Progressive Contextual Aggregation Empowered by Pixel-wise Dense Detector for Image Inpainting

Jinwoo Kim, Woojae Kim, Heeseok Oh, and Sanghoon Lee, *Senior Member, IEEE*

***Abstract*—Image inpainting methods generate alternative content by harnessing similarities from neighboring pixels. However, as the invisible region becomes larger, completed pixels in a deeper hole have difficulty in inferring from the surrounding pixel signals, easily leading to visual artifacts. To help fill this void, we adopt an alternative progressive hole-filling scheme that hierarchically fills the corrupted region in the feature and image spaces. This technique allows us to utilize reliable contextual information of the surrounding pixels, even for large hole samples, and then gradually complete the details as the resolution increases. To more realistically represent the completed region, we devise a pixel-wise dense detector. By distinguishing each pixel as a masked region and passing the gradients to all resolutions, the generator further reinforces the potential quality of the synthesis. The completed images of different resolutions are then merged using a structure transfer module (STM) that preserves the global continuity. By comparing our solutions qualitatively and quantitatively with state-of-the-art methods, we conclude that our model exhibits a significantly improved visual quality, even in the case of large holes.**

***Index Terms*—Image inpainting, Image completion, Free-form inpainting, Object removal, adversarial learning**

I. Introduction

IMAGE inpainting is a technique of composing missing parts into an alternative content with visually plausible quality. Although this is one of the challenging tasks in computer vision owing to the inherent ambiguity of natural images, it is an essential functionality deployed in various image processing and graphic applications, for example, object removal, image restoration, manipulation, retargeting, compositing, image-based rendering [1]–[3], and computational photography [2], [3].

Existing inpainting methods can be categorized into two approaches. The first approach is called *"fill through copying"*, which attempts to explicitly borrow content or textures from the surroundings to fill in the missing region. According to the content propagation type, these methods can be divided into diffusion-based [4]–[7] and patch-based [8] algorithms. Such conventional methods often achieve a texture synthesis, although the main problem is related to an understanding of image semantics to capture a global structure. Owing to a decade of advances in convolutional neural networks (CNNs), the second approach attempts to learn the implicit distribution of the image in a data-driven manner, which we call *"fill through modeling"*. This approach usually utilizes an encoder-decoder pipeline, which assumes that the network should capture both high-level semantics and low-level details at the feature level [9]–[13]. However, as the invisible region becomes larger, this assumption loses its validity because of the weak interrelationship between neighboring pixels. To cope with this, advanced attempts have been made to employ an attention module aiming to transfer long-range coherency between the visible and invisible regions [10], [12]–[14]. Despite the slight improvement, the simple attention mechanism is insufficient to propagate semantics because this method requires roughly completed features that are not guaranteed in the large hole case. Therefore, most state-of-the-art methods adopt an alternative scheme that fills the hole from the boundary to the center recursively [15], [16]. Unfortunately, these recursive processes easily lead to structural discontinuities owing to the straightforward mapping process from the abstracted feature to the image domain (i.e., RGB space) iteratively.

Essentially, the painting process is first built up of globally sketched semantics and the entire details are then filled in while considering the global-local continuity and consistency [5], [17]. Similarly, the image inpainting process should simultaneously reflect both high-level semantics and fine low-level details. To reflect the inherent nature of painting in image inpainting works, we devise a novel framework called *progressive inpainting*. In particular, this framework is remarkable when the three proposed networks (i.e., progressive generator, pixel-wise dense detector, and merge block) cooperate with each other to boost the capacity of reconstruction. The *progressive restoring network* (PRN) fills in the hole by gradually enlarging the resolution, which is inspired by the recent success of the progressive growing technique in the image generation fields [18]–[20]. As the key insight, the model organizes the global semantics from the low-resolution image prior, and then restores the fine details across different multi-scale resolutions. Fig. 1 shows examples of our completed results. As can be seen, even a large hole is gradually filled with a pleasing quality.

Toward this entire success of the progressive hole-filling scheme, an intermediate completion at each step has to intensify the texture details to convey wealth information across different resolutions. Here, the discriminator usually benefits the generator in terms of synthesizing more natural images [21], [22]. Because the discriminator focuses on distinguishing an image as real or fake in a holistic manner, the adversarial loss penalizes the generator, which is biased toward the most discriminative part (i.e., a global context) rather than the newly generated region. Thus, the discriminator commonly misses the fine details of the local region. The problem is amplified when the discriminator has to learn in a non-stationary pattern, such as an irregular hole-filling task. In this case, the position and size of the hole dynamically vary during the training procedure, which makes the generator prone to

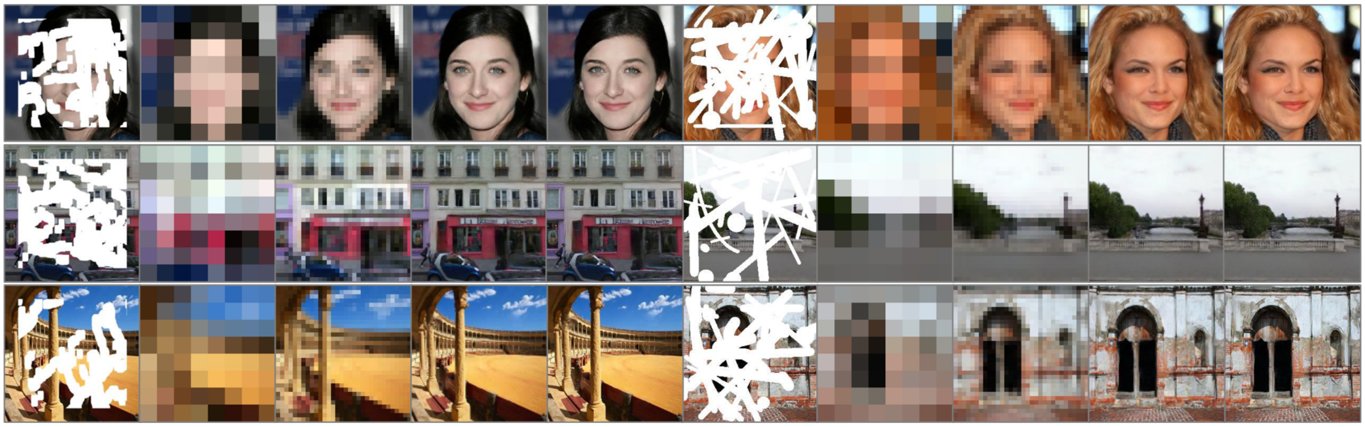


Fig. 1. Results of our method on a face, building, and natural scenery with various masks (missing region shown in white). For each group, the resolution of the generated images increases toward the right reaching the final output without post-processing (please zoom in to see the details).

forget the previous knowledge. Therefore, the discriminator is not incentivized to maintain a more powerful representation, learning both global and local image differences.

To resolve this issue, we propose a novel *pixel-wise dense detector* (PWDD) that efficiently penalizes the generator to enhance the quality of the newly generated region by acting as a discriminator. Here, the PWDD takes the role of both the global supervisor and the per-pixel classifier applicable to arbitrary holes. The global supervisor learns a representation that allows the regulation of the generator based on the most discriminative parts between the target and synthetic images, as in the standard adversarial loss setting. By contrast, the per-pixel classifier attempts to distinguish between the real and fake at the pixel level. Motivated by the U-Net based discriminator [22], we first cut out the synthetic pixels from the hole region and mix these pixels into the target image in the same region. Then, the binary mask maps are spatially combined with respect to the real and fake pixels. Empowered by a per-pixel feedback, we employ a pixel-wise segment map for consistency regularization, penalizing inconsistent predictions. These changes promote the PWDD to focus more on the semantic and structural changes by encouraging an invariant prediction to the perturbation of the mask type. Therefore, the overall visual quality can be dramatically improved.

To further ensure the final quality of the progressive inpainting, we should address the structural discontinuity arising from recursive feature mapping to the image domain. To this end, we propose a *merge block* that blends the completed multi-resolution images. Directly applying an in-network upscaling for the merge block is improper because these easily fail to capture long-range dependency over different resolutions, which leads to a semantical ambiguity. In this context, our merge block uses the structural transfer module (STM) to learn the region affinity between neighboring resolutions. Unlike the existing attention module [10], [23], the STM is utilized at different resolutions as a query, key, and value to enable scale-to-scale information transfer. Leveraged by the merge block, the level of structural consistency is enhanced in the final completed image.

In short, our main goal is to fill an extremely large hole while guaranteeing a visually plausible quality. To this end, the following three novel components comprising the proposed inpainting framework outperforms the other counterparts.

1) We design a progressive generator that transfers a roughly completed context to the missing pixels of the neighboring resolution to guide semantically coherent hole-filling across the multi-scale image.
2) We newly devise the PWDD to reinforce the potential quality over the completed region by identifying whether the pixel is masked, and by passing the gradients to all different resolutions.
3) We introduce the merge block that combines the completed images in multiple resolutions through consecutive STMs that transfer the region affinity features to the next resolution activation volumes.

The remainder of this paper is organized as follows. Section II introduces related works to review the latest algorithms. We describe the progressive inpainting, including the detailed architecture of the proposed model and the training procedure in Section III. In Section IV, we conduct extensive experiments to validate our state-of-the-art performance on multiple datasets. Finally, conclusions are provided in Section V.

II. Related Work

Traditionally, numerous image inpaintings have focused on propagating information from visible parts [1], [4], [5], [24], [25]. More recently, many researchers have utilized large image datasets to generate semantically consistent content by applying adversarial training. Therefore, most image inpainting methods produce a realistic image from a given corrupted input by defining it as a conditional generative problem. The context encoder is one of the early attempts to generate reasonable results based on feature learning [9]. In this case, the model was trained using the pixel-to-pixel loss and adversarial loss to restore the corrupted region.

A. Contextual Attention

Yu *et al.* [10] devised two consecutive learning systems that generate an initial coarse prediction and refine the missing region using the contextual attention layer, which computes the long-range spatial dependency during inpainting. Contextual attention propagates information from the background to foreground features through feature matching and aggregation phases. Shin *et al.* [11] offered a single shared contextual attention layer to create a lightweight network. Zeng *et al.* [26] extended this idea to multiple layers by using a pyramid of contextual attention. Instead of borrowing the long-term interaction capabilities of the attention module, Wang *et al.* fully utilize the information in the known area by distinguishing the corrupted region from the valid ones [27]. However, it is still inferior to filling the center of the large hole with coherent content.

B. Coarse-to-Fine Inpainting

Evolving from the above approaches, numerous studies have attempted to use the coarse result of the intermediate network as explicit guidance. Nazeri *et al.* [12] utilized a planar structure to rectify persceptively distorted textures. They first recover the edge maps and then filled in the missing regions with fine details. In addition, Xiong *et al.* [14] proposed a foreground-aware method that infers the contour of saliency objects in a coarse-to-fine fashion. Ren *et al.* [13] proposed a StructureFlow that restores edge-preserved smooth images and then reconstructs the texture details. With the recent success of coarse-to-fine inpainting methods, Lin *et al.* has extended the multi-stage inpainting methods to outpainting work [28]. Inspired by EdgeConnect [12], Lafin [29], and semantic regeneration network [30], they propose a 3-stage completion framework that contains an edge-generator. However, due to the limited representation ability to fill in the large hole, these methods may fail to infer the missing pixels conditioned through valid pixels. Rather than complete coarse information such as edge map and flow map from the corrupted images, Liao *et al.* introduce coherence prior by characterizing the relationships between the semantics and textures [31]. Although these methods have delivered considerable improvements, they remain suboptimal for a prior representation as they mapped the limited relationships between the textures and edges.

C. Recursive Inpainting

Image inpainting methods have recently adopted a recursive hole-filling scheme to cover a large hole. Zhang *et al.* [16] employed a UNet generator with an LSTM in a bottleneck. It takes a sequence of inputs with a large to small hole and generates a sequence of corresponding outputs. Guo *et al.* [15] proposed consecutive residual blocks to fill the hole gradually. They used partial convolutions [32] in these blocks and updated the hole mask according to the invalid pixels selected by partial convolutions. Similarly, Li *et al.* [33] proposed recurrent feature reasoning propagating the confidence region from the boundary to the center within the feature space. However, the recursive process has an unacceptably high computational cost, and recursive feature mapping results in a structure discontinuity output.

D. Pluralistic Image Inpainting

Starting from the *"Pluralistic Image Completion"* [17], diverse image inpainting task, which provide multiple solutions, has developed tremendously in the computer vision literatures. To obtain a diverse set for each masked input, Zheng *et al.* [17] propose a dual pipeline architecture implicitly modeling the data distribution by adopting the variational autoencoder framework [34]. Inspired by this work, Zhao *et al.* maps both features from the corrupted image and reference image into the same probabilistic space to achieve diversities [35]. However, variational training restricts diversity and the images completed through sampling does not change continuously. To this end, Liu *et al.* designs probabilistic diverse [36] by adopting a GAN framework [37], Wan *et al.* borrows transformer [23] representation ability to model a coarse global structure [38]. In our view, the main stream of image painting seems to be divided into two ways: providing a detailed deterministic solution through long-term interactions and providing a diverse solutions through noise sampling. Our attempt is closer to the former methods.

