Article

Diversity-Generated Image Inpainting with Style Extraction

Weiwei Cai and Zhanguo Wei *

Posted Date: 3 December 2019

doi: 10.20944/preprints201912.0028.v1

Keywords: deep learning; generative adversarial networks; image inpainting; diversity inpainting

Article

Diversity-Generated Image Inpainting with Style Extraction

Weiwei Cai [1] and Zhanguo Wei [2,*]

[1] Central South University of Forestry and Technology, Changsha 410004, China
[2] Central South University of Forestry and Technology, Changsha 410004, China
* Correspondence: t20110778@csuft.edu.cn

Abstract: The latest methods based on deep learning have achieved amazing results regarding the complex work of inpainting large missing areas in an image. This type of method generally attempts to generate one single "optimal" inpainting result, ignoring many other plausible results. However, considering the uncertainty of the inpainting task, one sole result can hardly be regarded as a desired regeneration of the missing area. In view of this weakness, which is related to the design of the previous algorithms, we propose a novel deep generative model equipped with a brand new style extractor which can extract the style noise (a latent vector) from the ground truth image. Once obtained, the extracted style noise and the ground truth image are both input into the generator. We also craft a consistency loss that guides the generated image to approximate the ground truth. Meanwhile, the same extractor captures the style noise from the generated image, which is forced to approach the input noise according to the consistency loss. After iterations, our generator is able to learn the styles corresponding to multiple sets of noise. The proposed model can generate a (sufficiently large) number of inpainting results consistent with the context semantics of the image. Moreover, we check the effectiveness of our model on three databases, i.e., CelebA, Agricultural Disease, and MauFlex. Compared to state-of-the-art inpainting methods, this model is able to offer desirable inpainting results with both a better quality and higher diversity. The code and model will be made available on https://github.com/vivitsai/SEGAN.

Keywords: Deep Learning; GAN; Image Inpainting; Diversity Inpainting

1. Introduction

Image inpainting requires a computer to fill in the missing area of an image according to the information found in the image itself or the area around the image, thus creating a plausible final inpainted image. However, in cases where the missing area of an image is too large, the uncertainty of the inpainting results increase greatly. For example, when inpainting a face image, the eyes may look in different directions and there may be glasses not, etc. Although a single inpainting result may seem reasonable, it is difficult to determine whether this result meets our expectations, as it is the only option. Therefore, driven by this observation, we hope to inpaint a variety of plausible results on a single missing region, which we call image diversity inpainting (as shown in Figure 1).

Some early researches [2,3,11] attempted to carry out image inpainting using the classical texture synthesis method, that is, by sampling similar pixel blocks from the undamaged area of the image to fill the area to be completed. However, the premise of these methods is that similar patches can be sampled from the undamaged area. When the inpainted area is designed with complex nonrepetitive structures (such as faces), these methods obviously cannot work (cannot capture high-level semantics). The vigorous development of deep generating models has promoted recent related research [16,39] which encodes the image into high-dimensional hidden space, and then decodes the feature into a whole inpainted image. Unfortunately, because the receptive field of the convolutional neural network is too small to obtain or borrow the information of distant spatial locations effectively,these CNN-based approaches typically generate boundary shadows, distorted results, and blurred textures inconsistent

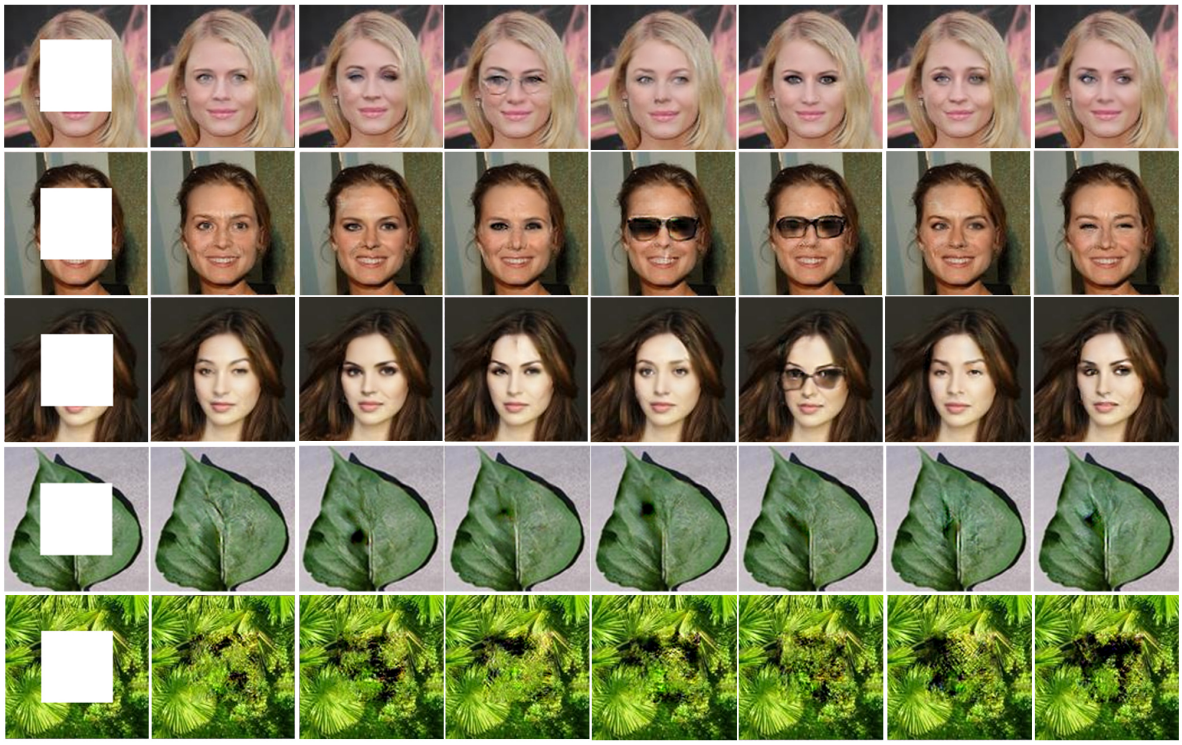


Figure 1. Examples of the inpainting results of our method on a face, leaf, and rainforest image (the missing regions are shown in white). The left is the masked input image, while the right is the diverse and plausible direct output of our trained model without any postprocessing.

with the surrounding region. Recently, some works [34,36] used spatial attention to recover the lost area using the surrounding image features as reference. These methods ensure the semantic consistency between the generated content and the context information. However, these existing methods are trying to inpaint a unique "perfect" result, but are unable to generate a variety of valuable and plausible results.

In order to obtain multiple diverse results, many methods based on CVAE [26] have been produced [14,27], but these methods are limited to specific fields, which need targeted attributes and may result in unreasonable images being inpainted.

To achieve better diversity of inpainting results, we add a new extractor in the generative adversarial network (GAN) [19], which is used to extract the style noise (a latent vector) of the ground truth image of the training set and the fake image generated by the generator. The encoder in CVAE-GAN [14] takes the extracted features of the ground truth image as the input of the generator directly. When a piece of label itself is a masked image, the number of labels matching each label in the training set is usually only one. Therefore, the results generated have very limited variations.

We proposed a novel deep generative model-based approach. In each round of iterative training, the extractor first extracts the style noise (a latent vector) of the ground truth image of the training set and inputs it to the generator together with the ground truth image. We use the consistency loss L1 to force the generated image to be as close to the ground truth image of the training set as possible; at the same time, we generate and input a random noise and masked image to the generator to get the output fake image, and use the consistency loss L1 to make the extracted style noise as close to the input noise as possible. After the iteration, the generator can learn the styles corresponding to multiple sets of input noise. We also minimize the KL(Kullback-Leibler) loss to reduce the gap between the prior distribution and the posterior distribution of the potential vectors extracted by the extractor.

