Progressively Unfreezing Perceptual GAN

Jinxuan Sun[1], Yang Chen[1], Junyu Dong[1], and Guoqiang Zhong[1]

Ocean University of China
gqzhong@ouc.edu.cn

Abstract. Generative adversarial networks (GANs) are widely used in image generation tasks, yet the generated images are usually lack of texture details. In this paper, we propose a general framework, called Progressively Unfreezing Perceptual GAN (PUPGAN), which can generate images with fine texture details. Particularly, we propose an adaptive perceptual discriminator with a pre-trained perceptual feature extractor, which can efficiently measure the discrepancy between multi-level features of the generated and real images. In addition, we propose a progressively unfreezing scheme for the adaptive perceptual discriminator, which ensures a smooth transfer process from a large scale classification task to a specified image generation task. The qualitative and quantitative experiments with comparison to the classical baselines on three image generation tasks, i.e. single image super-resolution, paired image-to-image translation and unpaired image-to-image translation demonstrate the superiority of PUPGAN over the compared approaches.

Keywords: Generative adversarial networks, Fine texture details, Image generation, Progressively unfreezing

1 Introduction

In recent years, generative adversarial networks (GANs) [4] have been widely applied to numerous types of image generation tasks, such as single image super-resolution [12,25], image-to-image translation [31,8], image inpainting [28,27] and image deblurring [11,14]. In a nutshell, GANs are a framework to produce a generated distribution to match a given target distribution. The architecture of GANs consists of a generator that produces the generated distribution and a discriminator that evaluates the discrepancy between the generated and real data distributions. The generator and the discriminator are trained alternatively with the adversarial loss. To improve the quality of the generated images, the perceptual loss [9] is usually used to measure the discrepancy of the high-level features on image generation tasks [12,25,30,11]. Nevertheless, the perceptual loss is measured by an external pre-trained convolutional neural network. It generally employs the VGGNet-16/19 [21] pre-trained on the ImageNet dataset [18]. In this case, the perceptual loss mainly focuses on the high-level features that contribute to the specific classification task, and therefore, may perform inferiorly on the other tasks. Moreover, utilizing an external network as the feature extractor ignores the fact that the discriminator in GANs can learn the representations of the images and measure the discrepancy between them.

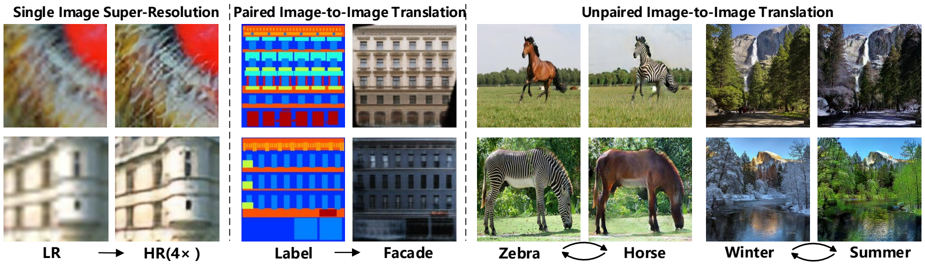
Single Image Super-Resolution
Paired Image-to-Image Translation
Unpaired Image-to-Image Translation
LR
HR(4×)
Label
Facade
Zebra
Horse
Winter
Summer

Fig. 1. The images generated by PUPGAN: (*left*) for single image super-resolution; (*center*) for paired image-to-image translation; (*right*) for unpaired image-to-image translation.

In addition, there is a persisting challenge in the training of GANs [20,1]. When the generated distribution and the real data distribution are perfectly distinguished by the discriminator, the training of the generator comes will be stopped, as the gradient produced by the discriminator is 0. A typical cause of this issue is that the discriminator rapidly overpowers the generator. To address this problem, Sajjadi et al. [19] propose a module to degrade the real data before feeding them to the discriminator, which balances the training of the generator and discriminator. However, this method greatly slows down the learning speed of the discriminator and the whole network.

In this paper, we propose a general framework, named Progressively Unfreezing Perceptual GAN (PUPGAN), which can generate images with fine texture details. Particularly, we propose an adaptive perceptual discriminator architecture, which utilizes a pre-trained dense block [5] as a perceptual feature extractor, and the capability encoded in the percetual feature extractor can be transferred to the current task. The last layer in the feature extractor obtains multi-level features from all preceding layers. Following the perceptual feature extractor, the discriminative learning layers are used to measure the discrepancy between the multi-level features of the generated and real images. In addition, we propose a progressively unfreezing scheme to stabilize the training of PUPGAN. Specifically, we unfreeze the parameters in the perceptual feature extractor layer-by-layer during the training process.

In summary, we make the following contributions in this work:

- We propose a general framework, called PUPGAN, for image generation with fine texture details. It can be applied to various scenarios to improve the texture details of the images generated by GANs-based models.
- We propose an adaptive perceptual discriminator to measure the discrepancy between multi-level features of the generated and real images. In contrast to traditional perceptual loss based on an external network, we fully exploit the capability of the discriminator for multi-level feature extraction.
- We propose a progressively unfreezing scheme for PUPGAN, to smoothly transfer external knowledge learned on large scale applications to the current task.

