

Received February 25, 2020, accepted March 5, 2020, date of publication March 9, 2020, date of current version March 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2979348

# PiiGAN: Generative Adversarial Networks for Pluralistic Image Inpainting

**WEIWEI CAI<sup>ID</sup>**, (Student Member, IEEE), AND **ZHANGUO WEI<sup>ID</sup>**

School of Logistics and Transportation, Central South University of Forestry and Technology, Changsha 410004, China

Corresponding author: Zhanguo Wei (t20110778@csuft.edu.cn)

This work was supported by the Hunan Key Laboratory of Intelligent Logistics Technology under Grant 2019TP1015.

**ABSTRACT** The latest methods based on deep learning have achieved amazing results regarding the complex work of inpainting large missing areas in an image. But this type of method generally attempts to generate one single “optimal” result, ignoring many other plausible results. 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 feature (latent vector) from the ground truth. Once obtained, the extracted style feature and the ground truth are both input into the generator. We also craft a consistency loss that guides the generated image to approximate the ground truth. After iterations, our generator is able to learn the mapping of styles corresponding to multiple sets of vectors. The proposed model can generate a large number of results consistent with the context semantics of the image. Moreover, we evaluated the effectiveness of our model on three datasets, i.e., CelebA, PlantVillage, and MauFlex. Compared to state-of-the-art inpainting methods, this model is able to offer desirable inpainting results with both better quality and higher diversity. The code and model will be made available on <https://github.com/vivitsai/PiiGAN>.

**INDEX TERMS** Deep learning, generative adversarial networks, image inpainting, diversity inpainting.

## I. 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 inpaint 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 expectation, as it is the only option. Therefore, driven by this observation, we hope to inpainting a variety of plausible results on a single missing region, which we call pluralistic image inpainting (as shown in Figure 1).

Early research [2] 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 inpaint 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 inpaint 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 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 inpainting a unique “optimal” 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

The associate editor coordinating the review of this manuscript and approving it for publication was Zhihan Lv<sup>ID</sup>.



**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.

targeted attributes and may result in unreasonable images being inpainting.

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 feature 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 feature of the ground truth of the training set and inputs it to the generator together with the ground truth. We use the consistency loss  $L1$  to force the generated image to be as close to the ground truth of the training set as possible. At the same time, we generate and input a random vectors and masked image to the generator to get the output fake image, and use the consistency loss  $L1$  to make the extracted style feature as close to the input vectors as possible. After the iteration, the generator can learn the mapping of styles corresponding to multiple sets of input vectors. 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], PlantVillage, 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. In addition, our model has practical application value in various fields such as art restoration, real-time inpainting of large-area missing images, and facial micro-reshaping.

The main contributions of this work are summarized as follows:

- We propose PiiGAN, a novel generative adversarial networks for pluralistic image inpainting that not only delivers higher quality results, but also produces a variety of realistic and reasonable outputs.
- We have designed a new extractor to improve GAN. The extractor extracts the style vectors of the training samples in each 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 inpainting the same missing regions with multiple results that are plausible and consistent with the high-level semantics of the image, and evaluated the effectiveness of our model on multiple datasets.

The rest of this paper is organized as follows. Section 2 provides related work on image inpainting. In addition, some existing studies on conditional image generation are introduced. In Section 3, we elaborate on the proposed model of pluralistic image inpainting (PiiGAN). Section 4 provides an evaluation. Finally, conclusions are given in Section 5.

## II. RELATED WORK

### A. IMAGE INPAINTING BY TRADITIONAL METHODS

The traditional method, which is based on diffusion, is to use the edge information of the area to be inpaint 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 and Sun [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. Wu *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.

### B. 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 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.* [35] 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], [29] 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. Other work [3], [11], [38], [49] also explores related applications.

### C. 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.

## III. PROPOSED APPROACH

We built our pluralistic image 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], [21], [36], [48], 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.



### A. 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  $2 \times 2$  pixels and  $5 \times 5$  kernels to reduce the image resolution while increasing the number of output filters. The two fully connected layers in the extractor will both output  $z_{var}$ , and they are equal, one of them is dedicated to KL loss, and the other will be input to the generator with the extracted latent vector  $\mathbf{Z}_r$ .

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

$$\mathbf{Z}_r^{(i)} = E(\mathbf{I}_{gt}^{(i)}) \quad (1)$$

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

$$\mathbf{Z}_f^{(i)} = E(\mathbf{I}_{cf}^{(i)}) \quad (2)$$

The extractor extracts the style feature of each training sample and outputs its mean and covariance, i.e.,  $\mu$  and  $\sigma$ . 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  $\mathbf{Z}$  be the centered isotropic multivariate Gaussian  $p_\theta(z) = \mathcal{N}(z; 0, I)$ . Assume  $p_\theta(I|z)$  is a multivariate Gaussian whose distribution parameters are computed from  $\mathbf{z}$  with the extractor network. We assume that the true posterior adopts an approximate Gaussian form and approximate diagonal covariance:

$$\log q(z|I) = \log N(z; \mu, \sigma^2 I) \quad (3)$$

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

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

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

$$\begin{aligned} \int q_\theta(z) \log q_\theta(z) dz &= \int \mathcal{N}(z; \mu_j, \sigma_j^2) \log \mathcal{N}(z; \mu_j, \sigma_j^2) dz \\ &= -\frac{J}{2} \log(2\pi) - \frac{1}{2} \sum_{j=1}^J (1 + \log(\sigma_j^2)) \end{aligned} \quad (5)$$

and

$$\begin{aligned} \int q_\theta(z) \log p_\theta(z) dz &= \int \mathcal{N}(z; \mu_j, \sigma_j^2) \log \mathcal{N}(z; 0, I) dz \\ &= -\frac{J}{2} \log(2\pi) - \frac{1}{2} \sum_{j=1}^J (\mu_j^2 + \sigma_j^2) \end{aligned} \quad (6)$$