E. Discriminator for Image Synthesis

For plausible image synthesis, the adversarial loss has recently shown impressive results in various computer vision tasks. They mainly focus on improving the discriminator by utilizing multiple [16], [39] and multi-resolution discriminator [20]. However, these methods only allow global feedback of the discriminator scores. To this end, Schonfeld *et al.* devises a u-net based discriminator that also provides global image feedback [22]. In image inpainting literature, there also have been attempts to improve the discriminator ability by focusing more on the newly generated region. In an early attempt, Iizuka *et al.* [21] proposed two discriminators that enforce both global and local consistencies. Shin [11] devised a region ensemble discriminator (RED) to integrate global and local discriminators. However, it is insufficient to provide semantically aware pixel-level feedback to the generator. As the role of a discriminator in image synthesis becomes important, Zhang *et al.* proposes pixel-wise dense detector inspired by [40]. Utilizing the adversarial loss, they localize the position of artifacts in a pixel-wise manner. In our work, we incorporate a pixel-wise feedback framework to the multi-resolution discriminator by scoring the confidence values to reinforce the potential quality of synthesis.

III. Proposed Method

For a better understanding of our proposed framework, we define the following notations:

- Let $\mathbf{x} = \{x_1, ..., x_N\}$ be the multi-resolution target image set where x_n represents the n^{th} downsampled version of the original target image x_0 by a factor of r^n ($r > 1$), and N is the total number of resolutions.
- Let $\mathbf{m} = \{m_1, ..., m_N\}$ denotes the multi-resolution binary mask set where the labels "0" and "1" represent the missing region and context, respectively. Here, m_n represents the down-sampled version of the mask m_0 by a factor of r^n, for some $r > 1$.

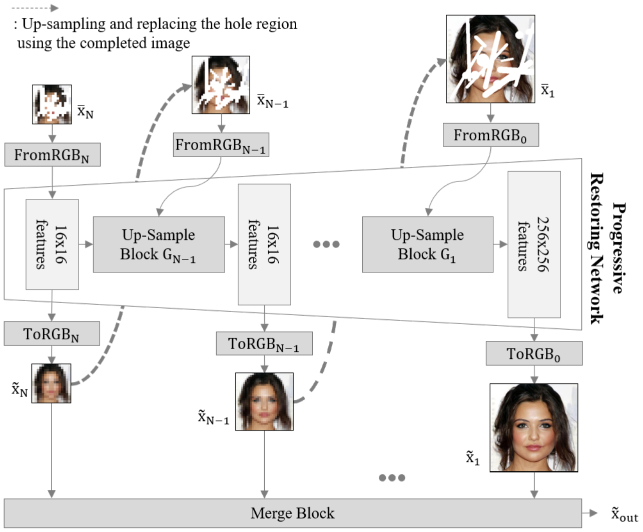


Fig. 2. Illustration of the progressive inpainting pipeline. The model fills the hole from the coarsest to the finest resolution, and the completed images across the different resolutions are then refined passing through the merge block.

- The multi-resolution corrupted image set $\bar{\mathbf{x}} = \{\bar{x}_1, ..., \bar{x}_N\}$ at the corresponding stage is represented as $\bar{x}_n = x_n \odot m_n$, where $\odot$ is the Hadamard product.

A nearest neighbor interpolation was used for down-sampling. To create a valid binary mask, after down-sampling, we round the values.

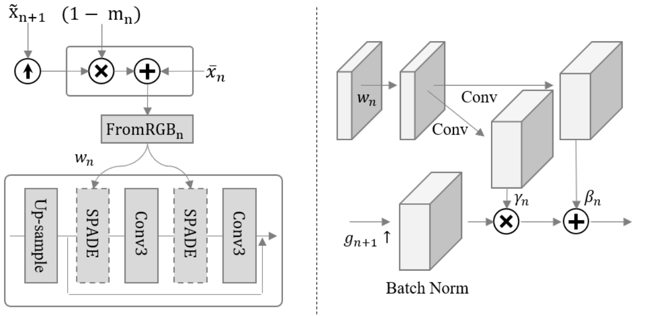
The proposed progressive inpainting consists of two main modules: PRN and PWDD. The PRN is responsible for completing invisible parts across different resolutions in both the image and feature spaces. The quality of the generated images is further reinforced using the PWDD. By estimating the confidence score at each pixel, the adversarial loss helps the PRN to focus more on the newly completed region. The merge block then blends the generated images to refine the structural discontinuity, as depicted in the lower part of Fig. 2. In Section III-A, we describe the overall workflow of the proposed model, and in Section III-B, we present our objective function in the training procedure.

A. Overall Workflow

1) Progressive restoring network: Fig. 2 shows the PRN pipeline where FromRGB$_N$ takes the lowest corrupted image $\bar{x}_N$ and produces feature maps w_N passing through an initial generator block G_N. Then, ToRGB$_N$ completes the hole and produces an initial generated image $\tilde{x}_N$, which is defined as follows:

$$\begin{aligned} &\tilde{x}_N = \text{ToRGB}_N(g_N), \\ &\text{where } g_N = G_N(w_N), \\ &\text{and } w_N = \text{FromRGB}_N(\bar{x}_N),\ \bar{x}_N \in \mathbb{R}^{16\times16\times3}. \end{aligned} \tag{1}$$

To preserve semantic level consistency, the previously generated image substitutes the invisible region of the adjacent resolution, and then feeds into the FromRGB$_n$ as shown in Fig. 3 (a). Thus, the activated feature map w_n is formulated as follows:



(a) Up-Sample Block G$_n$

(b) Spatially-adaptive denormalization

Fig. 3. (a) Details of the up-sample block. (b) The replaced images embedded in the main body of the up-sample block by adopting a SPADE.

$$w_n = \text{FromRGB}_n(\bar{x}_n + \tilde{x}_{n+1} \uparrow^r \odot (1 - m_n)),\ n < N. \tag{2}$$

Then, w_n is embedded in the main body of the up-sample block. Here, we denote each up-sample block attribute to the PRN as $\{G_1, ..., G_{N-1}\}$. To transfer w_n to the main body of the up-sample block G_n, we leverage a spatially adaptive denormalization (SPADE) [41], which is generally utilized to modulate a local feature for an image synthesis. The mathematical expression is briefly summarized as follows:

$$\text{SPADE}(g_{n+1}, \gamma_n, \beta_n) = \gamma_n \frac{g_{n+1} - \mu_{n+1}}{\sigma_{n+1}} + \beta_n, \tag{3}$$

where g_{n+1} is the activation that comes from the previous up-sample block G_{n+1} and, μ_{n+1} and σ_{n+1} are the mean and standard deviation of g_{n+1}, respectively, as shown in Fig. 3 (b). We compute the spatially invariant modulation parameters γ_n and β_n from w_n which are the learnable tensors of the normalization layer. By employing the SPADE layers, abnormal spatial information is recalibrated at the feature level.

After passing through each up-sample block G_n, the recalibrated features g_n are fed into ToRGB$_n$ to generate the completed image at each resolution:

$$\begin{aligned} &\tilde{x}_n = \text{ToRGB}_n(g_n), \\ &\text{where } g_n = G_n(g_{n+1}, w_n) \text{ and } n < N. \end{aligned} \tag{4}$$

The key motivation of the PRN is to transfer intermediate features to neighboring scales and to fill in the invisible parts in a consecutive manner. Using such a method, both visual and contextual coherence, which have to be implied in the invisible region, can be effectively recovered. In our implementation, both FromRGB$_n$ and ToRGB$_n$ consist of a single convolution layer followed by a nonlinear function.

2) Pixel-wise dense detector: Toward a more pleasing image inpainting, we use a newly formulated adversarial training scheme. Beyond a simple feedforward design of a conventional discriminator, the proposed method adopts multiple skip connection blocks following the MSG-GAN [20]. Thus, we can use a single discriminator that allows the gradients to flow at multiple resolutions simultaneously. By contrast to MSG-GAN, the proposed discriminator enables the determination of

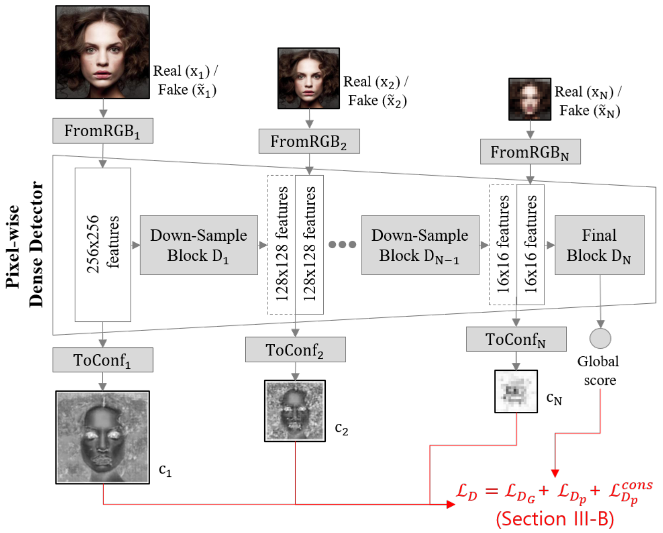

Real (x_1) / Fake ($\tilde{x}_1$)
Real (x_2) / Fake ($\tilde{x}_2$)
Real (x_N) / Fake ($\tilde{x}_N$)
FromRGB$_1$
FromRGB$_2$
FromRGB$_N$
Pixel-wise Dense Detector
256x256 features
Down-Sample Block D$_1$
128x128 features
128x128 features
Down-Sample Block D$_{N-1}$
16x16 features
16x16 features
Final Block D$_N$
ToConf$_1$
ToConf$_2$
ToConf$_N$
Global score
c$_1$
c$_2$
c$_N$
$\mathcal{L}_D = \mathcal{L}_{D_G} + \mathcal{L}_{D_p} + \mathcal{L}_{D_p}^{cons}$
(Section III-B)

Fig. 4. Illustration of the PWDD pipeline. The PWDD determines the input images on a global and local scale at the pixel level.

both global critics and pixel-level classification, which forces the PRN to focus intensively on the invisible hole. Fig. 4 illustrates the overall architecture of the PWDD.

We denote each down-sample block in the PWDD as $\{D_1, ..., D_N\}$. These blocks are designed similar to the up-sample block, but the SPADE is removed, and the last layer is replaced with an average pooling layer. Let d_{n-1} and $\tilde{w}_n$ represent the $(n-1)^{th}$ intermediate features of the PWDD and the embedded features from $\tilde{x}_n$, respectively. The output activation volume d_n of the n^{th} intermediate layer is defined as follows:

$$\begin{aligned} &d_n = D_n(d_{n-1}; \tilde{w}_n), \\ &\text{where } \tilde{w}_n = \text{FromRGB}_n(\tilde{x}_n) \text{ and } n > 1. \end{aligned} \tag{5}$$

In Eq. 5, “; ” denotes the channel-wise concatenation operator. The proposed method additionally estimates the confidence score for whether each pixel belongs to the masked region. To achieve this, we define the ToConf$_n$ block to segment the confidence map at the pixel level by considering the intermediate features of each resolution. Therefore, confidence map c_n is represented as follows:

$$c_n = \text{ToConf}_n(d_{n-1}; \tilde{w}_n), \;\; n > 1. \tag{6}$$

The optimization process using the confidence map is a fully self-supervised learning method in which the visible pixels are intended to be true values, and the invisible pixels become false through an adversarial loss. The implications of adding a per-pixel critic are discussed in Section III-B.

3) Merge Block: To address the structural discontinuities caused by a recursive feature mapping in the PRN, we blend the completed multi-resolution images by consecutively propagating the global structure across different resolutions. The merge block explicitly transfers the contextual representations from the low-resolution completed image into the spatially corresponding positions of the high-resolution completed image, as shown in Fig. 5 left. The self-attention mechanism [10], [23] has been applied as a paradigm capturing long-range interactions between an input and the contextual characteristic. However, the quadratic memory footprint has hindered its applicability in the transfer of scale-to-scale information. For example, applying a single multi-head attention layer to images with a pixel resolution of 128 × 128 with 8 batches still requires more than 32GB of memory, which is generally impractical.

To propagate the long-range interaction between adjacent resolutions, the proposed STM transforms the contexts lying in a low-resolution into individual linear functions, which are directly applied to an adjacent higher resolution as a query. The STM is a modified version of the lambda network [42]. The right side of Fig. 5 depicts the details of the STM. Let us denote $\tilde{\mathbf{x}} : \{\tilde{x}_1, \tilde{x}_2, ...\tilde{x}_N\}$ as a sequence of the generated image set. First, we encode high-level structural inputs through convolutional filters $\mathbf{s} : \{s_1, s_2, ...s_N\}$ $(s_n \in \mathbb{R}^{(H_n \times W_n) \times M_n})$, where H_n and W_n are the height and width of the kernel at the n^{th} level, and M_n is the number of embedding dimensions used to represent each entity. By contrast to conventional attention [10], [23], the STM maps an upper-resolution query q_n to its output p_n through matrix multiplication as $p_n = \lambda_n(p_{n+1})(q_n)$ for a certain linear function λ_n. Such a process can be achieved by defining three learnable weight matrices to transform the queries $w_n^q \in \mathbb{R}^{M_n \times K_n}$, keys $w_n^k \in \mathbb{R}^{M_n \times K_n}$, and values $w_n^v \in \mathbb{R}^{M_n \times U_n}$.

