Neural Processing Letters (2020) 51:2669–2684
https://doi.org/10.1007/s11063-020-10215-w

Gaussian Pyramid of Conditional Generative Adversarial Network for Real-World Noisy Image Denoising

Ruijun Ma[1] · Bob Zhang[1] · Haifeng Hu[2]

Published online: 6 March 2020

Abstract
Image denoising is an essential and important pre-processing step in digital imaging systems. However, most of existing methods are not adaptive in real-world applications due to the complexity of real noise. To address this problem, a novel pyramidal generative structural network (PGSN) is proposed for robust and efficient real-world noisy image denoising. Specifically, we consider the denoising problem as a process of image generation. The procedure is to first build a Gaussian pyramid where a cascade of encoder-decoder networks are used to adaptively capture multi-scale image features and progressively reconstruct the corresponding noise-free image from coarse to fine granularity. Then, we train a conditional form of GAN at each pyramid level. By integrating the conditional GAN approach into the Gaussian pyramid, the proposed network can well combine the image features from different pyramid levels, and an incremental distinction between the real noise and image details is dynamically built up, hence greatly boosting the denoising performance. Extensive experimental results demonstrate that our PGSN gives satisfactory denoising results, and achieves superior performance against the state-of-the-arts.

Keywords Image denoising · Real-world noisy images · Gaussian pyramid · Generative model

1 Introduction

Image denoising aims at recovering the clean image from its noisy observation. Over the past few decades, a considerable amount of methods have been extensively studied in literature, e.g., [1–15]. These studies mainly concentrate on additive white Gaussian noise (AWGN)

✉ Haifeng Hu
huhaif@mail.sysu.edu.cn

Ruijun Ma
yb97442@um.edu.mo

Bob Zhang
bobzhang@um.edu.mo

[1] PAMI Research Group, University of Macau, Macau, China

[2] School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou, China

removal, and assume that the noise can be modeled with Gaussian or mixture of Gaussian (MoG) distribution. Despite the promising denoising results, most of them can either be less effective, or lack flexibility for complex noise, especially when dealing with real-world noisy images. In fact, the noise in real-world noisy images is much more complex than AWGN and MoG noise. This is because that in the in-camera imaging process, the realistic noise comes from multiple sources and varies in different sensors, cameras, camera settings and the image acquiring environment [16–18]. As a result, most of existing denoising methods may still not be flexible enough to deal with real noisy images directly.

In recent years, several approaches [19–24] have been proposed to cope with realistic noise in real images. Among such methods, [19–22] focus mainly on noise modeling, where the noise model is estimated by using the multivariate Gaussian or mixture of Gaussian (MoG) distribution. However, these methods would remove noise incompletely, or introduce visible artifacts. More recently, dictionary learning and sparse coding (SC) have exhibited a remarkable capability for practical image denoising problems [23,24]. Their analysis follows a two-step framework. First, they employ weighted SC to better exploit sparsity priors of natural images. Then, by solving the sparse system, they are able to enforce sparse regularization on the noise information at each channel of color images, achieving better performance on removing unknown noise from real images. Nevertheless, even though the statistics of realistic noise can be characterized adaptively, the learned image priors are modeled on a specific model explicitly and heavily rely on human knowledge, providing some leeway to fully capture the fine-scale image characteristics.

Recently, deep neural networks have been widely used and advanced many computer vision tasks like image retrieval [25], image captioning [26], image recognition [27] and so on. Particularly, they could leverage the benefits of deep neural architecture and external large-scale datasets to effectively learn meaningful image features without the conjunction of human knowledge of image priors. For the denoising problem, deep neural network architectures based on discriminative learning have also been proposed to learn the distinction between image details and noise [15,28–31]. Therefore, the adaption of discriminative learning model can break through the limitations of the previous approaches, and contribute to the success of denoising. However, one nonnegligible drawback of many existing network methods is that they usually focus on more deeper and larger convolutional neural network (CNN) design. That is to say, they need to train a huge number of network parameters and gain the optimal solution to better learn the latent feature representations of noise. As a result, they may involve a complex rebalancing of computational efficiency as well as denoising quality, and can hardly satisfy the actual application.

To overcome the above drawbacks, we propose a pyramidal generative structural network (PGSN) for real-world noise removal. Our PGSN employs two modules: a denoising module and a discriminative module. The denoising module first repeatedly downsamples the noisy image by a factor of two, producing a series of decreasing downscaled images to construct a Gaussian pyramid. We start from the next to last pyramid level, and adopt an encoder-decoder network to combine the last two pyramid images to reconstruct the sub-band clean image. Subsequently, the reconstructed image is upsampled and fed into the subsequent level for improvement. We continue this process of reconstruction until we get back to the first level. Our discriminative module is utilized to enforce the features between the recovered and ground-truth noise-free images to be as similar as possible. In addition, we use the conditional generative adversarial network (CGAN) approach to better regularize the training process of the denoising. The network architecture of the proposed approach is simple yet flexible enough to handle real-world image denoising tasks. Figure 1 provides an illustration of denoising results. Another prominent characteristic of our PGSN is that we

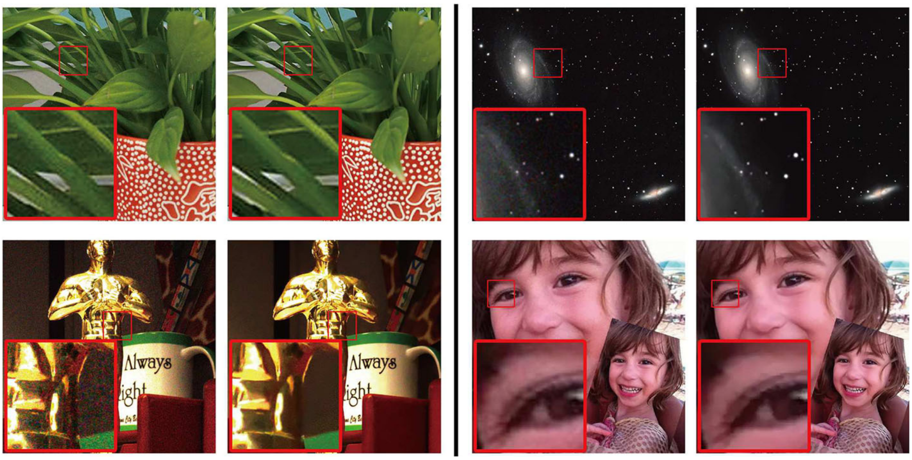


Fig. 1 Denoising results of our PGSN model. In each panel from left to right: noisy images, our denoising results. Note that the ground-truth clean image of the noisy input is not available. Better zoomed-in on screen

break the denoising into a succession of more manageable refinements, each of which has potential in reducing the complexity of the overall optimization procedure, allowing a more stable network training and enforcing better denoising performance. The contributions of this paper are summarized as follows:

- A stacked convolutional learning algorithm is developed for real-world noisy image denoising by combining the Gaussian pyramid with CGAN, which effectively and efficiently simplifies the denoising process and achieves excellent performance.
- We formulate the problem of image denoising as a process of image generation and devise a multi-scale decomposition and progressive reconstruction architecture. This approach can significantly improve the learning of the discriminative representation between image details and noise. More importantly, it provides a better guidance to the reconstruction of clean image from coarse to fine granularity.
- We evaluate the proposed model on three datasets. The further experimental results show that our PGSN performs robustly for real-world noisy images. And it's worth noting that PGSN outperforms the state-of-the-art methods, with which we can effectively preserve fine-scale structural details of image while removing realistic noise.