Finally, we can obtain

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

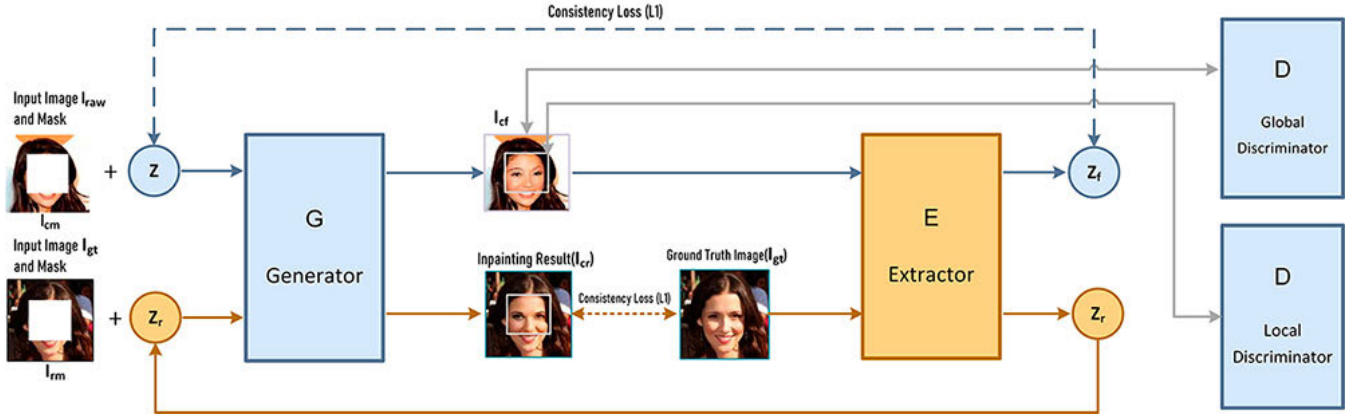
where the mean and s.d. of the approximate posterior,  $\mu$  and  $\sigma$ , are outputs of the extractor  $E$ , i.e. nonlinear functions of the generated sample  $x^{(i)}$  and the variational parameters  $\phi$ .

After this, the latent vector  $z \sim q_\phi(z|x)$  is sampled using  $g_\phi(x, \epsilon) = \mu + \sigma \odot \epsilon$  where  $\epsilon \sim \mathcal{N}(0, I)$  and  $\odot$  is an element-wise product. The obtained latent vector  $\mathbf{Z}$  is fed into the generator together with the masked input image.

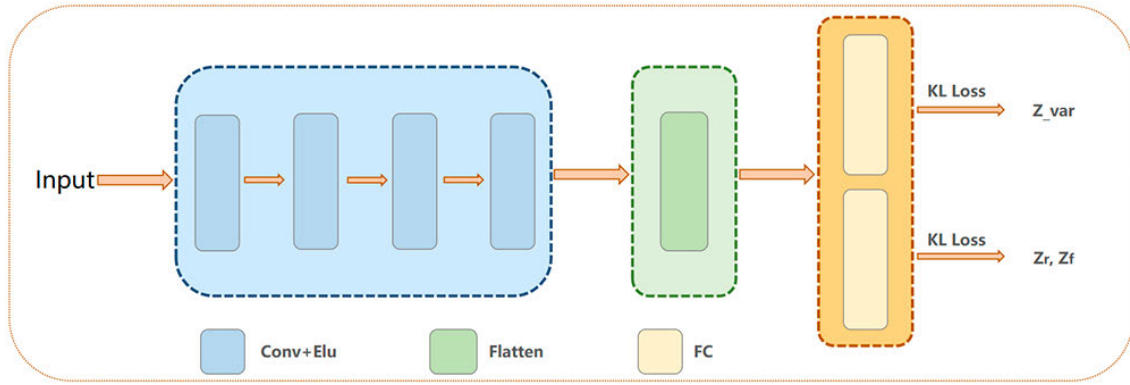
The outputs of the generator are processed by the extractor  $E$  again to get style feature, which is applied to another masked input image.

### B. PLURALISTIC IMAGE INPAINTING NETWORK: PiiGAN

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 latent vector  $\mathbf{Z}$ . We concatenated an image with white pixels as the missing regions and generated random vectors as input, then we output the inpainting fake image ( $\mathbf{I}_{cf}$ ). We input the fake image generated by the generator into the extractor to extract the style feature 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 feature of the ground truth image, and the ground truth image concatenated to the mask was input into the generator to obtain the generated image  $\mathbf{I}_{cr}$ . We wanted the extracted style feature to be as small as possible with the input vectors. It is also desirable that the generated image  $\mathbf{I}_{cr}$  is as close as possible to the ground truth image  $\mathbf{I}_{gt}$  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.



**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 feature as inputs and generates a fake image. The style feature is spatially replicated and concatenated with the input image. (b) E is used to extract the style feature 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.



**FIGURE 3.** The architecture of our extractor network.

### 1) CONSISTENCY LOSS

Since the perceptual loss [46] 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 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

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

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 feature,  $z_r = E(I_{gt}^{(i)})$ . For the separate generative path, the per-instance loss is

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

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.

### 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].

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

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:

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

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

### 3) DISTRIBUTIVE REGULARIZATION

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

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

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

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

### 4) OBJECTIVE

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

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

where  $\alpha_{KL}$ ,  $\alpha_c$ ,  $\alpha_{adv}$  are the tradeoff parameters for the KL, consistency, and adversarial losses respectively.

### C. TRAINING

For training, given a ground truth image  $I_{gt}$ , we used our proposed extractor to extract the style feature of the ground truth, and then concatenate the style feature 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 vectors 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 latent vector  $z_f$  of the generated image and forced  $z_f$  to be as close to  $z$  as possible to update the generator using consistency loss L1.

