# Image Multi-Inpainting via Progressive Generative Adversarial Networks

Jiayin Cai Changlin Li Xin Tao Yu-Wing Tai
Kuaishou Technology
{caijiayin, lichanglin, taoxin, daiyurong}@kuaishou.com

## Abstract

*Image inpainting aims to inpaint missing pixels of an image naturally and realistically. Previous deep learning approaches typically require specific design for different types of masks and cannot generalize well to multiple inpainting scenarios simultaneously. Thus on top of most common stroke-type mask approaches, we in this paper propose a unified framework to handle multiple types of masks simultaneously (*e.g. *strokes, object shapes, extrapolation, dense and periodic grids* et al*). We address this problem by proposing a progressive learning scheme to an Semantic Aware Generative Adversarial Network (SA-PatchGAN). Specifically, the overall training proceeds in multiple stages with different type of mask inputs, so that the model can gradually generate an output image from coarse to fine with mask independent property. In our experiments, we show that this strategy yields a large performance gain compared to the single-scale learning methods. We also introduce additional semantic conditioning to the discriminator which encourage high quality local style statistics, and show that this approach is effective on a wider scenario/tasks and could better adapt to various types of mask. Our method produces promising results on various mask types using one single model.*

## 1. Introduction

Image completion (also known as image inpainting) is a very useful editing tool to remove unwanted objects or to fill missing pixels based on surrounding context. Although this task has been studied for more than a decade, it remains an active computer vision research area due to its highly ill-posed nature, and the recent deep learning methods bring semantics into this task.

In order to improve performance, previous methods usually reduce the difficulty by solving a specific subset of inpainting problem. For example, they usually train on street view images, faces or paintings separately, to make networks focus on certain scenarios. Also, researchers designed different network structures to handle different mask types. Here we classify these mask types into 3 classes: a) **inpainting**: stroke, object or regular shapes b) **outpainting**: one or multiple image borders c) **interpolation**: dense and periodical missing pixels. We list seven types of mask into these 3 classes as shown in Fig. 1. In fact, they indeed require special design choices. Inpainting task aims to remove objects or scratches and fill with natural background patterns. Previous methods [18, 27, 30, 31] focus on establish correspondences between background the missing areas. Outpainting task expects to propagate and predict beyond image borders. Therefore, current approaches [22, 28] exploit large receptive fields to better understand context. Interpolation task focus more high quality local prediction.

We in this paper propose a new framework to solve various mask types in one single model. As mentioned above, the proposed network requires a large receptive field to better understand global structures, while it should also be able to pay enough attention on pixel-level fine details. It is excepted to find the best match from known pixels to recover and approximate missing ones, as well as to make good predictions beyond image borders.

In summary, we propose a progressive coarse to fine Generative Adversarial Network for Image Inpainting under the scenario of various kinds of masks. It applies the progressive learning to the network to make the overall training procedure more stable.This unified model can handle multiple types of masks simultaneously and mitigates the instability of training caused by extremely variance and challenging types of mask. Furthermore, our work also helps to avoid the quality degradation problem by performing the upsampling process progressively.

* Progressive learning are applied to the network to make the overall training procedure more stable.
* We use semantic information from a pretrained deep network to Enhanced semantic awareness of the discriminator in a Patch-GAN, which is a stabilization our training and improve our performance.
* Our method has achieved the 3rd place on the NTIRE [20] 2022 Inpainting public leaderboard (the 3rd on both PSNR and SSIM) and significantly outperforms existing methods on benchmark datasets.

