Generating Diverse Structure for Image Inpainting With Hierarchical VQ-VAE

Jialun Peng[1] Dong Liu[1*] Songcen Xu[2] Houqiang Li[1]
[1] University of Science and Technology of China [2] Noah's Ark Lab, Huawei Technologies Co., Ltd.
pjl@mail.ustc.edu.cn, {dongeliu, lihq}@ustc.edu.cn, xusongcen@huawei.com

Abstract

Given an incomplete image without additional constraint, image inpainting natively allows for multiple solutions as long as they appear plausible. Recently, multiple-solution inpainting methods have been proposed and shown the potential of generating diverse results. However, these methods have difficulty in ensuring the quality of each solution, e.g. they produce distorted structure and/or blurry texture. We propose a two-stage model for diverse inpainting, where the first stage generates multiple coarse results each of which has a different structure, and the second stage refines each coarse result separately by augmenting texture. The proposed model is inspired by the hierarchical vector quantized variational auto-encoder (VQ-VAE), whose hierarchical architecture disentangles structural and textural information. In addition, the vector quantization in VQ-VAE enables autoregressive modeling of the discrete distribution over the structural information. Sampling from the distribution can easily generate diverse and high-quality structures, making up the first stage of our model. In the second stage, we propose a structural attention module inside the texture generation network, where the module utilizes the structural information to capture distant correlations. We further reuse the VQ-VAE to calculate two feature losses, which help improve structure coherence and texture realism, respectively. Experimental results on CelebA-HQ, Places2, and ImageNet datasets show that our method not only enhances the diversity of the inpainting solutions but also improves the visual quality of the generated multiple images. Code and models are available at: https://github.com/USTC-JialunPeng/Diverse-Structure-Inpainting.

1. Introduction

Image inpainting refers to the task of filling in the missing region of an incomplete image so as to produce a complete and visually plausible image. Inpainting benefits a series of applications including object removal, photo restoration, and transmission error concealment. As an ill-posed problem, inpainting raises a great challenge especially when the missing region is large and contains complex content. As such, inpainting has attracted much research attention.

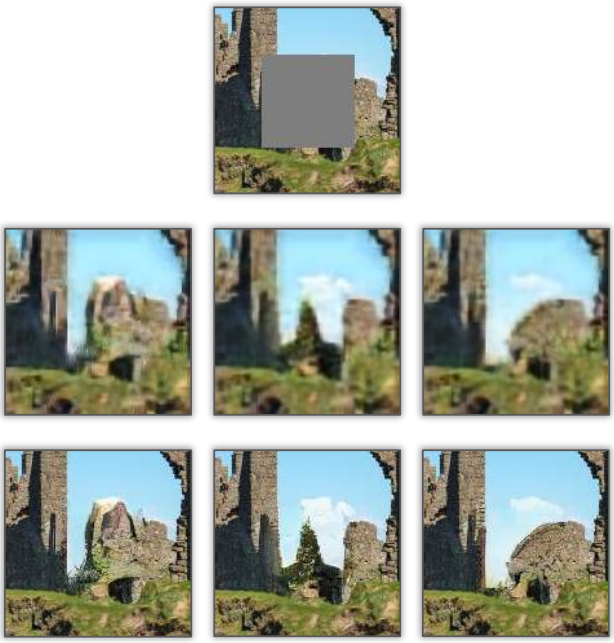

Figure 1. (Top) Input incomplete image, where the missing region is depicted in gray. (Middle) Visualization of the generated diverse structures. (Bottom) Output images of our method.

Recently, a series of deep learning-based methods are proposed for inpainting [10, 19]. They usually employ encoder-decoder architectures and train the networks with the combinations of reconstruction and adversarial losses. To enhance the visual quality of the results, a number of studies [23, 26, 32, 34, 36] adopt the contextual attention mechanisms to use the available content for generating the missing content. Also, several studies [13, 33, 35] propose modified convolutions in replacement of normal convolutions to reduce the artifacts.

The aforementioned methods all learn a deterministic mapping from an incomplete image to a complete im-

*This work was supported by the Natural Science Foundation of China under Grants 62036005 and 62022075. *(Corresponding author: Dong Liu.)*

age. However, in practice, the solution to inpainting is not unique. Without additional constraint, multiple inpainting results are equally/similarly plausible for an incomplete image, especially when the missing region is large and contains complex content (*e.g*. Figure 1). Moreover, for typical applications, providing multiple inpainting results may enable the user to select from them according to his/her own preference. It then motivates the design for multiple-solution inpainting.

In contrast to single-solution methods, multiple-solution inpainting shall build a probabilistic model of the missing content conditioned on the available content. Several recent studies [38, 39] employ variational auto-encoder (VAE) architectures and train the networks with the combinations of Kullback-Leibler (KL) divergences and adversarial losses. VAE-based methods [38,39] assume a Gaussian distribution over continuous latent variables. Sampling from the Gaussian distribution presents diverse latent features and leads to diverse inpainted images. Although these methods can generate multiple solutions, some of their solutions are of low quality due to distorted structures and/or blurry textures. It may be attributed to the limitation of the parametric (*e.g*. Gaussian) distribution when we try to model the complex natural image content. In addition, recent studies more and more demonstrate the importance of structural information, *e.g*. segmentation maps [12, 27], edges [17, 30], and smooth images [14, 21], for guiding image inpainting. Such structural information is yet to be incorporated into multiple-solution inpainting. Thus, VAE-based methods [38,39] tend to produce multiple results with limited structural diversity, which is called posterior collapse in [29].

