Image Inpainting Guided by Coherence Priors of Semantics and Textures

Liang Liao[1,2] Jing Xiao[1,*] Zheng Wang[1] Chia-Wen Lin[3] Shin'ichi Satoh[2]
[1] National Engineering Research Center for Multimedia Software,
School of Computer Science, Wuhan University
[2] National Institute of Informatics [3] National Tsinghua University
{liaoliangwhu, wangzwhu, jing}@whu.edu.cn, cwlin@ee.nthu.edu.tw, satoh@nii.ac.jp

Abstract

Existing inpainting methods have achieved promising performance in recovering defective images of specific scenes. However, filling holes involving multiple semantic categories remains challenging due to the obscure semantic boundaries and the mixture of different semantic textures. In this paper, we introduce coherence priors between the semantics and textures which make it possible to concentrate on completing separate textures in a semantic-wise manner. Specifically, we adopt a multi-scale joint optimization framework to first model the coherence priors and then accordingly interleaving optimize image inpainting and semantic segmentation in a coarse-to-fine manner. A Semantic-Wise Attention Propagation (SWAP) module is devised to refine completed image textures across scales by exploring non-local semantic coherence, which effectively mitigates the mix-up of textures. We also propose two coherence losses to constrain the consistency between the semantics and the inpainted image in terms of the overall structure and detailed textures. Experimental results demonstrate the superiority of our proposed method for challenging cases with complex holes.

1. Introduction

High-quality image inpainting aims to fill in missing regions with synthetic content [1, 2, 5]. It requires both semantically meaningful structures and visually pleasing textures. To this end, deep learning-based methods [24, 39, 42, 44, 46, 47] resort to encoder-decoder based networks to infer the context of a corrupted image and then refine the texture details in the initial inference of a missing region by some tools, such as non-local algorithms. Although current image inpainting methods have made significant progress, it still poses technical challenges in completing complex holes, particularly when a missing region involves multiple sub-regions with different semantic classes. The main reason falls in the failure of modeling the prior distributions of a mixture of different semantic regions, which usually result in blurry boundaries and unrealistic textures [15, 16].

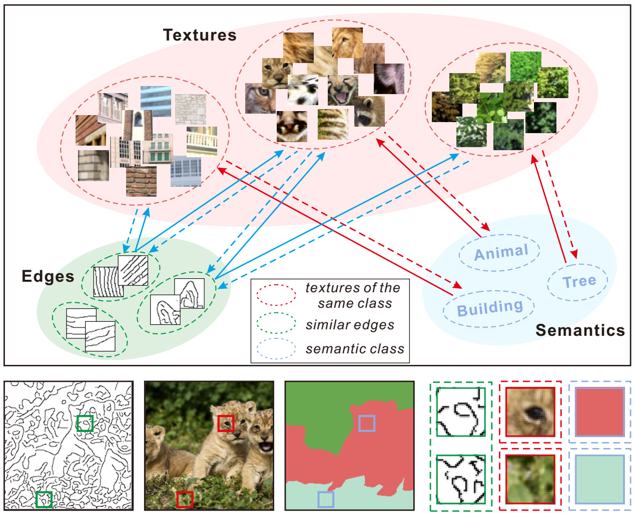


Figure 1: Upper part: Mapping between image textures and edges/semantics (Dot arrow - extraction of edges/semantics; solid arrow - texture generation). Notice that two similar edge patches in a green circle could be mapped to completely different semantic textures, but one semantic will be clearly mapped to a certain texture category. Lower part: an example showing two similar edge patches map to two different semantic textures.

A feasible approach is to adopt structural information, such as edges [12, 21], contours [38], and smooth images [26], as guidance to complete missing structures and textures in two steps. The assumption is that structures offer semantic clues for inferring an unknown scene, making them suitable for guiding the filling of textures. However, we notice that the correspondence between structural information and textures is not apparent, making the filled textures still highly rely on the local correlation around the missing region. Figure 1 demonstrates the ambiguity of the mapping from mid-level structures (e.g., the edges) to the textures, which can significantly degrade the visual authenticity of the generated textures.

Compared with mid-level structures, high-level semantic information offers more vital semantic clues to the object textures. For example, in Figure 1, the semantic *animal* leads to fluffy fur while the semantic *tree* leads to green leaf in the image, which cannot be usually distinguished solely from their mid-level structures. In contrast, object textures have been shown to provide sufficient information about the semantic classes [7, 11]. Thus we characterize the relationships between the semantics and textures of objects as **coherence priors** and build our inpainting method on the coherence priors to complete complex holes while ensuring the mutual consistency between the predicted semantics and textures.

Based on the above motivation, we propose to utilize coherence priors between semantics and textures to facilitate joint optimization of semantic segmentation and image inpainting. To this end, our framework extracts a shared feature to represent the common information of the two tasks, and characterize the interaction between scales to enable the utilization of coherence priors to optimize the two tasks jointly. Specifically, two novel designs are proposed: 1) A Semantic-Wise Attention Propagation (SWAP) module is used to explicitly capture the semantic relevance between an unknown (missing) area and the known regions. As a result, when mapping semantics to image textures, filling in an unknown patch only refers to those known patches with the same semantic, rather than to the entire image, to avoid irrelevant texture filling. 2) We devise two loss terms to learn the global and local coherence relationships, respectively. The image-level structure coherence loss is used to supervise the structural matching between the inpainted image and the corresponding segmentation map to generate clear boundaries in the inpainted image. Besides, the non-local patch-level coherence loss aims to assess the distribution of patch textures in the semantic domain to encourage the generated textures to be as similar as the matched known patch of the same semantics.

Unlike the existing semantics-guided inpainting methods, such as SPG-Net [28] and SGE-Net [15], which synthesize textures by convolution to involve the local semantic information, our proposed method predicts the textures and the semantics simultaneously, and borrows the known texture feature of the same semantic to fill in a missing region by a semantics-guided non-local means, which not only ensures realistic textures but also is valuable for semantic recognition.