We experimented on the open datasets CelebA [33], Agricultural Disease, and MauFlex [1]. Both quantitative and qualitative tests show that our model can generate not only higher quality results, but also a variety of plausible results.

Our contributions are as follows:

- We propose SEGAN, a novel diversity-generated adversarial network that not only delivers higher quality results, but also produces a variety of realistic and reasonable outputs compared to existing methods;
- We added a brand-new extractor to the GAN [19] that is specifically designed to extract the style of the training samples in each training iteration and introduce the consistency loss to guide the generator to learn a variety of styles that match to the semantics of the input image;
- We validated that our model can inpaint the same missing regions with multiple results that are plausible and consistent with the high-level semantics of the image, such as those shown in Figure 1.

2. Related Work

2.1. Image Inpainting

Image inpainting by traditional methods. The traditional method, which is based on diffusion, is to use the edge information of the area to be inpainted to determine the direction of diffusion, and spread the known information to the edge. For example, Ballester et al. [5] used the variational method, the histogram statistical method based on local features [6], and the fast marching method based on the level set application proposed by Telea [7]. However, this kind of method can only inpaint small-scale missing areas. In contrast to diffusion-based technologies, patch-based methods can perform texture synthesis [6,7], which can sample similar patches from undamaged areas and paste them into the missing areas. Bertalmio et al. [4] proposed a method of filling texture and structure in the area with missing image information at the same time; and Duan et al. [9] proposed a method of using local patch statistics to complete the image. However, these methods usually generate distorted structures and unreasonable textures.

Xu et al. [8] proposed a typical inpainting method which involves investigating the spatial distribution of image patches. This method can better distinguish the structure and texture, thus forcing the new patched area to become clear and consistent with the surrounding texture. Ting et al. [10] proposed a global region filling algorithm based on Markov random field energy minimization, which pays more attention to the context rationality of texture. However, the calculation complexity of this method is high. Barnes et al. [11] put forward a fast approximate nearest neighbor algorithm called PatchMatch, which can be used for advanced image editing. Shao et al. [12] put forward an algorithm based on the Poisson equation to decompose the image into texture and structure, which is effective in large-area completion. However, these methods can only obtain low-level features, and the obvious limitation is that they only extract texture and structure from the input image. If no texture can be found in the input image, these methods have a very limited effect and do not generate semantically reasonable results.

Image inpainting by deep generative models. Recently, using deep generative models to inpaint images has yielded exciting results. In addition, Image inpainting with generative adversarial networks (GAN) [19] has gained significant attention. Early works[13,15] trained CNNs for image denoising and restoration. The deep generative model named Context Encoder proposed by Pathak et al. [16] can be used for semantic inpainting tasks. The CNN-based inpainting is extended to the large mask, and a context encoder based on the generation adversarial network (GAN) is proposed for inpainting the learned features [18]. The guide loss is introduced to make the feature map generated in the decoder as close as possible to the feature map of the ground truth generated in the encoding

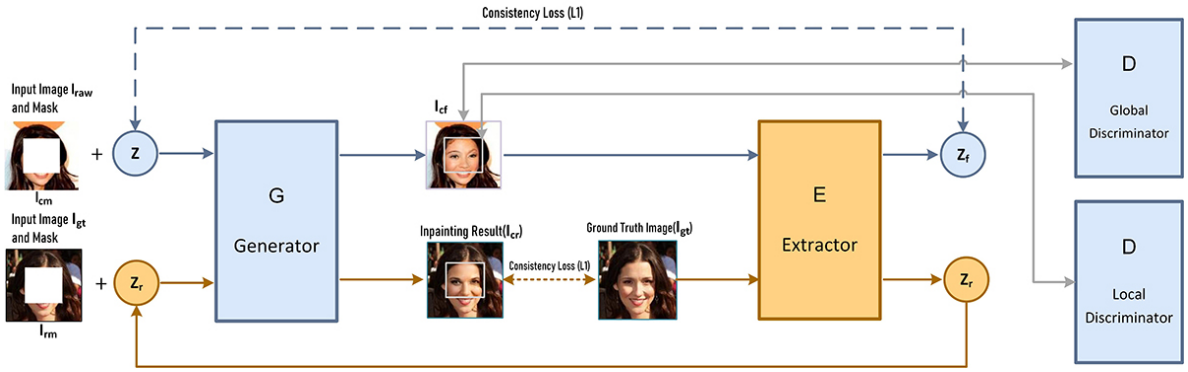


Figure 2. The architecture of our model. It consists of three modules: a generator G, an extractor E, and two discriminators D. (**a**) G takes in both the image with white holes and the style noise as inputs and generates a fake image. The style noise is spatially replicated and concatenated with the input image. (**b**) E is used to extract the style noise (a latent vector) of the input image. (**c**) The global discriminator [39] identifies the entire image, while the local discriminator [39] only discriminates the inpainting regions of the generator output.

process. Lizuka et al. [39] improved the image completion effect by introducing local and global discriminators as experience loss. The global discriminator is used to check the whole image and to evaluate its overall consistency, while the local discriminator is only used to check a small area to ensure the local consistency of the generated patch. Lizuka et al. [39] also proposed the concept of dilated convolutions to the reception field. However, this method needs a lot of computational resources. For this reason, Sagong et al. [21] proposed a structure (Pepsi) composed of a single shared coding network and a parallel decoding network with rough and patching paths, which can reduce the number of convolution operations. Recently, some works [20,23] have proposed the use of spatial attention [24,25] to obtain high-frequency details. Yu et al. [20] proposed a context attention layer, which fills the missing pixels with similar patches of undamaged areas. Isola et al. [22] tried to solve the problem of image restoration using a general image translation model. Using advanced semantic feature learning, the deep generation model can generate semantically consistent results for the missing areas. However, it is still very difficult to generate realistic results from the residual potential features.

Conditional Image Generation On the basis of VAE [31] and GAN [19], conditional image generation has been widely used in conditional image generation tasks, such as 3D modeling, image translation, and style generation. Sohn et al. [26] used random reasoning to generate diverse but realistic outputs based on the deep condition generation model of the Gaussian latent variable. The automatic encoder of conditional variation proposed by Walker et al. [27] can generate a variety of different predictions for the future. After that, the variant automatic encoder is combined with the generation countermeasure network to generate a specific class image by changing the fine-grained class label input into the generation model. In [28], different facial image restorations are achieved by specifying specific attributes (such as male and smile). However, this method is limited to specific areas and requires specific attributes.

3. Approach

We built our diversity inpainting network based on the current state-of-the-art image inpainting model [20], which has shown exciting results in terms of inpainting face, leaf, and rainforest images. However, similar to other existing methods [1,20,35,36,38], classic image completion methods attempt to inpaint missing regions of the original image in a deterministic manner, thus only producing a single result. Instead, our goal was to generate multiple reasonable results.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 December 2019 doi:10.20944/preprints201912.0028.v1

Peer-reviewed version available at IEEE Access 2020, 8, 48451-48463; doi:10.1109/ACCESS.2020.2979348

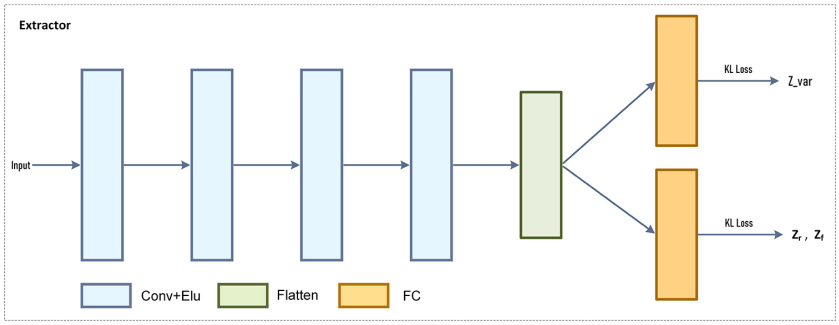


Figure 3. The architecture of our extractor. We added the extractor to the GAN [19] network.