## IV. EXPERIMENTS AND RESULTS

We evaluated our proposed model on three open datasets: CelebA faces [33], PlantVillage, and MauFlex [1]. The PlantVillage dataset is a publicly available dataset for researchers, and we manually downloaded all training and test set images from the PlantVillage page (<https://plantvillage.org>). And MauFlex is also an open dataset by Morales *et al.* The number of samples in the three datasets we obtained were 200k, 45k and 25k images, respectively. We randomly divide the data set into training and test sets, of which 15% of the data set is the test 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.

### Algorithm 1 Training Procedure of Our Proposed Model

---

```

1: while G has not converged do
2:   for  $i = 1 \rightarrow n$  do
3:     Input ground truth images  $I_{gt}$ ;
4:     Get style feature by extractor  $Z_r \leftarrow E(I_{gt})$ ;
5:     Concatenate inputs  $\tilde{I}_{rm} \leftarrow Z_r \odot I_{gt} \odot m$ ;
6:     Get predicted outputs  $I_{cr} \leftarrow G(Z_r, I_{gt}) \odot (1 - M)$ ;
7:     Update the generator G with L1 loss ( $I_{cr}, I_{gt}$ );
8:     Meanwhile,
9:     Sample image  $I_{raw}$  from training set data;
10:    Generate white mask  $m$  for  $I_{raw}$ ;
11:    Generate random vectors  $z$  for  $I_{raw}$ ;
12:    Concatenate inputs  $\tilde{I}_{cm} \leftarrow I_{raw} \odot m \odot z$ ;
13:    Get predictions  $I_{cf} \leftarrow G(I_{raw}, z)$ ;
14:    Get style feature by extractor  $z_f \leftarrow E(I_{cf})$ ;
15:    Update the generator G with L1 loss ( $z_f, z$ );
16:  end for
17: end while

```

---

Our method was compared to the following:

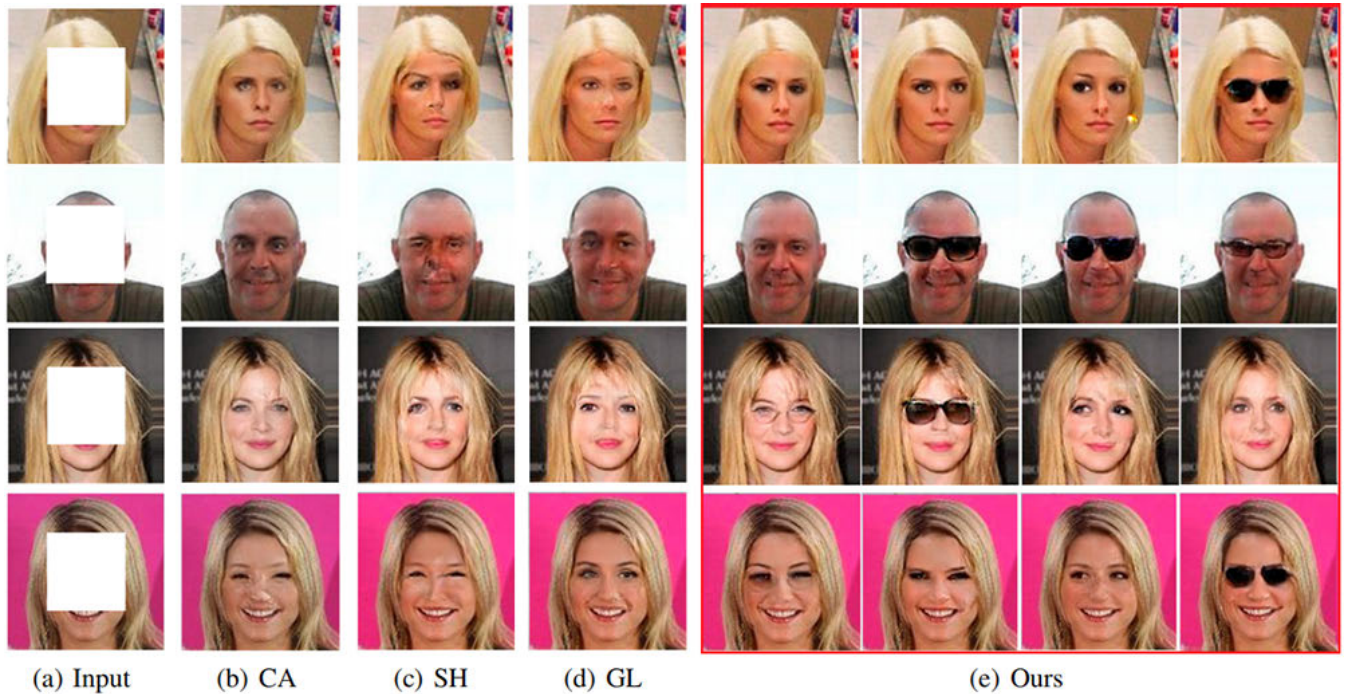
- CA Contextual Attention, proposed by Yu *et al.* [20]
- SH Shift-net, proposed by Yan *et al.* [17]
- GL Global and local, proposed by Iizuka *et al.* [39]

### A. 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. We use Tensorboard to visualize the training process to view various parameters of the training process in real time. It was confirmed that The training of CelebA [33] model, PlantVillage 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], PlantVillage, and MauFlex [1] validation sets. The size of all mask images were processed to  $128 \times 128$  for training and testing. We used the Adam algorithm [42] to optimize our model with a learning rate of  $2 \times 10^{-3}$  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. Compared with ReLUs, ELUs can be more robust to input changes or noise.





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

**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. <sup>-</sup>Lower is better. <sup>+</sup>Higher is better.

Method	L <sub>1</sub> <sup>-</sup> (%)	L <sub>2</sub> <sup>-</sup> (%)	SSIM <sup>+</sup>	PSNR <sup>+</sup>
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	<b>1.79</b>	<b>0.11</b>	<b>0.985</b>	<b>34.99</b>

## B. 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 Ratio (PSNR), and Structural SIMilarity (SSIM) [43]. We used a  $64 \times 64$  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.