2 Related Work

2.1 Image Denoising

A wide variety of approaches have been proposed for image denoising. Generally speaking, they fall into two groups, with one group mainly focusing on AWGN or MoG noise while the other adopting parameter estimation techniques or image prior learning for noise in real photographs.

For the first group, based on the properties of images, wavelet and curvelet transforms [1,2]are introduced for denoising. Later, [3–6] propose sparse coding and dictionary learning for more complicated image denoising tasks by exploiting the sparsity of natural images. In

[7,8], the authors propose non-local means algorithm for AWGN image denoising. Compared with predefining image priors in some transformed domains, learning image priors from natural images has been developed to recover the clean image from the noisy input and proved to achieve encouraging denoising results [9–12]. Recently, neural network has also gained much attention in the field of image denoising, among which the representative works include the methods proposed in [13–15]. In particular, Chen et al. [14] propose a trainable nonlinear reaction diffusion (TNRD) learning framework for Gaussian image denoising. In [15], the authors devise a residual learning scheme to distinct the noisy observation from the clean images by a feed-forward CNN, receiving state-of-the-art Gaussian denoising performance. Although the methods mentioned above have paid a lot of efforts in learning the correlation between noise and the clean image and facilitated the process of image denoising, they are still limited in AWGN or MoG noise removal and found inflexible to estimate the real noise model.

For the second group, recent papers have sought deeper understanding of realistic noise in real photography. Typically, Lebrun et al. [19] propose a multiscale noise clinic (NC) algorithm for image denoising by adopting the NL-bayes means filter [7]. In [20], the authors investigate the influence of in-camera image processing [32] on noise and use a data-driven approach to estimate the noise parameters for image denoising. Moreover, neat image (NI) [21] is developed to reduce the complex noise by estimating the noise model. Zhu et al. [22] build a “dependent Dirichlet process tree” and model each group of patches of the given noisy image with MoG distribution. However, these approaches [19–22] take the common idea that the noise in real images follows Gaussian or MoG distribution while it has been found that noise in real images can be more complex and may not be well modeled by explicit distributions. More recently, Xu et al. [23,24] exploit the beneficial relationship between realistic noise and image priors, and utilize it to perform real-world noisy image denoising. Despite their high denoising performance, both methods are defined based on a learned sparse model. The over-reliance on human knowledge of image priors makes them unable to faithfully explore full image textures and structural information when removing the real noise. Lately, Chen et al. [31] propose a GAN-CNN based blind denoiser (GCBD) for real noise removal task. This method leverages the benefits of a generative model for noise modelling then uses the generated noise samples for image denoising. However, their generated noisy samples are assumed to be additive noise with zero mean, which is too ideal for practical image denoising problems because it is still not entirely clear whether such an explicit distributions can well characterize the property of realistic noise.

2.2 Conditional Generative Adversarial Network (CGAN)

In recent years, generative adversarial network (GAN) [33] has made considerable progress on synthesizing impressive photorealistic images via a min-max two-player game between two networks: a generative network and a discriminative network. However, GAN is an unconditioned generative model, with which the generated samples cannot be controlled during training and is more likely to lack diversity. To conquer this dilemma, Mirza et al. [34] propose the conditional generative adversarial network (CGAN), which is an extension of original GAN. CGAN feeds additional information, i.e., class labels or data from other modalities, to both the generator and discriminator and aims at controlling the generative model with the conditioning variable. In contrast to GAN, CGAN can not only stabilize the learning processing but also enhance the descriptive power of generative network. Many successful vision tasks, such as image fusion [35], image synthesis [36], image deblurring [37], etc., benefit from CGAN and achieve significant success.

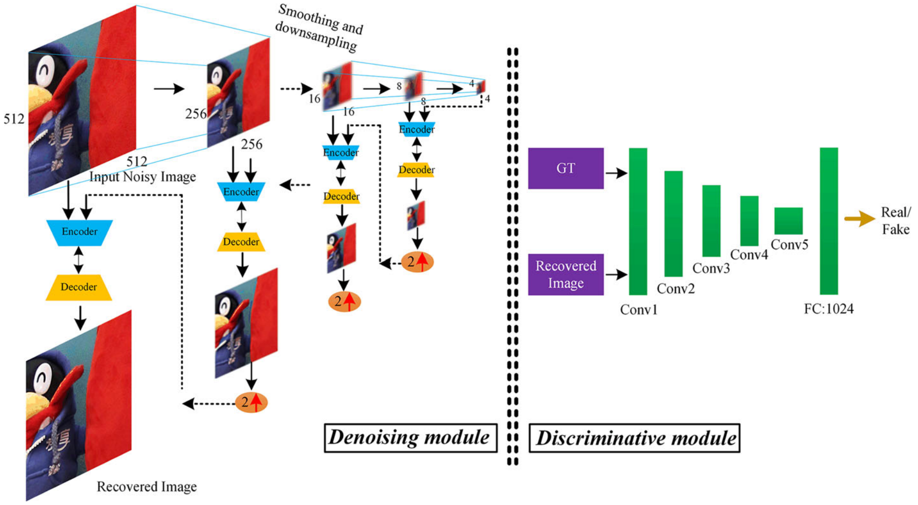


Fig. 2 The architecture of our PGSN

In our work, we apply the CGAN method and combines it with a Gaussian pyramid for real-world noisy image denoising. Such a mechanism has two prominent features. First, this framework makes it suitable to capture the multi-scale structure of images and denoise the real image in a coarse-to-fine manner, from which fine-scale image details can be faithfully preserved. Second, it allows us to break the denoising procedure into a succession of more manageable stages, making the denoising easier and more efficient. As we will see later in this paper, the integration of CGAN with pyramid-based neural network has a certain beneficial effect in achieving a favorable denoising performance and fast processing speed.