3.1. Extractor

Figure 3 shows the extractor network architecture of our proposed method. It has four convolution layers, one flattened layer, and two parallel fully connected layers. Each convolutional layer uses the Elus activation function. All the convolutional layers use a stride of 2 x 2 pixels and 5 x 5 kernels to reduce the image resolution while increasing the number of output filters.

Let $\mathbf{I}_{gt}$ be ground truth images; the extractor and style noise extracts from $\mathbf{I}_{gt}$ are denoted by $\mathbf{E}$ and $\mathbf{Z}_r$, respectively; we use the ground truth images $\mathbf{I}_{gt}$ as the input; and $\mathbf{Z}_r$ is the latent vector extracted from $\mathbf{I}_{gt}$ by the extractor.

$$Z_r^{(i)} = E(I_{gt}^{(i)}) \tag{1}$$

Let $\mathbf{I}_{cf}$ be the fake image generated by the generator. The extractor and style noise extracts from $\mathbf{I}_{cf}$ are denoted by $\mathbf{E}$ and $\mathbf{Z}_f$, respectively; we use the fake images $\mathbf{I}_{cf}$ as the input; and $\mathbf{Z}_f$ is the latent vector extracted from $\mathbf{I}_{cf}$ by the extractor.

$$Z_f^{(i)} = E(I_{cf}^{(i)}) \tag{2}$$

The extractor extracts the latent vector of each training sample and outputs its mean and covariance , i.e., μ and σ. Similar to VAEs, the KL loss is used to narrow the gap between the prior $p_\theta(z)$ and the Gaussian distribution $q_\phi(z|I)$.

Let the latent vector Z be the centered isotropic multivariate Gaussian $p_\theta(z) = \mathcal{N}(z; 0, \mathbf{I})$. Assume $p_\theta(I|z)$ is a multivariate Gaussian whose distribution parameters are computed from z with the extractor network. We assume that the true posterior adopts an approximate Gaussian form and approximate diagonal covariance:

$$\log q(z|I) = \log N\left(z; \mu, \sigma^2 \mathbf{I}\right). \tag{3}$$

Let σ and μ denote the s.d. and variational mean evaluated at datapoint i, and let μ_j and σ_j simply denote the j-th element of these vectors. Then, the KL divergence between the posterior distribution $q_\phi\left(z|I^{(i)}\right)$ and $p_\theta(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I})$ can be computed as

$$-D_{KL}\left((q_\phi(z)||p_\theta(z)\right) = \int q_\theta(z)\left(\log p_\theta(z) - \log q_\theta(z)\right) dz. \tag{4}$$

According to our assumptions, the prior $p_\theta(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, I)$ and the posterior approximation $q_\phi(\mathbf{z}|\mathbf{I})$ are Gaussian. Thus we have

$$\begin{aligned} \int q_\theta(\mathbf{z}) \log q_\theta(\mathbf{z}) d\mathbf{z} &= \int \mathcal{N}\left(\mathbf{z}; \mu_j, \sigma_j^2\right) \log \mathcal{N}\left(\mathbf{z}; \mu_j, \sigma_j^2\right) d\mathbf{z} \\ &= -\tfrac{J}{2}\log(2\pi) - \tfrac{1}{2}\sum_{j=1}^{J}\left(1 + \log(\sigma_j^2)\right), \end{aligned} \tag{5}$$

Peer-reviewed version available at IEEE Access 2020, 8, 48451-48463; doi:10.1109/ACCESS.2020.2979348

and

$$\begin{aligned}\int q_\theta(\mathbf{z}) \log p(\mathbf{z}) d\mathbf{z} &= \int \mathcal{N}\left(\mathbf{z}; \mu_j, \sigma_j^2\right) \log \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I}) d\mathbf{z} \\ &= -\frac{J}{2} \log(2\pi) - \frac{1}{2} \sum_{j=1}^{J} \left(\mu_j^2 + \sigma_j^2\right).\end{aligned} \tag{6}$$

Finally, we can obtain

$$\begin{aligned}-D_{KL}\left((q_\phi(z)||p_\theta(z)\right) &= -\frac{1}{2} \sum_{j=1}^{J} \log(2\pi) + \frac{1}{2} \sum_{j=1}^{J} \left(1 + \log(\sigma_j^2)\right) \\ &-[-\frac{1}{2} \sum_{j=1}^{J} \log(2\pi) + \frac{1}{2} \sum_{j=1}^{J} \left(\mu_j^2 + \sigma_j^2\right)] \\ &= \frac{1}{2} \sum_{j=1}^{J} \left(1 + \log(\sigma_j^2) - \mu_j^2 - \sigma_j^2\right),\end{aligned} \tag{7}$$

where the mean and s.d. of the approximate posterior, μ and σ , are outputs of the extractor E, i.e., nonlinear functions of the generated sample and the variational parameters.

After this, the latent vector is sampled using where and is an elementwise product. The obtained latent vector Z is fed into the generator together with the masked input image.

The outputs of the generator are processed by the extractor E again to get a latent vector, which is applied to another masked input image.

3.2. Diversity-Generated Inpainting Network: SEGAN

Figure 2 shows the network architecture of our proposed model. We added a novel network after the generator and named the network the extractor, which is responsible for extracting style noise Z (a latent vector). We concatenated an image with white pixels as the missing regions and generated random noise as input, then we output the inpainted fake image(I_{cf}). We input the fake image generated by the generator into the extractor to extract the style noise of the fake image. At the same time, a sample of the ground truth image input extractor was randomly taken from the training set to obtain the style noise of the ground truth image, and the ground truth image concatenated to the mask was input into the generator to obtain the generated image I_{cr}. We wanted the extracted style noise to be as small as possible with the input noise. It is also desirable that the generated image I_{cr} is as close as possible to the ground truth image I_{gt} to be able to continuously update the parameters and weights of the generator network. Here, we propose using the L1 loss function to minimize the sum of the absolute differences between the target value and the estimated value, because the minimum absolute deviation method is more robust than the least squares method.

3.2.1. Consistency Loss

Since the perceptual loss cannot directly optimize the convolutional layer and ensure consistency between the feature maps after the generator and the extractor. We adjusted the form of perceptual loss [46] and propose the consistency loss to handle this problem. As shown in Figure 3, we use the extractor to extract a high-level style space in the ground truth image. Our model also auto-encodes the visible inpainting results deterministically, and the loss function needs to meet this inpainting task. Therefore, the loss per instance here is

$$\mathcal{L}_c^{e,(i)} = \left\| I_{cr}^{(i)} - I_{gt}^{(i)} \right\|_1, \tag{8}$$

where $I_{cr}^{(i)} = G(Z_r^{(i)}, f_m)$ and $I_{gt}^{(i)}$ are the completed and ground truth images, respectively; G is the generator and E is our extractor; z_r is the extractor extracted latent vector we call style noise: z_r = E($I_{gt}^{(i)}$). For the separate generative path, the per-instance loss is

$$\mathcal{L}_c^{g,(i)} = \left\| I_{cf}^{(i)} - I_{raw}^{(i)} \right\|_1, \tag{9}$$

where $I_{cf}^{(i)} = G(Z_f^{(i)}, f_m)$ and $I_{raw}^{(i)}$ are the fake images completed by the generator and input raw images respectively.