## C. QUALITATIVE COMPARISONS

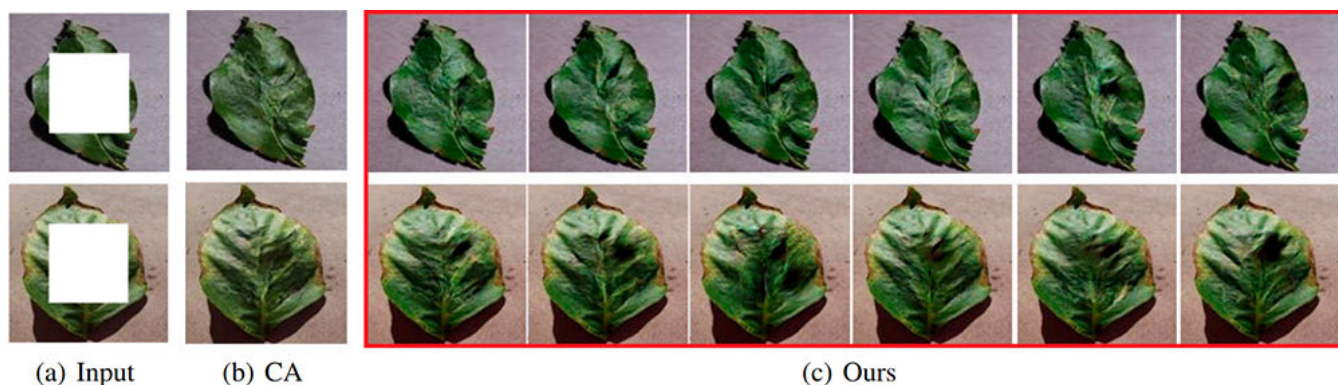
We first evaluated our proposed method on the CelebA [33] face dataset; Figure 4 shows the inpainting results with large missing regions, highlighting the diversity of the output of our model, especially in terms of high-level semantics. GL [39] can produce more natural images using local and global 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 PlantVillage 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.

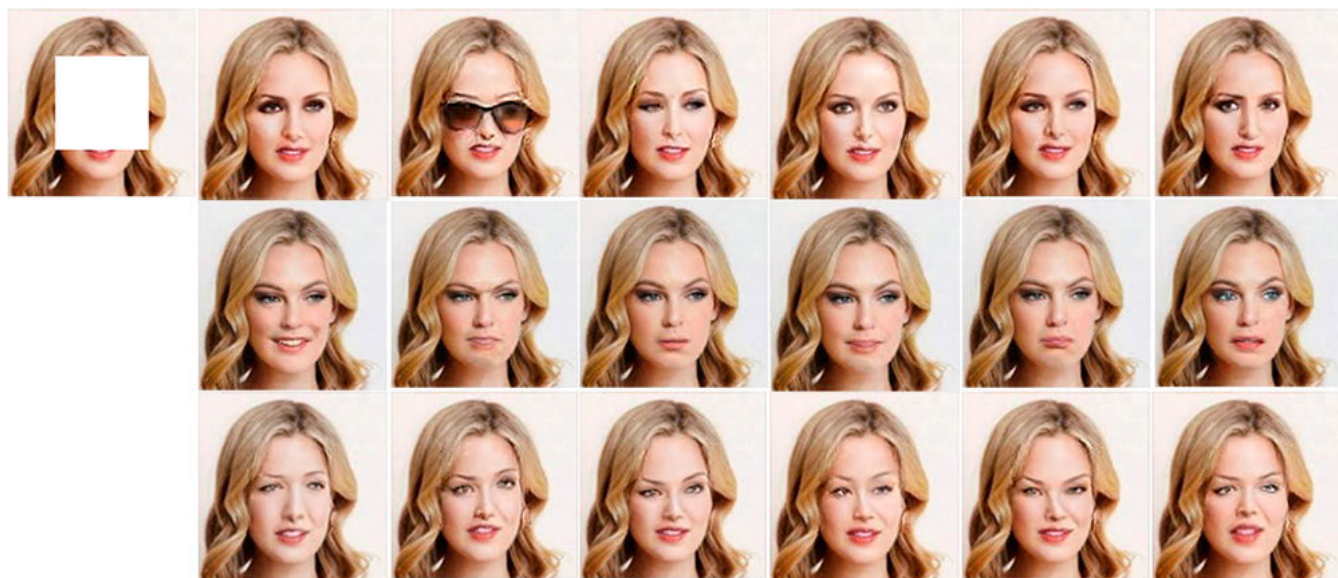
Finally, Figure 4 shows the various facial attribute results from the CelebA [20] dataset. We observed that existing models, such as GL [39], SH [17], and CA [20], can only generate a single facial attribute for each masked input. The results of our method on these test data provide higher visual quality 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.



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



**FIGURE 6.** Comparison of qualitative results with Contextual Attention (CA) [20] on the PlantVillage dataset.



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

#### D. OTHER COMPARISONS

Compared to some of the existing methods (BicycleGAN [40] and StarGAN [32]), we investigated the influence of using

our proposed extractor. We used the common parameters to train these three models. As shown in Figure 7, for BicycleGAN [40], the output was not good and the generated result



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

Method	$I_{global}$ (LPIPS)	$I_{local}$ (LPIPS)
pix2pix+noise	0.003	0.009
CVAE	0.013	0.051
BicycleGAN	0.028	0.067
Our method	<b>0.037</b>	<b>0.090</b>

**TABLE 3.** Quantitative comparisons of realism.

Method	AMT Fooling Rate(%)
pix2pix+noise	25.93±2.80 %
CVAE	22.50±3.27 %
BicycleGAN	31.17±3.69 %
Our method	<b>58.87±2.17 %</b>

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.