3 The Proposed Method

As shown in Fig. 2, the proposed model is composed of two parts: a denoising module and a discriminative module. We start with a noisy image and progressively downsample it by a factor of two within a pyramid framework, yielding a small spatial image at the final level. We then feed the last two pyramid images into a fully convolutional encoder-decoder network, which is designed as a generator used to synthesize clean image from its real-world noisy observation. For each of the subsequent level, we follow a two-stage processes: first, we upsample the generated image by a factor of two; second, we feed the upscaled image as well as another noise image from the previous scale into the encoder-decoder network to generate a finer clean image. Furthermore, a discriminative module is introduced to enforce the final clean synthesized image to draw near to the actual one.

3.1 Denoising Module

3.1.1 Model Design

The denoising module in our work aims to remove the real noise, and reconstruct the corresponding clean image. From the viewpoint of [38], the high-level image features extracted from an object image play an essential part in general image reconstruction. However, the noise embedding in the original input images may give rise to undesired correlated features, making the reconstructed image deviate from the target contents. So the key insight of our denoising module is to effectively transfer the high-level features extracted from the input noisy images to the output domain that is independent of the noise. When human try to obtain in-depth knowledge of complex objects, we never make any attempts to use one-step learning. On the contrary, we usually divide the process into several stages and move towards the goal step by step. Inspired by this cognition process, we propose to integrate the conditional GAN model into the architectural guideline of a Gaussian pyramid, and make it a promising approach to deal with realistic noise in real-world images.

According to the above analysis, given a noisy image, we first decompose it with a downsampling factor of two to build a Gaussian pyramid: $\mathcal{G}(\boldsymbol{I}_l^D) = \left[\boldsymbol{I}_0^D, \boldsymbol{I}_1^D, \boldsymbol{I}_2^D, ..., \boldsymbol{I}_l^D\right]$, where $\boldsymbol{I}_l^D$ is parameterized by l representing the downsampled image on the l-th pyramid levels, and $\boldsymbol{I}_0^D$ is the input noisy image. For example, as depicted in Fig. 2, the input is the full 512×512-pixel image and the framework consists of 8 pyramid levels, with which the final level of the pyramid is a 4×4-pixel small spatial slow-frequency image. Then, we start from the last two pyramid images, and then feed them into a generator which comprises an encoder extracting the high-level features of the input and a decoder inverting the extracted features to recover a clean image that has the same spatial size as the input. Our encoder is formed by the VGG-19 model with layers from conv1 to pool3, and we stack another two convolution layers as well as a fully-connected layer [39] after that. The decoder follows the same structure of the encoder but with fractional strides.

3.1.2 Denoising with CGAN and Gaussian Pyramid

In this part, we will describe in detail how the denoising module separates the image details from noisy observation via the combination of CGAN and Gaussina pyramid.

When building our Gaussian pyramid, the image pixel value at spatial location (i, j) on the l-th pyramid level could be expressed as:

$$\boldsymbol{I}_l^D(i, j) = \sum_{m=-c}^{c} \sum_{n=-c}^{c} \mathcal{G}(m, n) * \boldsymbol{I}_{l-1}^D(2i - 1 - m, 2j - 1 - n), \tag{1}$$

where $*$ denotes the convolution operation. $\mathcal{G}(m, n)$ is the Gaussian windown with a size of $(2c + 1) \times (2c + 1)$ and is defined as:

$$\mathcal{G}(m, n) = \frac{1}{2\pi\sigma^2} e^{-(m^2+n^2)/2\sigma^2}, \tag{2}$$

where σ is the Gaussian filter and is set as $\sqrt{2}$ in our work. Here, the Gaussian pyramid could be regarded as a low-pass filter, with which the noise can be reduced effectively. After repeated downsampling operations which blur and decimate the noisy input image, the final pyramid level has very small spatial extent with very little noise. Therefore, we roughly assume that the noise distribution of the small image is almost to zero.

Then, we take both the last two pyramid images into the encoder-decoder to recover the clean image. Note that initially, the small image at the final pyramid level is considered as the additional information of the encoder-decoder network since it can provide as many useful noise-free features as possible to help the network classify different features accurately into the corresponding class, i.e., the real noise and the image details. The reconstruction process is formulated as follows:

$$\hat{\boldsymbol{I}}_{l-1} = G_{l-1}(\boldsymbol{I}_{l-1}^{D}, u(\hat{\boldsymbol{I}}_l)) \tag{3}$$

where $\hat{\boldsymbol{I}}_{l-1}$ and G_{l-1} represent the recovered image and the encoder-decoder network on the $l-1$ pyramid levels, respectively. $u(.)$ is an up-sampling operator by a factor of two, and $u(\hat{\boldsymbol{I}}_l)$ is a conditioning variable for G_{l-1}. Then, we up-sample the output image by a factor of two by using use a transposed convolutional layer [40] (a shallow CNN). The upscaled image, in addition to the downsampled noisy image at the subsequent level, are then used as the input for the next encoder-decoder network to generate a finer clean image. Subsequent levels repeat the same procedure until we get to the first level.

Intuitively, the encoder-decoder network with conditional form distinguishes between image structure and noisy observation at multiple pyramid levels. This strategy brings the following two advantages: on the one hand, the output of the encoder-decoder network can be well constrained by the additional variable, say, a generated clean image, which makes it possible to strengthen the denoising capability for the encoder-decoder network by concatenating and integrating the contextual information. As such, the proposed network is of high effectiveness and reliability, as well as fine flexibility in capturing discriminative image features, thus further enforcing the generated clean image with more plausible fine details in different frequency bands. On the other hand, instead of directly starting from some random noise, our network generates images by taking into account both the noisy and clean images, which helps the proposed network to avoid learning irrelevant details and stabilize our training process.

In order to encourage the generated clean images to preserve prominent fine details of image structure, we apply a pixel-wise L2 norm regularization as our reconstruction loss function, and it takes the form:

$$\mathcal{L}_{rec} = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} \left\| (\hat{\boldsymbol{I}}_l)_{i,j} - (\hat{\boldsymbol{I}}_l^{GT})_{i,j} \right\|_2^F , \tag{4}$$

where W,H represent the width and height sizes of an image, and $\hat{\boldsymbol{I}}_l^{GT}$ is the corresponding downscaled ground-truth image on the l pyramid levels. The reconstruction loss is measured at the L2 distance between the generated output and the downscaled noise-free image. Although this simple loss is capable of removing noise with small reconstruction errors, it either remains noise or reconstructs over-smooth edges and textures as illustrated in Fig. 3b. This stems from the fact that the reconstruction loss penalizes the deviation of the generated image $\hat{\boldsymbol{I}}_l$ from the downscaled noise-free image $\hat{\boldsymbol{I}}_l^{GT}$, so that it encourages spatial smooth and blurry synthesized results to avoid heavy penalties.