In this paper we try to address the limitations of the existing multiple-solution inpainting methods. First, instead of parametric distribution modeling of continuous variables, we resort to autoregressive modeling of discrete variables. Second, we want to generate multiple structures in an explicit fashion, and then base the inpainting upon the generated structure. We find that the hierarchical vector quantized VAE (VQ-VAE) [20] is suitable for our study[1]. First, there is a vector quantization step in VQ-VAE making the latent variables to be all discrete; as noted in [29], these discrete latent variables allow the usage of powerful decoders to avoid the posterior collapse. Second, the hierarchical layout encourages the split of the image information into global and local parts; with proper design, it may disentangle structural features from textural features of an image.

Based on the hierarchical VQ-VAE, we propose a two-stage model for multiple-solution inpainting. The first stage is known as diverse structure generator, where sampling from a conditional autoregressive distribution produces multiple sets of structural features. The second stage is known as texture generator, where an encoder-decoder architecture is used to produce a complete image based on the guidance of a set of structural features. Note that each set of the generated structural features leads to a complete image (see Figure 1).

The main contributions we have made in this paper can be summarized as follows:

- We propose a multiple-solution image inpainting method based on hierarchical VQ-VAE. The method has two distinctions from previous multiple-solution methods: first, the model learns an autoregressive distribution over discrete latent variables; second, the model splits structural and textural features.
- We propose to learn a conditional autoregressive network for the distribution over structural features. The network manages to generate reasonable structures with high diversity.
- For texture generation we propose a structural attention module to capture distant correlations of structural features. We also propose two new feature losses to improve structure coherence and texture realism.
- Extensive experiments on three benchmark datasets including CelebA-HQ, Places2, and ImageNet demonstrate the superiority of our proposed method in both quality and diversity.

2. Related Work

2.1. Image Inpainting

Traditional image inpainting methods such as diffusion-based methods [1, 4] and patch-based methods [2, 5, 7, 8] borrow image-level patches from source images to fill in the missing regions. They are unable to generate unique content not found in the source images. Furthermore, these methods often generate unreasonable results without considering high-level semantics of the images.

Recently, learning-based methods which use deep convolutional networks are proposed to semantically predict the missing regions. Pathak *et al*. [19] first apply adversarial learning to the image inpainting. Iizuka *et al*. [10] introduce an extra discriminator to enforce the local consistency. Yan *et al*. [32] and Yu *et al*. [34] propose patch-swap and contextual attention to make use of distant feature patches for the higher inpainting quality. Liu *et al*. [13] and Yu *et al*. [35] introduce partial convolutions and gated convolutions to reduce visual artifacts caused by normal convolutions. In order to generate reasonable structures and realistic textures, Nazeri *et al*. [17] and Xu *et al*. [31] use edge maps as structural information to guide image inpainting. Ren *et al*. [21] propose to use edge-preserved smooth images instead of edge maps. Liu *et al*. [14] propose feature equalizations to improve the consistency between the structure and the texture. However, these learning-based meth-

[1]In this paper we use the basic model in [20], which is called VQ-VAE-2. Note that our method can use other hierarchical VQ-VAE models as well.

ods only generate one optimal result for each incomplete input. They focus on reconstructing ground truth rather than creating plausible results.

To obtain multiple inpainting solutions, Zheng *et al*. [39] propose a VAE-based model with two parallel paths, which trades off between reconstructing ground truth and maintaining the diversity of the inpainting results. Zhao *et al*. [38] propose a similar VAE-based model which uses instance images to improve the diversity. However, these methods do not effectively separate the structural and textural information, they often produce distorted structures and/or blurry textures.

2.2. VQ-VAE and Autoregressive Networks

The vector quantized variational auto-encoder (VQ-VAE) [29] is a discrete latent VAE model which relies on vector quantization layers to model discrete latent variables. The discrete latent variables allow a powerful autoregressive network such as PixelCNN [6, 18, 25, 28] to model latents without worrying about the posterior collapse problem [29]. Razavi *et al*. [20] propose a hierarchical VQ-VAE which uses a hierarchy of discrete latent variables to separate the structural and textural information. Then they use two PixelCNNs to model structural and textural information, respectively. However, the PixelCNNs are conditioned on the class label for image generation, while there is no class label in the image inpainting task. Besides, the generated textures of the PixelCNNs lack fine-grained details due to the lossy nature of VQ-VAE. It relieves the generation model from modeling negligible information, but it hinders the inpainting model from generating realistic textures consistent with the known regions. The PixelCNNs are thus not practical for image inpainting.

3. Method

As shown in Figure 2, the pipeline of our method consists of three parts: hierarchical VQ-VAE E_{vq}-D_{vq}, diverse structure generator G_s and texture generator G_t. The hierarchical encoder E_{vq} disentangles discrete structural features and discrete textural features of the ground truth $\mathbf{I}_{\text{gt}}$ and the decoder D_{vq} outputs the reconstructed image $\mathbf{I}_{\text{r}}$. The diverse structure generator G_s produces diverse discrete structural features given an input incomplete image $\mathbf{I}_{\text{in}}$. The texture generator G_t synthesizes the image texture given the discrete structural features and outputs the completion result $\mathbf{I}_{\text{comp}}$. We also use the pre-trained E_{vq} as an auxiliary evaluator to define two novel feature losses for better visual quality of the completion result.

3.1. Hierarchical VQ-VAE

In order to disentangle structural and textural information, we pre-train a hierarchical VQ-VAE E_{vq}-D_{vq} following [20]. The hierarchical encoder E_{vq} maps ground truth $\mathbf{I}_{\text{gt}}$ onto structural features $\mathbf{s}_{\text{gt}}$ and textural features $\mathbf{t}_{\text{gt}}$. The processing of E_{vq} can be written as $(\mathbf{s}_{\text{gt}}, \mathbf{t}_{\text{gt}}) = E_{vq}(\mathbf{I}_{\text{gt}})$. These features are then quantized to discrete features by two vector quantization layers. Each vector quantization layer has $K = 512$ prototype vectors in its codebook and the vector dimensionality is $D = 64$. As such, each vector of features is replaced by the nearest prototype vector based on Euclidean distance. The processing of vector quantization can be written as $\bar{\mathbf{s}}_{\text{gt}} = VQ_s(\mathbf{s}_{\text{gt}})$ and $\bar{\mathbf{t}}_{\text{gt}} = VQ_t(\mathbf{t}_{\text{gt}})$. Finally, the decoder D_{vq} reconstructs image from these two sets of discrete features. The processing of D_{vq} can be written as $\mathbf{I}_{\text{r}} = D_{vq}(\bar{\mathbf{s}}_{\text{gt}}, \bar{\mathbf{t}}_{\text{gt}})$.

