A deep cascade of neural networks for image inpainting, deblurring and denoising

Guoping Zhao[1] · Jiajun Liu[1] · Jiacheng Jiang[1] ·
Weiying Wang[2]

Received: 10 July 2017 / Revised: 23 September 2017 / Accepted: 19 October 2017

Abstract In recent years, we have witnessed the great success of deep learning on various problems both in low and high-level computer visions. The low-level vision problems, including inpainting, deblurring, denoising, super-resolution, and so on, are highly anticipated to occur in machine vision and image processing. Many deep learning based methods have been proposed to solve low-level vision problems. Most researches treat these problems independently; however, most of the time they appear concurrently. Motivated by the success of generative model in the field of image generation, we develop a deep cascade of neural networks to solve the inpainting, deblurring, denoising problems at the same time. Our model contains two networks: inpainting GAN and deblurring-denoising network. Inpainting GAN generates the coarse patches to fill the lost part in damaged image, and the deblurring-denoising network, stacked by a convolutional auto-encoder, will further refine them. Unlike other methods that handle each problem separately, our method jointly optimizes the two sub-networks. Because GAN training is not only unstable but also difficult, we adopt the Wasserstein distance as the loss function of the inpainting GAN and propose a gradual training strategy. Learning from the idea of residual learning, we utilize skip connections to pass image details from input to reconstruction layer. Experimental results have demonstrated that the

✉ Guoping Zhao
guopingzhao@ruc.edu.cn

Jiajun Liu
jiajunliu@ruc.edu.cn

Jiacheng Jiang
jiachengjiang@ruc.edu.cn

Weiying Wang
wangw17@miamioh.edu

[1] School of Information, Renmin University of China, Beijing 100872, China

[2] Department of Computer Science and Software Engineering, Miami University, Oxford, OH, USA

Published online: 03 November 2017

proposed model can achieve state-of-the-art performance. Through the experiments, we also demonstrated the effectiveness of the cascade architecture.

Keywords Inpainting · Deblurring · Denoising · Wasserstein GAN · Auto-encoder

1 Introduction

Image inpainting [5, 6, 10, 24, 25, 29], deblurring [3, 8, 9, 13, 21, 31] and denoising [3, 4, 7, 12, 14, 38, 43] are the widely-concerned ill-posed problems in machine vision and image processing. These problems have not been remarkably dealt with because the missing part of the image is indeed difficult to estimate. Inpainting is the process of reconstructing lost or deteriorated parts of images, in order to make the images look more natural and visually plausible. The inpainting technology is widely used to rebuild damaged photographs, remove unwanted objects or texts and replace objects. Motion blur is the result of the relative motion between the camera and the scene during image exposure time. Blur may come from the shaking of the camera at the time of imaging and also may come from the noise generated when saving images. Deblurring attempts to recover the origin sharp content, remove the noise and enhance the quality of images. Deblurring methods are trending topic due to its involvement of many challenges in regularization and optimization. Denoising algorithms seek to remove noise, errors, or perturbations from an image, while preserving as many image details as possible. Previous researches commonly assume that image noise is additive white Gaussian noise [38]. Yet in many cases, the noise is not stationary, and the variance of the noise is difficult to estimate. Figure 1 Shows the typical examples of these problems.

In practice, those three problems often arise concurrently, rather than exist solely. Such as, if the noise on the image is serious, it will cause the image to be blurred. If the blurred area on the image is concentrated somewhere, it becomes a inpainting problem. So it is necessary to consider these problems as a whole, and deal with them jointly. But most current research methods treat these problems separately, and mostly focused on solving one of the problems. In general, part of the content in the image after inpainting is certainly not clear, such as the lost