3.2 Discriminative Module

Following Goodfellow et al. [33], we further adopt a discriminative module to help the denoising results obtain better visual quality with consistent contents. The discriminative module contains five convolutional layers which use 3 × 3 kernels. Besides, we add a channel-wise fully-connected layer after that to output a 1024-dimensional feature vector. The adversarial loss is calculated as follows:

$$\mathcal{L}_{adv} = \mathcal{E}_{\boldsymbol{I}_{rec} \sim p(\boldsymbol{I}_{rec})} - \log D(\boldsymbol{I}_{rec}), \tag{5}$$

where D denotes our discriminative module and $p(\boldsymbol{I}_{rec})$ denotes the distributions of the final recovered image. Actually, adversarial loss is widely used to reflect how well the discriminator

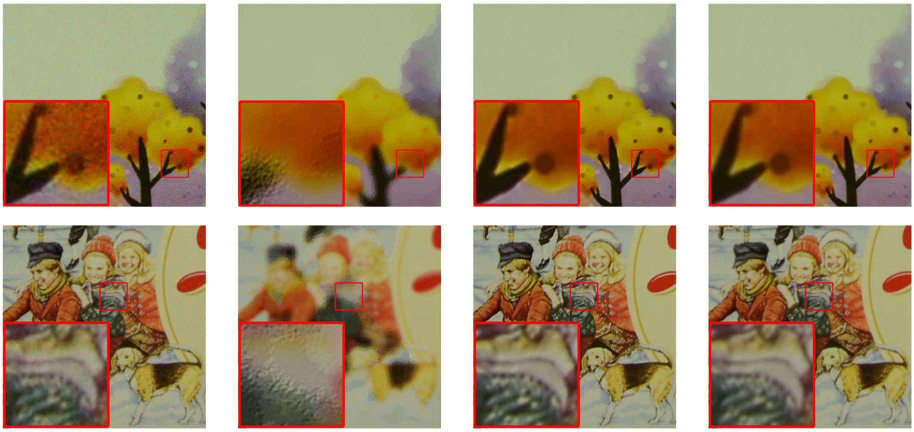

(a) Noisy input (b) $\mathcal{L}_{rec}$ (c) $\mathcal{L}_{rec} + \mathcal{L}_{adv}$ (d) Mean images

Fig. 3 Denoising results under different variants of our proposed network. **a** Input noisy images. **b** Denoising results with $\mathcal{L}_{rec}$. **c** Denoising results with $\mathcal{L}_{rec} + \mathcal{L}_{adv}$. **d** Mean images. Better zoomed-in on screen

can correctly distinguish between the recovered and real images and how realistic of the synthesized images are. From Fig. 3c, solving this min-max problem can enforce a more convincing denoising result with sharp edges and fine-scale textures while removing the noise, avoiding the spatial smoothness while using the reconstruction loss alone.

3.3 Objective Function

The objective loss function of our model is to minimize differences in feature representations between the noisy image and the corresponding clean one. Taking the loss functions defined above, the overall loss function is constructed as:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{rec} + \lambda_2 \mathcal{L}_{adv}, \tag{6}$$

where λ_1 and λ_2 are constant weights of different losses. Given a noisy image, the reconstruction loss focuses on features learning and noisy image denoising while the adversarial loss aims to regularize the global structure of entire image and boost much the image denoising performance. It is the combined loss that allows our model to preserve more faithfully the better image details as well as structure, and ensure that the new generated textures are visually realistic.

3.4 Training Details

Except for denoising a real-world noisy image to its corresponding clean counterpart, our proposed model also aims to faithfully characterize clear details of each object part. To effectively train our network, we follow the curriculum strategy [41] and use Adam [42] with an initial learning rate of 10^{-3}, decreased by a factor of 2 for 50 epochs. The training process is split into three steps. First, at the last two scales of the pyramid, an encoder-decoder network is trained using $\mathcal{L}_{rec}$ to obtain a coarse-scale denoising version. Then, we optimize the $\mathcal{L}_{rec}$ across the encoder-decoder networks at each pyramid level. At the last stage, we fine-tune the network with adversarial loss to further force the output of the generator to maximally fool the discriminator, leading to visually more satisfying denoising results.

3.5 Analysis on Complexity

The complexity of our proposed PGSN includes the model network learning and stable training of CGAN. First, learning deep convolutional network features for real-world image denoising may involve a complex optimization problem due to the complexity of real noise, resulting in an increased computation time. Second, since the training procedure of CGAN comprises jointly minimizing and maximizing conicting objectives, it would cause instability during learning. To remedy these limitations, we develop a pyramid-based neural architecture, which attempts to break the denoising procedure into more manageable successive refinements. In other words, we train a cascade of networks to progressively reconstruct clean image by inverting the extracted deep feature maps in multiple up-sampling levels from coarse to fine granularity. Such a learning strategy allows our PGSN to be easily trained for image features learning with improved accuracy. In the meanwhile, each pyramid level is capable of generating a clean image for the next one to improve. Consequently, it is far more stable to regularize the learning process and train the network to convergence.

4 Experiment

4.1 Experimental Settings

4.1.1 Datasets

In order to drive our model to learn high-level image feature representations comprehensively, we conduct experiments using three real-world noisy image datasets. Specifically, Dataset 1, taken from Nam et al. [20], contains 11 different static scenes, each with 500 images captured under a same camera model but with different ISO values. Images in this dataset are of 7360 × 4912 pixels and the ground true clean image is represented using the mean of the 500 images. Dataset 2, namely Darmstadt Noise Dataset (DND) [43], is composed of 50 different scenes and each scene is taken by 4 different consumer cameras. Each camera captures pairs of images under the same scene, where the image captured with higher ISO values and shorter exposure time is taken as the noisy image while the image under the opposite operation is taken as the noisy-free image. Dataset 3 is provided in Xu et al. [23], where the authors use two cameras, i.e., the Canon 80D and Sony A7II cameras, to capture 10 different scenes with more comprehensive and realistic objects. Similar to the Dataset 1, each camera captures 500 images per scene and the mean image of the 500 images is used as the ground-truth clean image.