3.2.2. Adversarial Loss

To enhance the training process and to inpaint higher quality images, Gulrajani et al. [47] proposed using gradient penalty terms to improve the Wasserstein GAN [37].

$$\begin{aligned} \mathcal{L}_{adv}^{G} = \mathbb{E}_{i_{raw}} \left[D\left(I_{raw}\right) \right] - \mathbb{E}_{i_{raw},z} \left[D\left(G(I_{cm}, z)\right) \right] \\ -\lambda \mathbb{E}_{\hat{i}} \left[\left(\left\| \nabla_{\hat{i}} D(\hat{i}) \right\|_2 - 1 \right)^2 \right], \end{aligned} \tag{10}$$

where $\hat{i}$ is sampled uniformly along a straight line between a pair of generated and input raw images. We used $\lambda = 10$ for all experiments.

For the image completion task, we only attempted to inpaint the missing regions, so for the local discriminator, we only applied the gradient penalty [47] to the pixels in the missing area. This can be achieved by multiplying the gradient by the input mask m as follows:

$$\begin{aligned} \mathcal{L}_{adv}^{L} = \mathbb{E}_{i_{raw}} \left[D\left(I_{raw}\right) \right] - \mathbb{E}_{i_{raw},z} \left[D\left(G(I_{cm}, z)\right) \right] \\ -\lambda \mathbb{E}_{\hat{i}} \left[\left(\left\| \nabla_{\hat{i}} D(\hat{i}) \odot (1 - m) \right\|_2 - 1 \right)^2 \right], \end{aligned} \tag{11}$$

where, for the pixels in that missing regions, the mask value is 0; for other locations, the mask value is 1.

3.2.3. Distributive Regularization

The KL divergence term serves to adjust the learned importance sampling function $q_\phi\left(z|I_{gt}\right)$ to a fixed potential prior $p\left(z_r\right)$. Defined as Gaussians, we get

$$\mathcal{L}_{KL}^{e,(i)} = -KL(q_\phi(z_r|I_{gt}^{(i)})||\mathcal{N}(0, \sigma^{2(i)}\mathbf{I})). \tag{12}$$

For the fake image output by the generator, the learned importance sampling function $q_\phi\left(z|I_{cf}\right)$ to a fixed potential prior $p\left(z_f\right)$ is also a Gaussian.

$$\mathcal{L}_{KL}^{g,(i)} = -KL(q_\phi(z_f|I_{cf}^{(i)})||\mathcal{N}(0, \sigma^{2(i)}\mathbf{I})) \tag{13}$$

3.2.4. Objective

Through the KL, the consistency, and adversarial losses obtained above, the overall objective of our diversity inpainting network is defined as

$$\mathcal{L} = \alpha_{KL}(\mathcal{L}_{KL}^{e} + \mathcal{L}_{KL}^{g}) + \alpha_c(\mathcal{L}_c^{e} + \mathcal{L}_c^{g}) + \alpha_{adv}(\mathcal{L}_{adv}^{G} + \mathcal{L}_{adv}^{L}), \tag{14}$$

where α_{KL}, α_c, α_{adv} are the tradeoff parameters for the KL, consistency, and adversarial losses, respectively.

Peer-reviewed version available at IEEE Access 2020, 8, 48451-48463; doi:10.1109/ACCESS.2020.2979348

3.3. Training

For training, given a ground truth image I_{gt}, we used our proposed extractor to extract the style noise (a latent vector) of the ground truth image, and then concatenate the style noise to the masked ground truth image. It was input to the generator G to obtain an image I_{cr} of the predicted output, and forced I_{cr} to be as close as possible to I_{gt} through the consistency loss L1 to update the parameters and weights of the generator. At the same time, Sample image I_{raw} from the training data, generate mask and random noise for I_{raw} and concatenate together to input generator G, to obtain the predicted output image I_{cf}, we used our proposed extractor to extract the style noise z_f of the generated image and forced z_f to be as close to z as possible to update the generator using consistency loss L1.

Algorithm 1 Training procedure of our proposed diversity inpainting network.

while G has not converged **do**
  for $i = 1 \rightarrow n$ **do**
    Input ground truth images I_{gt};
    Get style noise(a latent vector) by extractor $Z_r \leftarrow E(I_{gt})$;
    Concatenate inputs $\widetilde{I_{rm}} \leftarrow Z_r \odot I_{gt} \odot m$;
    Get predicted outputs $I_{cr} \leftarrow G(Z_r, I_{gt}) \odot (1 - M)$;
    Update the generator G with L1 loss (I_{cr}, I_{gt});
    Meanwhile;
    Sample image I_{raw} from training set data;
    Generate white mask m for I_{raw};
    Generate random noise z for I_{raw};
    Concatenate inputs $\widetilde{I_{cm}} \leftarrow I_{raw} \odot m \odot z$;
    Get predictions $I_{cf} \leftarrow G(I_{raw}, z)$;
    Get style noise by extractor $z_f \leftarrow E(I_{cf})$;
    Update the generator G with L1 loss (z_f, z);
  end for
end while

4. Experiments and Results

We evaluated our proposed model on three open datasets: CelebA faces [33], Agricultural Disease, and MauFlex [1]. AgriculturalDisease is an open dataset from AI Challenger 2018. We split the dataset into a training set, test set, and validation set. Since our method can inpaint countless results, we generated 100 images for each image with missing regions and selected 10 of them, each with different high-level semantic features. We compared the results with current state-of-the-art methods for quantitative and qualitative comparisons.

Our method was compared to the following:

- CA Contextual Attention, proposed by Yu et al. [20]
- SH Shift-net, proposed by Yan et al. [17]
- GL Global and local, proposed by Iizuka et al. [39]

4.1. Implementation Details

Our diversity-generation network was inspired by recent works [20,39], but with several significant modifications, including the extractor. Furthermore, our inpainting network which is implemented in TensorFlow [41] contains 47 million trainable parameters, and was trained on a single NVIDIA 1080 GPU (8GB) with a batch size of 12. The training of CelebA [33] model, Agricultural Disease model, and MauFlex [1] model took roughly 3 days, 2 days, and 1 day, respectively.

To fairly evaluate our method, we only conducted experiments on the centering hole. We compared our method with GL [39], CA [20], and SH [17] on images from the CelebA [33], Agricultural Disease, and MauFlex [1] validation sets. The size of all mask images were processed to 128×128 for training and testing. We used the Adam algorithm [42] to optimize our model with a learning rate of 2×10^3 and $\beta 1 = 0.5, \beta 2 = 0.9$. The tradeoff parameters were set as $\alpha_{KL} = 10, \alpha_{rec} = 0.9, \alpha_{adv} = 1$. For the nonlinearities in the network, we used the exponential linear units (ELUs) as the activation function to replace the commonly used rectified linear unit (ReLU). We found that the ELUs tried to speed up the learning by bringing the average value of the activation function close to zero. Moreover, it helped avoid the problem of gradient disappearance by positive value identification.

4.2. Quantitative comparisons

Quantitative measurement is difficult for the image diversity inpainting task, as our research is to generate diverse and plausible results for an image with missing regions. Comparisons should not be made based solely on a single inpainting result.

However, solely for the purpose of obtaining quantitative indicators, we randomly selected a single sample from our set of results that was close to the ground truth image and selected the best balance of quantitative indicators for comparison. The comparison was tested on 10,000 Celeba [33] test images, with quantitative measures of mean L1 loss, L2 loss, Peak Signal-To-Noise Ration (PSNR), and Structural SIMilarity (SSIM) [43]. We used a 64 x 64 mask in the center of the image. Table 1 lists the results of the evaluation with the centering mask. It is not difficult to see that our methods are superior to all other methods in terms of these quantitative tests.

Table 1. Results using the CelebA dataset with large missing regions, comparing global and local (GL) [39], Shift-net (SH) [17], Contextual Attention (CA) [20], and ours method. $^-$Lower is better. $^+$Higher is better.