- We have demonstrated the effectiveness of PUPGAN on several tasks, as shown in Fig. 1. In particular, we have applied PUPGAN to unpaired image-to-image translation and obtained promising results. To our best, this is the first work to efficiently exploit the external knowledge and adapt it to the GANs-based unpaired image-to-image translation.

2 Related Work

Since Goodfellow et al. [4] introduced GANs, they have received more and more attention from the deep learning community. Radford et al. [17] modified the architecture of GANs with convolutional layers [7] to improve their performance and stabilize their training. Following this methodology, Salimans et al. [20] proposed several techniques to encourage the convergence of GANs, such as feature matching and minibatch discrimination.

To further improve the learning capability of GANs, the researchers proposed to train GANs with additional loss functions, such as pixel-wise loss and perceptual loss [9]. The reason behind this is that additional measurement can help to better evaluate the discrepancy between the generated image distribution and the real one. Specifically, the perceptual loss encourages the generated images to have similar high-level features with the real image. Though integrating perceptual loss to GANs has produced impressive results, it depends on the external convolutional neural network (e.g., VGGNet-19 [21]) trained on a specific classification task. As a result, the perceptual loss mainly focuses on high-level features relevant to the original dataset, e.g., ImageNet [18]. Recently, Li et al. [13] proposed a perceptual GAN for small object detection. The discriminator of this perceptual GAN consists of a perception branch. In addition, Wang et al. [24] proposed a perceptual adversarial loss to train an image-to-image transformation network. The VGGNet was adopted as an internal perception function in the discriminator. Moreover, Sungatullina et al. [22] proposed a perceputal discriminator, which embeded a pre-trained feature extractor (VGGNet) inside the discriminator. However, the parameters in the pre-trained feature extractor were fixed during training. These approaches have improved the quality of the generated image to some extent. Nevertheless, the high-level features contribute to the specific classsication task and may perform inferiorly on the other tasks. In this work, we embed a well-trained dense block into the discriminator of the proposed PUPGAN. With the fine-tuning of the adaptive perceptual discriminator, PUPGAN can sufficiently leverage the external knowledge and adapt to the current task.

In order to overcome the unstable training problem of GANs due to the discriminator overpowering the generator, some efforts have been made. Sajjadi et al. [19] introduced a convolutional model to balance the generator and discriminator by gradually revealing the details of the real data. This method encourages a smooth training procedure. However, it slows down the training speed of the entire network. Karras et al. [10] proposed to grow the generator and discriminator progressively by adding new layers to both the generator and discriminator as the training progresses. In this work, we propose a progressively unfreezing scheme for PUPGAN, which is quite different from the previous approaches.

Similar to GANs, PUPGAN can be considered as a general framework for generating images with fine texture details. It can be used for various image generation tasks. Among others, in this work, we have applied PUPGAN to the unpaired image-to-image translation task, which is a very challenging image generation task. Recently, CycleGAN has been applied to this task [31]. However, the generated images are generally lack of texture details. On the contrary, with the adaptive perceptual discriminator, PUPGAN can perform well on the unpaired image-to-image translation task.

3 Progressively Unfreezing Perceptual GAN

In this section, we introduce the proposed Progressively Unfreezing Perceptual GAN (PUPGAN) in detail. We first present an overview of the PUPGAN framework. Then, we describe the architecture of the adaptive perceptual discriminator in PUPGAN. Next, we introduce the progressively unfreezing scheme for the training of PUPGAN. Finally, we specify how to apply PUPGAN to the unpaired image-to-image translation task.

3.1 The PUPGAN Framework

PUPGAN consists of a generator G and an adaptive perceptual discriminator D^{ap}. The generator G is trained to learn a mapping from the input distribution $x \sim p_x$ to the real data distribution p_{data}. For the architecture of the generator G, we just take the generator of the baseline method. Namely, there are no modifications to the generator G. The adaptive perceptual discriminator D^{ap}, which consists of a perceptual feature extractor and the discriminative learning layers, is trained to measure the multi-level feature discrepancy between the real images $y \sim p_{data}$ and the generated images $G(x)$. The learning objective can be written as follows:

$$\begin{aligned}\min_{G}\max_{D^{ap}} V(D^{ap}, G) =& E_{y\sim p_{data}}[\log D^{ap}(y)] \\ &+ E_{x\sim p_x}[\log(1 - D^{ap}(G(x)))].\end{aligned} \tag{1}$$

The generator G and the adaptive perceptual discriminator D^{ap} are trained adversarially to compete with each other. The generator is encouraged to simulate the real images y fed with the input images x. The adaptive perceptual discriminator is trained to distinguish the generated images $\tilde{y}$ $(G(x))$ from the real images y. Hence, the adversarial loss can be defined as:

$$L_{Adv} = -E_{x\sim p_{G(x)}}[\log(D^{ap}(G(x)))]. \tag{2}$$

The adversarial loss encourages the generated distributions to reside on the manifold of the real images by penalizing the discrepancy between the generated and real images.