To effectively train and evaluate our deep model, we cropped 112 000 images of 512 × 512 pixels from different shots in the three datasets. Considering that the ground-truth noise-free images of Dataset 2 have not been publicly available, we choose 100 000 cropped images from Dataset 1 and Dataset 3 for training. To demonstrate the efficiency of the proposed algorithm on dealing with real-world noisy images, we select 12 000 images for testing, plus extra testing images from CC [20] and NI [21].

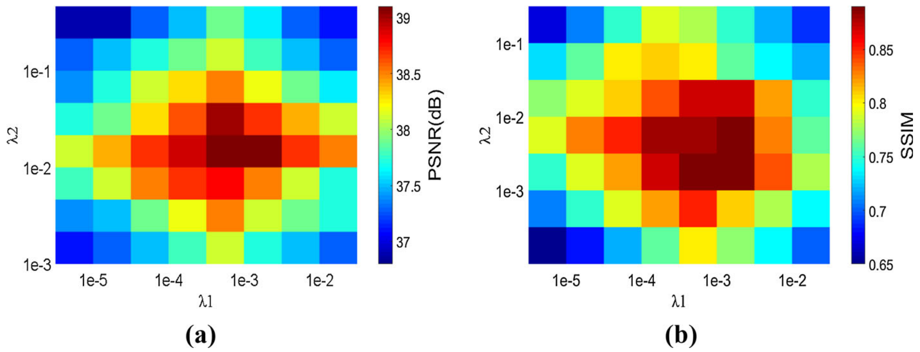


Fig. 4 The average PSNR and SSIM results according to different weighting parameters pairs on the entire testing dataset. Best viewed in color

4.1.2 Parameter Setting

The parameters of our method include the weighting parameters of the objective function (Eq. 4), i.e., λ_1 and λ_2. We jointly obtain parameters λ_1 and λ_2 by computing the peak signal-to-noise ratio (PSNR) and the structural similarity index (SSIM) values between the denoising results and the ground-truth noise-free images. Results are shown in Fig. 4.

Seen as a whole, the weighting parameters achieve the best quantitative evaluation within a certain range. Specifically, one can see that the proposed model yields the highest PSNR and SSIM result when λ_1 is set between 5×10^{-4} to 5×10^{-3}. Meanwhile, the PSNR and SSIM performance drop gradually from both sides. As a trade-off between the effects of two losses, we empirically set $\lambda_1 = 10^{-3}$ and $\lambda_2 = 0.03$. In our proposed method, we fix the weighting parameters throughout the experiments and find it robust and flexible enough to cope with various real-world noisy images. All the experiments are carried out in the MATLAB2017b environment using an Intel Core i7-8550u CPU of 3.7GHz with 6 cores and NVIDIA Titan X GPU.

4.2 Qualitative Results

We qualitatively compare the proposed method against several state-of-the-art image denoising methods whose codes or executable files are online available, such as CBM3D [4], Noise Clinic (NC) [19], Cross-Channel (CC) [20], Neat Image (NI) [21], Zhu et al. [22], Xu et al. [23] and GAN-CNN Based Blind Denoiser (GCBD) [31].

We first compare our approach with those of the competitors in Dataset 1. Since Xu et al. [23] perform experiments using 500 × 500-pixel images, we crop the testing images with the same size. The visual comparisons are shown in Fig. 5. From the results, we can see that CBM3D does not work well on real noisy images because the realistic noise is much more complex than Gaussian. It's not surprising to see that NC, CC, NI and GCBD perform better than CBM3D, whereas they would either remove noise incompletely or tend to generate much noise caused artifacts. This lies in the fact that the distribution of the real-world noise is usually unknown and is hard to be modeled by explicit distributions, because using Gaussian, MoG or zero-mean noise to estimate the model of real-world noisy images would lack flexibility and adaptability. It turns out that these methods are not good enough in handling real-world image denoising tasks and lack robustness against artifacts. From Fig. 5g, h, it can be found

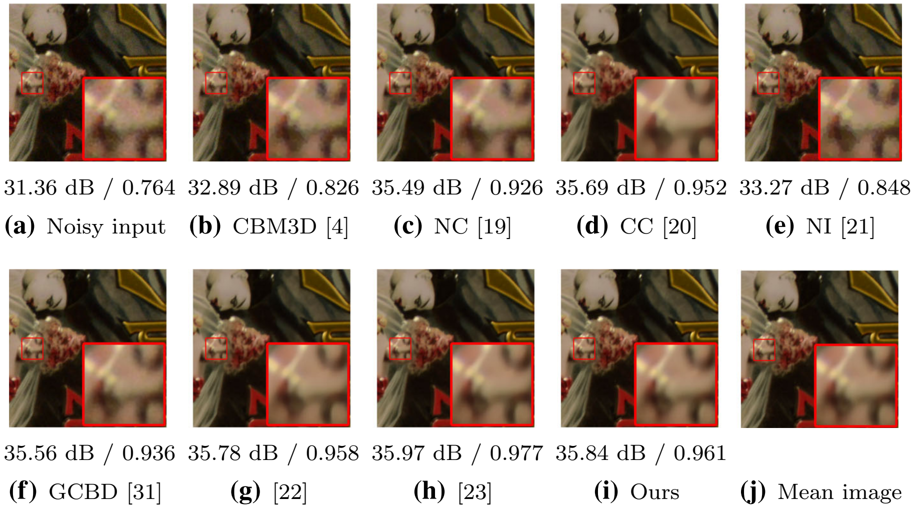


Fig. 5 Denoising results by different methods of the real-world noisy image on Dataset 1. Better zoomed-in on screen