The main contributions of our paper are three-fold:

- We introduce coherence priors that highlight the mutual consistency between the semantics and textures in image inpainting and devise two coherence losses to boost the consistency between semantic information and inpainted image in the global structure level and local texture level.
- We propose a novel semantic-wise attention propagation module, which generates semantically realistic textures by capturing distant relationships and referring to the texture feature of the same semantic in the feature maps.
- Our approach outperforms existing state-of-the-art image inpainting methods [15, 21, 43, 45] on completing complex hole with multiple semantic regions in terms of the sharpness of boundaries and the coherence and visual plausibility of textures.

2. Related Work

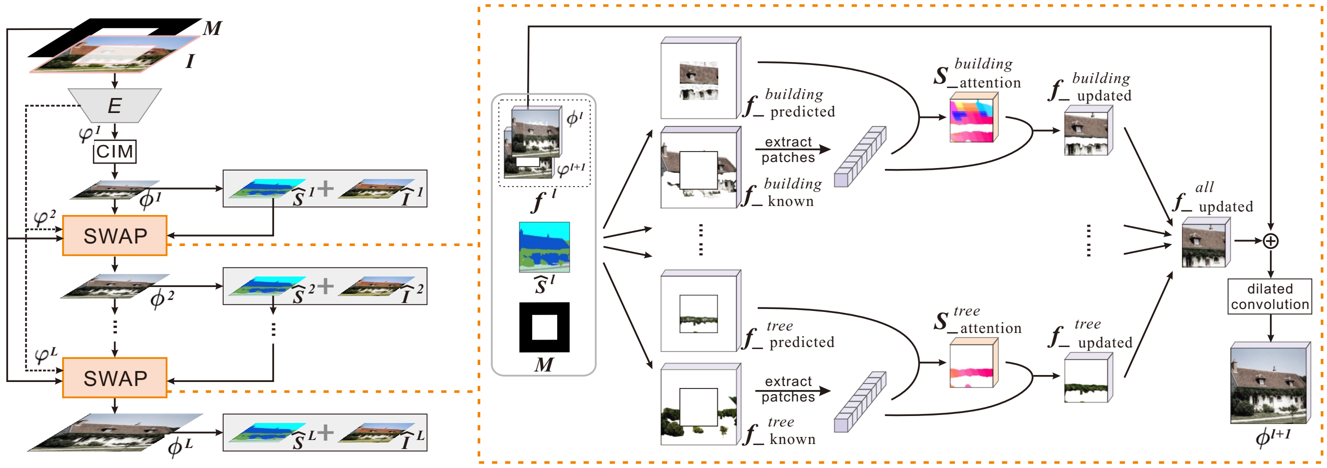
2.1. Image inpainting

Deep learning-based inpainting approaches have recently been proposed by understanding the images, which can generate meaningful content for filling in the missing region. Context Encoder [24] was first proposed to employ generative adversarial networks and demonstrated its potential for inpainting tasks. Based on it, efforts have been made to enhance the inpainting performance, including introducing specific losses [6, 34, 47], building recursive architectures for progressive refinement [13, 40], designing special convolutional layers to better handle irregular holes [18, 43], and involving the structural priors in a two-stage framework as guidance for structural consistency [14, 21, 26, 41]. However, these methods lack the ability to model long-term correlations between distant contexts, leading to blurry textures.

To better refine the inpainted image textures, non-local algorithms are adopted to borrow distant features from a known region, which contains fine textures, to the missing region. [42] firstly proposed to compute textural affinity within the same image to fill the corrupted area with more realistic texture patches from the available area. [45] devised a pyramid of contextual attention at multiple layers to refine the textures from high-level to low-level. [19] used a coherent semantic attention layer to ensure semantic relevance between nearby filled features. [37] extends the single attention map to bidirectional attention maps and re-normalizes the features to let the decoder concentrate on filling the holes. Although these methods have delivered considerable improvements, they failed to address the semantic ambiguity as they try to measure the texture affinity across all semantics.

2.2. Semantic-guided image processing

Recent research reveals that the semantic priors of the high-level vision tasks are useful in the guidance of low-level vision tasks [4, 17, 32, 35]. One hot research topic is semantic-guided image generation. It covers several research directions, including image translation from segmentation maps to realistic images [10, 31] and semantic image



(a) Overall Framework

(b) Semantic-wise Attention Propagation (SWAP) Module

Figure 2: Proposed network architecture. At each scale, both the inpainted image and segmentation map are output from two task-specific heads to control the predicted structures of the shared features. SWAP is added between scales to progressively optimize the texture details of contextual feature.

synthesis [4, 29, 32]. To avoid the vanishing of semantic priors in the generation process, [23] proposed a spatially adaptive normalization layer to propagate the semantic information to the synthesized images. With the useful semantic priors, it can generate high-quality images.

The semantic priors have also been applied to promote many conditional low-level vision tasks and demonstrated its effectiveness in constraining the plausible solution space in the ill-posed problems. For example, they are involved in the tasks of super-resolution [33], dehazing [25], denoising [17], style transfer [20, 30], image manipulation [9, 22]. Inspired by the successful assistance from semantic priors in conditional image generation, we also exploit the semantic guidance in completing a corrupted image, especially when the hole involves multiple semantic regions.