3.2 The Adaptive Perceptual Discriminator

Unlike previous GANs using an external classifier to compute the perceptual loss, we use the discriminator of PUPGAN to evaluate the feature discrepancy between the generated and real images.

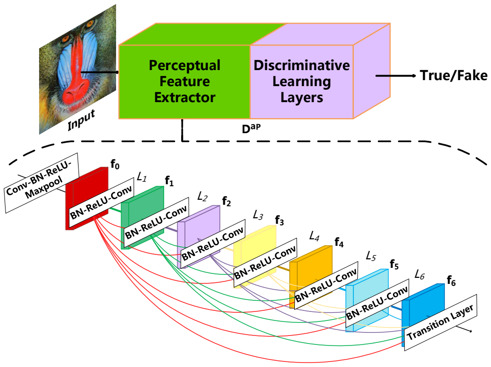

Fig. 2. The adaptive perceptual discriminator of PUPGAN.

In this paper, we propose an adaptive perceptual discriminator, which is composed of a perceptual feature extractor and some discriminative learning layers (as shown in Fig. 2). Specifically, the perceptual feature extractor employs the hidden layer of the discriminator to extract the perceptual features and the discriminative learning layers are used to measure the feature discrepancy. More importantly, the perceputal feature extractor in the adaptive perceptual discriminator can be learned along with the training of the PUPGAN.

The perceptual feature extractor consists of a 7×7 convolution, a 3×3 max pooling, and a 6-layer dense block where each layer consists of a sequence of 1×1 convolutions and a 3×3 convolutions. Before each convolution, a batch normalization and the ReLU activation are adopted. Specifically, the 6-layer dense block is taken from Dense-121 [5], which is pre-trained on the ImageNet classification task [18]. The dense skip connections guarantee the forwarding of multi-level features to the discriminative learning layers. Furthermore, for error back propagation, the skip connections ensure the gradients from the discriminative learning layers and unfreezed layers to pass on to the generator. Hence, the discriminative learning layers measure a multi-level discrepancy between the generated and real images.

The perceptual feature extractor in the discriminator performs a transfer learning for the current image generation task. As a result, the perceptual features extracted by the adaptive perceptual discriminator is more suitable for the current task than the traditional perceptual loss and models dirctly embeded the VGGNet into the discriminators. In addition, compared to the traditional perceptual loss with element-wise measuring methods, using the discriminative learning layers can better measure the discrepancy of the perceptual features.

3.3 Progressively Unfreezing Scheme

Three Ways for Employing the Feature Extractor The pre-trained perceptual feature extractor in the adaptive perceptual discriminator can be employed in three ways. The first way is to update all the parameters in the perceptual feature extractor with the entire model, which we called normal transformation. The second way is to freeze all the parameters in the perceptual feature extractor, which we called no transformation (same as [22]). The third way is a compromise between them, which progressively unfreezes the parameters in the perceptual feature extractor layer-by-layer. We call it progressive transformation.

For the first way, at the beginning of training, since the discriminator has strong discriminative capability, it is trial to find the split hyperplane between the generated images and the real images, whereas the generator with low generative capability cannot fool the discriminator (as mentioned in Section 1). As a result, the gradient produced by the discriminator may lead to the gradient vanishing or mode collapse problems.

For the second way, when the parameters of the pre-trained perceptual feature extractor are freezed, its learning capability is restricted. In other words, the provided perceptual feature extractor is pre-trained on the ImageNet classification task, its performance is limited on the current image perceptual feature extraction task.

Here, we choose the third way, i.e. progressive transformation, which progressively unfreezes the pre-trained perceptual feature extractor in the adaptive perceptual discriminator.

Progressively Unfreezing with the Unfreezing Factor φ To achieve the progressive transformation, we propose a progressively unfreezing scheme, which depends on an unfreezing factor φ.

The training process is shown in Algorithm 1. At the beginning of training, the parameters of the perceptual extractor are all freezed. As the training advances, the parameters of the bottom layer in the perceptual feature extractor that connects to the discriminative part are first unfreezed. Next, we progressively unfreeze the front layers one-by-one depending on a randomly generated probability. If the probability larger than a threshold (we empirically set the threshold $\varphi = 0.66$, please refer to the supplementary material), we unfreeze the parameters of one convolutional layer in front of the unfreezed layer. Namely, the unfreezed layers in the perceptual feature extractor and the discriminative learning layers remain trainable, while the parameters of the other layers in the perceptual feature extractor are freezed. Hence, the perceputal feature extractor smoothly adapt to the current image generation task.