The reconstruction loss of E_{vq}-D_{vq} is defined as:

$$\mathcal{L}_{\ell_2} = \|\mathbf{I}_{\text{r}} - \mathbf{I}_{\text{gt}}\|_2^2 \tag{1}$$

To back-propagate the gradient of the reconstruction loss through vector quantization, we use the straight-through gradient estimator [3]. The codebook prototype vectors are updated using the exponential moving average of the encoder output. As proposed in [29], we also use two commitment losses $\mathcal{L}_{sc}$ and $\mathcal{L}_{tc}$ to align the encoder output with the codebook prototype vectors for stable training. The commitment loss of structural features is defined as:

$$\mathcal{L}_{sc} = \|\mathbf{s}_{\text{gt}} - sg[\bar{\mathbf{s}}_{\text{gt}}]\|_2^2 \tag{2}$$

where sg denotes the stop-gradient operator [29]. The commitment loss of textural features (denoted as $\mathcal{L}_{tc}$) is similar to $\mathcal{L}_{sc}$. The total loss of E_{vq}-D_{vq} is defined as:

$$\mathcal{L}_{vq} = \alpha_{\ell_2}\mathcal{L}_{\ell_2} + \alpha_c(\mathcal{L}_{sc} + \mathcal{L}_{tc}) \tag{3}$$

where α_{ℓ_2} and α_c are loss weights.

For 256×256 images, the size of structural features is 32×32 and that of textural features is 64×64. We visualize their discrete representations using the decoder D_{vq}. The visualized results can be written as $D_{vq}(\bar{\mathbf{s}}_{\text{gt}}, \mathbf{0})$ and $D_{vq}(\mathbf{0}, \bar{\mathbf{t}}_{\text{gt}})$, where $\mathbf{0}$ is a zero tensor. As shown in Figure 2, the structural features model global information such as shapes and colors, and the textural features model local information such as details and textures.

3.2. Diverse Structure Generator

Previous multiple-solution inpainting methods [38, 39] often produce distorted structures and/or blurry textures, suggesting that these methods struggle to recover the structure and the texture simultaneously. Therefore, we first propose a diverse structure generator G_s which uses an autoregressive network to formulate a conditional distribution over the discrete structural features. Sampling from the distribution can produce diverse structural features.

Similar to the PixelCNN in [20], our autoregressive network consists of 20 residual gated convolution layers and