Method	$L_1{}^-$(%)	$L_2{}^-$(%)	Structural SIMilarity (SSIM)$^+$	Peak Signal-To-Noise Ration (PSNR)$^+$
GL	2.99	0.53	0.838	23.75
SH	2.64	0.47	0.882	26.38
CA	1.83	0.27	0.931	26.54
Our method	**1.79**	**0.11**	**0.985**	**34.99**

4.3. Qualitative Comparisons

We first evaluated our proposed method on the CelebA [33] face dataset; Figure 4 shows the inpainting results with large missing regions, highlighting the diversity of the output of our model, especially in terms of high-level semantics. GL [39] can produce more natural images using local and global discriminators to make images consistent. SH [17] has been improved in terms of the copy function, but its predictions are to some extent blurry and there is detail missing. In contrast, our method not only produces clearer and more plausible images, but also provides complementary results for multiple attributes.

As shown in Figures 5 and 6, we also evaluated our approach on the MauFlex [1] dataset and agricultural disease dataset to demonstrate the diversity of our output across different datasets. Contextual Attention (CA) [20], while producing reasonable completion results in many cases, can only produce a single result, and in some cases, a single solution is not enough. Our model produces a variety of reasonable results.

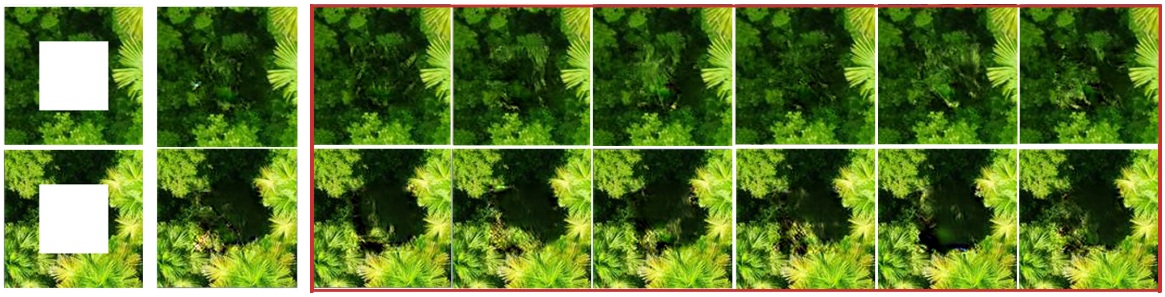
Finally, Figure 4 shows the various facial attribute results from the CelebA [20] dataset. We observed that existing models, such as GL [39], SH [17], and CA [20], can only generate a single facial attribute for each masked input. The results of our method on these test data provide higher visual quality and a variety of attributes, such as the gaze angle of the eye, whether or not glasses are worn, and the disease location on the blade. This is obviously better for image completion.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 December 2019 doi:10.20944/preprints201912.0028.v1

Peer-reviewed version available at IEEE Access 2020, 8, 48451-48463; doi:10.1109/ACCESS.2020.2979348

(a) Input (b) CA (c) SH (d) GL (e) Ours

Figure 4. Comparison of qualitative results with CA [20], SH [17] and GL [39] on the CelebA dataset.



(a) Input (b) CA (c) Ours

Figure 5. Comparison of qualitative results with Contextual Attention (CA) [20] on the MauFlex dataset.

(a) Input (b) CA (c) Ours

Figure 6. Comparison of qualitative results with Contextual Attention (CA) [20] on the AgriculturalDisease dataset.

4.4. Other comparisons

Compared to some of the existing methods (e.g., BicycleGAN [40] and StarGAN [32]), we investigated the influence of using our proposed extractor. We used the common parameters to

Peer-reviewed version available at IEEE Access 2020, 8, 48451-48463; doi:10.1109/ACCESS.2020.2979348

Figure 7. Our method (top), StarGAN [32] (middle), and BicycleGAN [40] (bottom).

train these three models. As shown in Figure 7, for BicycleGAN [40], the output was not good and the generated result was not natural. For StarGAN [32], although it can output a variety of results, this method is limited to specific targeted attributes for training, such as gender, age, happy, angry, etc.

Diversity In Table 2, we use the LPIPS metric proposed by [44] to calculate the diversity scores. For each approach, we calculated the average distance between the 10,000 pairs randomly generated from the 1000 center-masked image samples. I_{gobal} and I_{local} are the full inpainting results and mask-region inpainting results, respectively. It is worth emphasizing that although BicycleGAN [40] obtained relatively high diversity scores, this may indicate that unreasonable images were generated, resulting in worthless variations.

Table 2. We measure diversity using average LPIPS [44] distance.

Method	I_{gobal}(LPIPS)	I_{local}(LPIPS)
pix2pix+noise	0.003	0.009
CVAE	0.013	0.051
BicycleGAN	0.028	0.067
Our method	**0.037**	**0.090**

Realism Table 3 shows the realism across methods. In [45] and later in [22], in order to evaluate the visual realism of the output of these models, human judgment was used to judge the output. We also presented a variety of images generated by our model to a human in a random order, for one second each, asking them to judge the generated fake and measure the "spoofing" rate. The pix2pix $\times$ noise model [22] achieved a higher realism score. CAVE-GAN [14] helped to generate diversity, but because the distribution of potential space for learning is unclear, the generated samples were not reasonable. The BicycleGAN [40] model suffered from mode collapse and had a good realism score. However, our method adds the KL divergence loss in the latent vector extracted by the extractor, making the inpainting results more realistic, as well as producing the highest realism score.

Table 3. Quantitative comparisons of realism.

Method	AMT Fooling Rate(%)
pix2pix+noise	25.93$\pm$2.80 %
CVAE	22.50$\pm$3.27 %
BicycleGAN	31.17$\pm$3.69 %
Our method	**58.87$\pm$2.17 %**

5. Conclusion

In this paper, we proposed SEGAN, a novel diversity-generated image inpainting adversarial network with a newly designed style extractor for diversity image inpainting tasks. For a single input image with missing regions, our model can generate numerous diverse results with plausible content. Experiments on various datasets have shown that our results are diverse and natural, especially for images with large missing areas.

Author Contributions: Conceptualization, W.C.; Data curation, W.C.; Investigation, W.C. and Z.W.; Methodology, W.C.; Project administration, Z.W.; Software, W.C.; Supervision, Z.W.; Validation, W.C. and Z.W.; Visualization, W.C.; Writing-original draft, W.C.; Writing-review & editing, Z.W.

Funding: This work received no external funding.

Conflicts of Interest: The authors declare no conflflict of interest.

Appendix A. More Comparisons Results

More quantitative comparisons with CA [20], SH [17], and GL [39] on the CelebA [33], AgriculturalDisease, and MauFlex [1] datasets were also conducted. Table A1 and Table A2 list the evaluation results on the AgriculturalDisease and MauFlex datasets, respectively. It is obvious that our model is superior to current state-of-the-art methods on multiple datasets.

Table A1. Results using the AgriculturalDisease dataset with large missing regions, comparing GL [39], SH [17], CA [20], and ours method. $^{-}$Lower is better. $^{+}$Higher is better.

Method	$L_1{}^{-}$(%)	$L_2{}^{-}$(%)	SSIM^{+}	PSNR^{+}
GL	2.99	0.53	0.838	23.75
SH	2.64	0.47	0.882	26.38
CA	1.83	0.27	0.931	26.54
Our method	**1.53**	**0.09**	**0.994**	**32.12**

Table A2. Results using the MauFlex dataset with large missing regions, comparing GL [39], SH [17], CA [20], and our method. $^{-}$Lower is better. $^{+}$Higher is better.

Method	$L_1{}^{-}$(%)	$L_2{}^{-}$(%)	SSIM^{+}	PSNR^{+}
GL	2.99	0.53	0.838	23.75
SH	2.64	0.47	0.882	26.38
CA	1.83	0.27	0.931	26.54
Our method	**1.44**	**0.21**	**0.989**	**33.22**

More quantitative comparisons of realism with CA [20], SH [17], and GL [39] on the CelebA [33], AgriculturalDisease, and MauFlex [1] datasets were also conducted. Table A1 and Table A2 list the evaluation results on the AgriculturalDisease and MauFlex datasets, respectively. It is obvious that our model is superior to current state-of-the-art methods on multiple datasets.