There are several benefits for our progressively unfreezing scheme. First, the generation at the beginning is stable, because the adaptive perceptual discrim-

Algorithm 1 The progressively unfreezing scheme.

```
Require: $\theta_G$: $G$'s parameters; $\theta_p$: perceptual feature extractor's parameters: $\theta_d$: discriminative learning layers' parameters; $m$: batchsize; $\varphi$: unfreezing factor (empirically set to 0.66).
Require: Adam hyperparameters: $\alpha_p = 1 \times 10^{-6}$, $\alpha_d = 2 \times 10^{-4}$, $\beta_1 = 0.5$ and $\beta_1 = 0.999$.
Freeze all of the parameters $\theta_p$
for e=1, 2, ..., epoch do
    $p = \text{random}(1)$
    if $p > \varphi$ then
        Unfreeze one layer in the perceptual feature extractor which is the nearest neighbor to the discriminative learning layers.
    end if
    for j=1, 2, ..., datasize/$m$ do
        Sample $y_1, \ldots, y_m$ from the real data distribution
        Sample $x_1, \ldots, x_m$ from the input data distribution
        $L_{D^{ap}} \leftarrow \frac{1}{m} \sum [logD^{ap}(y_i) + log(1 - D^{ap}(G(x_i)))]$
        $L_G \leftarrow \frac{1}{m} \sum [logD^{ap}(G(x_i))]$
        $\theta_d, \theta_p = Adam(L_{D^{ap}}, \theta_d, \theta_p, \alpha_d, \alpha_p, \beta_1, \beta_2)$
        $\theta_G = Adam(L_{D^{ap}}, \theta_G, \beta_1, \beta_2)$
    end for
end for
```

inator with most parameters freezed is easy to fool by the generator. As the training progresses, the generator is asked for a more complicated question than the previous one by unfreezing the parameters in the adaptive perceptual discriminator. In addition, it is a reliable progressively training scheme, which can successfully avoid the gradient vanishing and model collapse problems.

Another benefit is that unfreezing the parameters of the perceptual feature extractor layer-by-layer can reduce the training time. Unlike normal transformation, when progressively unfreeze the parameters of the perceptual feature extractor, most parameters of the convolutional layers are freezed without update. In other words, the pre-trained feature extractor is incrementally transformed to that for the current task. In contrast to normal transformation, which needs to update all the parameters in the perceptual feature extractor, the progressive transformation method is more efficient.

In addition, the dense block is extremely suitable to the progressively unfreezing scheme. The skip connections between layers make it possible for information flow between the layers. Moreover, compared with VGGNet, the dense block has fewer parameters and better feature extraction capability.

3.4 PUPGAN for Unpaired Image-to-Image Translation

In GANs' research field, paired examples are generally required to train the network, as the discriminator needs both the generated images and the real images. CycleGAN [31] made a breakthrough on the unpaired image-to-image translation task, which introduced a cycle consistency loss to measure the discrepancy between the inputs with the reversely generated images. Although it has achieved competitive results on image-to-image translation task, the generated images are usually lack of texture details. Besides, the traditional perceptual loss, which needs both real images and generated images to measure the feature discrepancy, cannot work on the tasks without the pair-wise images.

The adaptive perceptual discriminator in PUPGAN extracts the multi-level perceptual features and measures the discrepancy between them. It overcomes the shortcomings of the traditional perceptual loss that needs pair-wise images to measure the feature discrepancy. Furthermore, the adaptive perceptual discriminator in PUPGAN efficiently extract and measure the multi-level feature discrepancy between the generated images and the real images. Hence, it can lead to better texture details and perceptual characteristics than the images generated by CycleGAN.

4 Experiments

To verify the universal validity of PUPGAN, we evaluate the performance of PUPGAN on several image generation tasks. Specifically, we choose three representative image generation tasks, i.e. single image super-resolution, paired image-to-image translation and unpaired image-to-image translation, for each task, we choose the classical baseline methods for comparison. For any other state-of-the-art GANs-based model, we can replace its discriminator to the adaptive perceptual discriminator designed in this work, and easily compare with it on the learning tasks.

In this section, we show some representative experimental results. For more implementation details and experimental results, please refer to the supplementary materials.

4.1 Datasets and Evaluation Metrics

For the single image super-resolution, we evaluated PUPGAN on several widely used benchmark datasets, i. e. Set5 [2], Set14 [29], BSD100 [15] and Urban100 [6]. All experiments were performed with a scale factor of 4× between the low-resolution and high-resolution images. Meanwhile, for quantitative comparison, we evaluated the generated images in terms of peak signal to noise ratio (PSNR) and structural similarity idex (SSIM) [26].

For the paired image-to-image translation, we conducted experiments on several tasks, such as aerial→map with training data scraped from Google Maps [8], label→facade with images from the CMP facade dataset [23]. Moreover, we evaluated the generated images by PUPGAN and the compared methods in terms of PSNR and SSIM.

For the unpaired image-to-image translation, we chose several tasks performed in CycleGAN [31]. The tasks included aerial↔map [8], label↔facade [23], horse↔zebra and orange↔apple with images downloaded from ImageNet [18] and summer↔winter Yosemite photos from Flickr (`http://www.flickr.com`). In addition, for quantitative comparison, we use the perceptual quality index (PQI) [3] to measure the quality of the generated images.