Given s_N, the corresponding output is represented by passing through a single convolutional layer, which is defined as $\lambda_N : s_N \mapsto p_N$, where $p_N \in \mathbb{R}^{(H_N \times W_N) \times M_N}$. Subsequently, the STM first up-samples the lower-resolution context p_{n+1} and then calculates keys and values through a linear projection to w_n^k and w_n^v. These are formulated as follows:

$$k_n = (p_{n+1} \uparrow^r) w_n^k \in \mathbb{R}^{(H_n \times W_n) \times K_n}, \tag{7}$$

$$v_n = (p_{n+1} \uparrow^r) w_n^v \in \mathbb{R}^{(H_n \times W_n) \times U_n}. \tag{8}$$

Here, we adopt the pixelshuffle (depth to space) technique [43] to up-sample its resolution which results in the output $(p_n) \uparrow^r \in \mathbb{R}^{(2H_{n+1} \times 2W_{n+1}) \times (K_{n+1}/4)}$. Subsequently, keys are normalized across the context positions through a softmax function, yielding the normalized keys $\bar{k}_n$. The linearly estimated context λ_n is obtained by summing the contributions from the keys and values as:

$$\lambda_n = \sum_i \bar{k}_n(i)^T v_n(i) \in \mathbb{R}^{K_n \times U_n}, \tag{9}$$

where i is the region belonging to $(H_n \times W_n)$. To apply λ_n, the higher resolution input s_n is transformed into queries such that $q_n = s_n w_n^q \in \mathbb{R}^{(H_n \times W_n) \times K_n}$ and the output of the STM is then obtained as:

$$p_n = q_n \lambda_n + \tilde{q}_n \in \mathbb{R}^{(H_n \times W_n) \times U_n}, \tag{10}$$

where $\tilde{q}_n$ is the skip connection of a single convolution layer to fit the final dimension of the channels.

As shown on the left-side of Fig. 5, after the merging process with the sequentially connected STM chains, the final context map p_0 is obtained. Using the enhanced context map, the final output $\tilde{x}_{out}$ is reconstructed using ToRGB$_M$ as:

$$\tilde{x}_{out} = \text{ToRGB}_M(p_0). \tag{11}$$

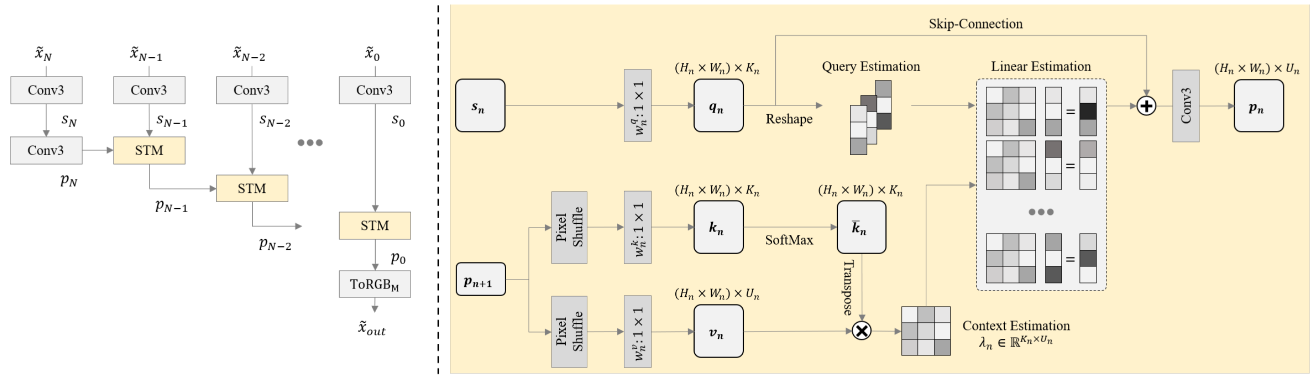


Fig. 5. Illustration of the merge block pipeline. The multi-resolution completed images are merged passing through STM (left). The STM layer transforms each context into a linear matrix that is applied to the corresponding query (right).

Note that the weight parameters of each merge block are not shared. In our implementation, W_n^q, W_n^k, and W_n^v are calculated using a 1×1 convolution filter.

B. Loss Functions

The proposed progressive architecture is simultaneously trained from the lowest to the finest resolution in an end-to-end manner. The objective function comprises adversarial and appearance-matching terms,

$$\min_G \max_D \mathcal{L}_{adv}(G, D) + \lambda \mathcal{L}_{app}(G). \tag{12}$$

The adversarial loss $\mathcal{L}_{adv}$ penalizes the inconsistency between the distribution of the generated images and the distribution in which the training set is drawn. The appearance matching loss $\mathcal{L}_{app}$ makes the model preserves the visible region , fills in the corrupted region, and is able to produce an important feature for semantic consistency.

1) Adversarial loss: While the general discriminator classifies the input images into real and fake, the PWDD additionally applies binary classification on a per-pixel basis (see Fig. 4). By separating the newly generated image into visible and invisible regions, this adversarial loss allows the progressive generator to learn the hallucinating context features.

Hereafter, we denote D_G as the module of the global supervisor and D_P^n as the module of the n^{th} per-pixel classifier, that is, the set of confidence maps is represented as $\mathbf{c} : \{c_1, c_2, ...c_N\} = \{D_P^1(\mathbf{x}), D_P^2(\mathbf{x}), ...D_P^N(\mathbf{x})\}$. Inspired by [26], we adopt the hinge version of the adversarial loss to work around the gradient vanishing problem. The new discriminator loss can now be formulated by taking decisions from both D_G and D_P:

$$\mathcal{L}_D = \mathcal{L}_{D_G} + \mathcal{L}_{D_P}. \tag{13}$$

The loss for the global supervisor $\mathcal{L}_{D_G}$ follows a conventional discriminator loss:

$$\begin{aligned} \mathcal{L}_{D_G} = &- \mathbb{E}\Big[\sum_{i,j} \min(0, -1 + [D_G(\mathbf{x})]_{ij})\Big] \\ &- \mathbb{E}\Big[\sum_{i,j} \min(0, -1 - [D_G(\tilde{\mathbf{x}})]_{ij})\Big], \end{aligned} \tag{14}$$

where $[D_G(\mathbf{x})]_{ij}$ and $[D_G(\tilde{\mathbf{x}})]_{ij}$ are the decisions of the discriminators on the target and generated images at pixel (i, j), respectively. To focus more on the newly generated region across the different resolutions, the visible part of the generated image $\tilde{x}_n$ is mixed with the corresponding target region x_n with the mask m_n:

$$\begin{aligned} \tilde{x}_n^{mix} &= \text{mix}(\tilde{x}_n, x_n, m_n), \\ &= \tilde{x}_n \odot m_n + x_n \odot (1 - m_n). \end{aligned} \tag{15}$$

The mixed sample $\tilde{x}_n^{mix}$, m_n acts as the ground-truth for the corresponding pixel of D_p^n by discriminating whether the pixel belongs to the masked region. To do this, the loss for the per-pixel classifier $\mathcal{L}_{D_P}$ is computed as follows:

$$\begin{aligned} \mathcal{L}_{D_P} = &- \mathbb{E}\Big[\sum_{n=0}^{N} \sum_{i,j \in \mathbb{R}_n} \min(0, -1 + [D_P^n(\tilde{x}_n^{mix})]_{ij})\Big] \\ &- \mathbb{E}\Big[\sum_{n=0}^{N} \sum_{i,j \notin \mathbb{R}_n} \min(0, -1 - [D_P^n(\tilde{x}_n^{mix})]_{ij}\Big], \end{aligned} \tag{16}$$

where $\mathbb{R}_n$ is the visible region, which is denoted as “1” in binary mask m_n. These multi-resolution per-pixel outputs of D_P are derived from global information based on high-level features, which enable the passage of gradients to all resolutions. The corresponding adversarial losses for the generator are as follows:

$$\mathcal{L}_{adv} = -\mathbb{E}\Big[\sum_{i,j} [D_G(\tilde{\mathbf{x}})]_{ij} + \sum_{n=0}^{N} \sum_{i,j \notin \mathbb{R}_n} [D_P(\hat{x}_n^{mix})]_{ij}\Big]. \tag{17}$$

Furthermore, we add consistency regularization to the PWDD. The pixel-wise decision by the well-trained D_p might be stabilized even under any perturbation of the mask. However, the adversarial loss is implicitly assured and the generator is prone to forget the previous task. To explicitly encourage the PWDD, we regularize the confidence map to sense semantic and structural changes between visible and invisible regions, and to pay less attention to arbitrary irregular masks. Similar to the U-Net based discriminator [22], we train the PWDD to output consistent per-pixel predictions, that is, $D_p^n(\tilde{x}_n \cdot m_n + x_n \cdot (1 - m_n)) \simeq D_p^n(\tilde{x}_n) \cdot m_n + D_p^n(x_n) \cdot (1 - m_n)$, by introducing consistency regularization in the loss of the discriminator:

$$\mathcal{L}^{cons}_{D_p} = \sum_{n=0}^{N} \Big\| \big[D_p^n(\tilde{x}_n \cdot m_n + x_n \cdot (1-m_n))\big] - \big[D_p^n(\tilde{x}_n) \cdot m_n + D_p^n(x_n) \cdot (1-m_n)\big] \Big\|_2. \tag{18}$$

Finally, we add a consistency loss in Eq. 13:

$$\mathcal{L}_D = \mathcal{L}_{D_G} + \mathcal{L}_{D_P} + \lambda_c \mathcal{L}^{cons}_{D_P}. \tag{19}$$

Empirically, we set the weighting hyperparameter λ_c as 0.1.

2) Appearance matching loss: With appearance matching loss, the aforementioned objective loss acts as a per-pixel reconstruction, although it can also maintain semantic consistency over the completed image. As such, we define the per-pixel losses for the global consistency, which are separated into visible loss $\mathcal{L}_v = \mathbb{E}[\sum_{n=0}^{N} \frac{1}{N_n} \|m_n \odot (x_n - \tilde{x}_n)\|_1]$ and hole loss $\mathcal{L}_h = \mathbb{E}[\sum_{n=0}^{N} \frac{1}{N_n} \|(1-m_n) \odot (x_n - \tilde{x}_n)\|_1]$, where N_n denotes the number of elements in x_n. For the final output $\tilde{x}_{out}$, the reconstruction loss is expressed as $\mathcal{L}_r = \mathbb{E}[\frac{1}{N_0} \|x_0 - \tilde{x}_{out}\|_1]$.

Next, we include the perceptual and style loss terms, which are widely utilized in the image inpainting task as in [12], [32], [44]. Perceptual loss is defined as the distance between the activation maps of the pre-trained VGG-19 network:

$$\mathcal{L}_p = \mathbb{E}\Big[\sum_m \frac{1}{N_m} \|\phi_m(x_0) - \phi_m(\tilde{x}_{out})\|_1\Big], \tag{20}$$

where ϕ_m is the m^{th} feature map of the pre-trained network (i.e., ReLu1_1, ReLu2_1, ReLu3_1, ReLu4_1 and ReLu5_1), and N_m denotes the number of elements in the feature map. The style loss is defined using a difference measure between the covariances of the activation maps. Let the Gram matrix G^{ϕ}_m operation be $C_m \times C_m$ from the feature map ϕ_m. This term is represented by the following:

$$\mathcal{L}_s = \mathbb{E}\Big[\sum_m \frac{1}{N_m} \|G^{\phi}_m(x_0) - G^{\phi}_m(\tilde{x}_{out})\|_1\Big]. \tag{21}$$

Finally, our appearance matching loss is given by

$$\mathcal{L}_{app} = \lambda_v \mathcal{L}_v + \lambda_h \mathcal{L}_h + \lambda_r \mathcal{L}_r + \lambda_p \mathcal{L}_p + \lambda_s \mathcal{L}_s, \tag{22}$$

where the hyperparameters are determined empirically, (i.e., λ_v, λ_h and λ_r are set to 250, λ_p is 0.1 and λ_s is 180).

IV. Experiments

A. Implementation Details

Datasets: Experiments are conducted on three commonly utilized datasets adopted in studies on image inpainting: Places2 [45], Paris StreetView [46], and CelebA-HQ [18]. We also adopt an external irregular mask dataset [47] to corrupt the target images. For fair comparisons, the same dataset setting is applied to our experiments and counterparts.

- **Places2:** A dataset containing over 1.6 million training images from 365 scene categories, which is extremely suitable for natural image inpainting as it enables the model to learn many different scenes. We follow the provided training, testing, and validation splits.
- **Paris StreetView:** A dataset containing 14,900 training images and 100 test images collected from the street views of Paris, which mainly focuses on buildings and structure information.
- **CelebA-HQ:** A dataset containing 30,000 celebrity facial images. We randomly divide the images into 90% for training and validation and 10% for testing.
- **Masks:** The mask is automatically generated during the training procedure following the free-form mask algorithm [48], which draws multiple lines and then erases pixels closer than an arbitrary distance from the lines. For the test, we employ an irregular mask dataset [47] that covers different hole-to-image area ratios.