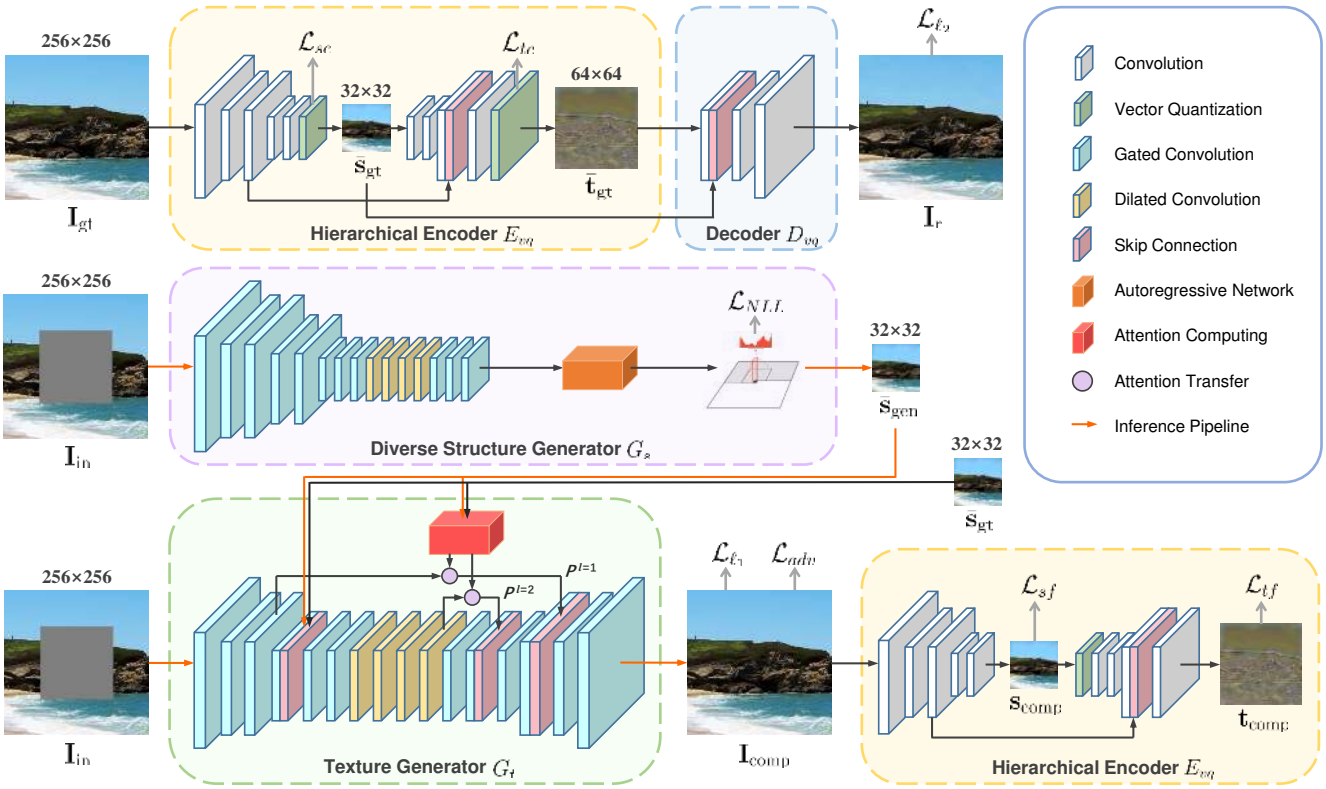

Figure 2. Overview of the proposed method. (Top) Hierarchical vector quantized variational auto-encoder (VQ-VAE) consists of hierarchical encoder E_{vq} and decoder D_{vq}. E_{vq} extracts discrete structural features $\bar{\mathbf{s}}_{\text{gt}}$ and discrete textural features $\bar{\mathbf{t}}_{\text{gt}}$. D_{vq} reconstructs the image from these two sets of discrete features. (Middle) Diverse structure generator G_s models the conditional distribution over the discrete structural features by using an autoregressive network, where $\bar{\mathbf{s}}_{\text{gt}}$ is used to calculate the loss $\mathcal{L}_{NLL}$. During inference, sampling from the distribution can generate multiple possible structural features $\bar{\mathbf{s}}_{\text{gen}}$. (Bottom) Texture generator G_t synthesizes the image texture given the discrete structural features ($\bar{\mathbf{s}}_{\text{gt}}$ in the training and $\bar{\mathbf{s}}_{\text{gen}}$ in the inference). The pre-trained E_{vq} is used as an auxiliary evaluator to improve image quality, where $\bar{\mathbf{s}}_{\text{gt}}$ and $\bar{\mathbf{t}}_{\text{gt}}$ are used to calculate the losses $\mathcal{L}_{sf}$ and $\mathcal{L}_{tf}$. During training, the hierarchical VQ-VAE is firstly trained, and then G_s and G_t are trained individually. During inference, only G_s and G_t are used.

4 casual multi-headed attention layers. Since the PixelCNN in [20] is conditioned on the class label for the image generation task, we make two modifications to make it practical for the image inpainting task. First, we stack gated convolution layers to map the input incomplete image $\mathbf{I}_{\text{in}}$ and its binary mask $\mathbf{M}$ to a condition. The condition is injected into each residual gated convolution layer of the autoregressive network. Second, we use a light-weight autoregressive network by reducing both the hidden units and the residual units to 128 for efficiency.

During training, G_s utilizes the input incomplete image as the condition and models the conditional distribution over $\bar{\mathbf{s}}_{\text{gt}}$. This distribution can be written as $p_\theta(\bar{\mathbf{s}}_{\text{gt}}|\mathbf{I}_{\text{in}}, \mathbf{M})$, where θ denotes network parameters of G_s. The training loss of G_s is defined as the negative log likelihood of $\bar{\mathbf{s}}_{\text{gt}}$:

$$\mathcal{L}_{NLL} = -\mathbb{E}_{\mathbf{I}_{\text{gt}} \sim p_{data}}[\log p_\theta(\bar{\mathbf{s}}_{\text{gt}}|\mathbf{I}_{\text{in}}, \mathbf{M})] \quad (4)$$

where p_{data} denotes the distribution of training dataset. During inference, G_s utilizes the input incomplete image as condition and outputs a conditional distribution for generating structural features. This distribution can be written as $p_\theta(\bar{\mathbf{s}}_{\text{gen}}|\mathbf{I}_{\text{in}}, \mathbf{M})$. Sampling from $p_\theta(\bar{\mathbf{s}}_{\text{gen}}|\mathbf{I}_{\text{in}}, \mathbf{M})$ sequentially can generate diverse discrete structural features $\bar{\mathbf{s}}_{\text{gen}}$.

Due to the low-resolution of structural features, our diverse structure generator can better capture global information. It thus helps generate reasonable global structures. In addition, the training objective of our diverse structure generator is to maximize likelihood of all samples in the training set without any additional loss. Thus, the generated structures do not suffer from GAN's known shortcomings such as mode collapse and lack of diversity.

3.3. Texture Generator

Network Architecture. After obtaining the generated structural features $\bar{\mathbf{s}}_{\text{gen}}$, our texture generator G_t synthesizes the image texture based on the guidance of $\bar{\mathbf{s}}_{\text{gen}}$. The network architecture of G_t is similar to the refine network in [33]. As shown in Figure 2, the network architecture consists of gated convolutions and dilated convolutions. Unlike

existing inpainting methods, our texture generator G_t utilizes the given structural features as guidance. The structural features are not only input to the first few layers of G_t, but also input to our structural attention module. The proposed structural attention module borrows distant information based on the correlations of the structural features. It thus ensures that the synthesized texture is consistent with the generated structure.

Let $\mathbf{I}_{\text{out}}$ denote the output of G_t. The final completion result $\mathbf{I}_{\text{comp}}$ is the output $\mathbf{I}_{\text{out}}$ with the non-masked pixels directly set to ground truth. During training, G_t takes the ground truth structural features $\bar{\mathbf{s}}_{\text{gt}}$ as input so that $\mathbf{I}_{\text{comp}}$ is the reconstruction of ground truth. During inference, G_t takes the generated structural features $\bar{\mathbf{s}}_{\text{gen}}$ as input so that $\mathbf{I}_{\text{comp}}$ is the inpainting result.