4.2 Parameter Settings

To demonstrate the effectiveness of PUPGAN, we conducted the ablation study. The adaptive perceptual discriminator in PUPGAN is represented as "Dense_D". For the perceptual feature extractor, we took a pre-trained VGGNet-19 as the variant, represented as "VGG_D". For the training scheme, we considered to progressively unfreeze the parameters in the perceptual extractor layer-by-layer or freeze all the parameters of the perceptual feature extractor, represented as "UF" or "F" respectively. While unfreezing all the parameters at the beginning of training may cause the gradient vanishing or mode collapse problems, it is not considered here. In addition, for the regularization mechanism, we conducted experiments that with or without spectral normalization [16] after each convolutional layer in the discriminator. The spectral normalization is represented as "SN".

The generators of PUPGAN for the applied tasks are the same as that of the baseline models. The adaptive perceptual discriminators of PUPGAN for different tasks are designed with their unique discriminative learning layers. The specific architectures are described in the following subsections and in the supplementray material.

4.3 Single Image Super-Resolution

For the single image super-resolution task, we take SRGAN [12] as the baseline. SRGAN is a famous GANs-based model for single image super-resolution. It utilizes a VGGNet loss which is based on a pre-trained VGGNet-19 network, and measures the high-level discrepancy with the Euclidean distance.

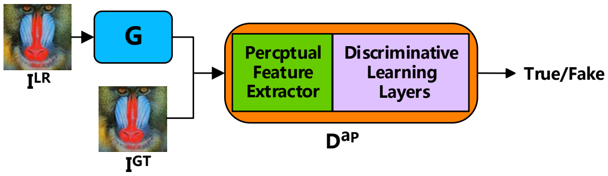

Fig. 3. The architecture of PUPGAN for single image super-resolution.

Fig. 3 illustrates the architecture of PUPGAN for single image super-resolution. The discriminative learning layers consist of three 3×3 convolutions with stride 2 followed by average pooling and two 1×1 convolutions with stride 1. The convolutional layers except the last one are followed by a LeakyReLU activation ($\alpha = 0.2$) and spectral normalization.

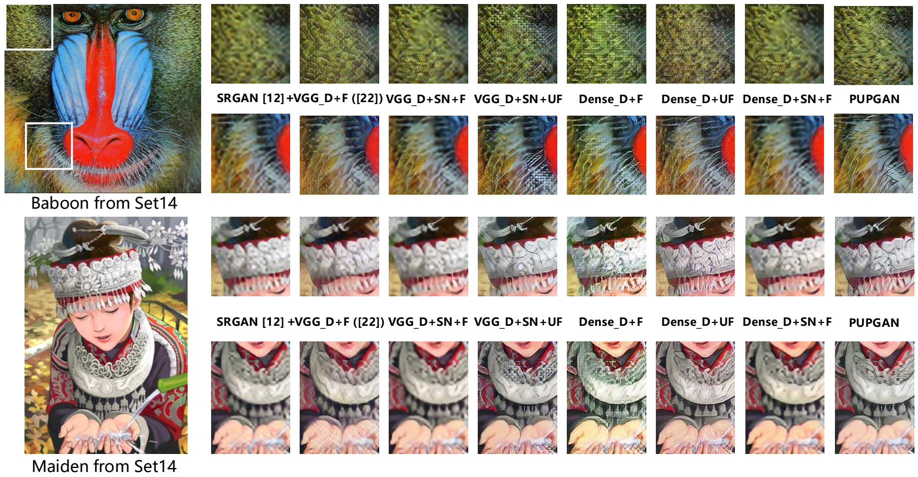

Fig. 4. From left to right: ground-truth, SRGAN [12], SRGAN + VGGNet-based discriminator with freezed parameters, SRGAN + VGGNet-based discriminator + SN with freezed parameters, SRGAN + VGGNet-based discriminator + SN with progressively unfreezing scheme, SRGAN + adaptive perceptual discriminator with freezed parameters, SRGAN + adaptive perceptual discriminator with unfreezed parameters, SRGAN + adaptive perceptual discriminator + SN with freezed parameters and PUPGAN.

Fig. 4 demonstrates the 4× upscaling super-resolved images generated by the baseline SRGAN, PUPGAN and the other compared approaches. It is obvious that the images generated by SRGAN are blurry and without enough texture details. In general, the images generated by the models with "VGG_D" is more blurry than those with "Dense_D". Furthermore, utilizing progressively unfreezing scheme further enhanced the generative capability of the networks. It is worth mentioning that PUPGAN achieved the best performance. It not only recovered the texture details of the images, but also kept the textures approximate to the real texture in the ground-truth images.

Table 1 illustrates the quantitative results on several benchmark datasets. PUPGAN obtained the highest score among the other compared approaches. Even though SRGAN received the second-highest score, the images generated by it is blurry (as shown in Fig. 4).

4.4 Paired Image-to-Image Translation

For the paired image-to-image task, we take Pix2Pix [8] as the baseline. Pix2Pix is a generic tool for paired image-to-image translation tasks, which utilized a conditional GAN-based model to achieve the reasonable results. It optimizes the entire model with a generative adversarial loss and an MAE loss between the generated images and the real images.

Table 1. Performance of the baseline SRGAN [12], PUPGAN and other variants for single image super-resolution in terms of PSNR/SSIM on the Set5, Set14, BSD100 and Uber100 datasets.