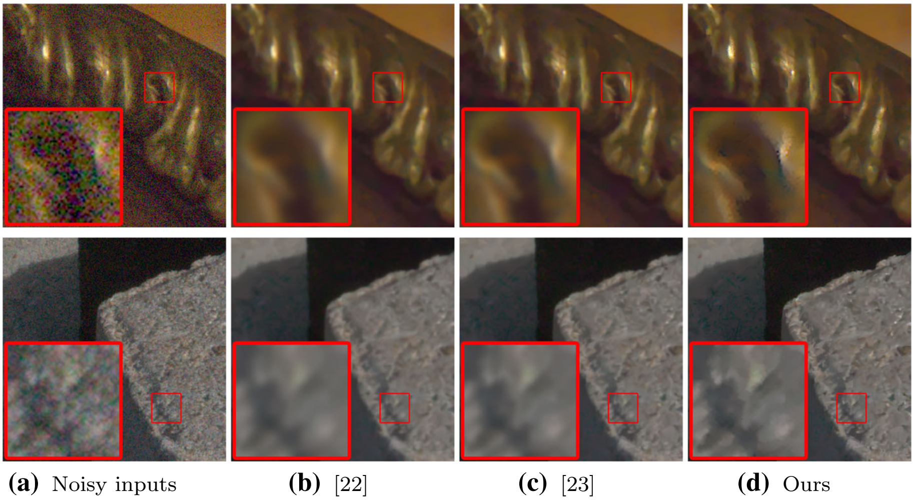
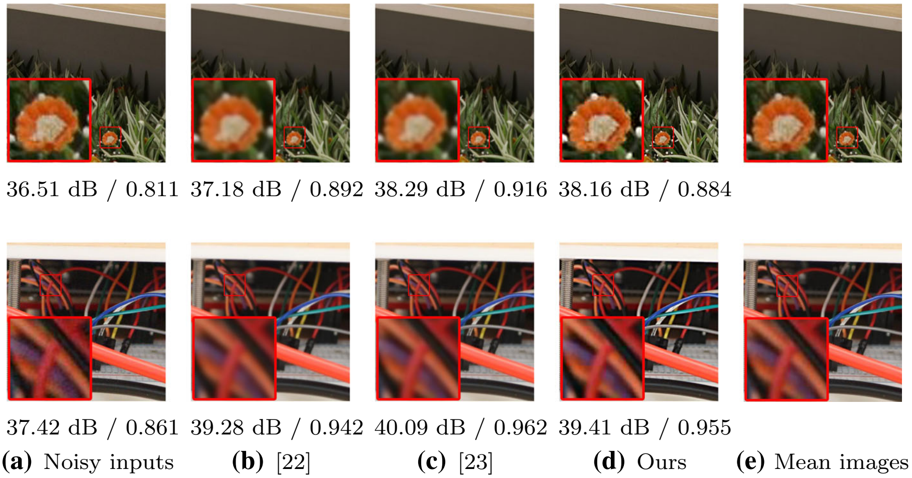


Fig. 6 Comparison with state-of-the-art denoising methods. Images are cropped from Dataset 2. Better zoomed-in on screen

that our PGSN obtains qualitatively similar results to Zhu's [22] and Xu's [23] model. One can see that these three approaches achieve more promising performance than CBM3D, NC, CC, NI and CGBD that they can remove unknown noise completely, resulting in natural clean images.

We then perform another denoising experiments, compared to Zhu's [22] and Xu's [23] model, to illustrate the superior denoising performance of our method. We specifically present more results on images cropped from Dataset 2 and Dataset 3 since these two datasets contain more realistic and comprehensive scenes. Results are shown in Figs. 6 and 7. Overall, the denoising results of our method are close to the state-of-the-arts. What's more, the proposed



(a) Noisy inputs **(b)** [22] **(c)** [23] **(d)** Ours **(e)** Mean images

Fig. 7 Comparison with state-of-the-art denoising methods. Images are cropped from Dataset 3. Better zoomed-in on screen

Table 1 Average PSNR (dB) and SSIM values of different methods under five different camera settings

Method	Camera setting				
	Canon 5D ISO = 3200	Nikon D600 ISO = 3200	Nikon D800 ISO = 1600	Nikon D800 ISO = 3200	Nikon D800 ISO = 6400
CBM3D	37.57/0.966	32.10/0.927	37.36/0.933	34.51/0.874	31.11/0.831
NC	36.66/0.953	37.18/0.949	38.42/0.951	36.85/0.928	33.05/0.908
CC	36.22/0.951	37.36/0.953	38.88/0.959	38.27/0.955	33.38/0.923
NI	35.77/0.946	36.05/0.938	37.93/0.945	36.32/0.910	31.35/0.824
GCBD	36.23/0.949	36.54/0.937	38.52/0.919	37.44/0.927	32.85/0.887
[22]	37.62/0.962	36.94/0.947	39.04/0.948	37.80/0.939	33.50/0.910
[23]	37.89/0.968	36.81/0.956	39.43/0.964	38.08/0.942	33.59/0.912
Ours	37.58/0.963	36.78/0.965	38.96/0.932	37.91/0.945	33.38/0.898

The test images are cropped from Dataset 1

PGSN shows very clear edges and details, such as the stone surface in Fig. 6d and the blooming flower in Fig. 7d, respectively. Although Zhu's [22] and Xu's [23] model demonstrate high-quality denoising performance, they do not sharpen edges discriminately when handling heavily structured objects. This suggests that our proposed method may be more aware of modeling high-level semantic representation of images. One explanation is that the encoder-decoder network based on VGG-19 network could implicitly learn to distinguish different types of semantic reasoning about the input scene. Besides, we take a set of encoder-decoder networks at different pyramid levels to better transfer semantic deep features to the domain of clean target images, therefore the training with CGAN approach encourages the recovered image to be semantically similar to the ground truth noise-free image.

4.3 Quantitative Results

To make a more comprehensive comparison among the competitors, we conduct quantitative evaluation using two metrics, i.e., the PSNR and SSIM. Considering that the code of CC [20]

Table 2 Average PSNR (dB) and SSIM values of different methods on Dataset 2 and dataset 3

Method	Dataset 2	Dataset 3	Method	Dataset 2	Dataset 3
CBM3D	32.14/0.925	37.14/0.756	[22]	36.21/0.968	37.29/0.903
NC	36.07/0.951	36.76/0.901	[23]	36.41/0.969	37.64/0.910
NI	35.11/0.924	35.70/0.878	Ours	36.12/0.962	37.15/0.897
GCBD	35.58/0.936	36.67/0.885	–	–	–