Structural Attention Module. Attention modules are widely used in the existing image inpainting methods. They generally calculate the attention scores on a low-resolution intermediate feature map of the network. However, due to the lack of direct supervision on the attention scores, the learned attention is insufficiently reliable [41]. These attention modules may refer to unsuitable features, resulting in poor inpainting quality. To address this problem, we propose a structural attention module which directly calculates the attention scores on the structural features. Intuitively, regions with similar structures should have similar textures. Calculating the attention scores on the structural features can model accurate long-range correlations of structural information, thereby improving the consistency between the synthesized texture and the generated structure.

In addition, the attention modules in the existing image inpainting methods distinguish the foreground features and the background features using the down-sampled mask. This hand-crafted design may incorrectly divide the feature map and produce artifacts in the inpainting result. Therefore, our structural attention module calculates full attention scores on the structural features. Unlike the foreground-background cross attention that only models correlations between the foreground and the background, our full attention learns full correlations regardless of the feature division. It thus maintains the global consistency of the inpainting result. Moreover, our full attention does not increase the amount of calculation compared to the cross attention.

Like [33], our structural attention module consists of an attention computing step and an attention transfer step. The attention computing step extracts 3×3 patches from the input structural features. Then, the truncated distance similarity score [23] between the patches at (x, y) and (x', y') is defined as:

$$\tilde{d}_{(x,y),(x',y')} = \tanh(-(\frac{d_{(x,y),(x',y')} - m}{\sigma})) \tag{5}$$

where $d_{(x,y),(x',y')}$ is the Euclidean distance, m and σ are the mean value and the standard deviation of $d_{(x,y),(x',y')}$. The truncated distance similarity scores are applied by a scaled sotfmax layer to output full attention scores:

$$s^*_{(x,y),(x',y')} = \text{softmax}(\lambda_1 \tilde{d}_{(x,y),(x',y')}) \tag{6}$$

where λ_1 = 50. After obtaining the full attention scores from the structural features, the attention transfer step reconstructs lower-level feature maps (P^l) by using the full attention scores as weights:

$$q^l_{(x',y')} = \sum_{x,y} s^*_{(x,y),(x',y')} p^l_{(x,y)} \tag{7}$$

where $l \in (1, 2)$ is the layer number and $p^l_{(x,y)}$ is the patch of P^l, and $q^l_{(x',y')}$ is the patch of the reconstructed feature map. The size of patches varies according to the size of feature map like [33]. Finally, the reconstructed feature maps supplement the decoder of G_t via skip connections.

Training Losses. The total loss of G_t consists of a reconstruction loss, an adversarial loss, and two feature losses. The reconstruction loss of G_t is defined as:

$$\mathcal{L}_{\ell_1} = \|\mathbf{I}_{\text{out}} - \mathbf{I}_{\text{gt}}\|_1 \tag{8}$$

We use SN-PatchGAN [35] as our discriminator D_t. The hinge version of the adversarial loss for D_t is defined as:

$$\begin{aligned}\mathcal{L}_d = &\mathbb{E}_{\mathbf{I}_{\text{gt}} \sim p_{data}}[\text{ReLU}(1 - D_t(\mathbf{I}_{\text{gt}}))] \\ &+ \mathbb{E}_{\mathbf{I}_{\text{comp}} \sim p_z}[\text{ReLU}(1 + D_t(\mathbf{I}_{\text{comp}}))]\end{aligned} \tag{9}$$

where p_z denotes the distribution of the inpainting results. The adversarial loss for G_t is defined as:

$$\mathcal{L}_{adv} = -\mathbb{E}_{\mathbf{I}_{\text{comp}} \sim p_z}[D_t(\mathbf{I}_{\text{comp}})] \tag{10}$$

Some inpainting methods such as [13, 14] use the pre-trained VGG-16 as an auxiliary evaluator to improve the perceptual quality of the results. They define a perceptual loss and a style loss based on the VGG features to train the generator. Inspired by these feature losses, we propose two novel feature losses by reusing our pre-trained hierarchical encoder E_{vq} as an auxiliary evaluator. As shown in Figure 2, E_{vq} maps $\mathbf{I}_{\text{comp}}$ onto structural features $\mathbf{s}_{\text{comp}}$ and textural features $\mathbf{t}_{\text{comp}}$. The structural feature loss of G_t is defined as the multi-class cross-entropy between $\mathbf{s}_{\text{comp}}$ and $\bar{\mathbf{s}}_{\text{gt}}$:

$$\mathcal{L}_{sf} = -\sum_{i,j} I_{ij} \log(\text{softmax}(\lambda_2 \tilde{d}_{ij})) \tag{11}$$

Here, we set λ_2 = 10. $\tilde{d}_{ij}$ denotes the truncated distance similarity score between the i^{th} feature vector of $\mathbf{s}_{\text{comp}}$ and the j^{th} prototype vector of the structural codebook. I_{ij} is an indicator of the prototype vector class. I_{ij} = 1 when the i^{th} feature vector of $\bar{\mathbf{s}}_{\text{gt}}$ belongs to the j^{th} class of the structural codebook, otherwise I_{ij} = 0. The textural feature loss (denoted as $\mathcal{L}_{tf}$) is similar to $\mathcal{L}_{sf}$. The total loss of G_t is defined as:

$$\mathcal{L}_{tg} = \alpha_{\ell 1}\mathcal{L}_{\ell 1} + \alpha_{adv}\mathcal{L}_{adv} + \alpha_f(\mathcal{L}_{sf} + \mathcal{L}_{tf}) \tag{12}$$

where $\alpha_{\ell 1}$, α_{adv}, and α_f are loss weights.

