation architecture, we compare it with a multi-task single-stream network, which is tailed by two branches to model the image structure and texture simultaneously. We enlarge its channels to make it have
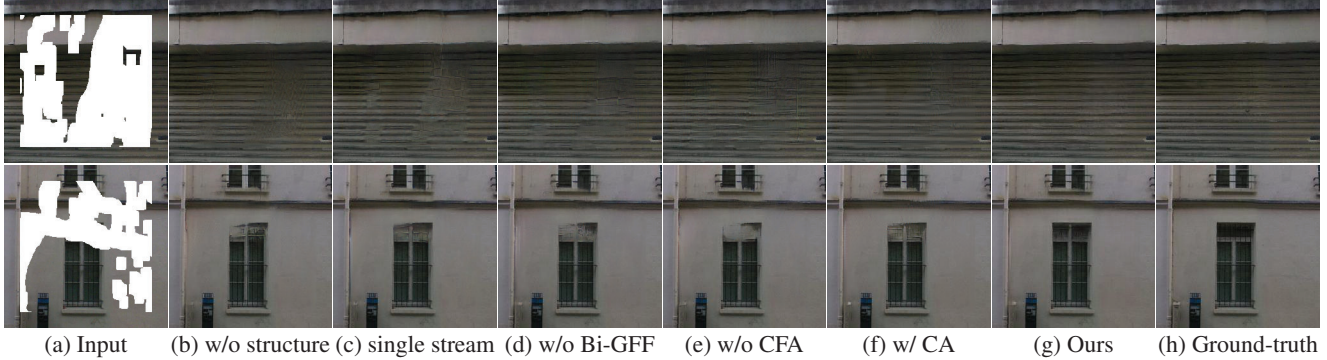
| (a) Input | (b) w/o structure | (c) single stream | (d) w/o Bi-GFF | (e) w/o CFA | (f) w/ CA | (g) Ours | (h) Ground-truth |

Figure 7: Visualization of the effects of network architecture and individual modules on Paris StreetView.

Metrics	LPIPS†			PSNR¶			SSIM¶		
Mask Ratio	0-20%	20-40%	40-60%	0-20%	20-40%	40-60%	0-20%	20-40%	40-60%
w/o structure priors	0.054	0.129	0.251	31.72	26.71	22.22	0.909	0.755	0.550
single-stream	0.051	0.122	0.245	32.27	27.03	22.59	0.913	0.764	0.558
w/o Bi-GFF	0.045	0.114	0.236	32.61	27.20	22.75	0.919	0.772	0.567
w/o CFA	0.049	0.119	0.243	32.34	27.09	22.64	0.914	0.766	0.561
w/ CA	0.043	0.115	0.240	32.54	27.15	22.69	0.920	0.769	0.566
Ours	**0.039**	**0.107**	**0.226**	**32.93**	**27.48**	**22.89**	**0.923**	**0.777**	**0.573**

Table 2: Quantitative ablation study on Paris StreetView.

the same amount of parameters as the proposed network. The Bi-GFF and CFA modules are embedded to refine generation as the proposed model. As shown in Figure 7 (c), the two-stream architecture exhibits superior performance with more visually reasonable structures and detailed textures. Quantitative results in Table 2 also validate the advantages of texture and structure dual generation.

4.5. Ablation Study

On Bi-directional Gated Feature Fusion. The Bi-GFF module is developed to enhance the consistency of the rebuilt structures and textures. For the results obtained using a simpler fusion module (a channel-wise concatenation followed by a convolution layer), blurred edges and unexpected noise can be observed in Figure 7 (d), especially around complex boundaries, such as the windows. To make the comparison more specific, quantitative results are given in Table 2, which indicate that Bi-GFF contributes to the performance gain.

On Contextual Feature Aggregation. The CFA module is introduced to enhance the correlation between local features and the overall image consistency. As shown in Figure 7 (e), the model without CFA renders low-quality images, and texture filling is sensitive to structure noise. Quantitative results in Table 2 also validate its necessity.

On Multi-scale Feature Aggregation in CFA. As our CFA module is updated from the contextual attention layer [35], we directly compare it with the original version to prove its

effectiveness. As shown in Figure 7 (f) and Table 2, we demonstrate that multi-scale feature aggregation obviously benefits the quality of the results, with consistent textures and better quantitative scores reported.

5. Conclusion

In this paper, we propose a novel two-stream image inpainting method, which recovers corrupted image by simultaneously modeling structure-constrained texture synthesis and texture-guided structure reconstruction. In this way, the two subtasks exchange useful information and thus facilitate each other. Furthermore, a Bi-directional Gated Feature Fusion module is introduced followed by a Contextual Feature Aggregation module to refine the results, with both semantically reasonable structures and detail-rich textures. Experiments show that this model is competent for this issue and outperforms the state-of-the-art counterparts.