3. Proposed Method

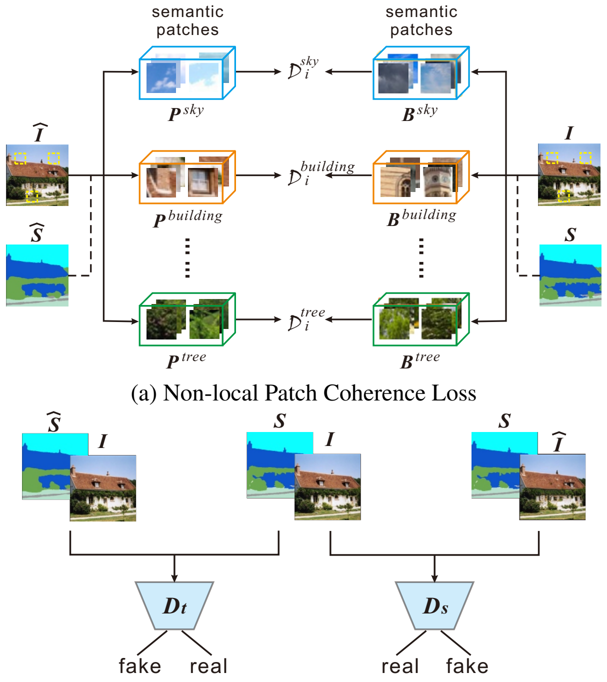
How to achieve high-quality inpainting results on both semantically reasonable structures and visually pleasing textures? We argue that such results should not only be able to reconstruct the structures of the semantic objects for the global structure, but also the textures of them should look realistic with the same semantic in the image for local pixel continuity. To this end, we build a multi-task learning framework on the coherence priors to explicitly reconstruct both the structures and textures of semantic objects. Moreover, we propose a new SWAP module to optimize the textures by semantically binding the textures between an inpainted region and the known regions based on the coherence priors. we also devise two losses to guide the learning of coherence relationships both in the global structure level and in the local patch level, respectively.

3.1. Framework Overview

We build our network on an alternating-optimization architecture to utilize the coherence priors to mutually assist image inpainting and semantic segmentation for a corrupted image. Specifically, we propose a multi-task learning framework by sharing features in the decoder for the two tasks (as shown in Figure 2). The encoder encodes the corrupted image and its mask into hierarchical contextual features, which are then fed into the decoder to predict the inpainted images and semantic segmentation maps across scales. Prior to feeding the encoded feature of the last layer into the decoder, we initially complete the feature via a Context Inference Module (CIM) based on the contextual inference method [15, 36].

In the decoder, the contextual feature at each scale is processed by two task-specific heads to predict the inpainted image and the segmentation map, respectively. Different from the method proposed in [15] that updates the contextual features by spatial adaptive normalization to capture the common properties of the same semantic, we propose a SWAP module to stress the realistic texture of each semantic patch by referring to the semantic relevant features from the known regions. In this way, the contextual features are learned to represent the global structure and refined with the semantic-aware texture details.

For brevity, we adopt the following notations. φ^l and ϕ^l denote the features from the encoder and decoder at scale l, respectively; $\hat{I}^l$ and $\hat{S}^l$ respectively represent the inpainted image and the predicted k-channel segmentation map from a inpainting head $h(\cdot)$ and a segmentation head $g(\cdot)$ at scale l, where k is the total number of semantic labels; l ranges from 1 (the coarsest layer) to 5 (the finest layer).



(a) Non-local Patch Coherence Loss

(b) Structure Coherence Loss

Figure 3: Proposed coherence losses. (a) non-local patch coherence loss encourages the generated texture to be as similar as any known patch of the same semantic in the real image; (b) structure coherence loss ensures the structural consistency between the entire segmentation map and the inpainted image.

3.2. Semantic-wise Attention Propagation (SWAP)

The SWAP module is designed to optimize the contextual features by enhancing the semantic authenticity of textures based on coherence priors. As shown in Figure 2(b), SWAP takes four inputs: two of them are the current-scale feature ϕ^l and the next-scale skip feature φ^{l+1} from the encoder. The third is the predicted segmentation probability map $\hat{S}^l$, that is used to guide the separation of features. The last is the missing-region mask M. The propagation process can be formulated as follows:

$$\phi^{l+1} = swap(\phi^l, \varphi^{l+1}, \hat{S}^l, M), \tag{1}$$

where $swap(\cdot)$ is the process of refining the contextual features in SWAP.

The attention-based approaches in [27, 37, 42] resort to contextual attention to pick known regions as references to complete a missing region, which, however, cannot distinguish patches of different semantics and thereby leads to blurry boundaries and semantic confusion during attention propagation. Unlike these approaches, SWAP calculates the attention scores by matching semantic-aware features of missing patches and the known patches based on the coherence priors. Specifically, we first split the contextual features f^l, which is generated from ϕ^l and φ^{l+1}, into different semantic parts according to the labels in the predicted segmentation map $\hat{S}^l$ of the l-th layer. We subsequently drop the superscript l for the sake of notational simplicity.

Within each semantic part, each patch's attention score is evaluated by the patch affinity between the missing region and a known region using the normalized inner product followed by a softmax operation:

$$Df^c_{i,j} = \langle \frac{p^c_i}{||p^c_i||_2}, \frac{p^c_j}{||p^c_j||_2} \rangle, \tag{2}$$

$$\omega^c_{j,i} = \frac{\exp(Df^c_{i,j})}{\sum_{i=1}^{N} \exp(Df^c_{i,j})}, \tag{3}$$

where p^c_i is the i-th patch extracted from semantic feature f^c of class c in the known region, p^c_j is the j-th patch extracted from f^c in the missing region. $Df^c_{i,j}$ is the affinity between them, and $\omega^c_{j,i}$ is the attention score representing the normalized affinities for each patch.

After obtaining the attention score from the known region, the feature of the j-th missing patch is updated by

$$p^c_j = \sum_{i=1}^{N^c} \omega^c_{j,i}\, p^c_i. \tag{4}$$

where N^c is the total number of patches of semantic class c in the known region.