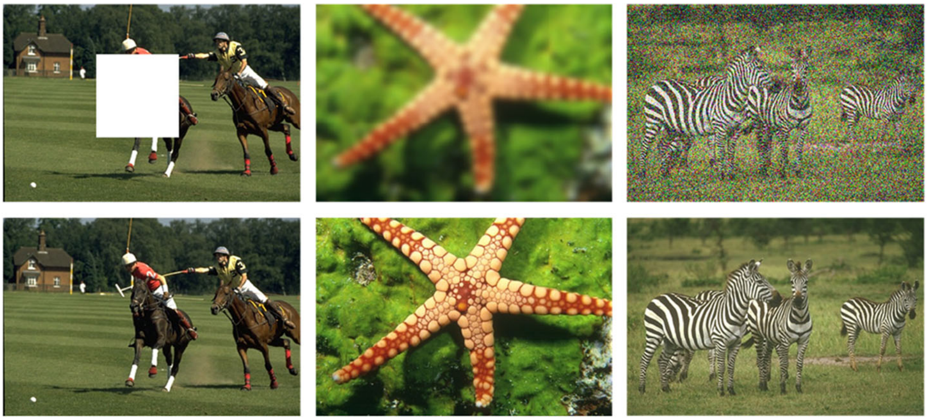


Fig. 1 Example of the mentioned three problems above. The upper-left is a picture losing the central part. The upper-middle is a blurred image, its visual effect is unclear. And the upper-right is a noising image. The lower images are the desired corresponded image after inpainting, deblurring and denoising

part and the twisty position where the obstructions removed. In order to get a visually plausible image, deblurring and denoising processes are necessary.

In this paper, we propose a deep cascade of neural networks to handle these multiple ill-posed image problems through an unique step. The model will learn how to fill the holes, deblurring and denoising the image at the same time. The obvious technical challenge is how to infer the details of an image that actually does not exist in the input data. Our approach is inspired by generative adversarial networks (GANs) [17], which is a powerful generative approach for probabilistic modeling. Although natural images are diversiform, in most cases, they are extremely structured and coherent. These properties make it possible that the GANs can capture the structure and pattern of the image through a well-trained model with bad image as the input. The method proposed in this paper establishes a specially designed cascade networks structure and set up a progressive training strategy, we can achieve the purpose that integrating the origin three processes into a holistic procedure.

Our contributions can be summarized as follows:

1. We propose a cascade of deep neural networks that deal with the inpainting, deblurring, and denoising through a unified process. In the training step, we can train the whole networks by pipelining the procedures, instead of training two separate models to accomplish these tasks. The model learns multiple tasks after training. Besides, in the inferring step, we can get not only the ultimate output image after inpainting, deblurring and denoising, but also the intermediate results after inpainting step.
2. We propose a gradual training strategy. At the first step, we only train the inpainting part networks, which is a GANs-like architecture. After the inpainting part being well trained, we commence to train the deblurring and denoising part of the networks. What has to be aware of is that at this step the parameters in the inpainting part are not frozen and jointly optimized with the latter part.
3. We evaluate the proposed model on several datasets and demonstrate that its performance is advanced. The ultimate output looks more natural, and the inpainting part looks smoother with its surrounding. This shows that the cascade of deep neural networks can learn the ability to handle these reverse vision tasks.

The rest of this paper is organized as following. Section 2 briefly reviews related work of the current methods that handle these problems. Section 3 presents the proposed method in detail, and Section 4 gives the experimental results to verify the effectiveness of the proposed method. Finally, we conclude the paper in Section 5.

2 Related work

A variety of techniques have been proposed to handle those tasks mentioned above. Such as in the inpainting and denoising fields, the traditional structural based and textural based methods have been studied for a long time. For the deblurring problem, the main methods are based on kernel estimation. In the recent years, Convolutional Neural Networks (CNNs) has shown outstanding performance in many tasks, including classification [23, 26, 32], object detection [16], segmentation [27], NLP [33], behavior analysis [34, 44] and so on. The deep learning basic approaches are also introduced to process the generative vision problems [17], such as image generation [30], image to image translation [20], and video prediction [39].