References

[1] Coloma Ballester, Marcelo Bertalmio, Vicent Caselles, Guillermo Sapiro, and Joan Verdera. Filling-in by joint interpolation of vector fields and gray levels. *IEEE TIP*, 10(8):1200–1211, 2001. 2

[2] 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. 1, 2, 6, 7

[3] Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. Image inpainting. In *SIGGRAPH*, 2000. 1, 2

[4] Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei Efros. What makes paris look like paris? *ACM TOG*, 31(4):101:1–101:9, 2012. 2, 6

[5] Alexei A Efros and William T Freeman. Image quilting for texture synthesis and transfer. In *SIGGRAPH*, 2001. 1

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[7] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Globally and locally consistent image completion. *ACM TOG*, 36(4):107:1–107:14, 2017. 1, 5

[8] Avisek Lahiri, Arnav Kumar Jain, Sanskar Agrawal, Pabitra Mitra, and Prabir Kumar Biswas. Prior guided gan based semantic inpainting. In *CVPR*, 2020. 2

[9] Chuan Li and Michael Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *ECCV*, 2016. 5

[10] Jingyuan Li, Fengxiang He, Lefei Zhang, Bo Du, and Dacheng Tao. Progressive reconstruction of visual structure for image inpainting. In *ICCV*, 2019. 2, 6, 7

[11] Jingyuan Li, Ning Wang, Lefei Zhang, Bo Du, and Dacheng Tao. Recurrent feature reasoning for image inpainting. In *CVPR*, 2020. 6, 7

[12] Liang Liao, Jing Xiao, Zheng Wang, Chia-wen Lin, and Shin'ichi Satoh. Guidance and evaluation: Semantic-aware image inpainting for mixed scenes. In *ECCV*, 2020. 2

[13] 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. 1, 6, 7

[14] 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. 2, 6, 7

[15] Hongyu Liu, Bin Jiang, Yi Xiao, and Chao Yang. Coherent semantic attention for image inpainting. In *ICCV*, 2019. 1

[16] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015. 2, 6

[17] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *ICLR*, 2018. 5

[18] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Qureshi, and Mehran Ebrahimi. Edgeconnect: Structure guided image inpainting using edge prediction. In *ICCVW*, 2019. 1, 2, 5, 6, 7

[19] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016. 1

[20] Yurui Ren, Xiaoming Yu, Ruonan Zhang, Thomas H Li, Shan Liu, and Ge Li. Structureflow: Image inpainting via structure-aware appearance flow. In *ICCV*, 2019. 1, 2

[21] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 5

[22] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015. 5

[23] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 5

[24] Yuhang Song, Chao Yang, Zhe Lin, Xiaofeng Liu, Qin Huang, Hao Li, and C-C Jay Kuo. Contextual-based image inpainting: Infer, match, and translate. In *ECCV*, 2018. 1

[25] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *CVPR*, 2018. 2

[26] Yi Wang, Ying-Cong Chen, Xin Tao, and Jiaya Jia. Vcnet: A robust approach to blind image inpainting. In *ECCV*, 2020. 2

[27] Chaohao Xie, Shaohui Liu, Chao Li, Ming-Ming Cheng, Wangmeng Zuo, Xiao Liu, Shilei Wen, and Errui Ding. Image inpainting with learnable bidirectional attention maps. In *ICCV*, 2019. 1

[28] Wei Xiong, Jiahui Yu, Zhe Lin, Jimei Yang, Xin Lu, Connelly Barnes, and Jiebo Luo. Foreground-aware image inpainting. In *CVPR*, 2019. 1, 2, 5

[29] Zongben Xu and Jian Sun. Image inpainting by patch propagation using patch sparsity. *IEEE TIP*, 19(5):1153–1165, 2010. 2

[30] Zhaoyi Yan, Xiaoming Li, Mu Li, Wangmeng Zuo, and Shiguang Shan. Shift-net: Image inpainting via deep feature rearrangement. In *ECCV*, pages 1–17, 2018. 1

[31] Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. High-resolution image inpainting using multi-scale neural patch synthesis. In *CVPR*, 2017. 1

[32] Jie Yang, Zhiquan Qi, and Yong Shi. Learning to incorporate structure knowledge for image inpainting. In *AAAI*, 2020. 2

[33] Raymond A Yeh, Chen Chen, Teck Yian Lim, Alexander G Schwing, Mark Hasegawa-Johnson, and Minh N Do. Semantic image inpainting with deep generative models. In *CVPR*, 2017. 1

[34] Zili Yi, Qiang Tang, Shekoofeh Azizi, Daesik Jang, and Zhan Xu. Contextual residual aggregation for ultra high-resolution image inpainting. In *CVPR*, 2020. 2

[35] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. In *CVPR*, 2018. 1, 2, 4, 5, 8

[36] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. In *ICCV*, 2019. 1, 2, 5, 6, 7

[37] Yu Zeng, Zhe Lin, Jimei Yang, Jianming Zhang, Eli Shechtman, and Huchuan Lu. High-resolution image inpainting with iterative confidence feedback and guided upsampling. In *ECCV*, 2020. 2

[38] Lei Zhao, Qihang Mo, Sihuan Lin, Zhizhong Wang, Zhiwen Zuo, Haibo Chen, Wei Xing, and Dongming Lu. Uctgan: Diverse image inpainting based on unsupervised cross-space translation. In *CVPR*, 2020. 2

[39] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE TPAMI*, 40(6):1452–1464, 2018. 2, 6

[40] Tong Zhou, Changxing Ding, Shaowen Lin, Xinchao Wang, and Dacheng Tao. Learning oracle attention for high-fidelity face completion. In *CVPR*, 2020. 2