The output feature of SWAP is generated by merging the updated features of all semantics, followed by four groups of dilated convolutions with different rates to improve the structural coherence in the final reconstructed features.

3.3. Coherence Losses

We devise new coherence losses between the semantics and textures as supervisions to guide the image inpainting and semantic segmentation to meet the following requirements: 1) the overall structures between the inpainted image and segmentation map should match each other; 2) the predicted textures of a certain semantic class should have the same distribution as that of the semantic textures in the known region. Under these considerations, we propose two coherence losses shown in Figure 3, a non-local patch coherence loss and a structure coherence loss, to respectively evaluate the patch similarity and the structural matching.

Non-local Patch Coherence Loss. Given the final inpainted image $\hat{I}$ and the predicted segmentation map $\hat{S}$, we aim to maximize the texture similarity between $\hat{I}$ and I. This is, the generated patches attributed to a specific class c should be similar to the realistic patches of the same class in the ground-truth image.

Similar to the attention propagation process, we first split $\hat{I}$ and I into different semantic images and extract patches to build the corresponding semantic patch sets $P^c = \{p^c_j\}$ and

$B^c = \{b^c_i\}$ according to $\hat{S}$ and S. For each patch in P^c, we randomly select one patch from B^c with the same semantic, and compute the pairwise cosine distances between them as

$$Di^c_{i,j} = 1 - \langle \frac{p^c_i - \mu^c_b}{||p^c_i - \mu^c_b||_2}, \frac{b^c_j - \mu^c_b}{\big|\big|b^c_j - \mu^c_b\big|\big|_2} \rangle, \tag{5}$$

where $\mu^c_b = \frac{1}{N_c}\sum_j b^c_j$, N_c is the number of patches in B_c.

The non-local patch coherence loss for each semantic class c aims to maximize the similarity between the patch couples:

$$\mathcal{L}^c_{\text{nlc}}(P^c, B^c) = -\log(\frac{1}{N^c_P}(\sum_i Di^c_{i,j})), \tag{6}$$

where N^c_P is the cardinality of the set of the generated patches with class label c. Our objective is defined as the sum of all the single-class non-local patch coherence losses over different classes found in $\hat{S}$:

$$\mathcal{L}_{\text{nlco}}(I, \hat{I}, S, \hat{S}) = \sum_c \mathcal{L}^c_{\text{nlco}}(P^c, B^c), \tag{7}$$

where c assumes all the class labels of mask in $\hat{S}$. Note that if the label value in $\hat{S}$ is not found in S, the coherence loss of the corresponding semantic patch set is set to 0.

Structure Coherence Loss. Besides the local patch similarity, we adopt a structure coherence loss to encourage the structural coherence between the inpainted image and the predicted segmentation map. In this work, we use two conditional discriminators to judge whether the semantics and the same image's textures are coherence. The texture-conditioned discriminator D_t is introduced to detect the predicted segmentation map's "fakes" given the real image, while the semantics-conditioned discriminator D_s is trained to detect the inpainted image's "fakes" given the real segmentation map. The structure coherence loss can be expressed as:

$$\begin{aligned}\mathcal{L}_{\text{sco}}(I, \hat{I}, S, \hat{S}) &= \mathcal{L}_{c_s}(I, \hat{I}, S) + \mathcal{L}_{c_t}(S, \hat{S}, I) \\ &= \mathbb{E}_{I,S}[\log D_s(I, S)] + \mathbb{E}_{\hat{I},S}[\log(1 - D_s(\hat{I}, S))] \\ &+ \mathbb{E}_{S,I}[\log D_t(S, I)] + \mathbb{E}_{\hat{S},I}[\log(1 - D_t(\hat{S}, I))].\end{aligned} \tag{8}$$

3.4. Objective Functions

We design appropriate supervised loss terms for learning the inpainting and segmentation tasks at each scale to obtain multi-scale predictions. We adopt the reconstruction loss, the adversarial loss and the proposed coherence losses to promote the fidelity of the inpainted images. The cross entropy loss is adopted for ensuring the accuracy of the predicted segmentation maps.

Reconstruction Loss. We use the $\mathcal{L}_1$ loss to encourage per-pixel reconstruction accuracy at all scales.

$$\mathcal{L}_1(I, \hat{I}) = \sum_l \Big|\Big|I - up(\hat{I}_l)\Big|\Big|. \tag{9}$$

where $up(\cdot)$ is to upsample $\hat{I}_l$ to the same size as I.

Adversarial Loss. We use a multi-scale PatchGAN [32] to classify the global and local patches of an image at different resolutions. The discriminator at each scale is identical and only the input is a differently scaled version of an image.

$$\mathcal{L}_\alpha(I, \hat{I}) = \sum_{k=1,2,3} (\mathbb{E}_I[\log D(p^k_I)] + \mathbb{E}_{\hat{I}}[(1 - \log D(p^k_{\hat{I}})]), \tag{10}$$

where $D(\cdot)$ is the discriminator, p^k_I and $p^k_{\hat{I}}$ are the patches in the k-th scaled versions of I and $\hat{I}$.

Cross-Entropy Loss. This loss is used to penalize the segmentation performance.

$$\mathcal{L}_{\text{xe}}(S, \hat{S}) = -\sum_l \sum_{p \in S} S(p) \log(up(\hat{S}^l)(p)), \tag{11}$$

where p is the pixel index for segmentation map S.

Overall Training Loss. The overall training loss function for our network is defined as the weighted sum of the above mentioned losses.

$$\begin{aligned}\mathcal{L}_{Final} &= \mathcal{L}_1(I, \hat{I}) + \lambda_\alpha \mathcal{L}_\alpha(I, \hat{I}) + \lambda_{\text{xe}} \mathcal{L}_{\text{xe}}(S, \hat{S}) \\ &+ \lambda_{\text{co}}(\mathcal{L}_{\text{nlco}}(I, \hat{I}, S, \hat{S}) + \mathcal{L}_{\text{sco}}(I, \hat{I}, S, \hat{S})),\end{aligned} \tag{12}$$

where λ_α, λ_{se} and λ_{co} are the weights for the adversarial loss, cross-entropy loss and coherence loss, respectively.