Training Procedure: Images with a resolution of 256×256 are utilized to train the proposed model. The color values of all images are linearly scaled to $[-1, 1]$ during all experiments. The scale of the multi-resolution images to be input into the PRN is set from 4 to 6 and the optimal N, the size of the lowest-resolution image, is set from 32×32 to 8×8. Then, the PWDD takes the same scale as the multi-resolution images as the PRN receives. Before the training procedure, we initialize all weights of the network using the normalized distribution $\mathcal{N}(0, 1)$. We conduct the optimization using the Adam optimizer [49] with $(\beta_1, \beta_2) = (0.0, 0.99)$ for both the generator and discriminator. We set the learning rate to $\lambda_g = 1e^{-4}$ for the generator which is then decreased by one order to $\lambda_d = 1e^{-1} \cdot \lambda_g$ for the discriminator because it is easy to distinguish whether the image is real or fake during the early training stage, resulting in unstable training [20].

The spectral normalization (SN) [50] is used to stabilize our model by scaling down the weight metrics with their largest singular values. Owing to the SN, both the PRN and PWDD are free from sudden changes in the parameters and gradient values. In general, SN has been widely applied in discriminators [50], [51]; however, we utilize it in a generator inspired by a recent study [52], which suggests that the generator profoundly benefits by limiting the sudden change in trainable parameters and gradient values.

Furthermore, we adopt a coordinate convolution [53] to the generator, which allows networks to learn either a complete translation invariance or varying degrees of translation dependency. This method is used to capture detailed information dependent on the local regions because the mask is randomly positioned over the target image in the training process. The entire training procedure is presented in Algorithm 1. Our full model is trained using a single GPU RTX 2080 Ti and powered in PyTorch v1.3.

Comparison Models: We compare our proposed model with four existing state-of-the-art image inpainting methods. We choose the EdgeConnect [12], Partial Convolution (PConv) [32], Gated Convolution (GatedConv) [48], and Recurrent Feature Reasoning (RFR) [33] for the qualitative and quantitative comparisons. These models are re-trained until convergence following the same experimental settings proposed in each study.

- **EdgeConnect:** This method first provides edge information of the missing region and recovers color and texture information by utilizing the generated edge information.

Algorithm 1: Training of our proposed network

```
Inputs  : $\bar{\mathbf{x}} : \{\bar{x}_1, \bar{x}_2, ..., \bar{x}_N\}$, Corrupted images;
          $\mathbf{m} : \{m_1, m_2, ..., m_N\}$, Masks;
Outputs: $\tilde{\mathbf{x}} : \{\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_N\}$, Outputs from the PRN;
          $\mathbf{c} : \{c_1, c_2, ..., c_N\}$, Confidence maps;
          $\tilde{x}_{out}$, Outputs from the merge block;
initialization;
while PRN has not converged do
    for $n$=$N$,$N$-1,...1 do
        if $n$=$N$ then
            $\tilde{x}_n \leftarrow \mathrm{PRN}_n(\bar{x}_n)$;
        else
            $\tilde{x}_n \leftarrow \mathrm{PRN}_n(\bar{x}_n + \tilde{x}_{n+1} \uparrow^r \odot (1 - m_n))$;
        end
    end
    $critics, \mathbf{c} \leftarrow \mathrm{PWDD}(\mathbf{x})$ or $\mathrm{PWDD}(\tilde{\mathbf{x}})$;
    $\tilde{x}_{out} \leftarrow \mathrm{MergeBlock}(\tilde{\mathbf{x}})$;
    Updating the PWDD with loss $\mathcal{L}_D$;
    Updating the PRN and MergeBlock with loss
      $\mathcal{L}_{adv} + \mathcal{L}_{app}$;
end
```

- **PConv:** This method is a classic approach that allows irregular hole filling by applying a novel convolutional layer that is aware of the mask shape and addresses the inductive locality problem of the inpainting.
- **GatedConv:** This method addresses the issue of a vanilla convolution by generalizing the PConv using a learnable dynamic feature selection mechanism.
- **RFR:** This method is devised to cover the large continuous hole filling scenarios by recurrently inferring the hole boundaries of the convolutional feature maps and then uses them as clues for further inference.

B. Performance Evaluation

Qualitative Comparisons: To demonstrate the superiority of the proposed method, we report the notable results to address the challenging problem in image inpainting studies, which hides most of the visible region by employing an extremely large hole. Figs. 6, 7, and 8 illustrate the completed outputs of the progressive inpainting with our previously described counterparts on the Places2, Paris StreetView, and CelebA-HQ datasets, respectively. EdgeConnect, PConv, and GatedConv perform well when the missing region is small and narrow (please refer to the studies of each method [12], [32], and [48]), but they have serious artifacts as the missing region becomes larger for all datasets.

Compared to the other methods, RFR, which adopts a recurrent hole-filling scheme, generates more semantically plausible results even in a large missing region. However, RFR occasionally produces a blurry texture in a heterogeneous region with high-frequency components. This implies that the RFR is vulnerable to the creation of alternative contextual details while maintaining structural consistency. With the help of a progressive hole-filling scheme and cross-scale STM, our model generates more semantically reasonable and visually plausible results with clear textures and consistent structures, even in cases of an extremely large hole.

TABLE I
STATISTICS OF USER STUDY. THE VALUE INDICATES THE RANK PERCENTAGE.

Method	EdgeConnect	PConv	GatedConv	RFR	Ours
percentage	10.48%	3.01%	7.88%	33.85%	**44.78%**

To further validate our superiority, we conducted a user study. We randomly collected $1,000$ images from Places2 and corrupted more than 60% of the visible regions using the test mask dataset. These corrupted images are completed from different models, for example, EdgeConnect, PConv, GatedConv, RFR, and Progressive Inpainting, and then shown to the volunteers anonymously. Over 20 volunteers are invited to evaluate the performance of the models and asked to choose the image that seems the most natural. Table I shows the statistical results for the votes. Our model is ranked better than the other models, and we can find that people prefer structurally clearer results than textual details.

Quantitative Comparisons: We conduct quantitative comparison in terms of the mean l_1, peak signal-to-noise ratio (PSNR), structural similarity index (SSIM), and Fréchet Inception Distance (FID) [54]. The first three metrics, e.g. mean l_1, PSNR and SSIM, assume pixel-wise independency, which may mark favorable scores to perceptually unreasonable results. Therefore, we employ the FID, which calculates the distance between the feature vectors calculated for the target and generated images using a pre-trained inception-V3 model [55]. Note that these statistics rely on the completed image, which mostly consists of the ground truth. Therefore, our reported FID scores are lower than those of the other generative models [56].

Table II shows the quantitative results on three datasets with mask ratio $(0.1, 0.2]$, $(0.3, 0.4]$, and $(0.5, 0.6]$. Our method produces excellent results and achieves the highest scores for all indicators in the Places2 and Paris StreetView. In particular, our model outperforms the other benchmarks by a large margin in the large hole case, whereas the recorded scores are compatible with our counterparts in the small hole case, thereby demonstrating the robustness of learning in an irregular hole-filling process.

C. Analysis of Progressive Generator

The progressive hole-filling scheme inherently assumes that once the global context is completed at an easier low-resolution, the model gradually adds texture details by enlarging the resolutions. During the training, all layers in the PRN were synchronized across the pre-processed resolutions early in the training. Therefore, the quality of the generated images is subsequently improved at all scales simultaneously. Fig. 1 shows the completed images at all resolutions. We can observe that global semantics at the lowest resolution are maintained throughout the training procedure, and the PRN incrementally improves the details across the different resolutions.

Furthermore, we investigate the influences of different values of N, which determine the number of input resolutions.

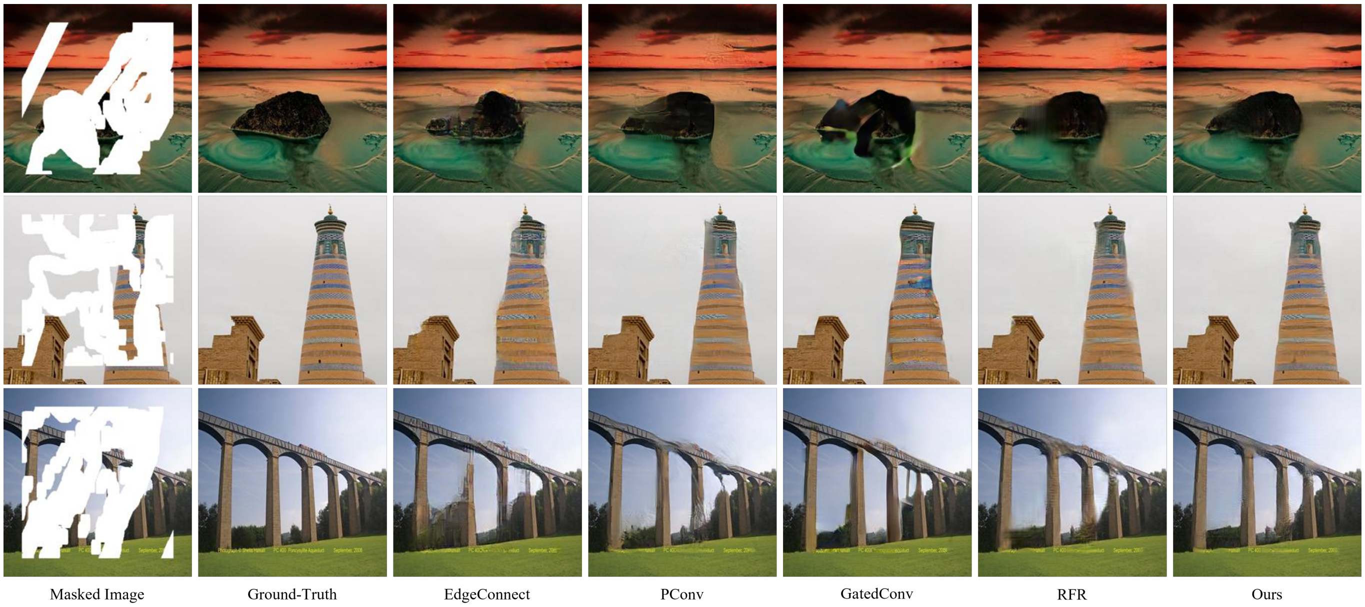


Fig. 6. Qualitative comparisons of Places2 dataset (please zoom in to see the detail).

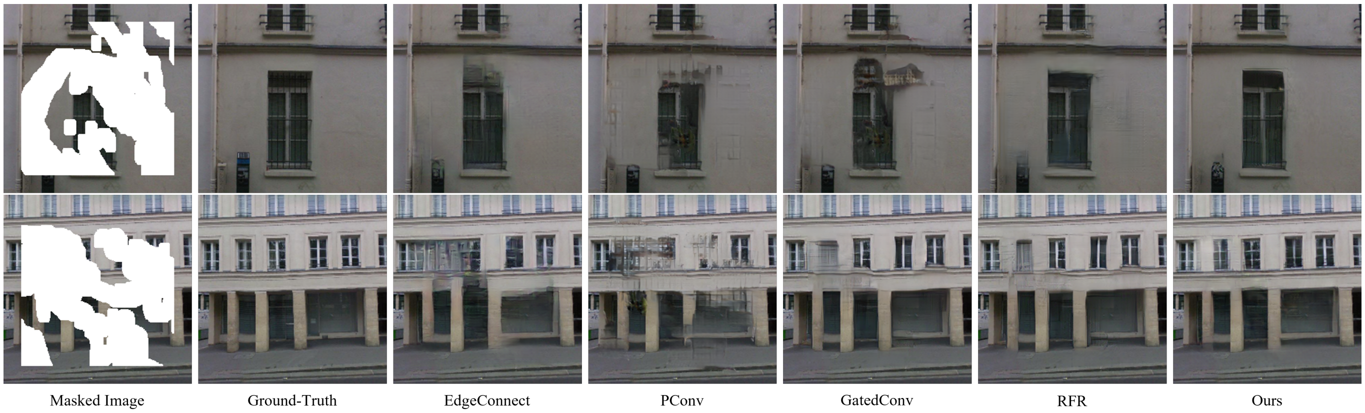


Fig. 7. Qualitative comparisons of Paris StreetView dataset (please zoom in to see the detail).

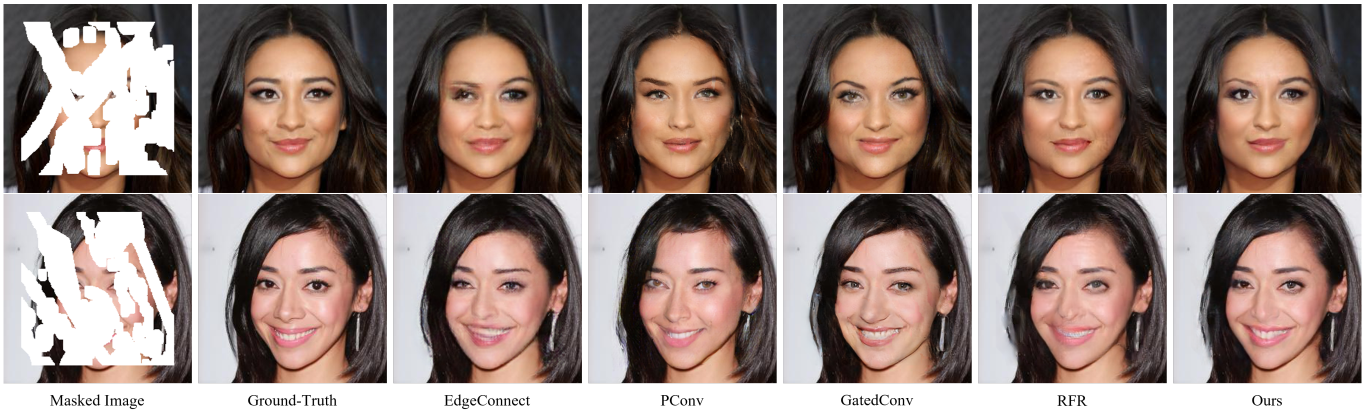


Fig. 8. Qualitative comparisons on CelebA-HQ dataset (please zoom in to see the detail).