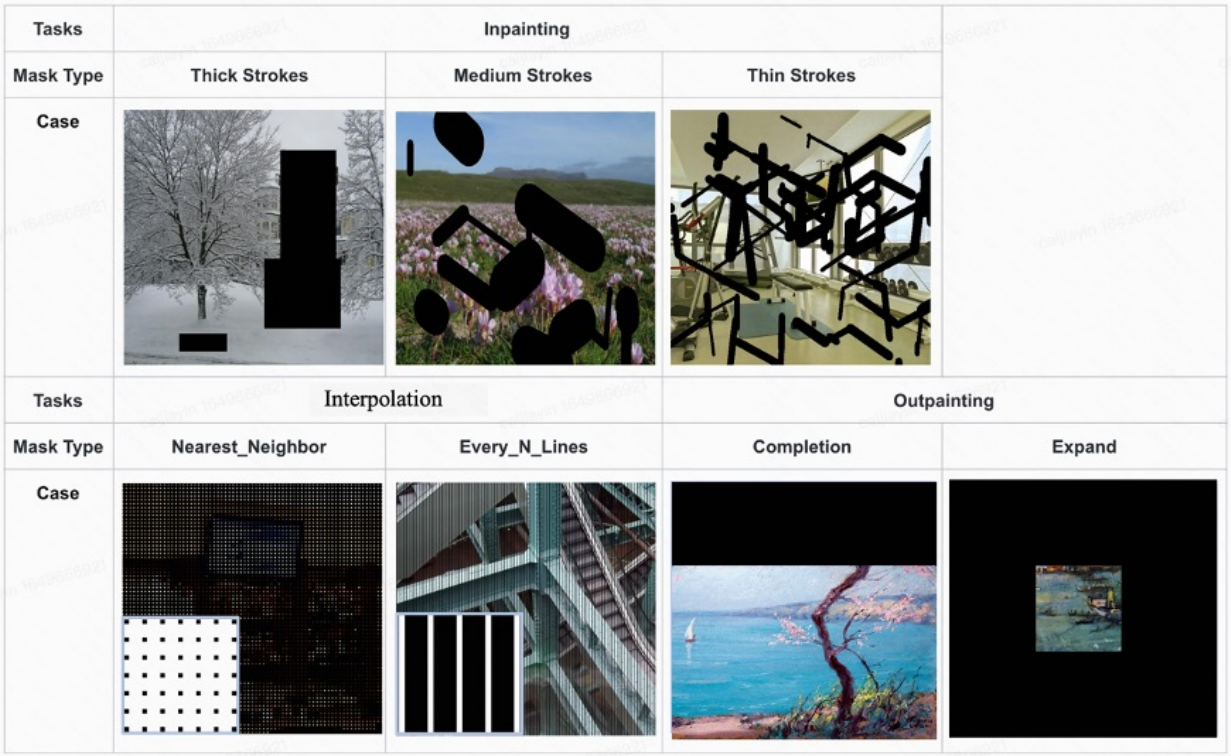


Figure 1. We list some examples of mask in 7 types and split these masks into three kinds of inpainting tasks: Inpainitng, Interpolation, Outpainting. We also plot part of the mask of Interpolation task in the blue close-up box, where white demonstrate the invalid pixels(mask area) and black demonstrate valid pixels(valid pixels). 15/16 of the pixels should be interpolate in Nearest Neighbour case and 1/4 pixels to be interpolate in Every N Lines case.

## 2. Related works

In recent years, many inpainting related works have been proposed to push the limit of this area. Here, we give a brief review of these methods most related to our method.

### 2.1. Image Inpainting

Patch-based [4] and diffusion-based [2] [12] methods were firstly proposed to handle the inpainting issues. After that, deep learning based methods are used with better performance and potentially architectures. Pathak *et al*. [18] used generative adversarial netowork for a realistic and stable image inpainting task. Later on, encoder and decoder with generative methods [30,31] are widely used to learn a latent space mapping, filling the missing holes into a feature level. Iizuka *et al*. [8] proposed a network with multi-discrimator and dilated convolutions to enhance the global consistency result. Liu *et al*. [14] utilized partical convolutions with style loss and perceputual loss in the inpainting task. Yu *et al*. [31] proposed a gated convolution with coarse-to-fine network to repair the image with irregular mask. [25] [13] [38] used multi-scale modules to obtain larger receptive field while handling large mask. And [26] estimates mask in a blind way. However, those methods can only filled the image with internal masks, leading chaotic contents and artifacts with image outpainting tasks.

### 2.2. Image Outpainting

Image extrapolation, same as image outpainting, tried to fill in the content outside the original image. [35] [24] described the outpainting task as matching and stitching problem from the training task. Thus, it led to limited results. Wang *et al*. [28] proposed a cGAN and contextual attention based network which handled a one side expand mask extrapolation problem. Teterwak *et al*. [22] used pre-trained InceptionV3 [21] output as semantic condition as the input of a discriminator for the extension result. However, those methods also suffered quality degradation along the extrapolate mask from the origin image.

### 2.3. Progressive Training

Progressive training adopted multi-stage networks to further enhance the quality of reconstructed image. Denton *et al*. [7] proposed a Laplacian pyramid to generate multi-size residual images. Then they were fed into the reconstruc-

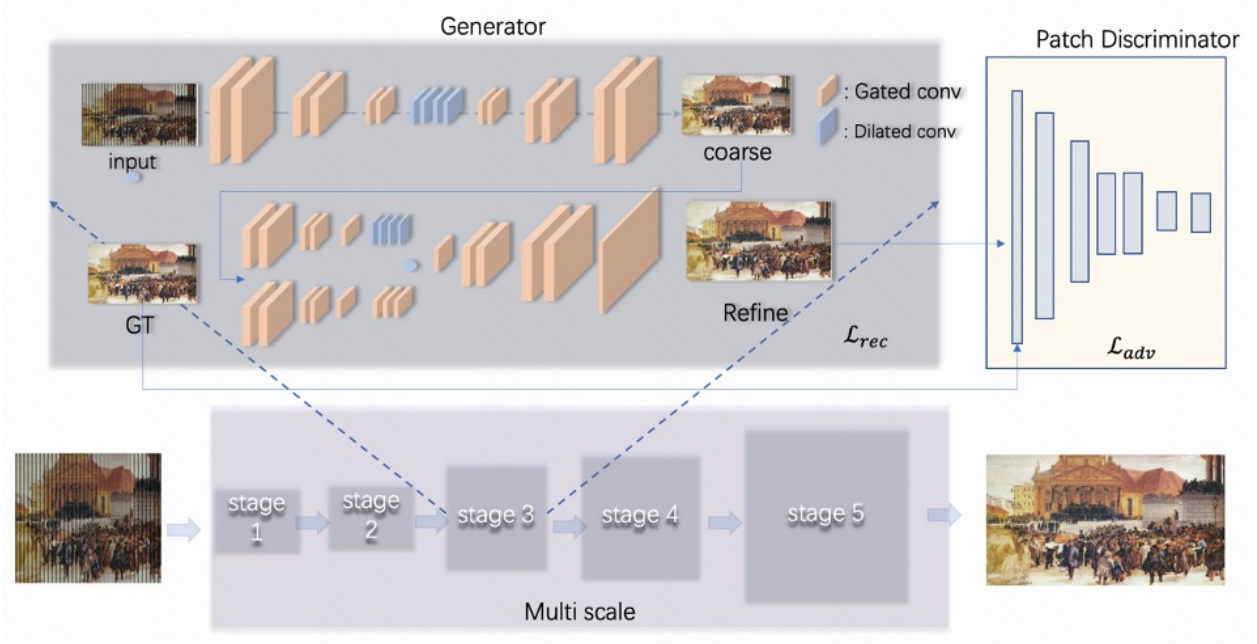


Figure 2. Total network. Illustration of progressive training for the 4 scale image inpainting task. The number that is denoted in each layer name means the image size in which the corresponding module dealing with. For instance, $\frac{1}{4}$ means the input and the output of this stage is $\frac{1}{4}$ size of the total network input.

tion phase along with upsampled original image. Karras *et al*. [10] proposed a progressive growing generative adversarial network from low spatial resolution. Lai *et al*. [11] started from low resolution image, progressively increased the resolution with residual images. The weight were shared along multi-level pyramids.