Appendix B. Network Architecture

As a supplement to the content in Section 3, in the following, we elaborate on the design of the proposed extractor. The specific architectural design of our proposed extractor network is shown in Table A3. We use the ELUs activation function after each convolutional layer. **N** is the number of output channels, **K** is the kernel size, **S** is the stride size, and **n** is the batch size.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 December 2019 doi:10.20944/preprints201912.0028.v1

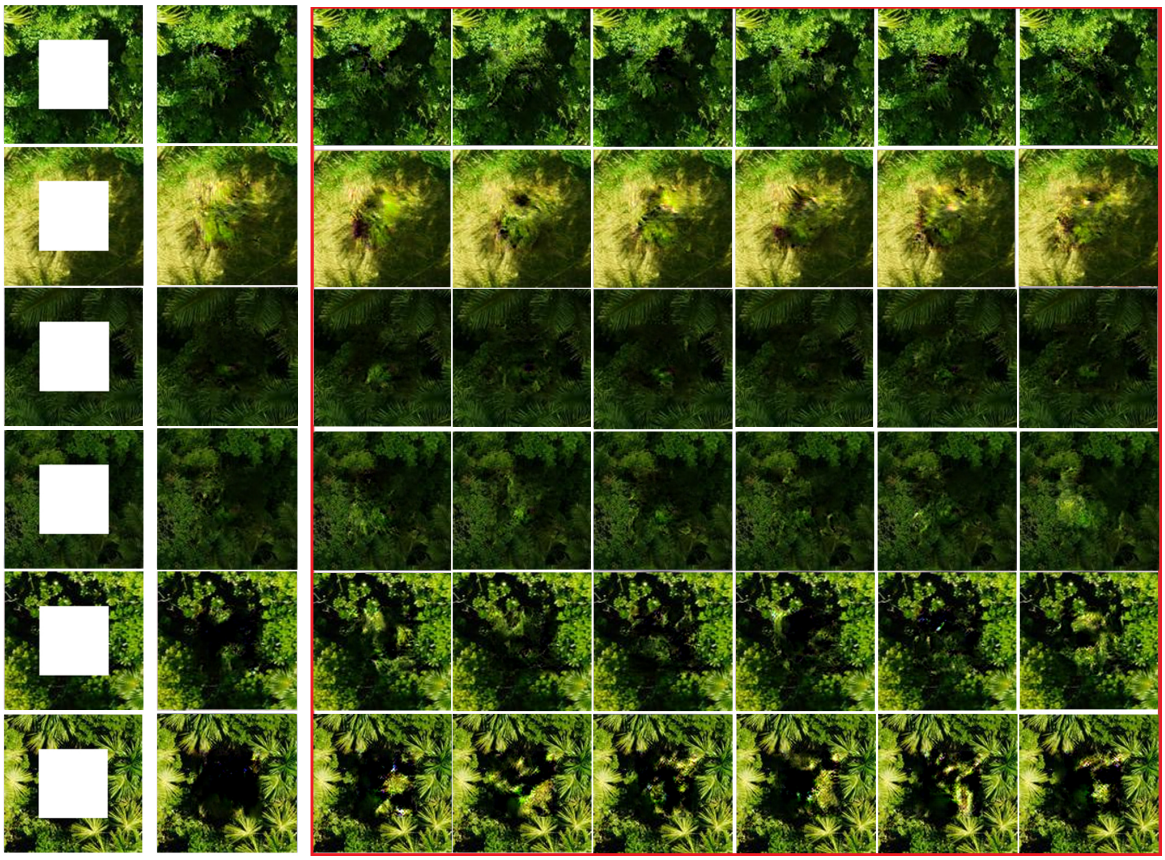
Peer-reviewed version available at IEEE Access 2020, 8, 48451-48463; doi:10.1109/ACCESS.2020.2979348

Table A3. The architecture of our extractor network.

Layer	Inout → Output Shape	Layer Information
Input Layer	(h, w, 3) → $(\frac{h}{2}, \frac{w}{2}, 64)$	Conv. (N64, K5x5, S2), stride=2; ELU;
Hidden Layer1	$(\frac{h}{2}, \frac{w}{2}, 64) \rightarrow (\frac{h}{4}, \frac{w}{4}, 128)$	Conv. (N128, K5x5, S2), stride=2; ELU
Hidden Layer2	$(\frac{h}{4}, \frac{w}{4}, 128) \rightarrow (\frac{h}{8}, \frac{w}{8}, 256)$	Conv. (N256, K5x5, S2), stride=2; ELU
Hidden Layer3	$(\frac{h}{8}, \frac{w}{8}, 256) \rightarrow (\frac{h}{8}, \frac{w}{8}, 256)$	Conv. (N256, K5x5, S2), stride=2; ELU
Hidden Layer4	$(\frac{h}{8}, \frac{w}{8}, 256) \rightarrow$ (n, 4096)	Flatten Layer
Output Layer1	(n, 4096) → (n, 4096)	FC Layer
Output Layer2	(n, 4096) → (n, 4096)	FC Layer

Appendix C. More diverse examples using the CelebA, AgriculturalDisease, and MauFlex datasets

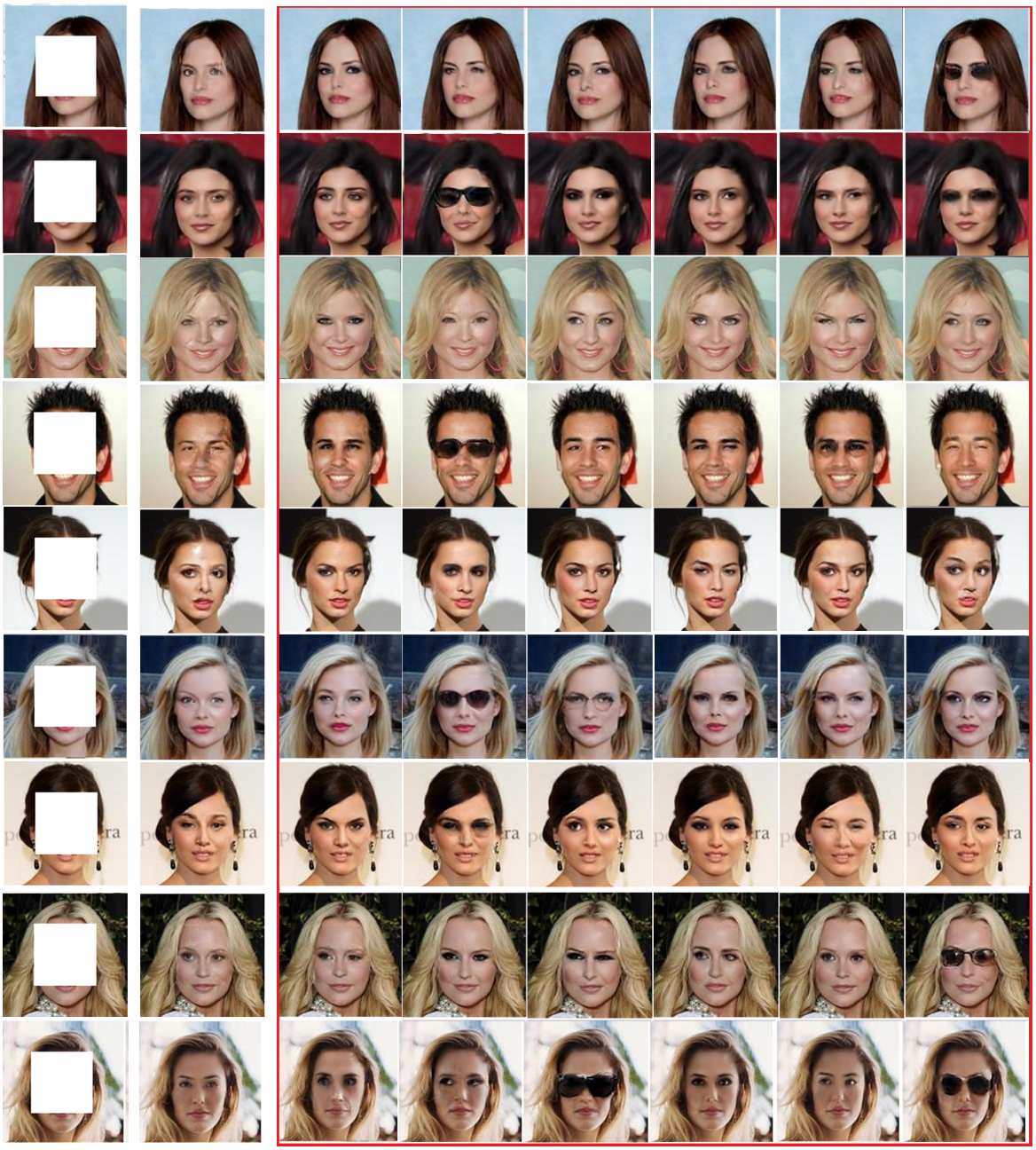
MauFlex Figure A1 shows the results of the qualitative analysis comparison of the models trained on the MauFlex [1] dataset. Our models also have more valuable diversity than existing methods. The MauFlex dataset is an open dataset published by Morales et al. [1] with an original image resolution of 513 x 513. We resized the images to 128 x 128 for training and evaluation.