4. Results

4.1. Experimental Settings

We evaluate our method on the **Outdoor Scenes** [33] and **Cityscapes** [48] datasets. **Outdoor Scenes** contains 9,900 training images and 300 test images. **Cityscapes** contains 5,000 street-view images in total. In order to enrich the training set of **Cityscapes**, we use 2,975 images from the training set and 1,525 images from the test set for training, and test on the 500 images from the validation set. Since the test set lacks human-labeled semantic annotations, we generate the annotations for training by using the state-of-the-art segmentation model Deeplab [3]. We resize each training image to ensure its minimal height/width to be 256 for **Outdoor Scenes** and 512 for **Cityscapes**, and then randomly crop sub-images of size 256×256 as inputs to our model. The fine annotations of segmentation labels for both datasets are also provided for training, in which **Outdoor Scenes** and **Cityscapes** are annotated to 8 and 20 categories,

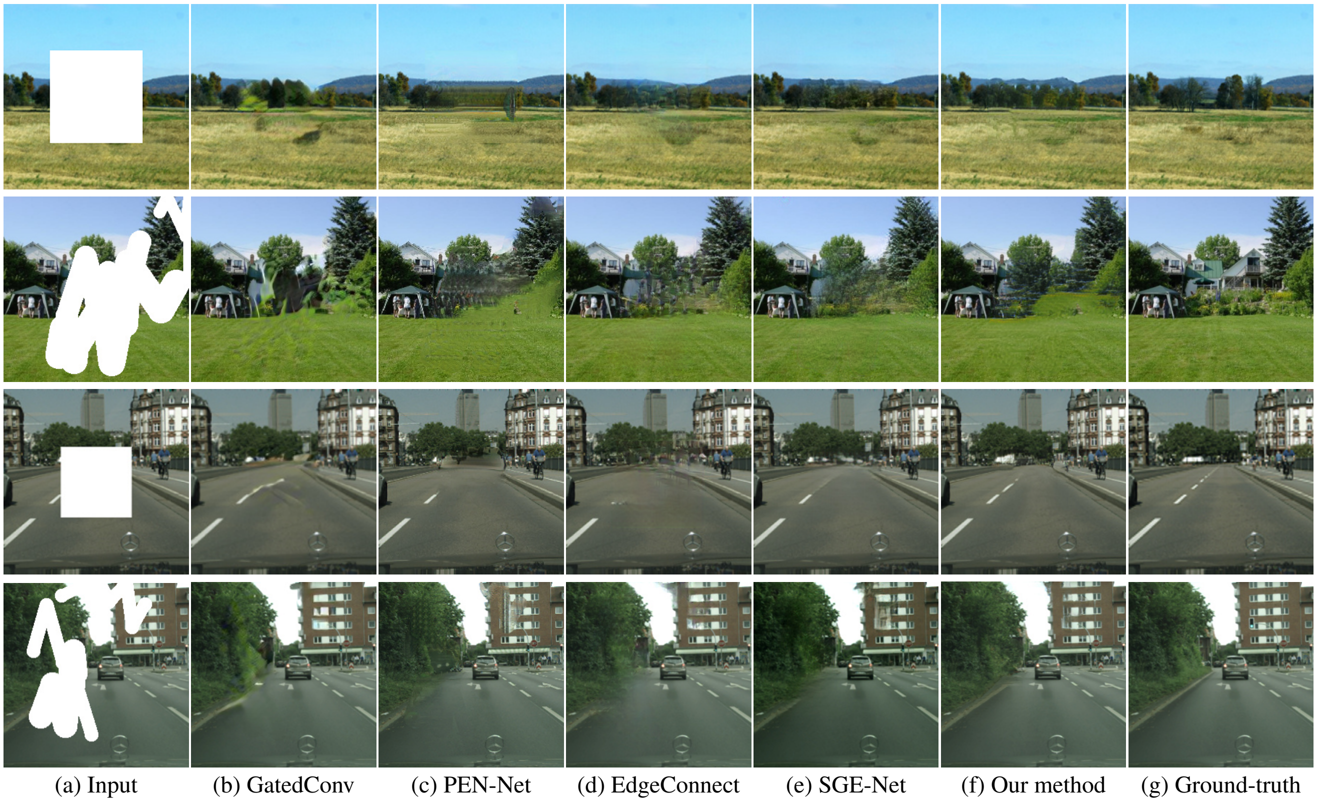


Figure 4: Qualitative comparison of inpainting results on image samples from **Outdoor Scenes** and **Cityscapes**.

respectively. Please note that the annotations can also be replaced by the extracted segmentation maps from the state-of-the-art segmentation models.

We compare our method with the following four learning-based inpainting methods: 1) GatedConv [43]: Contextual attention for leveraging the surrounding textures and structures. 2) EdgeConnect [21]: Two-stage inpainting framework with edges as low-level structural information. 3) PEN-Net [45]: Cross-layer attention transfer and pyramid filling in a multi-scale framework. 4) SGE-Net [15]: Semantic guidance for inpainting based on spatially adaptive normalization [23].