### 1) 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.  $I_{global}$  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.

### 2) 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 × 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 style feature extracted by the extractor, making the inpainting results more realistic, as well as producing the highest realism score.

## V. CONCLUSION

In this paper, we proposed PiiGAN, a novel generative adversarial networks with a newly designed style extractor for pluralistic 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,

**TABLE 4.** Results using the PlantVillage dataset with large missing regions, comparing GL [39], SH [17], CA [20], and ours method. <sup>-</sup>Lower is better. <sup>+</sup>Higher is better.

Method	L_1 <sup>-</sup> (%)	L_2 <sup>-</sup> (%)	SSIM <sup>+</sup>	PSNR <sup>+</sup>
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	<b>1.53</b>	<b>0.09</b>	<b>0.994</b>	<b>32.12</b>

**TABLE 5.** Results using the MauFlex dataset with large missing regions, comparing GL [39], SH [17], CA [20], and our method. <sup>-</sup>Lower is better. <sup>+</sup>Higher is better.

Method	L_1 <sup>-</sup> (%)	L_2 <sup>-</sup> (%)	SSIM <sup>+</sup>	PSNR <sup>+</sup>
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	<b>1.44</b>	<b>0.21</b>	<b>0.989</b>	<b>33.22</b>

especially for images with large missing areas. Our model can also be applied in the fields of art restoration, facial micro-shaping and image augmentation. In future work, we will further study the image inpainting of large irregular missing areas.

## APPENDIXES

### APPENDIX A

#### MORE COMPARISONS RESULTS

More quantitative comparisons with CA [20], SH [17], and GL [39] on the CelebA [33], PlantVillage, and MauFlex [1] datasets were also conducted. Table 4 and Table 5 list the evaluation results on the PlantVillage and MauFlex datasets, respectively. It is obvious that our model is superior to current state-of-the-art methods on multiple datasets.

More quantitative comparisons of realism with CA [20], SH [17], and GL [39] on the CelebA [33], PlantVillage, and MauFlex [1] datasets were also conducted. Table 4 and Table 5 list the evaluation results on the PlantVillage and MauFlex datasets, respectively. It is obvious that our model is superior to current state-of-the-art methods on multiple datasets.

### APPENDIX B

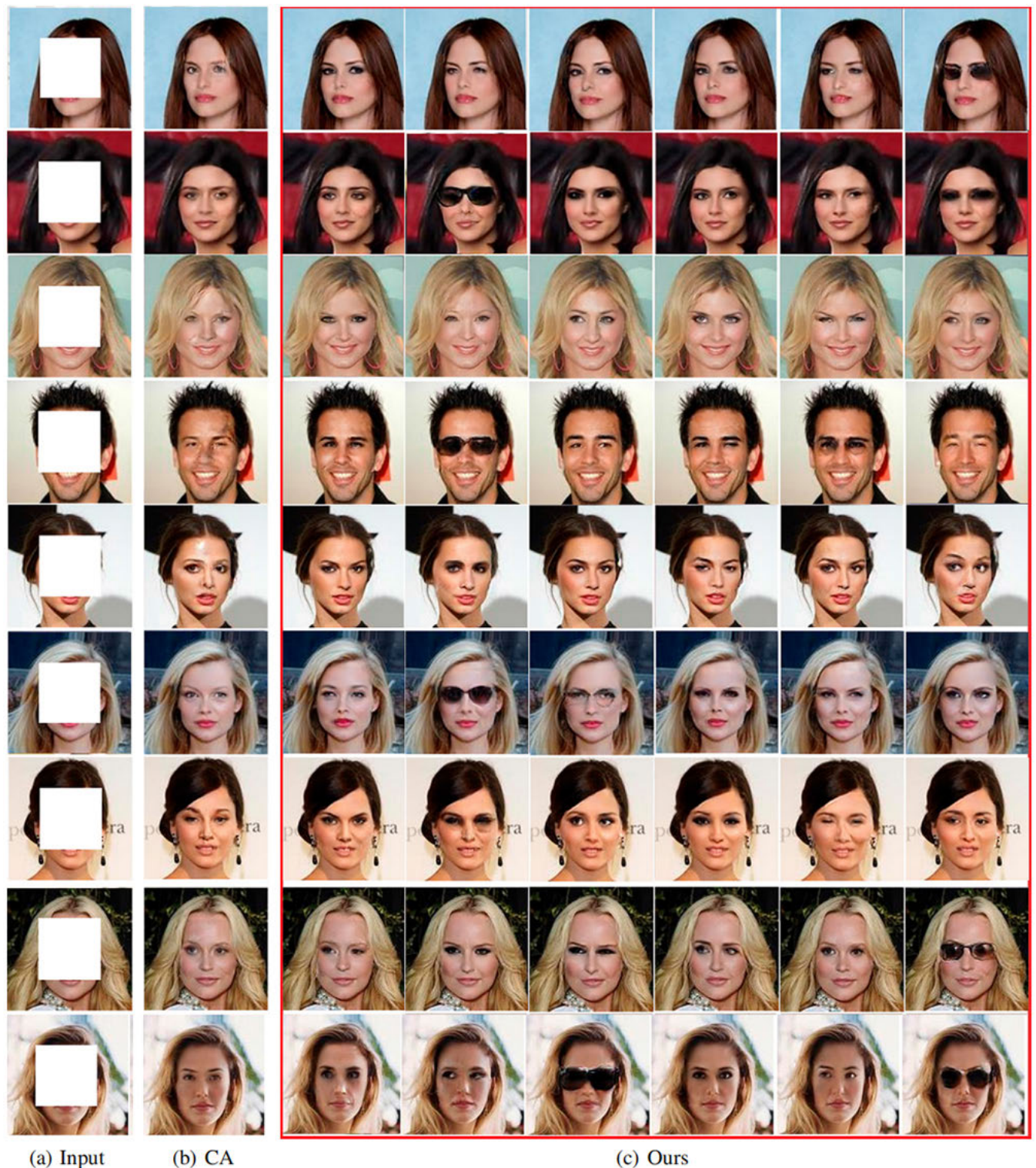
#### NETWORK ARCHITECTURE

As a supplement to the content in Section III, 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 6. 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.