Empirically, we set N to 4, 5, and 6, and the corresponding multi-resolution images are embedded in the PRN, i.e. the lowest resolution is 32×32, 16×16, and 8×8, respectively. Table III shows the results for the CelebA-HQ dataset corresponding to different N values after the same training iterations with a mask ratio of 30-40%. This ablation study reveals that our

TABLE II
QUANTITATIVE COMPARISON ON THREE DATASETS. THE BEST MEASURES ARE IN BOLD. † A LOWER VALUE IS BETTER. ∗ A HIGHER VALUE IS BETTER.

Dataset		Places2			Paris StreetView			CelebA-HQ		
Mask Ratio		10-20%	30-40%	50-60%	10-20%	30-40%	50-60%	10-20%	30-40%	50-60%
Mean$l_1^\dagger$	EdgeConnect	0.0157	0.0408	0.0821	0.0110	0.0281	0.0582	0.0086	0.0239	0.0551
	PConv	0.0154	0.0426	0.0874	0.0123	0.0313	0.0623	0.0081	0.0234	0.0524
	GatedConv	0.0150	0.0397	0.0814	0.0120	0.0309	0.0660	0.0085	0.0240	0.0541
	RFR	0.0142	0.0381	0.0761	**0.0110**	0.0275	0.0546	**0.0071**	**0.0209**	0.0467
	Ours	**0.0139**	**0.0368**	**0.0754**	0.0118	**0.0265**	**0.0544**	0.0079	0.0214	**0.0463**
PSNR*	EdgeConnect	27.17	22.18	18.35	31.19	26.04	21.89	32.76	26.52	22.28
	PConv	27.29	22.04	18.07	30.76	25.46	21.39	32.84	26.84	22.27
	GatedConv	27.18	22.31	18.39	31.32	25.52	20.61	32.56	26.79	22.08
	RFR	27.78	22.63	18.92	**31.71**	26.44	22.40	**33.71**	**27.83**	23.03
	Ours	**28.01**	**22.69**	**18.98**	31.48	**26.83**	**22.71**	33.49	27.61	**23.17**
SSIM*	EdgeConnect	0.933	0.802	0.553	0.950	0.849	0.646	0.965	0.915	0.759
	PConv	0.934	0.812	0.519	0.947	0.835	0.619	0.966	0.922	0.791
	GatedConv	0.933	0.803	0.555	0.953	0.849	0.621	0.963	0.914	0.767
	RFR	0.939	0.819	0.596	0.954	0.862	0.681	0.982	0.934	0.819
	Ours	**0.941**	**0.824**	**0.603**	**0.960**	**0.867**	**0.699**	**0.987**	**0.941**	**0.831**
FID†	EdgeConnect	2.42	9.05	15.39	2.38	8.97	15.29	2.27	8.83	15.15
	PConv	2.44	9.16	15.52	2.36	8.89	15.32	2.32	8.75	15.28
	GatedConv	2.40	9.07	15.47	2.33	8.93	15.33	2.29	8.80	15.21
	RFR	2.31	8.91	15.35	2.29	8.68	15.09	2.18	8.63	15.06
	Ours	**2.28**	**8.86**	**15.17**	**2.26**	**8.60**	**15.08**	**2.03**	**8.58**	**14.98**

TABLE III
INFLUENCES OF DIFFERENT N RESOLUTIONS. † A LOWER VALUE IS BETTER. ∗ A HIGHER VALUE IS BETTER.

N	4	5	6
SSIM*/ FID†	0.730/15.51	0.858/8.72	0.862/8.65

model is robust to changes in this parameter when N is set to 5 and 6. The results also show an improved performance compared with previous methods. However, when N is set to 4 and the lowest resolution is set to 32×32, the quantitative scores are dramatically decreasing. This can be interpreted that the 32×32 resolution is insufficient for the model to comprehend the global context first, owing to the limited size of the receptive filter. Therefore, N should be greater than 4 to ensure a sufficient visual quality of the completed images.

D. Analysis of Pixel-wise Dense Detector

In this section, we verify the effect of the PWDD, which helps the generator focus more on the generated region by introducing the newly formulated adversarial loss. To do this, comparisons are conducted from all images on the Paris StreetView and CelebA-HQ datasets.

Effectiveness of Pixel-wise Dense Detector: We first test our proposed PWDD by comparing its performance with two representative discriminators. One is the RED, which is newly introduced in PEPSI++ [11] for image inpainting tasks. The other is a discriminator from MSG-GAN (MSGD) [20], which allows a multi-scale gradient flow for generative adversarial networks. For fair comparisons, we employ each discriminator on a PRN without a merge block for the baseline network. As shown in Table IV, the proposed PWDD exhibits a better performance in terms of the SSIM and FID compared to existing discriminators.

TABLE IV
NUMERICAL COMPARISONS ACCORDING TO THE ABLATION OF DIFFERENT DISCRIMINATORS. † LOWER IS BETTER. ∗ HIGHER IS BETTER.

Dataset		Paris StreetView		CelebA-HQ	
Mask Ratio		30-40%	50-60%	30-40%	50-60%
RED	SSIM*	0.849	0.624	0.897	0.771
	FID†	8.97	15.55	8.99	15.38
MSGD	SSIM*	0.842	0.631	0.901	0.785
	FID†	8.90	15.49	8.93	15.33
PWDD	SSIM*	0.850	0.643	0.907	0.805
	FID†	8.77	15.31	8.84	15.26
PWDD (+consistency)	SSIM*	**0.854**	**0.650**	**0.929**	**0.813**
	FID†	**8.73**	**15.25**	**8.82**	**15.18**

To further demonstrate our PWDD, we show the qualitative comparisons according to each discriminator in Fig. 9. The examples show that the existing discriminators cannot provide satisfactory feedback to generate visually plausible images. They produce results with visual artifacts such as blurred or distorted images in the masked region. Although they are effective in the original studies, as the hole size becomes larger, a global-critic is insufficient to generate texture details into the deeper missing region. In contrast to the compared counterparts, because the PWDD allows per-pixel feedback to the PRN at a semantic level, the generator can maintain the global and local realism.

Effectiveness of Consistency Regularization: The above results show that the PWDD compares favorably with the state-of-the-art approaches on public datasets and shows the high potential of the reconstruction. However, the adversarial loss implicitly guarantees these potential qualities. To enable it explicitly, the PWDD should be regularized to focus more on the semantic and structural changes in the masked region. To

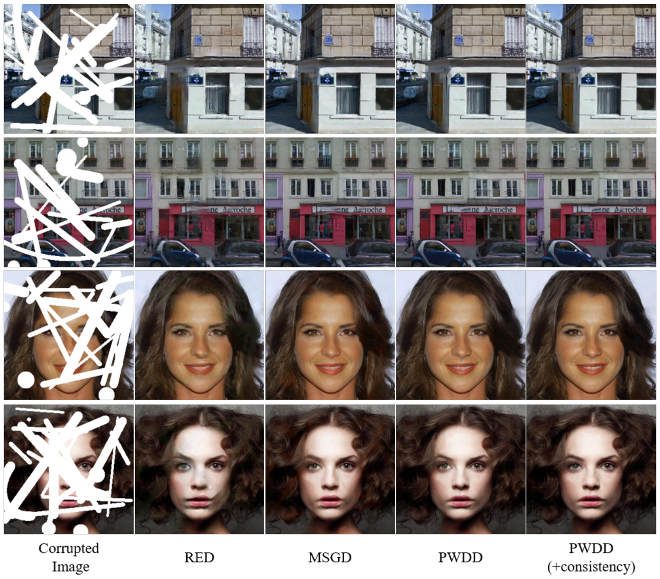


Fig. 9. Comparisons of RED, MSGD and PWDD (+consistency) on Paris StreetView and CelebA-HQ datasets. The PWDD (+consistency) preserves the contextual details better than our counterparts (zoom in to see the details).

inject this particular prior, we introduce the consistency regularization in Eq. 18. This strategy stems from our observation that consistency regularization [22] is most helpful for penalizing the per-pixel inconsistency prediction of the discriminator under the CutMix transformations [57]. As shown in Table IV, employing the proposed consistency regularization in the multi-resolution confidence maps enables us to obtain the most superior scores for our indicators and allows us to leverage a better per-pixel feedback of the PWDD without imposing much computational or memory costs. Overall, we observe that consistency regularization leads to an improved visual quality in terms of the quantitative indicators and qualitatively (Fig. 9).

Visualization of Confidence Map: With the help of pixel-wise feedback, including consistency regularization, our method brings a stronger restoration effect even in the case of extremely large holes. Experimentally, we observe that this provides a detailed and spatially coherent response to the PRN, which leads to further improvement in the output quality, as shown in Fig. 10. The brighter color refers to the pixel confidence in the visible region and darker in the invisible region. The influence of pixel-level feedback is significant during the training procedure. Fig. 11 shows the confidence maps of the finest resolution according to the training epoch. Intuitively, we can find that the unnaturally blurry corrupted region is recognized as fake by the PWDD during early training and is corrected by the PRN throughout the training. Therefore, at the end of the training epoch, the confidence scores are similarly distributed for all pixels, which implies that the visual quality of the newly generated region becomes indistinguishable from the visible region.

TABLE V

NUMERICAL COMPARISONS ACCORDING TO THE ABLATION OF THE MERGE BLOCKS. † A LOWER VALUE IS BETTER. ∗ A HIGHER VALUE IS BETTER.

Dataset		Paris StreetView		CelebA-HQ	
Mask Ratio		30-40%	50-60%	30-40%	50-60%
w/o Merge Block (PRN)	SSIM*	0.854	0.650	0.929	0.813
	FID†	8.73	15.25	8.82	15.18
w/ Merge Block (Conv Layer)	SSIM*	0.857	0.662	0.936	0.819
	FID†	8.68	15.22	8.71	15.08
w/ Merge Block (Attention)	SSIM*	0.859	0.679	0.940	0.824
	FID†	8.70	15.17	8.64	15.14
w/ Merge Block (STM)	SSIM*	**0.867**	**0.699**	**0.941**	**0.831**
	FID†	**8.60**	**15.08**	**8.58**	**14.98**

E. *Analysis of Merge Block*

In this section, we explore the ability of the merge block to refine the multiresolution outputs of the PRN. We conduct additional experiments combining multi-resolution images with different techniques. The first benchmark is an intuitive approach that adopts fully convolutional layers. Therefore, we replace the STM illustrated on the left side of Fig. 5 with the transposed convolution layer with a feature concatenation. The second benchmark employs the existing attention module [23] to enhance the relevant features from scale to scale. Considering the high computation cost, we introduce the existing attention module up to a resolution of 32×32 and replace the rest with the convolutional layers.

Effectiveness of Merge Block: We report in Table V the SSIM and FID scores calculated over the four different ablation models. Comparing the results with and without the merge block, both indicators show higher marks when using the merge block. This demonstrates that the merge block has a robust capability for generating realistic images. The improved scores of our model, which combines multi-resolution images, indicate that the merge block plays a beneficial role in generating realistic images by directly supervising the semantic correspondence between adjacent resolutions.

Effectiveness of Structural Transfer Module: To transfer scale to scale information, we devise a STM that enables long-range interaction by utilizing the linearly estimated function. The STM outperforms the other benchmarks by a large margin, as reported in Table V. Our experiments indicate that the STM achieves a superior refinement layer that provides semantically coherent knowledge across different resolutions. Fig. 12 illustrates the final inpainting images of the three differently constructed merge blocks. Compared to the benchmarks, the STM results show semantically consistent patches inside the missing region, which implies that the proposed module can be properly operated as scale-matching features. To further validate this, we visualize the structure similarity maps [58], which are defined as

$$\mathrm{SSIM}(x,y) = [l(x,y)]^{\alpha} \cdot [c(x,y)]^{\beta} \cdot [s(x,y)]^{\gamma}, \qquad (23)$$

where $\alpha = \beta = \gamma = 1$ and $l(x,y)$, $c(x,y)$, and $s(x,y)$ represent the luminance, contrast and structure maps, respectively, following the original literature [58]. In our experiment, x

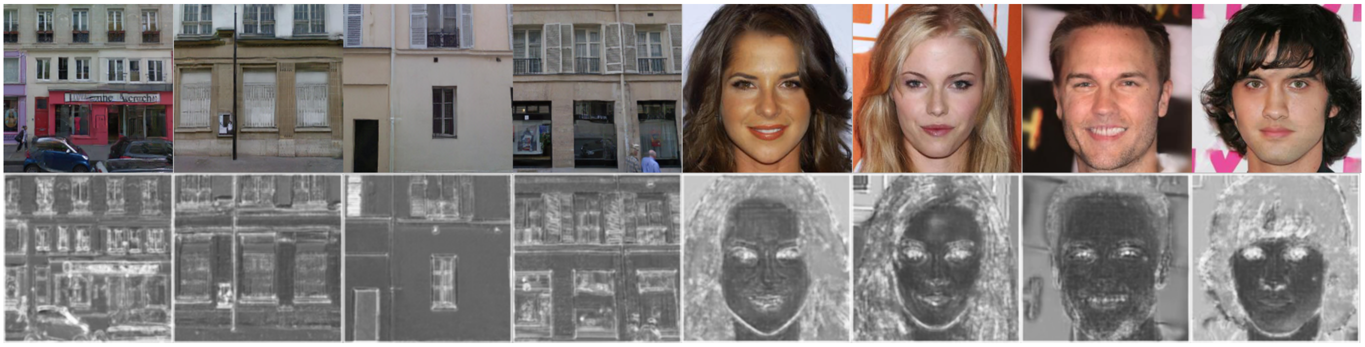


Fig. 10. Examples of inpainting results and corresponding confidence maps. Brighter colors correspond to the discriminator of the pixel as real and darker colors correspond to those as fake.