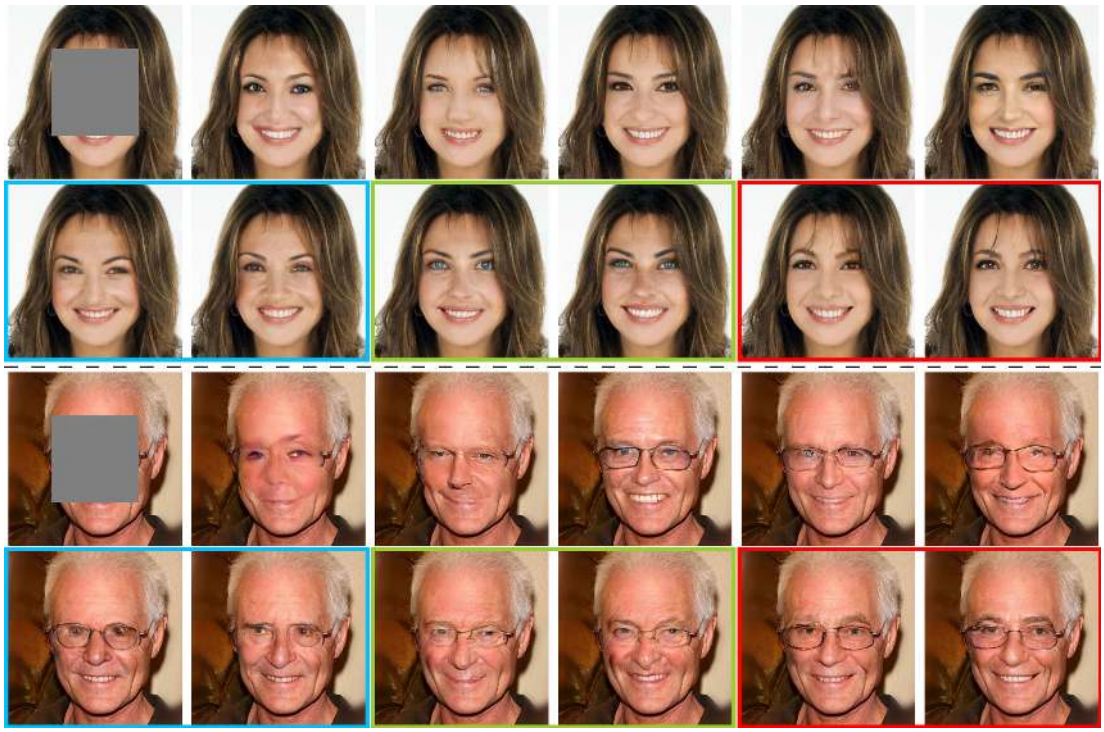

Figure 3. Qualitative comparison results on two test images of CelebA-HQ. For each group, from top to bottom, from left to right, the pictures are: incomplete image, results of CA [34], GC [35], CSA [15], SF [21], FE [14], results of PIC [39] (with blue box), results of UCTGAN [38] (with green box), and results of our method (with red box).

Figure 4. Qualitative comparison results on two test images of Places2. For each group, from top to bottom, from left to right, the pictures are: incomplete image, results of CA [34], GC [35], CSA [15], SF [21], FE [14], results of PIC [39] (with blue box), results of UCTGAN [38] (with green box), and results of our method (with red box).

Method		PSNR↑	SSIM↑	IS↑	MIS↑	FID↓
Single-Solution	CA [34]	23.65	0.8525	3.206	0.0207	16.64
	GC [35]	25.23	0.8713	3.384	0.0235	12.24
	CSA [15]	**25.26**	**0.8840**	3.408	0.0199	11.78
	SF [21]	25.05	0.8717	3.360	0.0229	10.59
	FE [14]	24.10	0.8632	3.357	0.0240	10.73
Multiple-Solution	PIC [39]	23.93	0.8567	3.357	0.0225	11.70
	UCTGAN [38]	24.39	0.8603	3.342	0.0237	11.74
	Ours	24.56	0.8675	**3.456**	**0.0245**	**9.784**

Table 1. Quantitative comparison of different methods on the CelebA-HQ test set. For multiple-solution methods, we sample 50 images for each incomplete image and report the average result. Note that PSNR and SSIM are full-reference metrics that compare the generated image with the ground truth, but IS, MIS, and FID are not. For each metric, the best score is highlighted in **bold**, and the best score within the other category is highlighted in red.

4. Experiments

4.1. Implementation Details

Our model is implemented in TensorFlow v1.12 and trained on two NVIDIA 2080 Ti GPUs. The batch size is 8. During optimization, the weights of different losses are set to $\alpha_{\ell_1} = \alpha_{\ell_2} = \alpha_{adv} = 1$, $\alpha_c = 0.25$, $\alpha_f = 0.1$. We use the Adam optimizer to train the three parts of our model. The learning rate of E_{vq}-D_{vq} is 10^{-4}. The learning rate of G_s follows the linear warm-up and square-root decay schedule used in [20]. The learning rate of G_t is 10^{-4} and $\beta_1 = 0.5$. We also use polyak exponential moving average (EMA) decay of 0.9997 when training E_{vq}-D_{vq} and G_s. Each part is trained for 1M iterations. During training, E_{vq}-D_{vq} is firstly trained, and then G_s and G_t are trained individually. During inference, only G_s and G_t are used.

4.2. Performance Evaluation

We evaluate our method on three datasets including CelebA-HQ [11], Places2 [40], and ImageNet [22]. We use the original training, testing, and validation splits for these three datasets. For CelebA-HQ, training images are downsampled to 256×256 and data augmentation is adopted. For Places2 and ImageNet, training images are randomly cropped to 256×256. The missing regions of the incomplete images can be regular or irregular. We compare our method with state-of-the-art single-solution and multiple-solution inpainting methods. The single-solution methods among them are CA [34], GC [35], CSA [15], SF [21] and FE [14]. The multiple-solution methods among them are PIC [39] and UCTGAN [38]

Qualitative Comparisons. Figure 3 and Figure 4 show the qualitative comparison results of center-hole inpainting on CelebA-HQ and Places2, respectively. It is difficult for CA [34], GC [35] and CSA [15] to generate reasonable structures without structural information acts as prior knowledge. SF [21] and FE [14] use edge-preserved smooth images to guide structure generation. However, they struggle to synthesize fine-grained textures, which indicates that their structural information provides limited help to texture generation. PIC [39] and UCTGAN [38] show high diversity. But their results are of low quality, especially for the challenging Places2 test images. Compared to these methods, the results of our method have more reasonable structures and more realistic textures, *e.g.* fine-grained hair and eyebrows in Figure 3. In addition, the diversity of our method is enhanced, *e.g.* different eye colors in Figure 3 and varying window sizes in Figure 4. More qualitative results and analyses of artifacts are presented in the supplementary material.