### APPENDIX C

#### MORE DIVERSE EXAMPLES USING THE CelebA, PlantVillage, AND MauFlex DATASETS

**CelebA** Figure 8 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

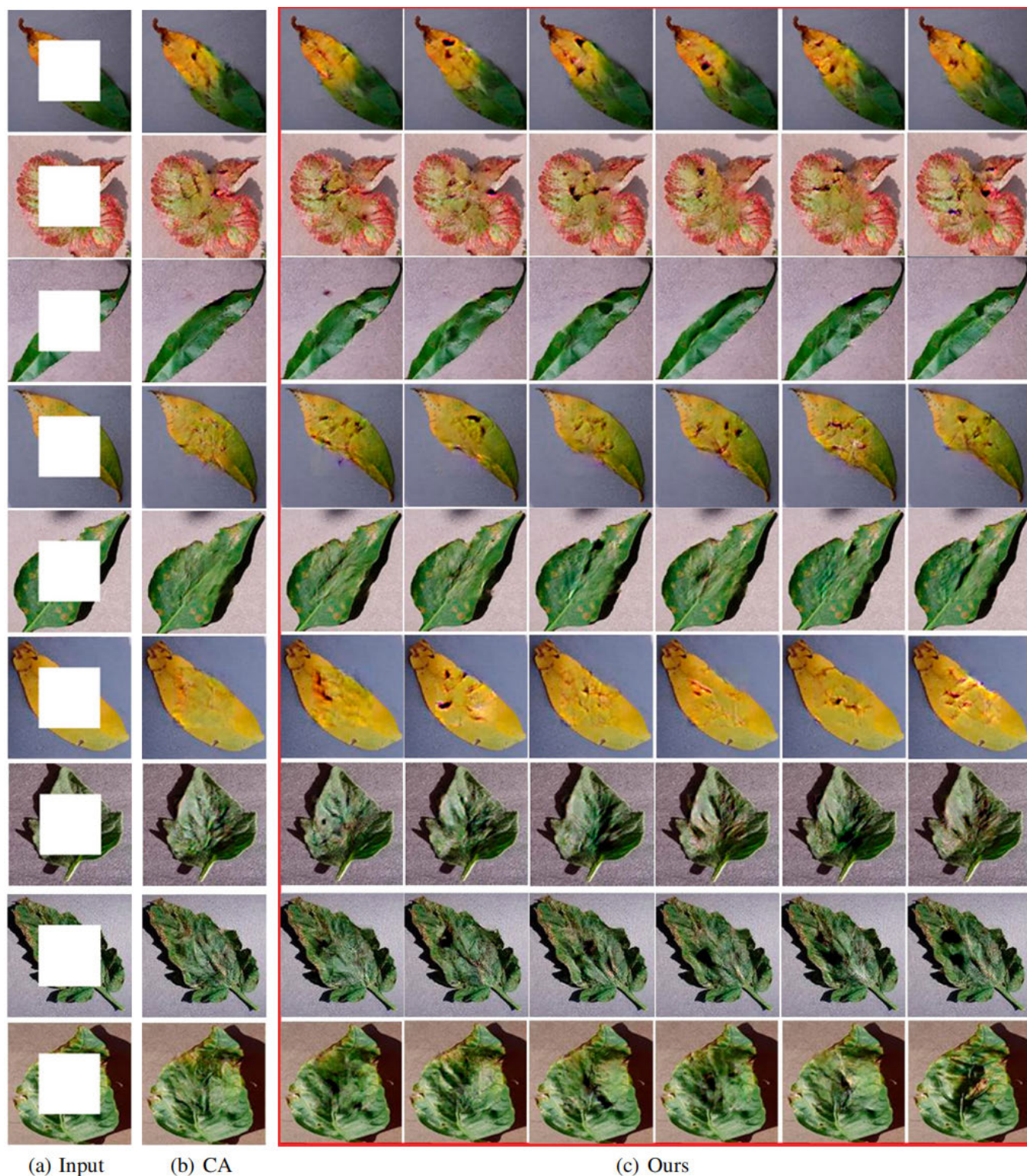


**FIGURE 8.** Additional examples of our model tested on the CelebA dataset. The examples have different genders, skin tones, and eyes. Because a large area of the image is missing, it is impossible to duplicate the content in the surrounding regions, so the Contextual Attention(CA) [20] method cannot 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.

diversity than the existing methods. The initial resolution of the CelebA dataset image was  $218 \times 178$ . We first randomly

cropped the images to a size of  $178 \times 178$ , and then resized the image to  $128 \times 128$  for both training and evaluation.