Existing methods address the inpainting problem can be divided into several categories such as structural inpainting [6, 25], textures synthesis [5, 6], and example-based methods [10, 11]. Structural inpainting uses geometric approaches to fill in the missing information in the region. Liu et al. [25] proposes a compression-oriented edge-based algorithm for inpainting, which focus on visual quality rather than pixel-wise fidelity. These algorithms focus on the consistency of the geometric structure. Textures synthesis inpainting algorithms uses similar textures approaches, under the constraint that image texture should be consistent. Bertalmio et al. [6] simultaneously utilizes structure and texture to fill-in the regions of missing image information. These classes of techniques are less effective in the case of large lost region due to missing global information. Example-based image inpainting attempts to infer the missing region through retrieving similar patches or learning-based model. Hays and Efros [18] retrieve semantically similar patches from a large photographs dataset and then use these patches to fill in the missing pixels. Pathak et al. [29] proposes an unsupervised learning algorithm named Context Encoders. That is a convolutional neural network trained to generate the contents of an arbitrary image region which is conditioned on its surroundings. Yang et al. [42] propose a multi-scale neural patch synthesis approach based on joint optimization of image content and texture constraints.

The task of image deblurring is to recover a clean image given only the blurry image. In order to generate clear image via image processing, a number of approaches have been proposed. Shan et al. [31] uses a unified probabilistic model of both blur kernel estimation and unblurred image restoration to deblur image. Cho and Lee [9] introduce a novel prediction step to accelerate both latent image estimation and kernel estimation in an iterative deblurring process. Cai et al. [8] removes motion blurring from a single image by formulating the blind blurring as a new joint optimization problem, which simultaneously maximizes the sparsity of the blur kernel and the sparsity of the clear image under certain suitable redundant tight frame systems, Sun et al. [37] utilizes deep learning approach to predict the probabilistic distribution of motion blur. Xu et al. [41] establishes a framework for robust deconvolution against artifacts through combining traditional optimization-based schemes and neural network.

Denoising is the process of reconstructing the original image by removing unwanted noise from a corrupted image. Image denoising approaches can be categorized as spatial domain, transform domain, and learning based methods [35]. Elad and Aharon [14] uses K-SVD to obtain an over-complete dictionary and describe the image content effectively. Most existing state-of-the-art image denoising algorithms are based on retrieving the similarities between a number of patches. The eminence method is block-matches with 3D filtering(BM3D) [12]. BM3D is based on effective filtering in 3D transform domain by combining sliding window transform processing with block-matching. Burger et al. [7] apply multi-layer perceptron (MLP) to image patches, directly learning the mapping from noising image to noise-free image. Stacked denoising auto-encoder [38], which is trained locally to denoise corrupted versions of their inputs, is one of the well-known deep neural network model used for denoising. The auto-encoder tries to restore the raw input without noise. Zhang et al. [43] utilizes the residual learning to train a denoising convolutional neural networks to handle Gaussian denoising with unknown noise level.

There are some researches which attempt to resolve more than one aspect of the reverse vision problems. Dong et al. [13] adds autoregressive models and nonlocal self-similarity regularization term to sparse-coding algorithm, achieving excellent results on both image deblurring and super-resolution. Meur et al. [24] introduces a framework involving a combination of multiple inpainting versions of the input picture followed by a single-image super-resolution method. Gharbi et al. [15] trains a deep neural network on a large corpus of image to jointly solve denoising and demosaicking. Unlike these methods treating each task separately, our approach is much general, learning multi-task through a whole neural network.