Datasets / Methods	Set5 [2]		Set14 [29]		BSD100 [15]		Uber100 [6]	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
SRGAN [12]	28.5694	0.8271	25.4943	0.7266	25.6282	0.6935	23.4113	0.7099
SRGAN [12]+VGG_D+F ([22])	24.2069	0.6185	22.7773	0.5455	22.8579	0.5262	21.3465	0.5761
SRGAN [12]+VGG_D+SN+F	27.5899	0.7686	25.1665	0.7249	25.3625	0.6769	23.0150	0.7020
SRGAN [12]+VGG_D+SN+UF	24.5858	0.7386	22.1574	0.6247	22.8857	0.6134	20.5188	0.6169
SRGAN [12]+Dense_D+F	23.3219	0.6699	21.4114	0.5920	21.8136	0.5455	19.7500	0.5463
SRGAN [12]+Dense_D+SN+F	27.6365	0.8092	25.0694	0.7184	25.1767	0.6832	22.9457	0.6892
SRGAN [12]+Dense_D+UF	24.3997	0.6643	22.4456	0.5933	22.5424	0.5569	20.9289	0.6037
PUPGAN	**29.0598**	**0.8495**	**25.7471**	**0.7454**	**25.8316**	**0.7074**	**23.5533**	**0.7297**

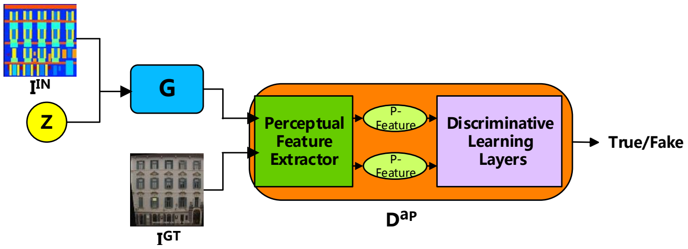

Fig. 5. The architecture of PUPGAN for paired image-to-image translation.

Fig. 5 illustrates the structure of PUPGAN for paired image-to-image translation. Notice that in the Pix2Pix model, the generated image and the real image are concatenated as the input to the discriminator. While for the adaptive perceptual discriminator, the generated image and the real image first go through the perceptual feature extractor, and then the perceptual features are concatenated as the input to the discriminative learning layers. Here, the discriminative learning layers consist of four 3×3 convolutions with stride 2 and a 4×4 convolution with stride 1. The convolutional layers are followed by a LeakyReLU activation ($\alpha = 0.2$) and spectral normalization.

Fig. 6 shows the images generated by the baseline Pix2Pix, PUPGAN and the other compared approaches on the label→facade and aerial→map tasks. The image details generated by PUPGAN is much better than the other compared approaches. It should be noted that the images generated by PUPGAN are the closest to the ground-truth image, no matter the color of the objects or the fine texture details.

Table 2 illustrates the quantitative results for the paired image translation tasks. PUPGAN obtained the highest score among the other compared approaches. In addition, we can see that taking the dense block as the perceptual

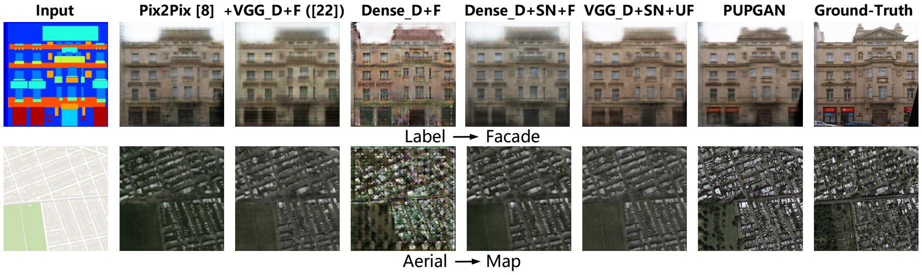

Fig. 6. The comparison between different methods for the paired image translation task. Pix2Pix [8] is the baseline method.

Table 2. Performance of PUPGAN, the baseline Pix2Pix [8] and other variants for the paired image translation tasks in terms of PSNR and SSIM.

Datasets Methods	Label→Facade		Aerial→Map	
	PSNR	SSIM	PSNR	SSIM
Pix2Pix [8]	18.8356	0.4955	17.4289	0.3749
Pix2Pix [8]+VGG_D+F ([22])	17.4117	0.3691	18.2702	0.3954
Pix2Pix [8]+Dense_D+F	16.8993	0.3560	15.2598	0.3554
Pix2Pix [8]+VGG_D+SN+F	18.8696	0.4794	18.2852	0.3971
Pix2Pix [8]+Dense_D+SN+F	19.4678	0.5234	17.8319	0.3980
Pix2Pix [8]+VGG_D+SN+UF	18.6036	0.4902	17.3374	0.3668
PUPGAN	**21.3647**	**0.6335**	**19.7952**	**0.48649**

feature extractor performs better than that taking the VGGNet as the perceptual feature extractor (from the last two lines in Table 2). It demonstrates that the dense connections in the feature extractor is extremely suitable to the progressively unfreezing scheme.