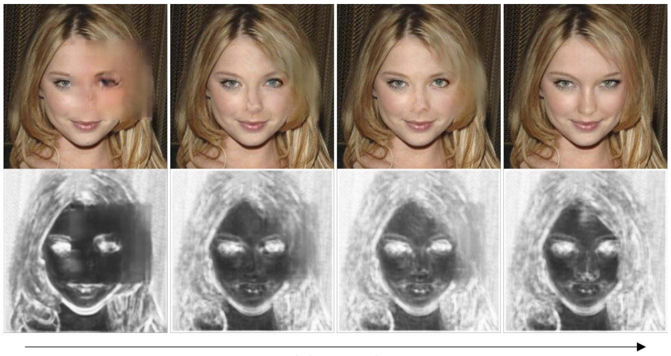


Fig. 11. Inpainting results throughout the training procedure and the corresponding confidence maps.

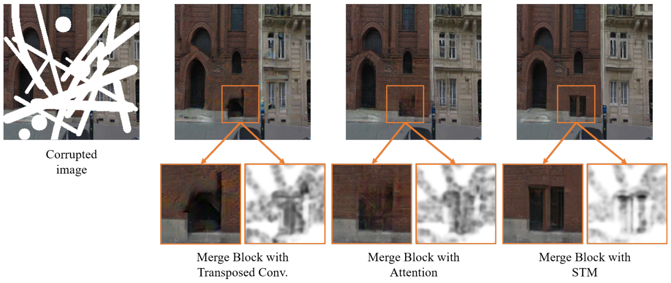


Fig. 12. Comparisons according to different settings of the merge block (please zoom in to see the details).

TABLE VI

Model efficiency of different networks and their quantitative performances on CelebA-HQ dataset of $40\% - 50\%$ mask. M is short for Milion.

Method	Metrics			Efficiency	
	PSNR*	SSIM*	FID†	Params	FLOPs
PConv	25.32	0.8803	10.73	26M	**19G**
GatedConv	25.78	0.8917	10.08	27M	138G
RFR	26.91	0.9062	9.92	31M	206G
PRN($Ours$)	27.03	0.9148	9.30	**19M**	22G
+Merge Block($Ours$)	**27.28**	**0.9204**	**8.90**	21M	40G

TABLE VII

Detailed combinations of the objective functions.

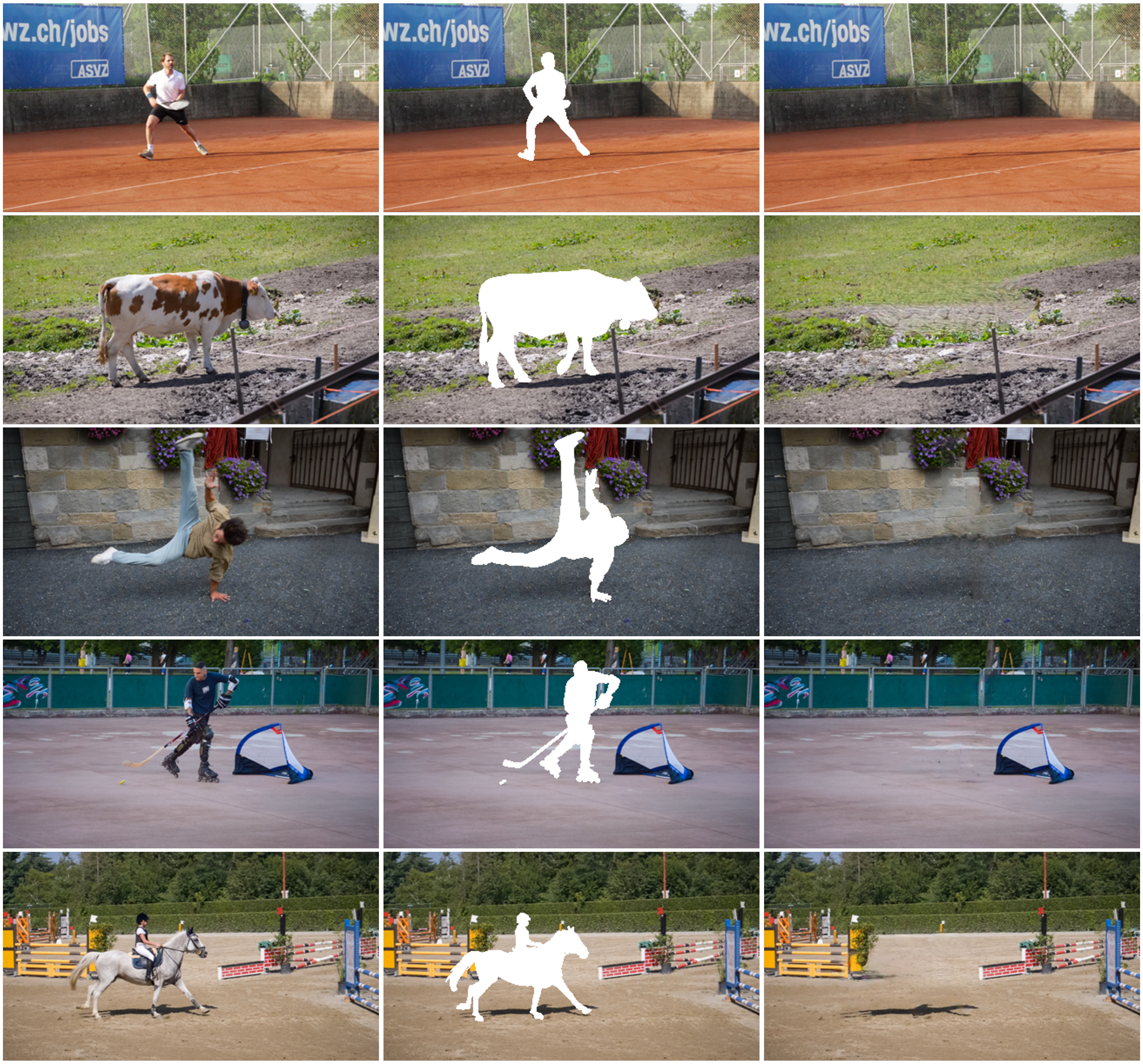
Configurations		Metrics		
$\mathcal{L}_{adv}$	$\mathcal{L}_{app}$	PSNR*	SSIM*	FID†
Hinge Loss (w/o $\mathcal{L}_{D_p}^{cons}$)	$\mathcal{L}_{rec}$	26.70	0.9115	9.58
Hinge Loss (w/o $\mathcal{L}_{D_p}^{cons}$)	$\mathcal{L}_{rec} + \mathcal{L}_{vgg}$	27.27	0.9153	9.44
Hinge Loss (w $\mathcal{L}_{D_p}^{cons}$)	$\mathcal{L}_{rec}$	**27.41**	0.9183	9.32
Hinge Loss (w $\mathcal{L}_{D_p}^{cons}$)	$\mathcal{L}_{rec} + \mathcal{L}_{vgg}$	27.28	**0.9204**	**8.90**
WGAN-GP (w/o $\mathcal{L}_{D_p}^{cons}$)	$\mathcal{L}_{rec}$	24.51	0.8948	9.85
WGAN-GP (w/o $\mathcal{L}_{D_p}^{cons}$)	$\mathcal{L}_{rec} + \mathcal{L}_{vgg}$	25.96	0.9033	9.74
WGAN-GP (w $\mathcal{L}_{D_p}^{cons}$)	$\mathcal{L}_{rec}$	25.84	0.9162	9.80
WGAN-GP (w $\mathcal{L}_{D_p}^{cons}$)	$\mathcal{L}_{rec} + \mathcal{L}_{vgg}$	26.03	0.9149	9.73

and y are the target and inpainting images $\tilde{x}_{out}$, respectively. Fig. 12 shows the magnified local region and corresponding structural similarity maps indicating that a darker color shows a loss of structure. These results demonstrate that the STM can semantically generate reasonable and fine-detailed results by aggregating contextual information through multiple resolutions.

F. Addtional Analysis

Model Efficiency: As shown in Table VI, the proposed model (with PRN and merge block) has fewer parameters than the widely used PConv [32], GatedConv [48], and RFR [33]. Specifically, we require less memory but have higher FLOPs than PConv due to unoptimized long-term interaction implementation, e.g., merge block. On the other hand, PConv is based on CNN which is a highly optimized computation. In addition, the inference time of our model (PRN+merge block) for each image is usually between 72 and 84 ms, which is also faster than several benchmarks. Considering qualitative results and model efficiency simultaneously, we can conclude that the proposed model efficient than several state-of-the-art methods.

Specific Setting of Objective Function Combinations: As shown in Eq. (12), the proposed loss function can be divided as $\mathcal{L}_{adv}$ and $\mathcal{L}_{app}$. In particular, $\mathcal{L}_{adv}$ consists of conventional adversarial loss $\mathcal{L}_{D_G}$ and PWDD term $\mathcal{L}_{D_P}$ (with and without consistency regularization $\mathcal{L}_{D_p}^{cons}$), and $\mathcal{L}_{app}$ includes reconstruction loss ($\mathcal{L}_{rec} = \mathcal{L}_v + \mathcal{L}_h + \mathcal{L}_r$) and VGG loss ($\mathcal{L}_{vgg} = \mathcal{L}_p + \mathcal{L}_s$). Table VII shows qualitative scores for various loss function combinations on CelebA-HQ degraded by $40\% - 50\%$ mask ratio. Here, we compared



(a) Original

(b) Input

(c) Output

Fig. 13. Additional resutls on DAVIS dataset for object removal. All results are reported at 432 × 240 resolution.

two conventionally used adversarial losses which are hinge loss and WGAN-GP [59] and applied Adam optimizer for all combinations with the same learning rate. Each weighting hyperparameters was set according to the previous studies [11], [12]. In our model, the hinge loss was the best fit, and a noticeable improvement in perceptual metrics was observed when the VGG loss was added.

Additional Visualization: Moreover, we study important real use case of image inpainting. In Fig. 13, we illustrate some examples for object removal using a DAVIS dataset [60]. Our model works well on the natural and outdoor scenes and could synthesize sharp and clear appearance by preserving the background textures in invisible or occluded regions.

V. Conclusion

This paper presents a novel progressive image inpainting scheme that integrates a progressive generator, pixel-wise dense detector, and merge block. With the help of the collaborative functionality of the three proposed networks, the completed images show visually plausible results even in a large hole. The evaluation results demonstrate that our method exhibits a superior performance on diverse datasets, which implies that our method has significant potential for practical applications.

REFERENCES

[1] C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman, "Patchmatch: A randomized correspondence algorithm for structural image editing," in *ACM Transactions on Graphics*, vol. 28, no. 3. ACM, 2009, p. 24.

[2] A. Levin, A. Zomet, S. Peleg, and Y. Weiss, "Seamless image stitching in the gradient domain," in *ECCV*. Springer, 2004, pp. 377–389.

[3] E. Park, J. Yang, E. Yumer, D. Ceylan, and A. C. Berg, "Transformation-grounded image generation network for novel 3d view synthesis," in *CVPR*, 2017, pp. 3500–3509.

[4] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera, "Filling-in by joint interpolation of vector fields and gray levels," *IEEE Transactions on Image Processing*, vol. 10, no. 8, pp. 1200–1211, 2001.

[5] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *SIGGRAPH*. ACM, 2000, pp. 417–424.

[6] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, 2001, pp. 341–346.

[7] A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2. IEEE, 1999, pp. 1033–1038.

[8] I. Drori, D. Cohen-Or, and H. Yeshurun, "Fragment-based image completion," in *ACM SIGGRAPH 2003 Papers*, 2003, pp. 303–312.

[9] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *CVPR*, 2016, pp. 2536–2544.

[10] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *CVPR*, 2018, pp. 5505–5514.

[11] Y.-G. Shin, M.-C. Sagong, Y.-J. Yeo, S.-W. Kim, and S.-J. Ko, "Pepsi++: Fast and lightweight network for image inpainting," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[12] K. Nazeri, E. Ng, T. Joseph, F. Qureshi, and M. Ebrahimi, "Edgeconnect: Generative image inpainting with adversarial edge learning," *arXiv preprint arXiv:1901.00212*, 2019.