## 3. Methods

In this section, we describe the methodology of the proposed model that uses progressive learning based on Deepfillv2 [30, 31]. We select Deepfillv2 because it achieves a good balance between efficiency and performance. In Section 3.1, we first overview the model, before presenting our progressive SA-PatchGAN model in Section 3.2.

### 3.1. Coarse-to-Fine deep inpainting network

Inspired by [30, 31], we use Coarse-to-fine network architecture with gated conv as our backbone. The network architecture of our improved model is shown in Figure 2. The model is based on gated convolutions which is used to learn a dynamic feature selection mechanism for each channel at each spatial location across all layers, significantly improving the color consistency and inpainting quality of free-form masks and inputs.

### 3.2. Semantic aware patch GAN (SA-PatchGAN)

The objective of the discriminator network is to determine whether an image is generator-produced or real. In our problem setup, the concern is not just whether the output of G appears real, but also that it is a plausible extension of G's inputs.

SA-PatchGAN is proposed for the reason that multi-type masks may appear anywhere in images with any shape. Previously introduced global and local GANs [8] designed for a single rectangular mask are not applicable. To address this, we add another form of conditioning, which is a modified version of the conditional projection discriminator (cGAN) [16]. In the original cGAN paper, a one-hot class label y is passed into the discriminator in addition to the image x to be classified as real or fake. The discriminator output is:

$$\boldsymbol{D}\left(\boldsymbol{x}^{*}, \boldsymbol{y}\right)=\boldsymbol{f}_{\phi}\left(\boldsymbol{\phi}\left(\boldsymbol{x}^{*}\right)\right)+\left\langle \boldsymbol{\phi}\left(\boldsymbol{x}^{*}\right), \boldsymbol{f}_{\boldsymbol{y}}(\boldsymbol{y})\right\rangle \quad (1)$$

where $\boldsymbol{\phi}$ is a learned function mapping an image to a vector, $\boldsymbol{f}_{\phi}$ is a learned fully-connected layer that maps that vector to a scalar, $\boldsymbol{f}_{y}$ is a learned fully-connected layer mapping y to a vector of the same size as the output of $\boldsymbol{\phi}$. The cGAN paper shows that this parameterization of the GAN objective enables the model to simultaneously learn the distributions $p(x)$ and $p(y|x)$. However, sometimes class labels are not available, and we also want our conditioning vectors to contain more information than class labels would

provide. Previous work on perceptual metrics [9, 34] replaced y with the activation of a pretrained image classification network, C, when applied to x (the ground truth image). We chose to instantiate C as an InceptionV3 [21] network trained on ImageNet [6] with the final softmax removed. We found that it helps to normalize these activation by subtracting the mean activation over the dataset and then dividing the result by its l2 norm. We change equ 1 to add semantic condition:

$$\boldsymbol{D}\left(\boldsymbol{x}^{*}, \boldsymbol{M}, \boldsymbol{x}\right)=\boldsymbol{f}_{\phi}\left(\boldsymbol{\phi}\left(\boldsymbol{x}^{*}, \boldsymbol{M}\right)\right)+\left\langle\boldsymbol{\phi}\left(\boldsymbol{x}^{*}, \boldsymbol{M}\right), \boldsymbol{f}_{\boldsymbol{C}}(\boldsymbol{C}(\boldsymbol{x}))\right\rangle \tag{2}$$

Where $\boldsymbol{M}$ is the input mask. The architecture of $\boldsymbol{\phi}$ is based on [5, 31] which consists of six strided convolutional layers, followed by a fully connected layer. The output dimensions of $\boldsymbol{\phi}$ and $\boldsymbol{f}_{\boldsymbol{C}}$ are both 256.

The total loss is the weighted summation of L1 loss, adversarial loss, perceptual loss, and style loss:

$$\mathcal{L}_{\text{total}}=\lambda_{rec} \cdot \mathcal{L}_{\text{rec}}+\lambda_{adv} \cdot \mathcal{L}_{\text{adv}}+\lambda_{per} \cdot \mathcal{L}_{\text{per}}+\lambda_{sty} \cdot \mathcal{L}_{\text{sty}} \tag{3}$$

Where the reconstruction loss: $\mathcal{L}_{rec}=\|\boldsymbol{x}-\boldsymbol{G}(\boldsymbol{z}, \boldsymbol{M})\|_{1}$.

### 3.3. Progressive learning Method

We progressively apply our backbone in different scales at different image size in a coarse-to-fine manner. We use multi scales strategy in our method. Firstly, we set the number of stages as three in the 1/4, 1/2, 1/1 scale inpainting task. That is, in each stage, the model performs $\frac{1}{4}$x $\rightarrow \frac{1}{4}$x , $\frac{1}{2}$x $\rightarrow \frac{1}{2}$ x, 1x $\rightarrow$ 1x inpainting tasks sequentially.

The training starts from stage one, which produces the $\frac{1}{4}$x scale image from the first stage. After the end of the first stage, we upsample it to $\frac{1}{2}$x scale and combine the it with background pixels from $\frac{1}{2}$x scale input as the input of stage 2. We freeze the stage 1 parameters when train stage 2. When we train stage3the procedure is the same with stage2.

We also found that adding two extra scale $\frac{\sqrt{2}}{4}$x and $\frac{\sqrt{2}}{2}$ x after stage 1 and stage 2 could improve model performance.

We proposed to progressive learning scheme [10] to effectively reconstruct uncompleted images. The key concept of the methodology is similar to that of Karras*et al*. [10], but we adapt this scheme for our multi-painting task as shown in Figure 2

### 3.4. Ensembles and fusion strategies

Using ensemble strategies on different mask can provide performance gain. However, we did not contain any ensemble strategies, in order to demonstrate the robustness and stability of our method on different type of mask and different kinds of tasks. We are able to handle various of inpainting / outpainting / interpolation tasks with a single model.