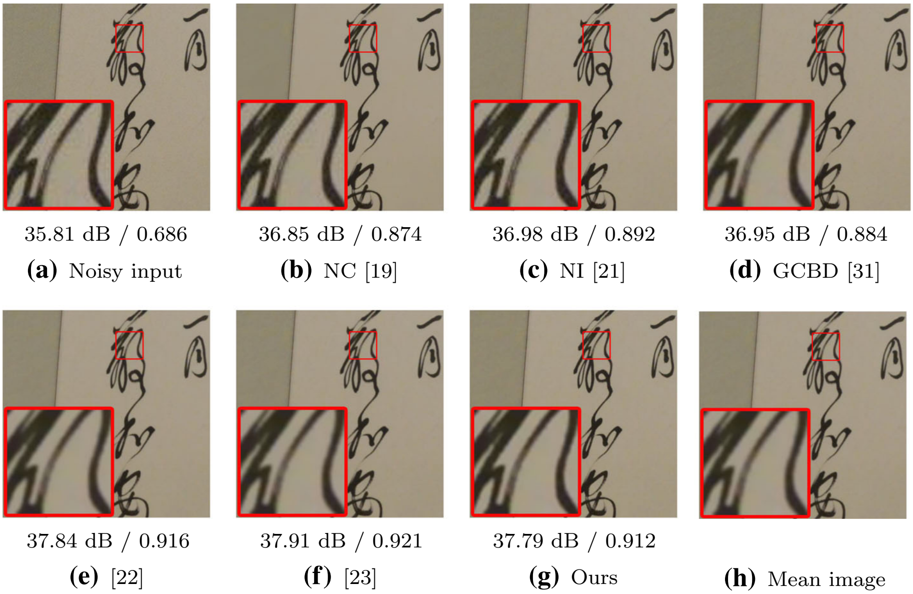

35.81 dB / 0.686
(a) Noisy input
36.85 dB / 0.874
(b) NC [19]
36.98 dB / 0.892
(c) NI [21]
36.95 dB / 0.884
(d) GCBD [31]
37.84 dB / 0.916
(e) [22]
37.91 dB / 0.921
(f) [23]
37.79 dB / 0.912
(g) Ours
(h) Mean image

Fig. 8 Denoising results of the Chinese calligraphy by different methods. The test image is cropped from Dataset 3. Better zoomed-in on screen

is not available and direct comparison with a single dataset has nonnegligible limitations, for concreteness, we perform quantitative comparison on three datasets separately for fair quantitative evaluation of different denoising algorithms.

We first perform experiments on 15 cropped images from Dataset 1 and the average PSNR and SSIM values are listed in Table 1. We can see that on 3 out of the 5 camera settings, Xu's model [23] obtains better PSNR and SSIM values in most cases while CC achieves the best PSNR and SSIM values on 2 out of the 5 camera settings. In Table 2, we list the average PSNR and SSIM results of the denoising methods on Dataset 2 and Dataset 3, respectively. Again we can see that Xu's model obtains better PSNR and SSIM results.

Although our PGSN has not yet achieved the best quantitative evaluation, we obtain relatively higher PSNR and SSIM values. For example, in Table 1, on average, our method has 0.238 dB PSNR and 0.0078 SSIM reduction over the best method, but has much higher PSNR and SSIM gains over other competing methods. Meanwhile, the proposed PGSN generally yields to better perceptual quality. We provide an additional example to show the denoising results both quantitatively and qualitatively, as shown in Fig. 8. Judging from the figure, it is clear that our method shows visually pleasant output where we reproduce clearly the image details, but corresponds to lower PSNR and SSIM values. More results can be

Table 3 Average speed (sec.) results of different methods on testing images cropped from Dataset 3

Method	CBM3D	NC	NI	GCBD	[22]	[23]	Ours
Time	6.872	15.502	0.594	24.682	26.741	23.988	18.953

seen in Figs. 6 and 7. Here, we give an explanation of this issue. Xu's model [23] has higher PSNR and SSIM values in most cases, as they use dictionary learning and sparse coding for universal image denoising problems and train their model by minimizing the mean square error. Nevertheless, we invert deep extracted learning features to reconstruct clean signals. Generally, the CGAN approach ensures the generated contents to have a high-level semantic similarity to the target image, but at the same time, our model trained for feature reconstruction based on VGG-19 netwrok encourages the denoising results to reconstruct more clear edges and textures than the ground-truth noise-free images, which may therefore harm the PSNR and SSIM values [44,45]. Similar phenomenon also occurs in [46,47].

4.4 Comparative Analysis on Efficiency

In addition to visual quality and quantitative evaluation, we also compare the average computational time of all competing methods except for CC. We perform efficiency comparison on Datatset 3 since the testing images contain various realistic objects and more comprehensive scenes. For a fair comparison, all experiments are conducted on the same machine with an Intel Core i7-8550u CPU (32G RAM) of 3.7GHz. Moreover, we evaluate each testing image for 20 times and provide the average computational time in Table 3.

From the results, we can observe that NI costs about 0.594 second to process an image and achieves the fastest running time. Compared to CBM3D and NC, one can observe that GCBD, Zhu's model, Xu's model and ours entail larger computational costs as these four methods involve either complex dictionary learning or deep network training. However, it should be noted that CBM3D, NC, CC, NI and GCBD are not capable enough for some practical image denoising applications, while the other methods can better eliminate the real noise effectively. Compared to those two state-of-the-art methods, the speed of our PGSN achieves very appealing computational efficiency.

5 Conclusion

In this paper, we made a good attempt for real-world noisy image denoising by using a deep generative model based on a Gaussian pyramid framework. Our network first extracts features from the multi-scale pyramid framework and then progressively generates clean images in a coarse-to-fine manner. Furthermore, we incorporate the conditional form of GAN model to train our network to improve the visual quality of the denoising results, with which the proposed method does a very good job at recovering the full characteristics of image texture and structures while removing realistic noise. Experiments on three datasets demonstrate that our PGSN can not only be robust for complex realistic noise, but also achieve better denoising efficiency than other competitors.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China (61673402, 61273270 and 60802069), the Natural Science Foundation of Guangdong Province

(2017A030311029, 2016B010123005 and 2017B090909005), the Science and Technology Program of Guangzhou of China (201704020180 and 201604020024), and the Fundamental Research Funds for the Central Universities of China (17lgzd08), and the University of Macau (MYRG2019-00006-FST).