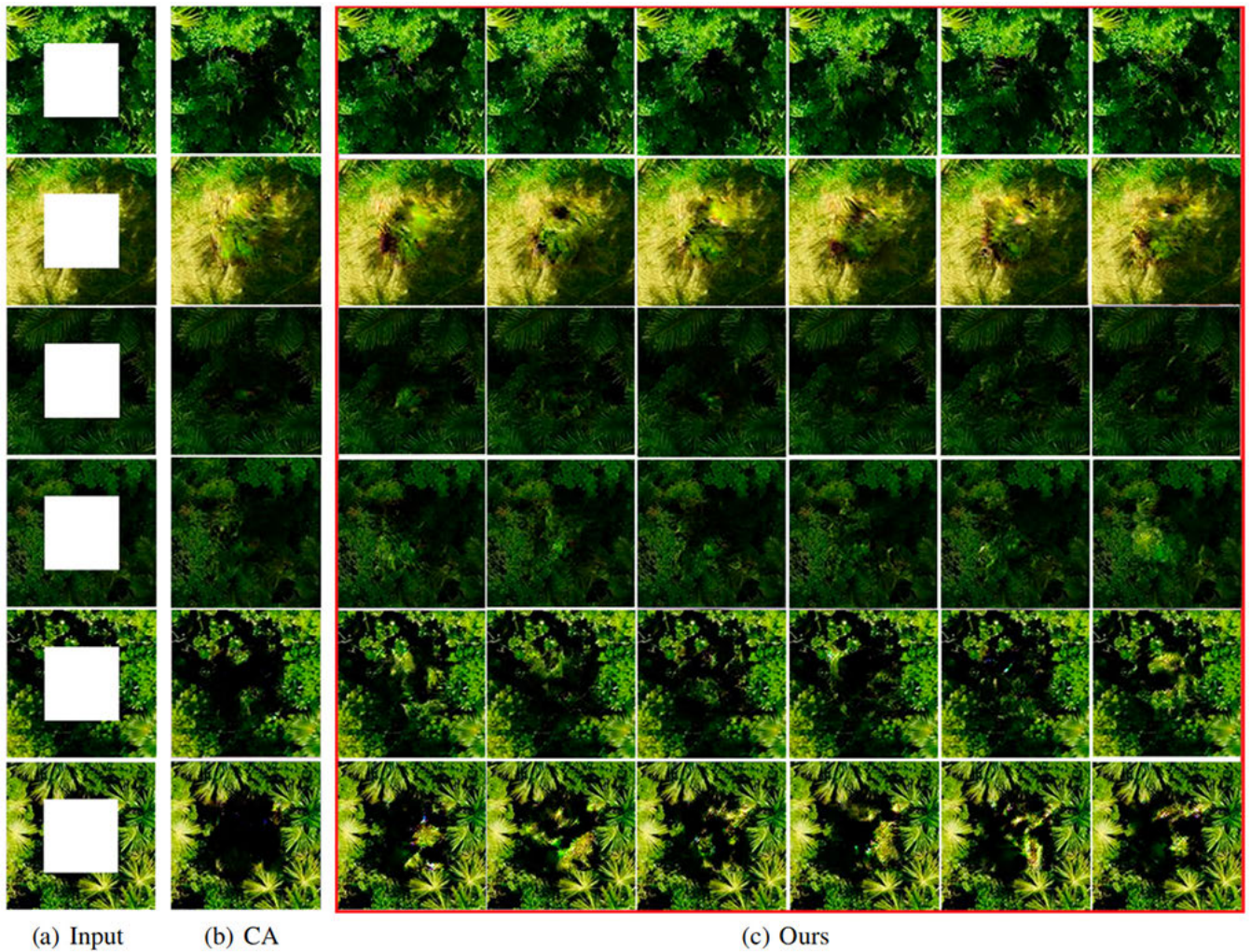


**FIGURE 9.** Additional examples of our model tested on the PlantVillage dataset. Examples have blades of different kinds and colors. Since the existing CA [20] method cannot find repeated leaf lesions around the missing area, it is difficult to generate a reasonable diseased leaf. Our model is capable of generating a wide variety of leaves with different lesion locations. In addition, we did not apply any attribute labels when training our model.

**PlantVillage** Figure 9 shows the results of the qualitative analysis comparison of the models trained on the PlantVillage dataset. Our models also have more valuable diversity than

existing methods. The PlantVillage dataset is an open dataset whose original image resolution is irregular. We resized the images to  $128 \times 128$  for training and evaluation.





**FIGURE 10.** 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.

**TABLE 6.** The architecture of our extractor network.

Layer	Inout $\rightarrow$ Output Shape	Layer Information
Input Layer	$(h, w, 3) \rightarrow (\frac{h}{2}, \frac{w}{2}, 64)$	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	$(n, 4096) \rightarrow (n, 4096)$	FC Layer
Output Layer2	$(n, 4096) \rightarrow (n, 4096)$	FC Layer

**MauFlex** Figure 10 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  $513 \times 513$ . We resized the images to  $128 \times 128$  for training and evaluation.

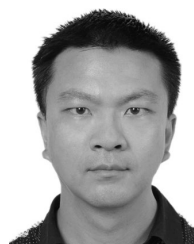
## REFERENCES

- [1] G. Morales, G. Kemper, G. Sevillano, D. Arteaga, I. Ortega, and J. Telles, "Automatic segmentation of *Mauritia flexuosa* in unmanned aerial vehicle (UAV) imagery using deep learning," *Forests*, vol. 9, no. 12, p. 736, 2018.
- [2] A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling," in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, vol. 2, Sep. 1999, pp. 1033–1038.
- [3] Z. Chen, H. Cai, Y. Zhang, C. Wu, M. Mu, Z. Li, and M. A. Sotelo, "A novel sparse representation model for pedestrian abnormal trajectory understanding," *Expert Syst. Appl.*, vol. 138, Dec. 2019, Art. no. 112753.
- [4] M. Bertalmio, L. Vese, G. Sapiro, and S. Osher, "Simultaneous structure and texture image inpainting," *IEEE Trans. Image Process.*, vol. 12, no. 8, pp. 882–889, Aug. 2003.
- [5] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera, "Filling-in by joint interpolation of vector fields and gray levels," *IEEE Trans. Image Process.*, vol. 10, no. 8, pp. 1200–1211, Aug. 2001.
- [6] A. Levin, A. Zomet, and Y. Weiss, "Learning how to inpaint from global image statistics," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, Oct. 2003, p. 305.
- [7] A. Telea, "An image inpainting technique based on the fast marching method," *J. Graph. Tools*, vol. 9, no. 1, pp. 23–34, Jan. 2004.