## 4. Experiments

Our proposed solutions is robust and stable dealing with diverse type of mask and multi-inpainting tasks. We are able to handle various of inpainting / outpainting / interpolation tasks with a single model. We didn't use ensemble strategies on different mask although it can provide performance gain. We demonstrate that for multi-inpainting task, our mask agnostic network design is capable of producing high-quality results with the best perceptual quality with respect to the ground truth.

### 4.1. Data preparation

**Masks** In addition to the typical strokes, in this paper, we aim at more generalizable solutions. We use seven types of masks in this paper as shown in Fig 1, and separate these seven types of masks into three classes representing three tasks: Thick, Medium and Thin Strokes represent traditional inpainting task. Nearest Neighbor and Every_N_Lines represent Image interpolation. Completion and Expand represent Outpainting task.

**Datasets** Following a common practice in Image Inpainting methods, we use three popular datasets for our challenge: FFHQ, Places, and ImageNet. Additionally, to explore a new benchmark, we also use the WikiArt dataset to tackle inpainting towards art creation.

### 4.2. Inplementation and training details

Our implementation is based on Pytorch with Nvidia Tesla V100 32GB GPU. The networks are trained with Adam optimizer. The method trained for about 15 hours on one V100 GPU. All images are random cropped and resized to 512×512 pixels. The batch size is set to 20. The training process is fast and converges in about 150K iterations: 15 hours on one GPU. The testing process is also fast which cost 0.38s to infer per image in four datasets (Places, ImageNet, FFHQ, WikiArt) on average. No human effort are required for implementation in training or validation, and the performance is stability during training and testing.

**Training description** We train the network using a joint loss consisting of a reconstruction l1 loss, adversarial loss, perceptual loss, and style loss. We use multi scales strategy in our method, the scale of each stage is 0.25, $\frac{\sqrt{2}}{4}$ ,0.5, $\frac{\sqrt{2}}{2}$, 1 respectively.

For Places2 dataset, our model is trained on places2 training set from scratch, and test the model on Places2 validation and test set. For ImageNet/ WikiArt/ FFHQ datasets, we initialise the model using parameters pretrained on Places2 dataset, then finetune the model on the training set of ImageNet/ WikiArt/ FFHQ datasets respectively.

**Testing description** For testing Places dataset, we use model trained on Places2 training set. For testing ImageNet

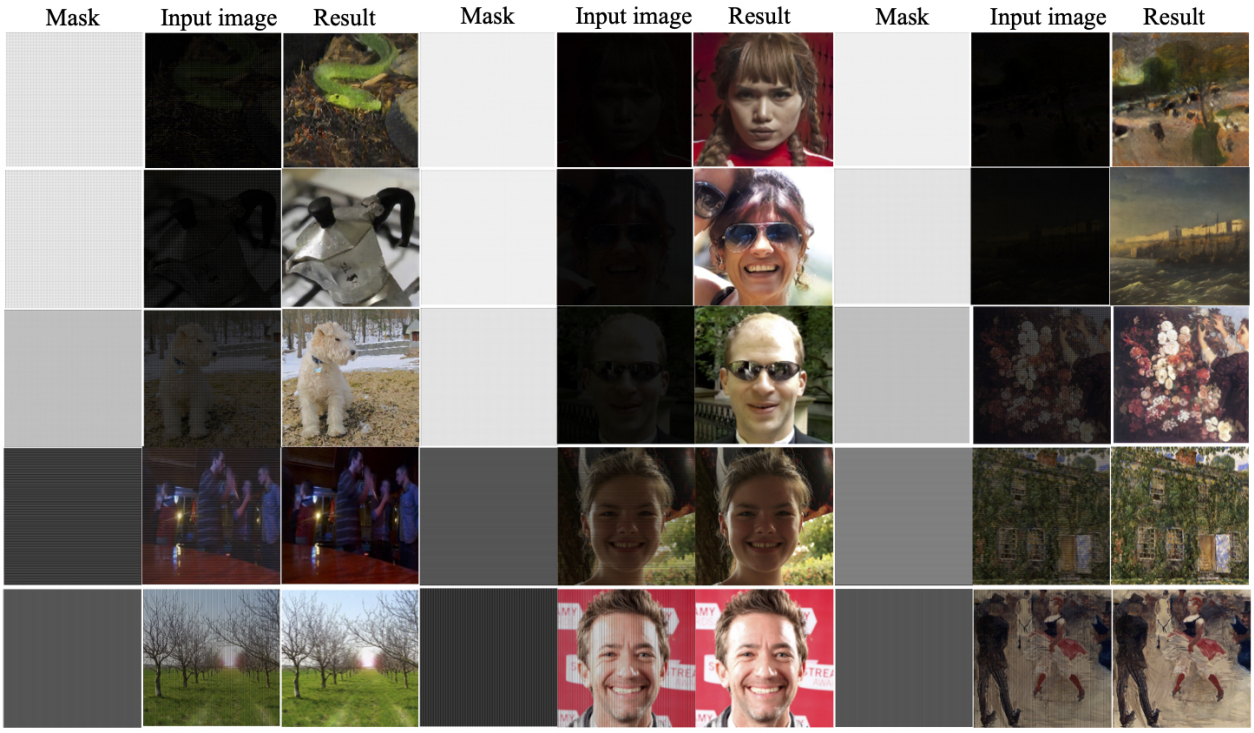


Figure 3. Interpolation-inpainting Task for Every N Line and Nearest Neighbor on four datasets.