4.2. Qualitative Comparisons

Figure 4 shows the qualitative comparisons of our method with all the baselines. The corrupted area is simulated by sampling a central hole (128 × 128 for **Outdoor Scenes** and 96 × 96 for **Cityscapes**) or randomly placing multiple irregular masks based on [43]. As shown in the figure, the baselines usually suffer from artifacts and unsatisfactory boundaries while completing complex holes. GatedConv and PEN-Net adopt contextual attention to bring in the features of the known region, but they usually distort the structures when referencing to incorrect semantic textures from the surrounding, especially in completing the complex holes. EdgeConnect and SGE-Net are able to recover correct structures owning to the use of structure priors. However, EdgeConnect may generate mixed edges, making it difficult to generate correct textures, whereas the textures of SGE-Net are often over-smoothed without texture refinement. In contrast, our method generates more realistic textures and better boundaries delineating semantic regions than all the baselines thanks to the coherence priors between semantics and textures.

4.3. Quantitative Comparisons

Table 1 shows the quantitative comparisons on **Outdoor Scenes** and **Cityscapes** datasets based on three quality metrics: Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM) and Fréchet Inception Distance (FID) [8]. In general, the proposed method achieves significantly better objective scores than the baselines, especially in PSNR and SSIM.

4.4. User Study

We randomly select 100 images from the two datasets (50 from **Outdoor Scenes** and 50 from **Cityscapes**) and invite 20 subjects with image processing expertise to rank the visual qualities of images inpainted by the five inpainting methods (GatedConv, PEN-Net, EdgeConnect, SGE-Net,

Table 1: Objective quality comparison of five methods in terms of PSNR, SSIM, and FID on **Outdoor Scenes** and **Cityscapes** (↑: Higher is better; ↓: Lower is better).

	Outdoor Scenes						Cityscapes					
	centering holes			irregular holes			centering holes			irregular holes		
	PSNR↑	SSIM↑	FID↓	PSNR↑	SSIM↑	FID↓	PSNR↑	SSIM↑	FID↓	PSNR↑	SSIM↑	FID↓
GatedConv	19.06	0.73	42.34	18.47	0.74	44.15	21.13	0.74	20.03	17.13	0.67	43.14
PEN-Net	18.58	0.75	44.12	17.56	0.69	48.95	20.48	0.72	22.34	16.37	0.66	47.87
EdgeConnect	19.32	0.76	41.25	19.12	0.74	42.27	21.71	0.76	19.87	17.63	0.72	39.04
SGE-Net	20.53	**0.81**	40.67	19.46	0.76	39.14	23.41	**0.85**	18.67	17.78	0.74	41.45
Ours	**21.18**	**0.81**	**38.15**	**20.31**	**0.80**	**36.74**	**23.89**	0.84	**18.14**	**17.86**	**0.76**	**38.18**

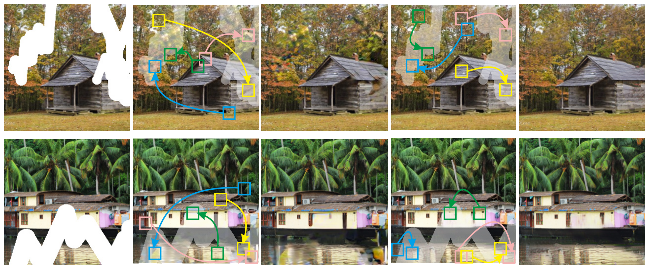


Figure 5: Comparisons between different attention modules. From left to right: input, the most matched patches in existing attention module, result from existing attention module, the most matched patches in SWAP module, and result from SWAP module. The arrows in columns 2 and 4 indicate the matched patch from known region to the missing region.

and our method). They are not informed of any mask information. For each test image, its five inpainting results are presented in a random order, and each subject is asked to rank the five methods from the best to the worst. The result shows that our method receives 51.8 % favorite votes (*i.e.*, the top-1 in 1,036 out of 2,000 comparisons), surpassing 21.3 % with SEG-Net, 13.5 % with EdgeConnect, 7.8 % with GatedConv, and 5.6 % with PEN-Net. Note, the higher percentage of favorite votes, the better subjective evaluation. Hence, our method outperforms the other methods.

4.5. Ablation Studies

4.5.1 Effectiveness of SWAP

We verify the effectiveness of SWAP by comparing it with the contextual attention module from GatedConv [43]. To show the difference, we highlight the location of the best-match patch for a patch in the missing area. As shown in Figure 5, since the existing attention module refers to the whole known region without any semantic guide, it usually matches wrong texture patches to the missing area, leading to ambiguous textures. In contrast, benefiting from SWAP, our method matches the patches within the same semantic class, which effectively improves the fidelity of matched reference textures so as to generate more realistic textures.

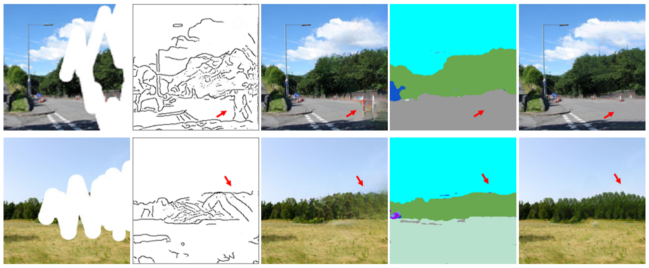


Figure 6: Visual quality comparisons between EdgeConnect and our method. From left to right: input, reconstructed edges and images by EdgeConnect, reconstructed segmentation maps and images by our method. The red arrows highlight the unrealistic regions generated from Edgeconnect compared with ours.

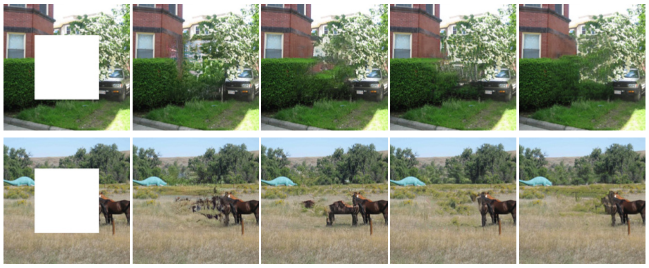


Figure 7: Visual quality comparisons on four variants to show the effectiveness of SWAP and Coherence Loss. From left to right: input, results of Ours (Base), Ours (Att), Ours (SWAP), and Ours (Full).