(a) Input (b) CA (c) Ours

Figure A1. Additional examples of our model tested on the MauFlex [1] dataset. The examples have different tree types. Since the existing CA [20] method cannot find duplicate tree content around the missing area, it is difficult to generate reasonable trees images. Our model is capable of generating a variety of trees with different locations. In addition, we did not apply any attribute labels when training our model.

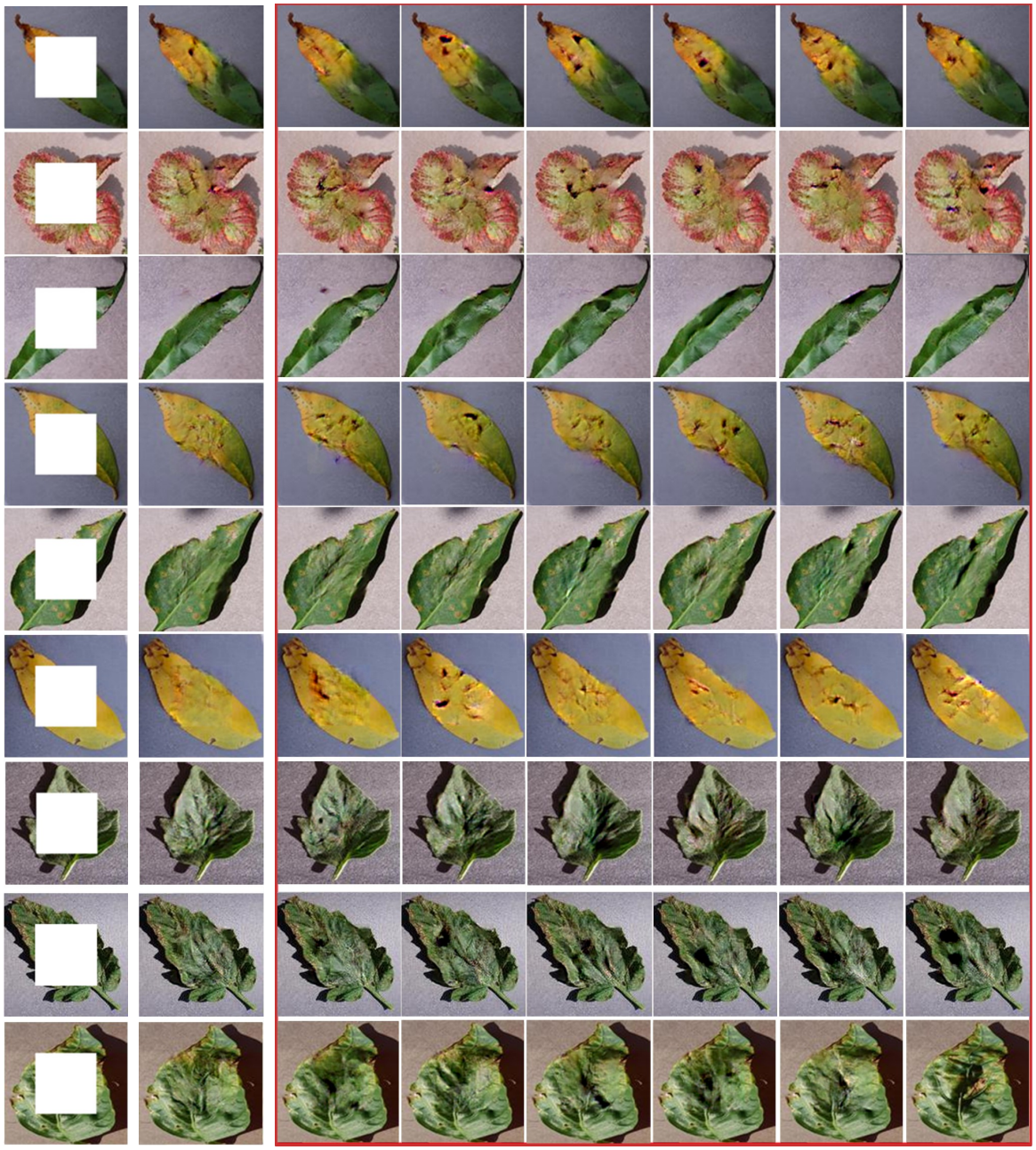
CelebA Figure A2 shows the results of the qualitative analysis comparison of the models trained on the CelebA [33] dataset. The direct output of our model shows a more valuable diversity than the existing methods. The initial resolution of the CelebA dataset image was 218 x 178. We first randomly cropped the images to a size of 178 x 178, and then resized the image to 128 x 128 for both training and evaluation.



(a) Input (b) CA (c) Ours

Figure A2. Additional examples of our model tested on the CelebA dataset. The examples have different genders, skin tones, and eyes. Because a large area of the image is missing, it is impossible to duplicate the content in the surrounding regions, so the Contextual Attention(CA) [20] method cannot generate visually realistic results like ours. In addition, our diversity inpainting results have different gaze angles for the eyes and variation in whether glasses are worn or not. It is important to emphasize that we did not apply any attribute labels when training our model.

AgriculturalDisease Figure A3 shows the results of the qualitative analysis comparison of the models trained on the AgriculturalDisease dataset. Our models also have more valuable diversity than existing methods. The AgriculturalDisease dataset is an open dataset whose original image resolution is irregular. We resized the images to 128 x 128 for training and evaluation.



(a) Input (b) CA (c) Ours

Figure A3. Additional examples of our model tested on the AgriculturalDisease dataset. Examples have blades of different kinds and colors. Since the existing CA [20] method cannot find repeated leaf lesions around the missing area, it is difficult to generate a reasonable diseased leaf. Our model is capable of generating a wide variety of leaves with different lesion locations. In addition, we did not apply any attribute labels when training our model.