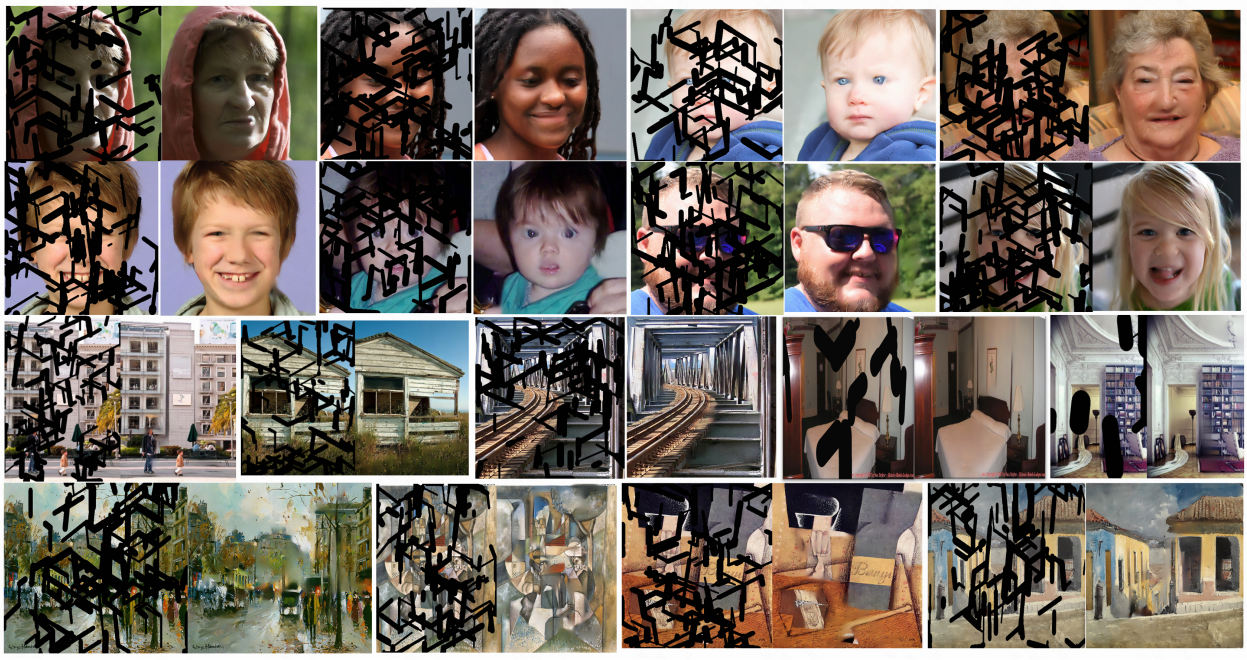

Figure 4. Conventional inpainting Task for stroke masks on four datasets.

dataset, we use the model trained on ImageNet training set which finetuned from the Places2 pretrained model. The WikiArt and FFHQ dataset is the same strategy with ImageNet. Since the first stage in our model is 1/4 x scale of the model and the image size should be divided evenly by 8, the input size(height and width) for the whole image divisible

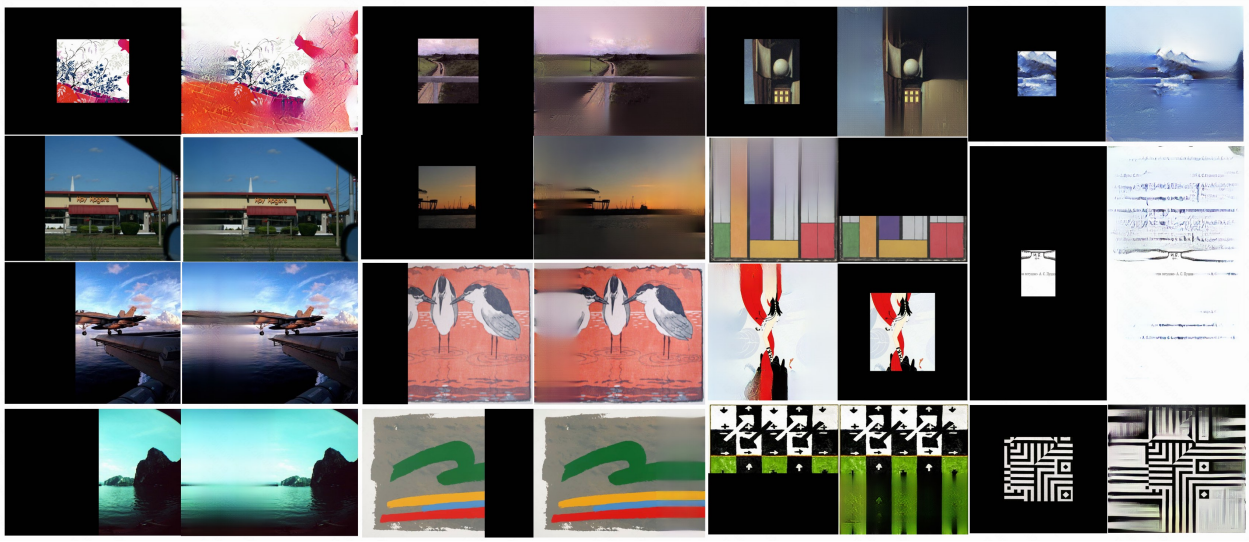

Figure 5. Out-painting Task for Completion and Expend masks on four datasets.

| | PSNR ↑ | | SSIM ↑ | | LPIPS ↓ | | FID ↓ |
|---|---|---|---|---|---|---|---|
| Datasets | mean | std | mean | std | mean | std | mean |
| FFHQ | 25.06 | 8.669 | 0.838 | 0.147 | 0.239 | 0.173 | 21.345 |
| Places | 23.41 | 7.892 | 0.787 | 0.195 | 0.255 | 0.193 | 18.334 |
| ImageNet | 23.804 | 8.781 | 0.776 | 0.221 | 0.249 | 0.213 | 18.854 |
| WikiArt | 23.142 | 7.305 | 0.759 | 0.204 | 0.276 | 0.185 | 26.395 |

Table 1. Quantitative results of our proposed model on all of the datasets for four datasets.

by 32. The test image are reshape to an integral number multiple of 32.

**Execution time** The execution time is evaluated on a machine with one Nvidia Tesla V100 GPUs. The speed of our model is 2.5 frame per second (FPS).