- [8] Z. Xu and J. Sun, "Image inpainting by patch propagation using patch sparsity," *IEEE Trans. Image Process.*, vol. 19, no. 5, pp. 1153–1165, May 2010.
- [9] K. Duan, Y. Gong, and N. Hu, "Automatic image inpainting using local patch statistics," U.S. Patent 10 127 631, Nov. 13, 2018.
- [10] H. Ting, S. Chen, J. Liu, and X. Tang, "Image inpainting by global structure and texture propagation," in *Proc. 15th Int. Conf. Multimedia*, 2007, pp. 517–520.
- [11] B. Wu, T. Cheng, T. L. Yip, and Y. Wang, "Fuzzy logic based dynamic decision-making system for intelligent navigation strategy within inland traffic separation schemes," *Ocean Eng.*, vol. 197, Feb. 2020, Art. no. 106909.
- [12] X. Shao, Z. Liu, and H. Li, "An image inpainting approach based on the Poisson equation," in *Proc. 2nd Int. Conf. Document Image Anal. Libraries (DIAL)*, Apr. 2006, p. 5.
- [13] J. Xie, L. Xu, and E. Chen, "Image denoising and inpainting with deep neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 341–349.
- [14] J. Bao, D. Chen, F. Wen, H. Li, and G. Hua, "CVAE-GAN: Fine-grained image generation through asymmetric training," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2745–2754.
- [15] L. Xu, J. S. Ren, C. Liu, and J. Jia, "Deep convolutional neural network for image deconvolution," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1790–1798.
- [16] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2536–2544.
- [17] Z. Yan, X. Li, M. Li, W. Zuo, and S. Shan, "Shift-net: Image inpainting via deep feature rearrangement," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 1–17.
- [18] Y. Wang, X. Tao, X. Qi, X. Shen, and J. Jia, "Image inpainting via generative multi-column convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 331–340.
- [19] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
- [20] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5505–5514.
- [21] H. Liu, B. Jiang, Y. Xiao, and C. Yang, "Coherent semantic attention for image inpainting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4170–4179.
- [22] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1125–1134.
- [23] Y. Song, C. Yang, Z. Lin, X. Liu, Q. Huang, H. Li, and C.-C. J. Kuo, "Contextual-based image inpainting: Infer, match, and translate," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.
- [24] M. Jaderberg, K. Simonyan, and A. Zisserman, "Spatial transformer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2017–2025.
- [25] T. Zhou, S. Tulsiani, W. Sun, J. Malik, and A. A. Efros, "View synthesis by appearance flow," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 286–301.
- [26] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 3483–3491.
- [27] J. Walker, C. Doersch, A. Gupta, and M. Hebert, "An uncertain future: Forecasting from static images using variational autoencoders," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 835–851.
- [28] Z. Chen, S. Nie, T. Wu, and C. G. Healey, "High resolution face completion with multiple controllable attributes via fully end-to-end progressive generative adversarial networks," 2018, *arXiv:1801.07632*. [Online]. Available: <http://arxiv.org/abs/1801.07632>
- [29] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. Huang, "Free-form image inpainting with gated convolution," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4471–4480.
- [30] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. A. Efros, "What makes Paris look like Paris?" *Commun. ACM*, vol. 58, no. 12, pp. 103–110, Nov. 2015.
- [31] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*. [Online]. Available: <http://arxiv.org/abs/1312.6114>
- [32] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8789–8797.
- [33] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1452–1464, Jun. 2018.
- [34] R. A. Yeh, C. Chen, T. Y. Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do, "Semantic image inpainting with deep generative models," 2016, *arXiv:1607.07539*. [Online]. Available: <http://arxiv.org/abs/1607.07539>
- [35] M.-C. Sagong, Y.-G. Shin, S.-W. Kim, S. Park, and S.-J. Ko, "PEPSI: Fast image inpainting with parallel decoding network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11360–11368.
- [36] Y. Zeng, J. Fu, H. Chao, and B. Guo, "Learning pyramid-context encoder network for high-quality image inpainting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1486–1494.
- [37] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," 2017, *arXiv:1701.07875*. [Online]. Available: <http://arxiv.org/abs/1701.07875>
- [38] Z. Huang, X. Xu, J. Ni, H. Zhu, and C. Wang, "Multimodal representation learning for recommendation in Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10675–10685, Dec. 2019.
- [39] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 1–14, Jul. 2017.
- [40] J. Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman, "Toward multimodal image-to-image translation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 465–476.
- [41] M. Abadi et al., "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," 2016, *arXiv:1603.04467*. [Online]. Available: <http://arxiv.org/abs/1603.04467>
- [42] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: <http://arxiv.org/abs/1412.6980>
- [43] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
- [44] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 586–595.
- [45] R. Y. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 649–666.
- [46] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 694–711.
- [47] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5767–5777.
- [48] K. Nazeri, E. Ng, T. Joseph, F. Z. Qureshi, and M. Ebrahimi, "EdgeConnect: Generative image inpainting with adversarial edge learning," 2019, *arXiv:1901.00212*. [Online]. Available: <http://arxiv.org/abs/1901.00212>
- [49] Z. Huang, X. Xu, H. Zhu, and M. Zhou, "An efficient group recommendation model with multiattention-based neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published.



**WEIWEI CAI** (Student Member, IEEE) is currently pursuing the master's degree with the Central South University of Forestry and Technology, Changsha, China. Prior to that, he worked with IT industry for more than ten years in the roles of an Enterprise Architect and the Program Manager. His research interests include machine learning, deep learning, and computer vision.



**ZHANGUO WEI** received the Ph.D. degree from the School of Technology, Beijing Forestry University, Beijing, China. He has been working as an Associate Professor with the Central South University of Forestry and Technology, Changsha, China, since 2011. His research interests include information retrieval, data mining, big data, and deep learning. His most work mainly promotes the development of the logistics engineering through the applications of data mining and deep learning.

...