Quantitative Comparisons. Following previous image inpainting methods, we use common evaluation metrics such as peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) to measure the similarity between the inpainting result and ground truth. However, these full-reference metrics are not suitable for the image inpainting task because there are multiple plausible solutions for an incomplete image. The image inpainting methods are supposed to focus on generating realistic results rather than merely approximating ground truth. Therefore, we also use Inception Score (IS) [24], Modified Inception Score (MIS) [38], and Fréchet Inception Distance (FID) [9] as perceptual quality metrics. These metrics are consistent with human judgment [9]. FID can also detect GAN's known shortcomings such as mode collapse and mode dropping [16].

Unlike previous multiple-solution methods [38, 39] that use discriminator to select samples for quantitative evaluation, we use all samples for fair comparison. The comparison is conducted on CelebA-HQ 1000 testing images with 128×128 center holes. As shown in Table 1, multiple-solution methods score relatively low on PSNR and SSIM because they generate highly diverse results instead of approximating ground truth. Still, our method outperforms PIC [39] and UCTGAN [38] on these two metrics. Furthermore, our method outperforms all the other methods in terms of IS, MIS and FID.

(a) (b) (c) (d) (e)

Figure 5. Results of the ablation study on the structural attention module. (a) Incomplete image. (b) Using cross attention on the learned features. (c) Using full attention on the learned features. (d) Using cross attention on the structural features. (e) Using full attention on the structural features. [Best viewed with zoom-in.]

(a) (b) (c) (d) (e) (f)

Figure 6. Results of the ablation study on the auxiliary losses. (a) Incomplete image. (b) Using no feature losses (but still using $\mathcal{L}_{\ell 1}$ and $\mathcal{L}_{adv}$). (c) Using $\mathcal{L}_{sf}$. (d) Using $\mathcal{L}_{tf}$. (e) Using both $\mathcal{L}_{sf}$ and $\mathcal{L}_{tf}$. (f) Using the perceptual and style losses as in [13, 14]. [Best viewed with zoom-in.]

Full	Structural	SSIM$^{\uparrow}$	IS$^{\uparrow}$	FID$^{\downarrow}$
		0.8606	3.402	11.55
✓		0.8645	3.436	11.26
	✓	0.8675	3.416	10.12
✓	✓	**0.8676**	**3.467**	**9.670**

Table 2. Quantitative comparison for the ablation study on the structural attention module. Refer to Section 4.3 and Figure 5 for description.

$\mathcal{L}_{sf}$	$\mathcal{L}_{tf}$	SSIM$^{\uparrow}$	IS$^{\uparrow}$	FID$^{\downarrow}$
		0.8581	3.383	11.02
✓		0.8589	3.367	11.16
	✓	0.8625	3.414	10.34
✓	✓	**0.8676**	**3.467**	**9.670**
$\mathcal{L}_{perceptual}$ + $\mathcal{L}_{style}$ [13, 14]		0.8638	3.388	9.672

Table 3. Quantitative comparison for the ablation study on the auxiliary losses. Refer to Section 4.3 and Figure 6 for description.

We also evaluate the diversity of our method using the LPIPS [37] metric. The average score is calculated between consecutive pairs of 50K results which are sampled from 1K incomplete images. Higher score indicates higher diversity. The reported scores of PIC [39] and UCTGAN [38] are 0.029 and 0.030, respectively. Our method achieves a comparable score of 0.029. Please refer to the supplementary material for some discussions about the diversity.

4.3. Ablation Study

We conduct an ablation study on CelebA-HQ to show the effect of different components of the texture generator G_t. Since the structure generator G_s can produce diverse structural features, we randomly sample a set of generated structural features. Then we use it across all the following experiments for fair comparison.

Effect of structural attention module. We compare the effect of different attention modules. The attention module in [33] calculates cross attention scores on the learned features, which often results in texture artifacts (see Figure 5(b)). Using full attention instead of cross attention maintains global consistency (see Figure 5(c)). Using the structural features instead of the learned features improves the consistency between structures and textures (see Figure 5(d)). Our structural attention module calculates full attention scores on the structural features, which can synthesize realistic textures, such as symmetric eyes and eyebrows (see Figure 5(e)). The quantitative results in Table 2 also demonstrate the benefits of our structural attention module.

Effect of our feature losses. We compare the effect of different auxiliary losses. Without any auxiliary loss, the perceptual quality of the inpainting result is not satisfactory (see Figure 6(b)). Using our structural feature loss $\mathcal{L}_{sf}$ can improve structure coherence, such as the shape of nose and mouth (see Figure 6(c)). Using our textural feature loss $\mathcal{L}_{tf}$ can improve texture realism, such as the luster of facial skin (see Figure 6(d)). Using both $\mathcal{L}_{sf}$ and $\mathcal{L}_{tf}$ can generate more natural images (see Figure 6(e)). Compared to our feature losses, perceptual and style losses used in [13, 14] may produce a distorted structure or an inconsistent texture (see Figure 6(f)). The quantitative results in Table 3 also demonstrate the benefits of our feature losses.