4.5 Unpaired Image-to-Image Translation

For the unpaired image-to-image translation task, we take CycleGAN [31] as the baseline. CycleGAN is a well-known tool for unpaired image-to-image translation task. It contains both forward and reverse mapping functions between the source domain and target domain. It is optimized with two adversarial losses and a cycle consistency loss. However, in this case, the traditional perceptual loss cannot work, as the target image corresponding to a generated image is unknown. Although Sungatullina et al. [22] embeded a pre-trained VGGNet inside the discriminator to extract the high-level features, the parameters in the pre-trained network were fixed, whereas the adaptive perceptual discriminator can help to learn the suitable high-level perceptual features adapted to the current task and generate images with fine texture details.

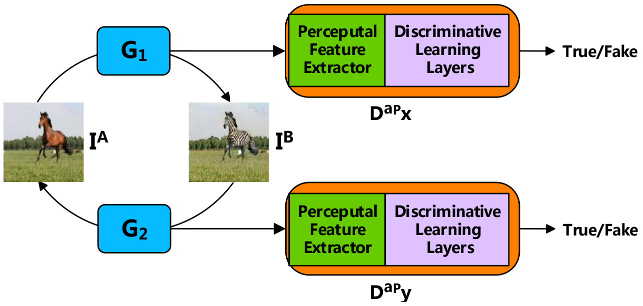
G1
G2
I^A
I^B
Perceptual Feature Extractor
Discriminative Learning Layers
True/Fake
D^ap x
Perceptual Feature Extractor
Discriminative Learning Layers
True/Fake
D^ap y

Fig. 7. The architecture of PUPGAN for unpaired image-to-image translation.

Fig. 7 illustrates the structure of PUPGAN for unpaired image-to-image translation. The structures of the discriminative learning layers are as same as we illustrated in Section 4.4.

We first test the performance of PUPGAN and the compared approaches for unpaired image-to-image translation on paired datasets, where ground-truth input-output pairs are available for evaluation. For training, we shuffled the images to made them unpaired. Fig. 8 shows the label→facade results obtained by PUPGAN and the other compared approaches. It is easy to see that, the images generated by PUPGAN are much better than that generated by CycleGAN and the compared approaches, whether from the contour of the facades or the texture details in the generated images.

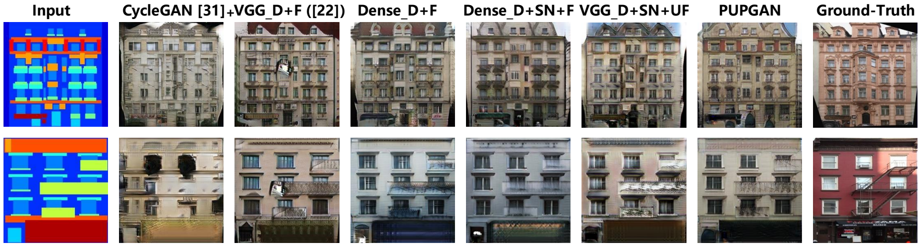

Fig. 8. The comparison for the unpaired image-to-image translation on label→facade task. We took CycleGAN [31] as the baseline and conduct extensive experiments on the variants of PUPGAN.

In addition, we conducted experiments on the unpaired image-to-image translation task, horse↔zebra. As shown in Fig. 9, images generated by PUPGAN have fine texture details and correct color. Specifically, even the zebra's mane was painted in white and black by PUPGAN, while the compared approaches did not have this property.

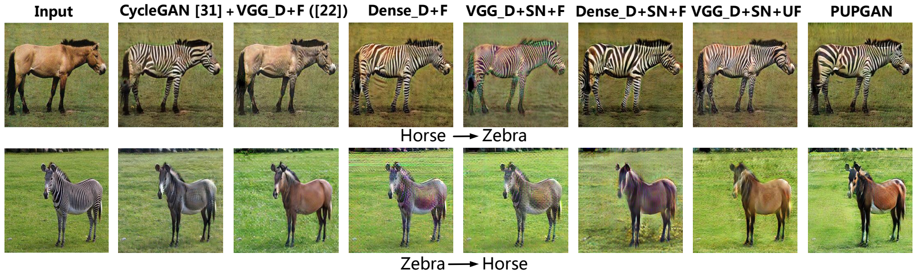

Fig. 9. The comparison for the unpaired image-to-image translation on horse↔zebra task, where paired data does not exist.

Table 3. Performance of PUPGAN, the baseline CycleGAN [31] and other variants for unpaired image translation tasks in terms of PQI.

PQI	CycleGAN [31]	VGG_D+F ([22])	Dense_D+F	VGG_D+SN+F	VGG_D+SN+UF	Dense_D+SN+F	PUPGAN
Label→Facade	5.0085	4.8144	5.8095	4.4230	4.1031	5.2549	**4.0308**
Facade→Label	6.6129	4.9482	5.2549	4.6094	4.4423	4.4885	**4.3692**
Horse→Zebra	3.3219	3.2865	3.2363	3.4095	3.5321	3.2862	**2.8842**
Zebra→Horse	3.3278	3.2679	3.2979	3.525	3.2938	3.2862	**3.2109**

Table 3 shows the quantitative results for the label↔facade and horse→zebra tasks in terms of PQI. The lower the PQI value, the higher the perceptual quality recovered by the method. It obvious that PUPGAN obtained the best result among the other compared methods. For more experimental results, please refer to the supplementary materials.