### 4.3. Comparison with state-of-the-art methods

Our model performs well on three kinds of masks (conventional inpainting mask, outpainting mask and interpolation mask), which is robustness for different mask types. We compare our methods with SOTA outpainting tasks on Places dataset in table 4. We provide We provide FID scores for FID correlates with perceptual quality best. We also compared our methods with SOTA inpainting tasks on Places2 dataset in table 5. Our results are tested from 1000 test images from Places 2 with 7 types of masks which is from Colab from NTIRE 2022 image inpainting challenge [19].

### 4.4. Pros and cons for each type of mask

**Mask of Strokes** Inpainting with stroke mask is the conventional image inpainting setting. Our model could handle this circumstance well and the reconstruct region match the valid area at the textural, structural and semantic levels. The result of our model on four datasets (FFHQ, Places, ImageNet, Wikiart) are shown in Fig 4. We also found the mask attributes substantially impact the difficulty of conventional image painting(with strokes). in particular, widely distributed free-form masks lead to better performance which are often non-contiguous and non-convex, although some of them has over $50\%$ invalid pixels. So when invalid percentage is the same, thin strokes performs better than thick strokes. However, in addition to the typical strokes, we aim at more generalizable solutions dealing with multi-inpainting task, namely various types of masks simultaneously(for instance, strokes, half completion, nearest neighbor up-sampling.)

**Mask of completion and expand** Completion Mask in out-painting tasks is a challenge task since it aims at creating new contents according to the semantic information rather than filling in partial regions guided by available surrounding pixels. So out-painting task requires a substantial understanding of scenes and semantic information of the input image. On the other side, out-painting can be achieved in more diverse ways since the problem is less constrained by the surrounding pixels. So traditional quantitative result used in inpainting model (such like PSNR, SSIM) is not in

| | | Strokes | | | Interpolation | | Completion | |
|---|---|---|---|---|---|---|---|---|
| | Mean | Thick | Medium | Thin | Every N Lines | Nearest Neighbor | Completion | Expand |
| PSNR ↑ | 22.89 | 23.330 | 23.992 | 27.284 | 31.772 | 24.873 | 16.130 | 12.877 |
| SSIM ↑ | 0.785 | 0.866 | 0.879 | 0.910 | 0.940 | 0.757 | 0.688 | 0.454 |
| LPIPS ↓ | 0.248 | 0.158 | 0.134 | 0.112 | 0.147 | 0.347 | 0.522 | 0.313 |
| FID ↓ | 20.314 | 15.213 | 12.341 | 10.214 | 14.906 | 21.924 | 37.484 | 30.098 |

Table 2. Quantitative results of our proposed model on Partial of Places datasets with different mask types. Our Partial test set contains $1,000 \times 7 \times 4$ images for seven type of mask on four dataset.


Figure 6. Quantitative results of our model on Interpolation task compared to Boundless [22] and CV2 Bicubic interpolation.

direct proportion to visual quality. So we use FID score in Table 4 since FID correlates with perceptual quality best. From qualitative and quantitative results, our model performs well on mask of completion and expand by appearing a plausible extension, demonstrate that we could not only deal with context and texture but also using semantic info to creating new contents. We also tried other outpainting methods [22, 28, 29] which GAN models which formulate the problem as an image-to-image task. We retrained these methods specifically for these three kinds of tasks. However, [28, 29] can not adapt to interpolation-inpainting task and tend to produce blur results, which is not a generalizable solutions for multi-inpainting task. [22] tends to create overly-smoothed results with raindrop-shaped artifacts.

**Nearest neighbor up-sampling Mask.** Nearest neighbor up-sampling Mask is the easiest task among these mask types. We found that our model could perform well on mask with $V : M = 1 : 4, 1 : 9, 1 : 16$. Where $V$ and $P$ and number of valid pixels before and after inapinting. We found that Deep network [30, 31] performs well on Interpolation task since it is good at deal with context and texture. Although the missing pixels don't have a larger amount of available surrounding pixels serving as the boundary conditions like Strokes Mask inpainting, the sparse valid pixels serves as boundary conditions and can also provide crucial guidance for interpolational-inpainting. We found that our model performs better on Interpolation task and good at deal with context and texture since the missing pixels have a few available surrounding pixels, serving as the boundary conditions and providing crucial guidance for inpainting.

### 4.5. Ablation Study

We perform ablation study on the effect of multi stage and semantic aware PatchGAN. We list out results in Table 3. $B$ means backbone of our model, while $B_3 s$ demonstrate 3 stage multi scale progressive learning. and $B_5 s$ demonstrate 5 stage backbone (adding two extra scale $\frac{\sqrt{2}}{4}$x and $\frac{\sqrt{2}}{2}$ x after stage 1 and stage 2). $SA$ means add semantic aware Patch GAN discriminator in the model. Multi scale experiences substantial gains with the origin three stage, and performs better with increased two extra stage. SA helps to capture semantic information and helps to better explore the global structures.

### 4.6. Results of the comparison to other approaches

For Nearest neighbour and Enery-N-Line masks, we compare our method with cv2.resize method, we extract the valid pixels and resize them to the same size as input image using cv2.resize with INTER_CUBIC interpolation. As shown in Figure 6. Our methods performs better compared with cv2 on both qualitative results and quantitative results. However, Bicubic method can not deal with outpainting task with We also compare our method with Boundless [22]. Boundless [22] tend to produce color difference results compared to input valid pixels on Interpolation task. Our method could converge easier than them and performs more stable on different types of masks.