5. Conclusion

We have proposed a multiple-solution inpainting method for generating diverse and high-quality images using hierarchical VQ-VAE. Our method first formulates an autoregressive distribution to generate diverse structures, then synthesizes the image texture for each kind of structure. We propose a structural attention module to ensure that the synthesized texture is consistent with the generated structure. We further propose two feature losses to improve structure coherence and texture realism, respectively. Extensive qualitative and quantitative comparisons show the superiority of our method in both quality and diversity. We demonstrate that the structural information extracted by the hierarchical VQ-VAE is of great benefit for the inpainting task. As for future work, we plan to extend our method to other conditional image generation tasks including style transfer, image super-resolution, and guided editing.

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 Trans. Image Process.*, 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 Trans. Graph.*, 28(3):24–33, 2009. 2

[3] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013. 3

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

[5] Marcelo Bertalmio, Luminita Vese, Guillermo Sapiro, and Stanley Osher. Simultaneous structure and texture image inpainting. *IEEE Trans. Image Process.*, 12(8):882–889, 2003. 2

[6] Xi Chen, Nikhil Mishra, Mostafa Rohaninejad, and Pieter Abbeel. PixelSNAIL: An improved autoregressive generative model. In *ICML*, pages 864–872, 2018. 3

[7] 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. 2

[8] Iddo Drori, Daniel Cohen-Or, and Hezy Yeshurun. Fragment-based image completion. In *ACM SIGGRAPH*, pages 303–312. 2003. 2

[9] 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 *NIPS*, pages 6626–6637, 2017. 7

[10] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Globally and locally consistent image completion. *ACM Trans. Graph.*, 36(4):1–14, 2017. 1, 2

[11] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 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*, pages 1–17, 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*, pages 85–100, 2018. 1, 2, 5, 8

[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*, pages 725–741, 2020. 2, 5, 6, 7, 8

[15] Hongyu Liu, Bin Jiang, Yi Xiao, and Chao Yang. Coherent semantic attention for image inpainting. In *ICCV*, pages 4170–4179, 2019. 6, 7

[16] Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are GANs created equal? A large-scale study. In *NIPS*, pages 700–709, 2018. 7

[17] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Z. Qureshi, and Mehran Ebrahimi. EdgeConnect: Structure guided image inpainting using edge prediction. In *ICCV Workshops*, pages 1–10, 2019. 2

[18] Aaron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *ICML*, pages 1747–1756, 2016. 3

[19] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. In *CVPR*, pages 2536–2544, 2016. 1, 2

[20] Ali Razavi, Aaron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with VQ-VAE-2. In *NeurIPS*, pages 14866–14876, 2019. 2, 3, 4, 7

[21] Yurui Ren, Xiaoming Yu, Ruonan Zhang, Thomas H. Li, Shan Liu, and Ge Li. StructureFlow: Image inpainting via structure-aware appearance flow. In *ICCV*, pages 181–190, 2019. 2, 6, 7

[22] Olga Russakovsky, Jia Deng, Hao Su, et al. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115(3):211–252, 2015. 7

[23] Min-cheol Sagong, Yong-goo Shin, Seung-wook Kim, Seung Park, and Sung-jea Ko. PEPSI: Fast image inpainting with parallel decoding network. In *CVPR*, pages 11360–11368, 2019. 1, 5

[24] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. In *NIPS*, pages 2234–2242, 2016. 7

[25] Tim Salimans, Andrej Karpathy, Xi Chen, and Diederik P. Kingma. PixelCNN++: Improving the PixelCNN with discretized logistic mixture likelihood and other modifications. *arXiv preprint arXiv:1701.05517*, 2017. 3

[26] 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*, pages 3–19, 2018. 1

[27] 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. In *BMVC*, pages 97–110, 2018. 2

[28] Aaron van den Oord, Nal Kalchbrenner, Oriol Vinyals, Lasse Espeholt, Alex Graves, and Koray Kavukcuoglu. Conditional image generation with PixelCNN decoders. In *NIPS*, pages 4790–4798, 2016. 3

[29] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *NIPS*, pages 6306–6315, 2017. 2, 3

[30] Wei Xiong, Jiahui Yu, Zhe Lin, Jimei Yang, Xin Lu, Connelly Barnes, and Jiebo Luo. Foreground-aware image inpainting. In *CVPR*, pages 5840–5848, 2019. 2

[31] Shunxin Xu, Dong Liu, and Zhiwei Xiong. E2I: Generative inpainting from edge to image. *IEEE Trans. Circuit Syst. Video Technol.*, 2020. 2

[32] 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, 2

[33] Zili Yi, Qiang Tang, Shekoofeh Azizi, Daesik Jang, and Zhan Xu. Contextual residual aggregation for ultra high-resolution image inpainting. In *CVPR*, pages 7508–7517, 2020. 1, 4, 5, 8

[34] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S. Huang. Generative image inpainting with contextual attention. In *CVPR*, pages 5505–5514, 2018. 1, 2, 6, 7

[35] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S. Huang. Free-form image inpainting with gated convolution. In *ICCV*, pages 4471–4480, 2019. 1, 2, 5, 6, 7

[36] Yanhong Zeng, Jianlong Fu, Hongyang Chao, and Baining Guo. Learning pyramid-context encoder network for high-quality image inpainting. In *CVPR*, pages 1486–1494, 2019. 1

[37] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pages 586–595, 2018. 8

[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*, pages 5741–5750, 2020. 2, 3, 6, 7, 8

[39] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. Pluralistic image completion. In *CVPR*, pages 1438–1447, 2019. 2, 3, 6, 7, 8

[40] 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. 7

[41] Tong Zhou, Changxing Ding, Shaowen Lin, Xinchao Wang, and Dacheng Tao. Learning oracle attention for high-fidelity face completion. In *CVPR*, pages 7680–7689, 2020. 5