4.5.2 Edge *vs.* Semantic Segmentation

Our work assumes that semantic segmentation labels offer tighter clues to the textures than edges. To validate it, we compare the reconstructed structures of EdgeConnect and our model in Figure 6. We find that the edges of different objects may be mixed up in the edge maps, making EdgeConnect fill in incorrect texture details for some missing areas. In contrast, the inferred semantic segmentation labels from our model help well delineate the layout of images, and the semantics can guide the filling of the textures, result in more photo-realistic results.

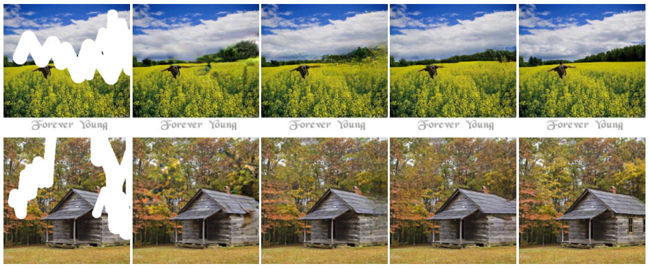


Figure 8: Visual quality comparisons on image samples from **Places2**. From left to right: input, results of GatedConv, EdgeConnect, and Our method, ground-truth.

Table 2: Statistical comparison on semantic segmentation accuracy between the semantic-guided inpainting methods, namely SPG-Net [28], SGE-Net[15] and our method on **Outdoor Scenes** and **Cityscapes**.

Outdoor Scenes		Cityscapes	
Methods	mIoU%	Methods	mIoU%
SPG-Net	0.51	SPG-Net	0.39
SGE-Net	0.68	SGE-Net	0.53
Ours	**0.71**	**Ours**	**0.57**

Table 3: Comparisons on the performance gains with SWAP and Coherence Loss (Co-Loss) in terms of three metrics.

	SWAP	Co-Loss	PSNR↑	SSIM↑	FID↓
Ours (Base)	✗	✗	17.43	0.65	57.31
Ours (Att)	✗	✗	19.77	0.76	43.54
Ours (SWAP)	✓	✗	20.58	0.79	39.46
Ours (Full)	✓	✓	**21.18**	**0.81**	**38.15**

4.5.3 Comparison of Segmentation Accuracy

In order to further validate the coherence priors between semantics and the textures, We also conduct experiments to compare the generated segmentation maps from SPG-Net, SGE-Net, and our method. Due to the alternative optimization of the inpainting and the segmentation tasks, we can generate high-quality segmentation maps, which in turn improve the inpainting results. Table 2 shows that our method outperforms the SPG-Net and the SGE-Net in semantic segmentation. The once-forward process from SPG-Net is hard to generate reliable semantic labels for the large missing areas, while the SGE-Net does not explicitly exploit the interaction between segmentation and inpainting.

4.5.4 Performance Gains with SWAP and Coherence Losses

In our method, the two core components, SWAP and coherence loss, are devised to improve the inpainting performance. In order to investigate their effectiveness, we conduct an ablation study on four variants: a) Ours (Base), with only joint optimization of inpainting and segmentation in a multi-scale framework; b) Ours (Att), adopting the attention module [42] to measure the texture affinity across all semantics; c) Ours (SWAP), with SWAP; d) Ours (Full), with both SWAP and Coherence loss.

The visual and numeric comparisons on **Outdoor Scenes** are shown in Figure 7 and Table 3, respectively. In general, the inpainting performance increases with the number of added modules. Specifically, the joint framework helps learn a more accurate scene layout, and the contextual attention does a good job of generating detailed content. Our SWAP can identify more relevant textures thanks to the predicted semantics. Moreover, the coherence losses further improve the texture details of inpainted regions.

4.5.5 Additional Results on Places2

We also conduct performance evaluation on the **Places2** dataset [48] without semantic annotation, which was used in the assessment by both GatedConv and EdgeConnect. We use our model trained on **Outdoor Scenes** to complete the images with similar scenes in **Places2**. The subjective results in Figure 8 show that our model is still able to generate proper semantic structures and textures, owing to the supervision of the coherence loss, which provides better prior knowledge about the scenes.

5. Conclusion

We proposed a novel joint optimization framework of semantic segmentation and image inpainting to exploit the coherence priors that existed between semantics and textures for solving the complex holes inpainting problem. To address the irrelevant texture filling, we proposed a semantic-wise attention propagation module to optimize the predicted textures from the same semantic region and two coherence losses to constrain the consistency of the semantic and texture in the same image. Experimental results demonstrate that our method can effectively generate promising semantic structures and texture details.

Acknowledgements This work was supported by National Key R&D Project (1502 211100026), Hubei Province Technological Innovation Major Project (2018CFA024), National Natural Science Foundation of China (61671336), and Natural Science Foundation of Jiangsu Province (BK20180234).