[13] Y. Ren, X. Yu, R. Zhang, T. H. Li, S. Liu, and G. Li, "Structureflow: Image inpainting via structure-aware appearance flow," in *ICCV*, 2019, pp. 181–190.

[14] W. Xiong, J. Yu, Z. Lin, J. Yang, X. Lu, C. Barnes, and J. Luo, "Foreground-aware image inpainting," in *CVPR*, 2019, pp. 5840–5848.

[15] Z. Guo, Z. Chen, T. Yu, J. Chen, and S. Liu, "Progressive image inpainting with full-resolution residual network," in *ACM Multimedia*, 2019, pp. 2496–2504.

[16] H. Zhang, Z. Hu, C. Luo, W. Zuo, and M. Wang, "Semantic image inpainting with progressive generative networks," in *ACM Multimedia*, 2018, pp. 1939–1947.

[17] C. Zheng, T.-J. Cham, and J. Cai, "Pluralistic image completion," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1438–1447.

[18] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

[19] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4401–4410.

[20] A. Karnewar and O. Wang, "Msg-gan: Multi-scale gradients for generative adversarial networks," in *CVPR*, 2020, pp. 7799–7808.

[21] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Transactions on Graphics*, vol. 36, no. 4, p. 107, 2017.

[22] E. Schonfeld, B. Schiele, and A. Khoreva, "A u-net based discriminator for generative adversarial networks," in *CVPR*, 2020, pp. 8207–8216.

[23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[24] J.-B. Huang, S. B. Kang, N. Ahuja, and J. Kopf, "Image completion using planar structure guidance," *ACM Transactions on Graphics*, vol. 33, no. 4, p. 129, 2014.

[25] J. C. Hung, C.-H. Huang, Y.-C. Liao, N. C. Tang, and T.-J. Chen, "Exemplar-based image inpainting base on structure construction." *JSW*, vol. 3, no. 8, pp. 57–64, 2008.

[26] Y. Zeng, J. Fu, H. Chao, and B. Guo, "Learning pyramid-context encoder network for high-quality image inpainting," in *CVPR*, 2019, pp. 1486–1494.

[27] N. Wang, Y. Zhang, and L. Zhang, "Dynamic selection network for image inpainting," *IEEE Transactions on Image Processing*, vol. 30, pp. 1784–1798, 2021.

[28] H. Lin, M. Pagnucco, and Y. Song, "Edge guided progressively generative image outpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 806–815.

[29] Y. Yang, X. Guo, J. Ma, L. Ma, and H. Ling, "Lafin: Generative landmark guided face inpainting," *arXiv preprint arXiv:1911.11394*, 2019.

[30] Y. Wang, X. Tao, X. Shen, and J. Jia, "Wide-context semantic image extrapolation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1399–1408.

[31] L. Liao, J. Xiao, Z. Wang, C.-W. Lin, and S. Satoh, "Image inpainting guided by coherence priors of semantics and textures," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 6539–6548.

[32] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro, "Image inpainting for irregular holes using partial convolutions," in *ECCV*, 2018, pp. 85–100.

[33] J. Li, N. Wang, L. Zhang, B. Du, and D. Tao, "Recurrent feature reasoning for image inpainting," in *CVPR*, 2020, pp. 7760–7768.

[34] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[35] L. Zhao, Q. Mo, S. Lin, Z. Wang, Z. Zuo, H. Chen, W. Xing, and D. Lu, "Uctgan: Diverse image inpainting based on unsupervised cross-space translation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5741–5750.

[36] H. Liu, Z. Wan, W. Huang, Y. Song, X. Han, and J. Liao, "Pd-gan: Probabilistic diverse gan for image inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9371–9381.

[37] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[38] Z. Wan, J. Zhang, D. Chen, and J. Liao, "High-fidelity pluralistic image completion with transformers," *arXiv preprint arXiv:2103.14031*, 2021.

[39] T. R. Shaham, T. Dekel, and T. Michaeli, "Singan: Learning a generative model from a single natural image," in *ICCV*, 2019, pp. 4570–4580.

[40] R. Zhang, W. Quan, B. Wu, Z. Li, and D.-M. Yan, "Pixel-wise dense detector for image inpainting," in *Computer Graphics Forum*, vol. 39, no. 7. Wiley Online Library, 2020, pp. 471–482.

[41] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, "Semantic image synthesis with spatially-adaptive normalization," in *CVPR*, 2019, pp. 2337–2346.

[42] I. Bello, "Lambdanetworks: Modeling long-range interactions without attention," *arXiv preprint arXiv:2102.08602*, 2021.

[43] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1874–1883.

[44] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," *arXiv preprint arXiv:1508.06576*, 2015.

[45] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 6, pp. 1452–1464, 2017.

[46] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. Efros, "What makes paris look like paris?" 2012.

[47] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro, "Nvidia irregular mask dataset," in *https://nv-adlr.github.io/publication/partialconv-inpainting*, 2018.

[48] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Free-form image inpainting with gated convolution," in *ICCV*, 2019, pp. 4471–4480.

[49] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[50] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," *arXiv preprint arXiv:1802.05957*, 2018.

[51] Y. Jo and J. Park, "Sc-fegan: Face editing generative adversarial network with user's sketch and color," in *ICCV*, 2019, pp. 1745–1753.

[52] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," *arXiv preprint arXiv:1805.08318*, 2018.

[53] R. Liu, J. Lehman, P. Molino, F. P. Such, E. Frank, A. Sergeev, and J. Yosinski, “An intriguing failing of convolutional neural networks and the coordconv solution,” in *Advances in Neural Information Processing Systems*, 2018, pp. 9605–9616.

[54] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” in *NIPS*, 2017, pp. 6626–6637.

[55] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2818–2826.

[56] M. Lucic, K. Kurach, M. Michalski, S. Gelly, and O. Bousquet, “Are gans created equal? a large-scale study,” *arXiv preprint arXiv:1711.10337*, 2017.

[57] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, “Cutmix: Regularization strategy to train strong classifiers with localizable features,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 6023–6032.

[58] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[59] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, “Improved training of wasserstein gans,” *arXiv preprint arXiv:1704.00028*, 2017.

[60] S. Caelles, A. Montes, K.-K. Maninis, Y. Chen, L. Van Gool, F. Perazzi, and J. Pont-Tuset, “The 2018 davis challenge on video object segmentation,” *arXiv preprint arXiv:1803.00557*, 2018.

Progressive Contextual Aggregation Empowered by Pixel-wise Dense Detector for Image Inpainting

Dear editors and reviewers,

We wish to thank the editors and reviewers for their careful review and comments on our manuscript "Progressive Contextual Aggregation Empowered by Pixel-wise Dense Detector for Image Inpainting" (TIP-24789-2021). To improve the quality of our manuscript, we have made significant revisions since our prior submission. Please find enclosed reply to the reviewer's comments we shall look forward to hearing from you at your earliest convenience. Thank you for your advice again.

Sincerely,

Jinwoo Kim

Woojae Kim

Heeseok Oh

Sanghoon Lee

Reviewer: 1
We'd like to thank the reviewer for their valuable remarks, which we used to upgrade the manuscript. After careful consideration, we have revised our paper accordingly. Our answers follow each italicized copy of the reviewer's comments, explaining how we have changed.

Specific Comments:

1. In RGB color space, have tried other color space, e.g. YUV?

Answer: Thanks for the constructive comment. As the reviewer indicated, we did not compare image inpainting results regarding various color spaces, because previous literatures related with this field [9], [11]–[13], [33] have only dealt with the RGB color space. To reflect the reviewer's suggestion, an additional test was conducted to verify the effectiveness of the proposed model over different color spaces. In Table VIII, quantitative scores are tabulated in accordance with various color spaces (e.g., YUV, HSV, and HED) by employing the model trained with RGB images. The scores are calculated for all images in Paris StreetView given the mask ratio of $30\% - 40\%$, and then the result shows that the best performance is achieved with the RGB color space. It is noteworthy that even though our model was trained on the RGB space, but the completion performances on the other color spaces are almost equivalent.

TABLE VIII
COMPARISON OF QUANTITATIVE SCORES ON DIFFERENT COLOR SPACES.

Color space	Mean$l_1^{\dagger}$	PSNR*	SSIM*	FID†
YUV	0.0294	26.28	0.859	8.86
HSV	0.0338	25.94	0.862	8.95
HED	0.0372	25.50	0.847	9.71
RGB	**0.0265**	**26.83**	**0.867**	**8.60**

2. How did you set the weighting parameter λ_c in Eq. (19)?

Answer: Thank you for mentioning the important question of scaling. In fact, the prior scaling of the objectives (λ_c) is necessary, not optional. But we missed out specifying the weighting parameter atthe initial submission. As the reviewer knows, such weighting parameter totally depends upon the stochastic processes based on the sampled data and their derivatives according to the objective functions, thus it is hardly determined in an analytic way. Therefore, we found the optimal weighting parameter by increasing the number discretely, i.e., $\lambda_c = \{0.01, 0.05, 0.1, 1.0\}$. Empirically, $\lambda_c = 0.1$ was the optimal value in terms of both performance and model stability. To address the reviewer's concern, we stated it clearly in Section III-B.

3. The groundtruth should be provided in Fig. 6, 7, and 8 for better visualization.

Answer: This is a constructive comment. Our concern was that showing the original image may hinder the understanding of the major contributions. By adopting a multiscale completion framework, even corrupt images with large missing holes can be covered by our generative model. That is, the images completed from large holes may be statistically different from original images. This is why we did not visualize ground-truth in the initial version. We agree with the reviewer's opinion, thus ground-truth images were added to Figs. 6-8.

Reviewer: 2
We'd like to thank the reviewer for their valuable remarks, which we used to upgrade the manuscript. After careful consideration, we have revised our paper accordingly. Our answers follow each italicized copy of the reviewer's comments, explaining how we have changed.

Specific Comments:

1. To provide a complete view of the inpainting community, please include the discussions of the following recent papers.
[1] Dynamic selection network for image inpainting, TIP 2021
[2] PD-GAN: Probabilistic Diverse GAN for Image Inpainting, CVPR 2021
[3] Image Inpainting Guided by Coherence Priors of Semantic and Textures, CVPR 2021
[4] Edge Guided Progressive Generative Image Outpainting, CVPR 2021

Answer: Thanks for the constructive comment. We sincerely reviewed all the papers above and added the discussion of them to the related work section. However, we are cautious with including the diverse image inpainting task (PD-GAN: Probabilistic Diverse GAN for Image Inpainting) to our previous categories. In our view, the mainstream of image painting seems to be divided into two ways: providing a detailed deterministic solution and providing a diverse solutions through noise sampling. Our attempt is closer to the former. To address the reviewer's concern, we newly added the detailed sub-section (Pluralistic Image Inpainting), and discussed the following papers.
[1] Pluralistic Image Completion, CVPR 2019
[2] UTCGAN: Diverse Image Inpainting based on Unsupervised Cross-Space Translation, CVPR 2020
[3] High-Fidelity Pluralistic Image Completion with Transformers, arXiv 2021
[4] In&Out: Diverse Image Outpainting via GAN Inversion, arXiv 2021

2. Please compare the method's efficiency (e.g., PARAMS, FLOPS, FPS) with the baselines. Which part is the bottleneck?

Answer: Thanks for the constructive commentary, and we agree with the reviewer's comment that adding the analysis of model efficiency might help the reader to understand our strengths. According to the reviewer's comment, we further compared the model efficiency with qualitative scores in Section IV-F. The result shows that our model is cost-efficient design rather than most state-of-the-art networks except PConv [32]. Here, PConv is a network consists of pure convolution and element-wise operations implemented as a highly optimized computation. On the other hand, recent works including our model requires much more matrix operations to capture the long-term interactions of features. Please note that our model costs about half FLOPs of the attention-based models, but achieves superior qualitative performance.

3. The ablation studies mainly focus on the model design. Please conduct ablation studies on the loss functions. In fact, there are lots of loss functions with weighting coefficients that need to be tuned, which degrades the model's generality.

Answer: Thanks for making a careful reading of our paper. With the reviewers' constructive comments, we are confident that our paper can be enriched experimentally. In Section IV-F, we evaluated the performance of loss function by studying various combinations. Specifically, our final object function can be divided into $L_{adv}(G, D)$ and $L_{app}(G)$ as represented in Eq. (12). Following the reviewer's comment, we conducted ablation studies to find the best loss function setting and weighting parameters (please note reviewer1 question2).

4. I see the model trained and tested in 255×255***. Up to what scale the model can perform inpainting? In other words, can the model perform well with higher resolution images? Please conduct quantitative experiments.***

5. Including several qualitative examples on actual uses cases. For example, the authors can apply the model on COCO or VOS dataset to erase the objects. In this way, we can compare the methods with other baselines qualitatively at least.

Answer: Here, we reply both comments 4 and 5. Thank you for these valuable comments. As the reviewer indicated, it is necessary to validate what scale the model can perform inpainting. Toward this, we visualize additional examples in Section IV-F. To be honest, the proposed model showed the competitive performance up to 512^2 resolution. However, our model was insufficient to cover HD or UHD size. In our view, the main issues for completion of higher resolution are that the memory usage becomes intractable and long-term interaction in the feature space deteriorates rapidly when the input size is up to $8K$. Now we are planning the next study on high-resolution image inpainting, and we hope that the qualified results will be achieved and the detailed analysis can be conducted in the next study. Furthermore, to visualize actual uses cases as the reviewer commented, we utilize DAVIS dataset which contains high-quality videos with object-like mask. Please refer newly added Fig. 13.