| | | | Strokes | | | Interpolation | | Completion | |
|---|---|---|---|---|---|---|---|---|---|
| | | Mean | Thick | Medium | Thin | N Lines | Neighbor | Comp | Expand |
| whole model | PSNR↑ | **22.89** | 23.330 | 23.992 | 27.284 | 31.772 | 24.873 | 16.130 | 12.877 |
| | SSIM↑ | **0.785** | 0.866 | 0.879 | 0.910 | 0.940 | 0.757 | 0.688 | 0.454 |
| | LPIPS↓ | **0.248** | 0.158 | 0.134 | 0.112 | 0.147 | 0.347 | 0.522 | 0.313 |
| | FID↓ | **20.314** | 15.213 | 12.341 | 10.214 | 14.906 | 21.924 | 37.484 | 30.098 |
| | | Mean | Thick | Medium | Thin | N Lines | Neighbor | Comp | Expand |
| $B_{5s}$+SA | PSNR↑ | 22.01 | 22.506 | 23.505 | 27.127 | 30.739 | 22.709 | 16.280 | 13.192 |
| | SSIM↑ | 0.776 | 0.862 | 0.877 | 0.907 | 0.922 | 0.675 | 0.662 | 0.471 |
| | LPIPS↓ | 0.260 | 0.180 | 0.142 | 0.131 | 0.166 | 0.359 | 0.540 | 0.301 |
| | FID↓ | 21.613 | 17.201 | 13.251 | 12.005 | 16.211 | 23.014 | 38.129 | 31.482 |
| | | Mean | Thick | Medium | Thin | N Lines | Neighbor | Comp | Expand |
| $B_{3s}$+SA | PSNR↑ | 21.89 | 22.368 | 23.383 | 27.022 | 30.422 | 22.704 | 16.295 | 12.944 |
| | SSIM↑ | 0.774 | 0.862 | 0.876 | 0.906 | 0.917 | 0.670 | 0.676 | 0.397 |
| | LPIPS↓ | 0.263 | 0.184 | 0.145 | 0.133 | 0.171 | 0.361 | 0.520 | 0.330 |
| | FID↓ | 21.939 | 17.512 | 13.492 | 12.395 | 16.592 | 23.288 | 38.529 | 31.771 |
| | | Mean | Thick | Medium | Thin | N Lines | Neighbor | Comp | Expand |
| B+SA | PSNR↑ | 21.66 | 23.508 | 24.066 | 26.527 | 28.241 | 19.778 | 15.363 | 12.126 |
| | SSIM↑ | 0.774 | 0.856 | 0.866 | 0.891 | 0.876 | 0.498 | 0.702 | 0.488 |
| | LPIPS↓ | 0.289 | 0.199 | 0.166 | 0.147 | 0.179 | 0.371 | 0.577 | 0.383 |
| | FID↓ | 22.530 | 17.892 | 13.625 | 12.766 | 16.983 | 23.504 | 39.733 | 33.207 |
| | | Mean | Thick | Medium | Thin | N Lines | Neighbor | Comp | Expand |
| B | PSNR↑ | 20.9 | 23.196 | 23.623 | 25.489 | 27.227 | 17.627 | 15.327 | 12.056 |
| | SSIM↑ | 0.670 | 0.855 | 0.858 | 0.875 | 0.841 | 0.395 | 0.703 | 0.487 |
| | LPIPS↓ | 0.302 | 0.204 | 0.173 | 0.161 | 0.199 | 0.394 | 0.588 | 0.401 |
| | FID↓ | 22.912 | 18.533 | 13.935 | 12.805 | 17.029 | 23.881 | 40.428 | 33.771 |

Table 3. Ablation study on Partial of Places test set. The test set contains 1,000 images for each type of mask for each dataset. $B$ means our backbone. $B_{3s}$ demonstrate 3 stage multi-scale progressive learning. and $B_{5s}$ demonstrate 5 stage backbone (adding two extra scale $\frac{\sqrt{2}}{4}$x and $\frac{\sqrt{2}}{2}$ x after stage 1 and stage 2). $SA$ means add semantic aware Patch GAN discriminator in the model.

| Method | FID ↓ |
|---|---|
| Boundless [22] | 35.02 |
| NS-outpaint [29] | 50.68 |
| DeepFillv2 [30, 31] | 56.14 |
| Image2StyleGAN [1] | 25.36 |
| In&Out [3] | 23.57 |
| Very_Long [29] | **13.71** |
| Ours | 18.33 |

Table 4. Comparing our methods with SOTA **outpainting** tasks on Places dataset. We provide We provide FID scores for outpainting task since FID correlates with perceptual quality best.

| | PSNR ↑ | | SSIM ↑ | | FID ↓ |
|---|---|---|---|---|---|
| Method | Thin | Thick | Thin | Thick | |
| EC† [17] | 26.52 | 22.23 | 0.880 | 0.731 | 30.13 |
| GC† [31] | 26.53 | 21.19 | 0.881 | 0.729 | 30.13 |
| MEDFE† [15] | 26.47 | 22.27 | 0.877 | 0.717 | 31.40 |
| PIC† [36] | 26.10 | 21.50 | 0.865 | 0.680 | 33.47 |
| ICT† [23] | 26.6 | 23.32 | 0.880 | 0.724 | 25.42 |
| AOT-GAN [33] | 26.03 | 22.62 | 0.890 | 0.804 | **5.47** |
| BAT-Fill† [32] | 26.47 | 21.74 | 0.879 | 0.704 | 22.16 |
| pluralistic [36] | 26.47 | 21.74 | 0.879 | 0.704 | 25.42 |
| Ours | **27.28** | **23.33** | **0.910** | **0.866** | 18.33 |

Table 5. Quantitative comparison of our model with state-of-the-art conventional inpainting methods on Places2 [37] validation images (1,000) with irregular masks. † denotes the results are copy from [32]

## 5. Conclusion

In this work, we proposed a progressive cascading Semantic Aware GAN network that can perform image inpainting accurately even in an various types of tasks in complex scenario. The main idea behind our work is to apply progressive learning scheme to Semantic Aware GAN network. By using the progressive scheme, the training process becomes much easier and more stable, since the model first learns the coarse structure and gradually learns how to restore details in the later stages. Our experiment shows that employing this idea leads to better performance on various benchmark datasets compared to the non-progressive approaches. We also introduce semantic conditioning to the discriminator of the GAN which only penalizes structure at the scale of image patches, to capture local style statistics, and show that this approach is effective on a wider scenario/tasks and could better adapt to various types of mask.