References

[1] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B. Goldman. PatchMatch: a randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.*, 28(3):24, 2009. 1

[2] Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. Image inpainting. In *Proc. ACM SIGGRAPH*, pages 417–424, 2000. 1

[3] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018. 5

[4] Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In *ICCV*, pages 1520–1529, 2017. 2, 3

[5] Antonio Criminisi, Patrick Pérez, and Kentaro Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE Trans. Image Process.*, 13(9):1200–1212, 2004. 1

[6] Alexey Dosovitskiy and Thomas Brox. Generating images with perceptual similarity metrics based on deep networks. In *NeurIPS*, 2016. 2

[7] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In *ICLR*, 2019. 2

[8] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *NeurIPS*, 2017. 6

[9] Seunghoon Hong, Xinchen Yan, Thomas E. Huang, and Honglak Lee. Learning hierarchical semantic image manipulation through structured representations. In *NeurIPS*, pages 2713–2723, 2018. 3

[10] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, pages 5967–5976, 2017. 2

[11] Christoph Kamann and Carsten Rother. Benchmarking the robustness of semantic segmentation models. In *CVPR*, pages 8825–8835. IEEE, 2020. 2

[12] Jingyuan Li, Fengxiang He, Lefei Zhang, Bo Du, and Dacheng Tao. Progressive reconstruction of visual structure for image inpainting. In *ICCV*, pages 5961–5970. IEEE, 2019. 1

[13] Jingyuan Li, Ning Wang, Lefei Zhang, Bo Du, and Dacheng Tao. Recurrent feature reasoning for image inpainting. In *CVPR*, pages 7757–7765, 2020. 2

[14] Liang Liao, Ruimin Hu, Jing Xiao, and Zhongyuan Wang. Edge-aware context encoder for image inpainting. In *ICASSP*, 2018. 2

[15] 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. 1, 2, 3, 6, 8

[16] Liang Liao, Jing Xiao, Zheng Wang, Chia-Wen Lin, and Shin'ichi Satoh. Uncertainty-aware semantic guidance and estimation for image inpainting. *IEEE J. Sel. Top. Signal Process.*, 15(2):310–323, 2021. 1

[17] Ding Liu, Bihan Wen, Jianbo Jiao, Xianming Liu, Zhangyang Wang, and Thomas S. Huang. Connecting image denoising and high-level vision tasks via deep learning. *IEEE Trans. Image Process.*, 29:3695–3706, 2020. 2, 3

[18] 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. 2

[19] Hongyu Liu, Bin Jiang, Yi Xiao, and Chao Yang. Coherent semantic attention for image inpainting. *arXiv preprint arXiv:1905.12384*, 2019. 2

[20] Zhuoqi Ma, Jie Li, Nannan Wang, and Xinbo Gao. Semantic-related image style transfer with dual-consistency loss. *Neurocomputing*, 406:135–149, 2020. 3

[21] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Qureshi, and Mehran Ebrahimi. EdgeConnect: Generative image inpainting with adversarial edge learning. *arXiv preprint arXiv:1901.00212*, 2019. 1, 2, 6

[22] Evangelos Ntavelis, Andrés Romero, Iason Kastanis, Luc Van Gool, and Radu Timofte. SESAME: semantic editing of scenes by adding, manipulating or erasing objects. In *ECCV*, pages 394–411. Springer, 2020. 3

[23] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *CVPR*, 2019. 3, 6

[24] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016. 1, 2

[25] Wenqi Ren, Jingang Zhang, Xiangyu Xu, Lin Ma, Xiaochun Cao, Gaofeng Meng, and Wei Liu. Deep video dehazing with semantic segmentation. *IEEE Trans. Image Process.*, 28(4):1895–1908, 2019. 3

[26] 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

[27] 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. 4

[28] Yuhang Song, Chao Yang, Yeji Shen, Peng Wang, Qin Huang, and C-C Jay Kuo. SPG-Net: Segmentation prediction and guidance network for image inpainting. *arXiv preprint arXiv:1805.03356*, 2018. 2, 8

[29] Hao Tang, Dan Xu, Yan Yan, Philip H. S. Torr, and Nicu Sebe. Local class-specific and global image-level generative adversarial networks for semantic-guided scene generation. In *CVPR*, pages 7867–7876, 2020. 3

[30] Matteo Tomei, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. Art2real: Unfolding the reality of artworks via semantically-aware image-to-image translation. In *CVPR*, pages 5849–5859, 2019. 3

[31] Chaoyue Wang, Chang Xu, Chaohui Wang, and Dacheng Tao. Perceptual adversarial networks for image-to-image transformation. *IEEE Trans. Image Process.*, 27(8):4066–4079, 2018. 2

[32] 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*, pages 8798–8807, 2018. 2, 3, 5

[33] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *CVPR*, 2018. 3, 5

[34] Yi Wang, Xin Tao, Xiaojuan Qi, Xiaoyong Shen, and Jiaya Jia. Image inpainting via generative multi-column convolutional neural networks. In *NeurIPS*, pages 329–338, 2018. 2

[35] Jing Xiao, Ruimin Hu, Liang Liao, Yu Chen, Zhongyuan Wang, and Zixiang Xiong. Knowledge-based coding of objects for multisource surveillance video data. *IEEE Trans. Multim.*, 18(9):1691–1706, 2016. 2

[36] Jing Xiao, Liang Liao, Qiegen Liu, and Ruimin Hu. CISI-Net: Explicit latent content inference and imitated style rendering for image inpainting. In *AAAI*, 2019. 3

[37] 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*, pages 8857–8866, 2019. 2, 4

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

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

[40] 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. 2

[41] Jie Yang, Zhiquan Qi, and Yong Shi. Learning to incorporate structure knowledge for image inpainting. In *AAAI*, pages 12605–12612, 2020. 2

[42] 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, 8

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

[44] Tao Yu, Zongyu Guo, Xin Jin, Shilin Wu, Zhibo Chen, Weiping Li, Zhizheng Zhang, and Sen Liu. Region normalization for image inpainting. In *AAAI*, pages 12733–12740, 2020. 1

[45] Yanhong Zeng, Jianlong Fu, Hongyang Chao, and Baining Guo. Learning pyramid-context encoder network for high-quality image inpainting. In *CVPR*, 2019. 2, 6

[46] 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*, pages 1–17, 2020. 1

[47] S. Zhang, R. He, Z. Sun, and T. Tan. Demeshnet: Blind face inpainting for deep meshface verification. *IEEE Trans. Inf. Forensics Security*, 13(3):637–647, 2018. 1, 2

[48] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(6):1452–1464, 2017. 5, 8