Reviewer: 3

We'd like to thank the reviewer for their valuable remarks, which we used to upgrade the manuscript. After careful consideration, we have revised our paper accordingly. Our answers follow each italicized copy of the reviewer's comments, explaining how we have changed.

Specific Comments:

1. The idea of "Pixel-wise Dense Detector" has been discussed and proposed in earlier papers. While I did not track whether they have been accepted to any venues, they have discussed very similar ideas as the Pixel-wise Dense Detector as presented in this paper. In the first paper [1], its name is "Pixel-wise Dense Detector for Image Inpainting", therefore, I believe these papers need to be carefully discussed.

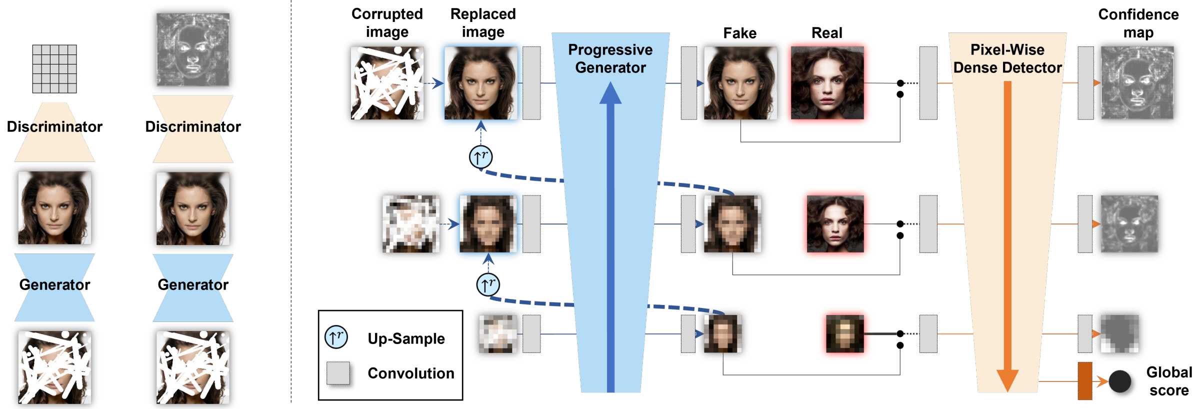
[1] https://arxiv.org/abs/2011.02293

[2] https://arxiv.org/abs/2104.01431

Answer: Thank you for this valuable comments. After sincerely reviewed the mentioned two papers, we realized the seriousness of the problem. To be honest, we did not be aware of the two papers when we were writing our paper. We were also embarrassed when we encountered the title of the first paper which is exactly the same as our pixel-wise dense detector. However, after careful review, we are convinced that our proposed approach is completely different from the above two papers (the detailed are below). Nevertheless, since the title may lead to a major misunderstanding in the future, we would like to change the title to *"Progressive Contextual Aggregation Empowered by Pixel-wise Adversarial Confidence Scoring"* if the reviewer agrees.

One of the major remaining challenges in image synthesis tasks is the ability to recover globally and locally consistent images with object shapes and textures indistinguishable from ground-truth images. To address this issue, it is a well-known approach to provide detailed per-pixel feedback to the generator while maintaining the global consistency of synthesized images, by providing the global image feedback as well. Please note "A U-Net Based Discriminator for Generative Adversarial Networks". In this paper, the authors proposed U-Net based discriminator which allows providing the detailed per-pixel feedback to the generator with the modified adversarial loss. In our view, the commented papers might be inspired by this paper. To reflect the reviewer's suggestion, we carefully discuss the differentiations between the model in Section II.

Furthermore, the proposed architecture and the models of the above two papers were re-drawn to help easy understanding. As shown below Figures (a) and (b), the studies noted by the reviewer assume that a conventional encoder-decoder structure performs both roles: high-level recognition and low-level pixel synthesis. However, this is still insufficient to fill the extremely large hole because the correlation between neighboring pixels is weakened in the *feature level*. To this end, most SOTAs [15], [16], [33] adopt an alternative scheme that fills the hole from the boundary to the center *recursively*. However, this recursive process easily suffers structural discontinuity problems since it directly maps the extracted feature map into the RGB space repeatedly. Therefore, we adopt the progressive hole-filling strategy which fills the region starting from the low- to the high-resolution regarding both *image and feature levels* simultaneously, as shown in below Figure (c). Therefore, using a multi-scale framework as an advantage, the proposed discriminator further reinforces the potential quality of synthesis by passing gradients to all the scales. This scale-wise per-pixel critic loss helps our model more focus on the newly completed region at each stage, and then the overall visual quality can be dramatically improved.



(a) Patch-wise discriminator

(b) Pixel-wise discriminator

(c) Multi-scale pixel-wise discriminator with global critic

2. Progressive learning has been commonly used in inpainting as well since the non-deep learning era. How could the progressive learning boost the results comparing to merely using PWDD in one scale? In section III-B, using PWDD in an adversarial loss is roughly mentioned, but how exactly this was used in a progressive training setting? My best guess is that you are using one adversarial loss for each scale as you mentioned $\{c_1, c_2, ..., c_N\}$**.**

Answer: Thanks for making a careful reading of our paper. This is a valuable comment. Before discussing the contribution of the pixel-wise dense detector (or pixel-wise confidence scoring) in our model, we would like to briefly remark the discriminator in the generative adversarial network (i.e., generative model). To enhance the quality of generated samples, there are a lot of works focusing on improvement of the discriminator by exploiting multiple and multi-resolution discriminators. As the reviewer knows, PG-GAN is a representative study employing multi-resolution discriminators which proposes a curriculum learning strategy to gradually increase the resolution of the generated images. With a success of this strategy, many SOTA methods including StyleGAN, SinGAN, and HRFC had adopted multi-resolution discriminators and had achieved superior performance.

Much like PG-GAN, MSG-GAN notes that multi-scale gradients account for a remarkable improvements to generate perceptual images. Unlike PG-GAN works, MSG-GAN allows the discriminator to look at not only the final output (highest resolution) of the generator, but also at the outputs of the intermediate resolutions. As a result, the discriminator becomes a function of multiple scale outputs and importantly, *passes gradients* to all the scales to the generator simultaneously. There are several advantages to adopting this strategy. Compared to multi-resolution discriminators which cannot share information across scales, MSG-GAN allows the gradients to flow at multiple resolutions making the task being explicit. Besides, this approach is free from additional constraints (e.g., an external color consistency regularization in StackGAN). Recent SOTA studies demonstrate that adopting a multi-scale feedback framework is more powerful to produce visually plausible results in comparison to utilizing a single-scale discriminator in the image synthesis fields (StyleGAN2, AnycostGAN and TFill).

However, in our view, a discriminator being a Bernoulli classifier is insufficient to identify the most discriminative difference between real and synthetic images. Since the previous discriminators criticize the generator using a single average score over holistic image or local patch, they are often biased to understand the global context which are not incentivized to enforce local consistency in the predicted image. That is, it is easy to lose local details when an environment is non-stationary that the synthesizing process varies randomly through training. To mitigate this problem, we propose an alternative discriminator architecture (PWDD), which provides both per-pixel (*across all the resolution*) and global (*global score*) decision over images to generator (please check above Figure (c)).

As the reviewer indicated, our generator is trained by their corresponding per-pixel feedback using a single adversarial loss for each scale ($\{c_1, c_2, ..., c_N\}$). Empowered by the per-pixel response of the PWDD, we further propose a consistency regularization, penalizing inconsistent prediction (please refer to Eq. (18)). By taking a role of both classifier and segmenter to PWDD, the completed results are boosted to preserve globally and locally coherent texture components than other benchmarks. Our contribution is that the proposed model not only passes the gradient flow to all scales through backpropagation but also scores the confidence by identifying whether the pixel is masked or not. By this, the discriminator is encouraged to maintain a more powerful representation. To address the reviewer's concern, we conducted additional rigorous simulations to demonstrate the effectiveness of the PWDD (or PWACS). First, we additionally implemented a single-scale discriminator with pixel-wise critics following the previous work, SingleScale. Specifically, only the final resolution image is passed through the discriminator. As shown in Table IX, PWDD with consistency shows superior performance for SSIM and FID on the Paris StreetView and CelebA-HQ datasets.

TABLE IX
COMPARISION OF QUANTITATIVE SCORES ON DIFFERENT DISCRIMINATORS.

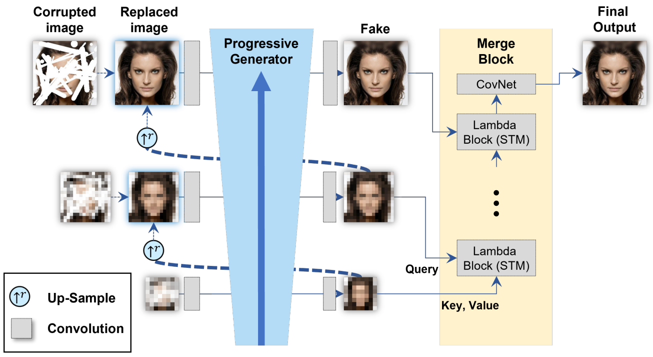
Dataset		Paris StreetView		CelebA-HQ	
Mask Ratio		30-40%	50-60%	30-40%	50-60%
SingleScale	SSIM*	0.837	0.633	0.890	0.794
	FID†	9.17	16,58	9.34	15.81
PWDD	SSIM*	0.850	0.643	0.907	0.805
	FID†	8.77	15.31	8.84	15.26
PWDD (+consistency)	SSIM*	**0.854**	**0.650**	**0.929**	**0.813**
	FID†	**8.73**	**15.25**	**8.82**	**15.18**

3. The lambda net is a great paper and it's great to see the inpainting work could leverage the lambda net. However, the contribution of the merge block is not clear given the experimental results. Even though I've zoomed hard to check out Figure 9, the differences are minor between PWDD and PWDD (+consistency). I understand that metric-wise, the merge net gives a slightly better score, however, what are those structures that could be corrected through this merge block? if not, then it is confusing to claim this merge block as one to address the structural discontinuity issues. Becauses at each resolution, an input of that resolution (downsampled from the original input) is fed into the network, so this in not a strict recursive setting. It might be possible that the structural discontinuity is not severe in this case.

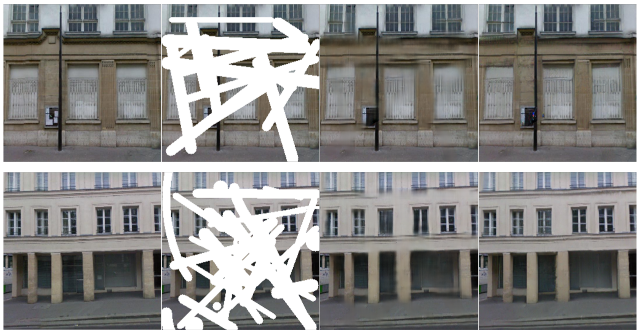
Answer: Thanks for the valuable comment. Above all, Fig. 9 in the draft is not an ablation study for a merge block but an ablation for PWDD (we have already discussed it at above comment), and the mentioned visualization is perhaps Fig. 12 which shows the effectiveness of the merge block.

As the reviewer mentioned, the lambda net is a great paper. This paper presents a computational efficient method to reflect long-range interaction. Here, the lambda layer drops nonlinearity from the original attention operation and makes the matrix multiplication be independent to the context. Hereby, the model is able to avoid expensive computation and burden to store the large attention maps. By leveraging the lambda layer, our key contribution is the propagation of the long-range interaction between adjacent resolutions. For better understanding, the proposed generator consisting of two independent networks has been redrawn below. The proposed framework is a kind of coarse-to-fine refinement approach.

Specifically, the progressive generator initially fills the hole starting from the lowest resolution to the highest resolution. After that, the merge block blends the multi-resolution completed images to construct the final output. Unlike the lambda layer which extracts query, key and value from the same context (like self-attention), our STM is utilized at each different resolution as query, key, and value to enable scale-to-scale information transfer as shown in right side of the below figure. By utilizing the proposed merge-block, a series of lambda layers can be improved in terms of 1) exploiting a multi-scale architecture to reduce computational cost for generation of high-resolution images, and 2) sequentially conveying coarse features (from lower resolution) to finer level (i.e., to adjacent higher resolution) where each block learns the region affinity between neighboring resolutions. Therefore, our merge block is robust in preserving structure continuity, as it gradually refines the global structure from the lowest resolution. As tabulated in Table VI, by adopting the merge block, we achieved a 5.59% gain of SSIM score compared to the other models using only convolutional filters on the Paris StreetView dataset. We visualize additional results below. As shown in figure, without the merge block, the resulted images degraded by some boundary artifacts. This is because the inconsistent feature maps across the different resolutions, and which might lead shadow-like artifacts. After the feature merging process representing the feature interaction between adjacent resolutions, the proposed model is able to refine the structure discontinuity with having clear textures even in cases of a large hole.



Progressive Generator and Merge Block



(a) Original
(b) Input
(c) Without merge block
(d) With merge block