References

1. Morales, G.; Kemper, G.; Sevillano, G.; Arteaga, D.; Ortega, I.; Telles, J. Automatic Segmentation of Mauritia flexuosa in Unmanned Aerial Vehicle (UAV) Imagery Using Deep Learning. *Forests*. **2018**, 9, 736.
2. Efros, A.A.; Leung, T.K. Texture synthesis by non-parametric sampling. In Proceedings of the seventh IEEE international conference on computer vision, September 1999; Volume 2, pp. 1033-1038.
3. Efros, A.A.; Freeman, W.T. Image quilting for texture synthesis and transfer. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques, August 2001; pp. 341-346.
4. Bertalmio, M.; Vese, L.; Sapiro, G.; Osher, S. Simultaneous structure and texture image inpainting. *IEEE transactions on image processing*. **2003**, *12*, 882-889.
5. Ballester, C.; Bertalmio, M.; Caselles, V.; Sapiro, G.; Verdera, J. Filling-in by joint interpolation of vector fields and gray levels. *IEEE transactions on image processing*. **2001**, *10*, 1200-1211.
6. Levin, A.; Zomet, A.; Weiss, Y. Learning how to inpaint from global image statistics. IEEE, October 2003; p. 305.
7. Telea, A. An image inpainting technique based on the fast marching method. *Journal of graphics tools*. **2004**, *9*, 23-34.
8. Xu, Z.; Sun, J. Image inpainting by patch propagation using patch sparsity. *IEEE transactions on image processing*. **2010**, *19*, 1153-1165.
9. Duan, K.; Gong, Y.; Hu, N. Automatic image inpainting using local patch statistics. *U.S. Patent Application*. **2018**, *10*, 631.
10. Ting, H.; Chen, S.; Liu, J.; Tang, X. Image inpainting by global structure and texture propagation. In Proceedings of the 15th ACM international conference on Multimedia, September 2007; pp. 517-520.
11. Barnes, C.; Shechtman, E.; Finkelstein, A.; Goldman, D.B. PatchMatch: A randomized correspondence algorithm for structural image editing. In ACM Transactions on Graphics (ToG), July 2009; Volume 28, p. 24.
12. Shao, X.; Liu, Z.; Li, H. An image inpainting approach based on the poisson equation. In Second International Conference on Document Image Analysis for Libraries, April 2006; pp. 5.
13. Xie, J.; Xu, L.; Chen, E. Image denoising and inpainting with deep neural networks. *In Advances in neural information processing systems*. **2012**, pp. 341-349.
14. Bao, J.; Chen, D.; Wen, F.; Li, H.; Hua, G. CVAE-GAN: fine-grained image generation through asymmetric training. In Proceedings of the IEEE International Conference on Computer Vision, 2017; pp. 2745-2754.
15. Xu, L.; Ren, J.S.; Liu, C.; Jia, J. Deep convolutional neural network for image deconvolution. *In Advances in neural information processing systems*. **2014**, pp. 1790-1798.
16. Pathak, D.; Krahenbuhl, P.; Donahue, J.; Darrell, T.; Efros, A.A. Context encoders: Feature learning by inpainting. In Proceedings of the IEEE conference on computer vision and pattern recognition, 2016; pp. 2536-2544.
17. Yan, Z.; Li, X.; Li, M.; Zuo, W.; Shan, S. Shift-net: Image inpainting via deep feature rearrangement. In Proceedings of the European Conference on Computer Vision, 2018; pp. 1-17.
18. Wang, Y.; Tao, X.; Qi, X.; Shen, X.; Jia, J. Image inpainting via generative multi-column convolutional neural networks. *In Advances in Neural Information Processing Systems*. **2018**, pp. 331-340.
19. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M., Xu, B.; Warde-Farley, D.; Ozair, S.; Bengio, Y. Generative adversarial nets. *In Advances in neural information processing systems*. **2014**, pp. 2672-2680.
20. Yu, J.; Lin, Z.; Yang, J.; Shen, X.; Lu, X.; Huang, T.S. Generative image inpainting with contextual attention. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018; pp. 5505-5514.
21. Sagong, M.C.; Shin, Y.G.; Kim, S.W.; Park, S.; Ko, S.J. Pepsi: Fast image inpainting with parallel decoding network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019; pp. 11360-11368.
22. Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, 2017; pp. 1125-1134.
23. Song, Y.; Yang, C.; Lin, Z.; Liu, X.; Huang, Q.; Li, H.; Jay Kuo, C.C. Contextual-based image inpainting: Infer, match, and translate. In Proceedings of the European Conference on Computer Vision, 2018; pp. 3-19.
24. Jaderberg, M.; Simonyan, K.; Zisserman, A. Spatial transformer networks. *In Advances in neural information processing systems*. **2015**, pp. 2017-2025.

25. Zhou, T.; Tulsiani, S.; Sun, W.; Malik, J.; Efros, A.A. View synthesis by appearance flow. In European conference on computer vision, October 2016; pp. 286-301.
26. Sohn, K.; Lee, H.; Yan, X. Learning structured output representation using deep conditional generative models. *In Advances in neural information processing systems*. **2015**, pp. 3483-3491.
27. Walker, J.; Doersch, C.; Gupta, A.; Hebert, M. An uncertain future: Forecasting from static images using variational autoencoders. In European Conference on Computer Vision, October 2016; pp. 835-851.
28. Chen, Z.; Nie, S.; Wu, T.; Healey, C.G. High resolution face completion with multiple controllable attributes via fully end-to-end progressive generative adversarial networks. 2018, arXiv:1801.07632.
29. Karras, T.; Laine, S.; Aila, T. 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.
30. Doersch, C.; Singh, S.; Gupta, A.; Sivic, J.; Efros, A.A. What makes Paris look like Paris?. *Communications of the ACM*. **2015**, 58, 103-110.
31. Kingma, D.P.; Welling, M. Auto-encoding variational bayes. arXiv 2013, arXiv:1312.6114.
32. Choi, Y.; Choi, M.; Kim, M.; Ha, J.W.; Kim, S.; Choo, J. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018; pp. 8789-8797.
33. Zhou, B.; Lapedriza, A.; Khosla, A.; Oliva, A.; Torralba, A. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*. **2017**, 40, 1452-1464.
34. Yeh, R.; Chen, C.; Lim, T.Y.; Hasegawa-Johnson, M.; Do, M.N. Semantic image inpainting with perceptual and contextual losses. arXiv 2016, arXiv:1607.07539.
35. Liu, H.; Jiang, B.; Xiao, Y.; Yang, C. Coherent Semantic Attention for Image Inpainting. arXiv 2019, arXiv:1905.12384.
36. Zeng, Y.; Fu, J.; Chao, H.; Guo, B. Learning Pyramid-Context Encoder Network for High-Quality Image Inpainting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019; pp. 1486-1494.
37. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein gan. arXiv 2017, arXiv: 1701.07875.
38. Nazeri, K.; Ng, E.; Joseph, T.; Qureshi, F.; Ebrahimi, M. Edgeconnect: Generative image inpainting with adversarial edge learning. arXiv 2019, arXiv:1901.00212.
39. Iizuka, S.; Simo-Serra, E.; Ishikawa, H. Globally and locally consistent image completion. *ACM Transactions on Graphics*. **2017**, *36*, 107.
40. Zhu, J.Y.; Zhang, R.; Pathak, D.; Darrell, T.; Efros, A.A.; Wang, O.; Shechtman, E. Toward multimodal image-to-image translation. *In Advances in Neural Information Processing Systems*. **2017**, pp. 465-476.
41. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Ghemawat, S. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv 2016, arXiv:1603.04467.
42. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. arXiv 2014, arXiv:1412.6980.
43. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*. **2004**, 13, 600-612.
44. Zhang, R.; Isola, P.; Efros, A.A.; Shechtman, E.; Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018; pp. 586-595.
45. Zhang, R.; Isola, P.; Efros, A.A. Colorful image colorization. In European conference on computer vision, October 2016; pp. 649-666.
46. Johnson, J.; Alahi, A.; Fei-Fei, L. Perceptual losses for real-time style transfer and super-resolution. In European conference on computer vision, October 2016; pp. 694-711.
47. Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; Courville, A.C. Improved training of wasserstein gans. *In Advances in neural information processing systems*. **2017**, pp. 5767-5777.