References

1. Chang SG, Yu B, Vetterli M (2000) Adaptive wavelet thresholding for image denoising and compression. IEEE Trans Image Process 9(9):1532–1546
2. Starck JL, Candès EJ, Donoho DL (2002) The curvelet transform for image denoising. IEEE Trans Image Process 11(6):670–684
3. Elad M, Aharon M (2006) Image denoising via sparse and redundant representations over learned dictionaries. IEEE Trans Image Process 15(12):3736–3745
4. Dabov K, Foi A, Katkovnik V, Egiazarian K (2007) Color image denoising via sparse 3d collaborative filtering with grouping constraint in luminance-chrominance space. In: ICIP, pp 313–316
5. Zhou M, Chen H, Paisley J, Ren L, Li L, Xing Z, Dunson D, Sapiro G, Carin L (2012) Nonparametric Bayesian dictionary learning for analysis of noisy and incomplete images. Trans Image Process 21(1):130–144
6. Dong W, Zhang L, Shi G, Li X (2013) Nonlocally centralized sparse representation for image restoration. IEEE Trans Image Process 22(4):1620–1630
7. Marc L, Buades A, Morel JM (2013) A nonlocal bayesian image denoising algorithm. SIAM J Imaging Sci 6(3):1665–1688
8. Gu S, Zhang L, Zuo W, Feng X (2014) Weighted nuclear norm minimization with application to image denoising. In: CVPR, pp 2862–2869
9. Xu J, Zhang L, Zuo W, Zhang D, Feng X (2015) Patch group based nonlocal self-similarity prior learning for image denoising. In: ICCV, pp 244–252
10. Chen F, Zhang L, Yu H (2015) External patch prior guided internal clustering for image denoising. In: ICCV, pp 603–611
11. Luo E, Chan SH, Nguyen TQ (2015) Adaptive image denoising by targeted databases. IEEE Trans Image Process 24(7):2167–2181
12. Mosseri I, Zontak M, Irani M (2013) Combining the power of internal and external denoising. In: ICCP, pp 1–9
13. Burger HC, Schuler CJ, Harmeling S (2012) Image denoising: Can plain neural networks compete with BM3D? In: CVPR, pp 2392–2399
14. Chen Y, Pock T (2017) Trainable nonlinear reaction diffusion: a flexible framework for fast and effective image restoration. IEEE Trans Pattern Anal Mach Intell 39(6):1256–1272
15. Zhang K, Zuo W, Chen Y, Meng D, Zhang L (2017) Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising. IEEE Trans Image Process 26(7):3142–3155
16. Healey GE, Kondepudy R (1994) Radiometric CCD camera calibration and noise estimation. IEEE Trans Pattern Anal Mach Intell 16(3):267–276
17. Tsin Y, Ramesh V, Kanade T (2001) Statistical calibration of CCD imaging process. In: ICCV, pp 480–487
18. Liu C, Szeliski R, Bing Kang S, Zitnick CL, Freeman WT (2008) Automatic estimation and removal of noise from a single image. IEEE Trans Pattern Anal Mach Intell 30(2):299–314
19. Colom M, Lebrun M, Morel J (2015) Multiscale image blind denoising. IEEE Trans Image Process 24(10):3149–3161
20. Nam S, Hwang Y, Matsushita Y, Kim SJ (2016) A holistic approach to cross-channel image noise modeling and its application to image denoising. In: CVPR, pp 1683–1691
21. Neatlab ABSoft. Neat image. https://ni.neatvideo.com/home
22. Zhu F, Chen G, Hao J, Heng P (2017) Blind image denoising via dependent Dirichlet process tree. IEEE Trans Pattern Anal Mach Intell 39(8):1518–1531
23. Xu J, Zhang L, Zhang D (2018) External prior guided internal prior learning for real-world noisy image denoising. IEEE Trans Image Process 27(6):2996–3010
24. Xu J, Zhang L, Zhang D (2018) A trilateral weighted sparse coding scheme for real-world image denoising. In: ECCV, pp 21–38
25. Yu J, Tao D, Wang M, Rui Y (2015) Learning to rank using user clicks and visual features for image retrieval. IEEE Trans Cybern 45(4):767–779
26. Yu J, Tan M, Zhang H, Tao D, Rui Y (2019) Hierarchical deep click feature prediction for fine-grained image recognition. IEEE Trans Pattern Anal Mach Intell. https://doi.org/10.1109/TPAMI.2019.2932058

27. Yu J, Li J, Yu Z, Huang Q (2019) Multimodal transformer with multi-view visual representation for image captioning. IEEE Trans Circuits Syst Video Technol. https://doi.org/10.1109/TCSVT.2019.2947482
28. Zhang K, Zuo W, Gu S, Zhang Lei (2018) Learning deep CNN denoiser prior for image restoration. In: CVPR, pp 21–38
29. Lefkimmiatis S (2017) Non-local color image denoising with convolutional neural networks. In: CVPR, pp 5882–5891
30. Lefkimmiatis S (2018) Universal denoising networks: a novel CNN architecture for image denoising. In: CVPR, pp 3204–3213
31. Chen J, Chen J, Chao H, Yang M (2018) Image blind denoising with generative adversarial network based noise modeling. In: CVPR, pp 3155–3164
32. Chakrabarti A, Xiong Y, Sun B, Darrell T, Scharstein D, Zickler T, Saenko K (2014) Modeling radiometric uncertainty for vision with tone-mapped color images. IEEE Trans Pattern Anal Mach Intell 36(11):2185–2198
33. Goodfellow IJ, Pouget-Abadie J, Mirza M, Warde-Farley B, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. In: NIPS, pp 2672–2680
34. Mirza M, Osindero S (2014) Conditional generative adversarial nets. arXiv:1411.1784
35. Guo X, Nie R, Cao J, Zhou D, Mei L, He K (2019) Fusegan: learning to fuse multi-focus image via conditional generative adversarial network. IEEE Trans Multimed 21(8):1982–1996
36. Dar SU, Yurt M, Karacan L, Erdem A, Erdem E, Çukur T (2019) Image synthesis in multi-contrast MRI with conditional generative adversarial networks. IEEE Trans Med Imaging 38(10):2375–2388
37. Liu J, Sun W, Li M (2019) Recurrent conditional generative adversarial network for image deblurring. IEEE Access 7:6186–6193
38. Dosovitskiy A, Brox T (2016) Inverting visual representations with convolutional networks. In: CVPR, pp 4829–4837
39. Donahue J, Darrell T, Pathak D, Krähenbühl P, Efros AA (2016) Context encoders: feature learning by inpainting. In: CVPR, pp 2536–2544
40. Vincent D, Francesco V. A guide to convolution arithmetic for deep learning. arXiv:1603.07285
41. Yoshua B, Jérôme L, Ronan C, Jason W (2009) Curriculum learning. In: ICML, p 6
42. Olga R, Jia D, Hao S, Jonathan K, Sanjeev S, Sean M, Zhiheng H, Andrej K, Aditya K, Michael B (2014) Imagenet large scale visual recognition challenge. Int J Comput Vis 115(3):211–252
43. Plötz T, Roth S (2017) Benchmarking denoising algorithms with real photographs. CVPR 115(3):2750–2759
44. Sabir MF, Sheikh HR, Bovik AC (2006) A statistical evaluation of recent full reference image quality assessment algorithms. IEEE Trans Image Process 15(11):3440–3451
45. Kundu D, Evans BL (2016) Full-reference visual quality assessment for synthetic images: a subjective study. In: ICIP, pp 2374–2378
46. Huszár F, Caballero J, Cunningham A, Acosta A, Aitken A, Tejani A, Totz J, Wang Z, Ledig C, Theis L, Shi W (2017) Photo-realistic single image super-resolution using a generative adversarial network. In: CVPR, pp 105–114
47. Alahi A, Johnson J, Fei-Fei L (2016) Perceptual losses for real-time style transfer and super-resolution. In: ECCV, pp 694–711