5 Conclusion

In this paper, we propose a new framework called PUPGAN, which can generate images with fine texture details. Its adaptive perceptual discriminator, which utilizes a pre-trained dense block as the perceptual feature extractor and transfers it to the current task, efficiently measures the multi-level discrepancy between the generated and real images. In addition, we propose a progressively unfreezing scheme to smoothly improve the generator's image generation capability. The qualitative and quantitative experiments demonstrate the effectiveness of PUPGAN for texture details generation on three representative image generation tasks, i.e. single image super-resolution, paired image-to-image translation and unpaired image-to-image translation. Last but not the least, PUPGAN can be considered as a general framework for image generation with fine texture details, it can be applied to more image generation tasks other than that we performed here.

References

1. Arjovsky, M., Bottou, L.: Towards principled methods for training generative adversarial networks. In: ICLR (2017)
2. Bevilacqua, M., Roumy, A., Guillemot, C., Alberi-Morel, M.: Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In: BMVC. pp. 1–10 (2012)
3. Blau, Y., Michaeli, T.: The perception-distortion tradeoff. In: CVPR. pp. 6228–6237. IEEE Computer Society (2018)
4. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A.C., Bengio, Y.: Generative adversarial networks. CoRR **abs/1406.2661** (2014)
5. Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: CVPR. pp. 2261–2269 (2017)
6. Huang, J., Singh, A., Ahuja, N.: Single image super-resolution from transformed self-exemplars. In: CVPR. pp. 5197–5206 (2015)
7. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: ICML. pp. 448–456 (2015)
8. Isola, P., Zhu, J., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: CVPR. pp. 5967–5976 (2017)
9. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: ECCV. pp. 694–711 (2016)
10. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. In: ICLR (2018)
11. Kupyn, O., Budzan, V., Mykhailych, M., Mishkin, D., Matas, J.: Deblurgan: Blind motion deblurring using conditional adversarial networks. In: CVPR. pp. 8183–8192 (2018)
12. Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A.P., Tejani, A., Totz, J., Wang, Z., Shi, W.: Photo-realistic single image super-resolution using a generative adversarial network. In: CVPR. pp. 105–114 (2017)
13. Li, J., Liang, X., Wei, Y., Xu, T., Feng, J., Yan, S.: Perceptual generative adversarial networks for small object detection. In: CVPR. pp. 1951–1959 (2017)
14. Lu, B., Chen, J., Chellappa, R.: Unsupervised domain-specific deblurring via disentangled representations. In: CVPR. pp. 10225–10234 (2019)
15. Martin, D.R., Fowlkes, C.C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: ICCV. pp. 416–425 (2001)
16. Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral normalization for generative adversarial networks. In: ICLR (2018)
17. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. In: ICLR (2016)
18. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M.S., Berg, A.C., Li, F.: Imagenet large scale visual recognition challenge. International Journal of Computer Vision **115**(3), 211–252 (2015)
19. Sajjadi, M.S.M., Parascandolo, G., Mehrjou, A., Schölkopf, B.: Tempered adversarial networks. In: ICML. pp. 4448–4456 (2018)
20. Salimans, T., Goodfellow, I.J., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. In: NIPS. pp. 2226–2234 (2016)

21. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015)
22. Sungatullina, D., Zakharov, E., Ulyanov, D., Lempitsky, V.S.: Image manipulation with perceptual discriminators. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV. Lecture Notes in Computer Science, vol. 11210, pp. 587–602. Springer (2018)
23. Tylecek, R., Sára, R.: Spatial pattern templates for recognition of objects with regular structure. In: GCPR. pp. 364–374 (2013)
24. Wang, C., Xu, C., Wang, C., Tao, D.: Perceptual adversarial networks for image-to-image transformation. IEEE Trans. Image Processing **27**(8), 4066–4079 (2018)
25. Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y., Loy, C.C.: ESRGAN: enhanced super-resolution generative adversarial networks. In: ECCV. pp. 63–79 (2018)
26. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Processing **13**(4), 600–612 (2004)
27. Yeh, R.A., Chen, C., Lim, T., Schwing, A.G., Hasegawa-Johnson, M., Do, M.N.: Semantic image inpainting with deep generative models. In: CVPR. pp. 6882–6890 (2017)
28. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S.: Generative image inpainting with contextual attention. In: CVPR. pp. 5505–5514 (2018)
29. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: ICCS. pp. 711–730 (2010)
30. Zhang, H., Sindagi, V., Patel, V.M.: Image de-raining using a conditional generative adversarial network. CoRR **abs/1701.05957** (2017)
31. Zhu, J., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: ICCV. pp. 2242–2251 (2017)