## References


[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan++: How to edit the embedded images? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8296–8305, 2020. 8

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

[3] Yen-Chi Cheng, Chieh Hubert Lin, Hsin-Ying Lee, Jian Ren, Sergey Tulyakov, and Ming-Hsuan Yang. In&out: Diverse image outpainting via gan inversion. *arXiv preprint arXiv:2104.00675*, 2021. 8

[4] Soheil Darabi, Eli Shechtman, Connelly Barnes, Dan B Goldman, and Pradeep Sen. Image melding: Combining inconsistent images using patch-based synthesis. *ACM Transactions on graphics (TOG)*, 31(4):1–10, 2012. 2

[5] Ugur Demir and Gozde Unal. Patch-based image inpainting with generative adversarial networks. *arXiv preprint arXiv:1803.07422*, 2018. 4

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 4

[7] Emily L Denton, Soumith Chintala, Rob Fergus, et al. Deep generative image models using a laplacian pyramid of adversarial networks. *Advances in neural information processing systems*, 28, 2015. 2

[8] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Globally and locally consistent image completion. *ACM Transactions on Graphics (ToG)*, 36(4):1–14, 2017. 2, 3

[9] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 4

[10] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 3, 4

[11] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 624–632, 2017. 3

[12] Anat Levin, Assaf Zomet, and Yair Weiss. Learning how to inpaint from global image statistics. In *ICCV*, volume 1, pages 305–312, 2003. 2

[13] Jingyuan Li, Ning Wang, Lefei Zhang, Bo Du, and Dacheng Tao. Recurrent feature reasoning for image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7760–7768, 2020. 2

[14] Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *ECCV*, pages 85–100, 2018. 2

[15] Hongyu Liu, Bin Jiang, Yibing Song, Wei Huang, and Chao Yang. Rethinking image inpainting via a mutual encoder-decoder with feature equalizations. In *European Conference on Computer Vision*, pages 725–741. Springer, 2020. 8

[16] Takeru Miyato and Masanori Koyama. cgans with projection discriminator. *arXiv preprint arXiv:1802.05637*, 2018. 3

[17] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Z Qureshi, and Mehran Ebrahimi. Edgeconnect: Generative image inpainting with adversarial edge learning. *arXiv preprint arXiv:1901.00212*, 2019. 8

[18] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016. 1, 2

[19] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, pages 724–732, 2016. 6

[20] Andres Romero, Angela Castillo, Jose M Abril-Nova, Radu Timofte, et al. NTIRE 2022 image inpainting challenge: Report. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2022. 1

[21] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 2, 4

[22] Piotr Teterwak, Aaron Sarna, Dilip Krishnan, Aaron Maschinot, David Belanger, Ce Liu, and William T Freeman. Boundless: Generative adversarial networks for image extension. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10521–10530, 2019. 1, 2, 7, 8

[23] Ziyu Wan, Jingbo Zhang, Dongdong Chen, and Jing Liao. High-fidelity pluralistic image completion with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4692–4701, 2021. 8

[24] Miao Wang, Yu-Kun Lai, Yuan Liang, Ralph R Martin, and Shi-Min Hu. Biggerpicture: data-driven image extrapolation using graph matching. *ACM Transactions on Graphics*, 33(6), 2014. 2

[25] Ning Wang, Jingyuan Li, Lefei Zhang, and Bo Du. Musical: Multi-scale image contextual attention learning for inpainting. In *IJCAI*, pages 3748–3754, 2019. 2

[26] Yi Wang, Ying-Cong Chen, Xin Tao, and Jiaya Jia. Vcnet: A robust approach to blind image inpainting. In *European Conference on Computer Vision*, pages 752–768. Springer, 2020. 2

[27] Yi Wang, Xin Tao, Xiaojuan Qi, Xiaoyong Shen, and Jiaya Jia. Image inpainting via generative multi-column convolutional neural networks. *Advances in neural information processing systems*, 31, 2018. 1

[28] Yi Wang, Xin Tao, Xiaoyong Shen, and Jiaya Jia. Wide-context semantic image extrapolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1399–1408, 2019. 1, 2, 7

[29] Zongxin Yang, Jian Dong, Ping Liu, Yi Yang, and Shuicheng Yan. Very long natural scenery image prediction by outpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10561–10570, 2019. 7, 8

[30] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5505–5514, 2018. 1, 2, 3, 7, 8

[31] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. In *ICCV*, pages 4471–4480, 2019. 1, 2, 3, 4, 7, 8

[32] Yingchen Yu, Fangneng Zhan, Rongliang Wu, Jianxiong Pan, Kaiwen Cui, Shijian Lu, Feiying Ma, Xuansong Xie, and Chunyan Miao. Diverse image inpainting with bidirectional and autoregressive transformers. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 69–78, 2021. 8

[33] Yanhong Zeng, Jianlong Fu, Hongyang Chao, and Baining Guo. Aggregated contextual transformations for high-resolution image inpainting. *IEEE Transactions on Visualization and Computer Graphics*, 2022. 8

[34] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 4

[35] Yinda Zhang, Jianxiong Xiao, James Hays, and Ping Tan. Framebreak: Dramatic image extrapolation by guided shift-maps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1171–1178, 2013. 2

[36] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. Pluralistic image completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1438–1447, 2019. 8

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

[38] Manyu Zhu, Dongliang He, Xin Li, Chao Li, Fu Li, Xiao Liu, Errui Ding, and Zhaoxiang Zhang. Image inpainting by end-to-end cascaded refinement with mask awareness. *IEEE Transactions on Image Processing*, 30:4855–4866, 2021. 2