3 Method

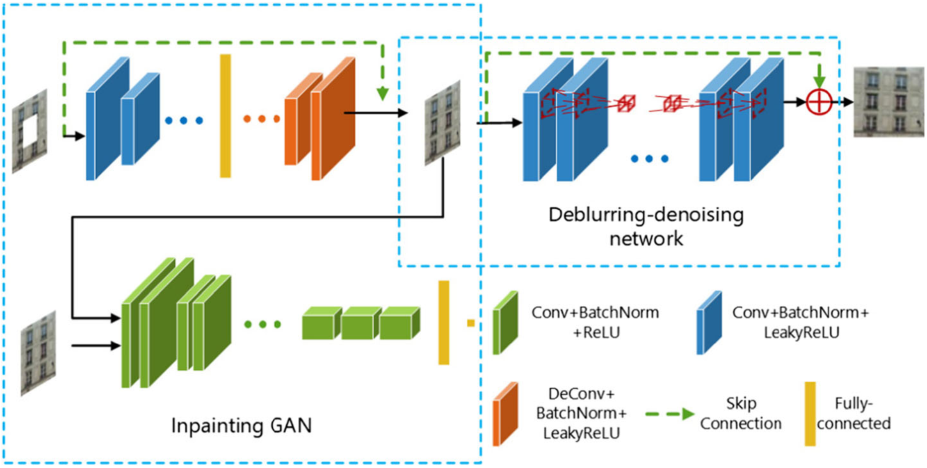
In this section, we first introduce the overall architecture of the deep cascade neural networks. Then we present the details of inpainting GAN and deblurring-denoising network. Finally, the gradually training strategy is introduced.

3.1 Framework overview

To deal with the above mentioned multiple ill-posed vision tasks, our deep cascade of neural networks is illustrated in Fig. 2. This framework mainly contains two parts: inpainting GAN and deblurring-denoising network. The corrupted image serves as the input to our method and the output of deblurring-denoising network is the final resulting image.

The first part named inpainting network is based on generative adversarial networks. The GAN generates meaningful visual blocks to fill in the vacancy or replace deteriorated parts through the competition between the generator and discriminator. The output of inpainting GAN is an image with complete content, but the filled area is blurred. The reason is that although the generator can produce images that look natural, the noise is inevitably mixed. The generated image is directly entered into the deblurring-denoising network. The intention of the deblurring-denoising network is to make the filled area clear. Inspired by deep residual networks [19] and Stacked denoising auto-encoders [38], the structure of the deblurring-denoising network is deep convolutional Auto-Encoder with skip connections.

We call our model deep cascade of neural networks because the two parts in the model are directly connected, and errors can be back propagated from deblurring-denoising network to



Deblurring-denoising network
Inpainting GAN
Conv+BatchNorm +ReLU
Conv+BatchNorm+ LeakyReLU
DeConv+ BatchNorm+ LeakyReLU
Skip Connection
Fully-connected

Fig. 2 Architecture of the cascade neural networks. It consists of two parts, the Inpainting GAN and deblurring-denoising network. The Inpainting GAN simultaneously train two networks: a generator and a discriminator. The resulting image of the generator is further processed by deblurring-denoising network in order to remove the blur and noise

the generator of inpainting GAN. The two sub-networks are joint as an integration. We firstly train the inpainting GAN network, and make the generator obtain the ability to generate coarse blocks to fill in the missing parts. The inpainting GAN enforces the generated image to be coherent and to look natural. Then we pre-train the deblurring-denoising network. After this step, we jointly optimize the deblurring-denoising network and inpainting GAN. The loss of the deblurring-denoising network will affect the parameters of the inpainting GAN. The stepwise training process is specially designed in order to optimize the complex model. A detailed description of training steps will be introduced in 3.4 section.

3.2 Inpainting GAN

Deep generative models attempt to capture the probability distributions of the given data. Generative adversarial networks, which have been proposed by Goodfellow et al. [17], aim to estimate the generative models via an adversarial process. The GANs simultaneously train two networks: a generative network G which wants to captures the input data distribution, and a discriminative network D which wants to correctly distinguish the sample came from the training data or model G. Unlike [30] generating image from noise prior, in our model the generative network generates the image $G(x)$ given the input image x. The x is the corrupted input image, and its corresponding ground-truth image is denoted by y. In the discriminator, $G(x)$ and y are presented as inputs. With the adversarial process, the generator can learn to create similar patches to fill in the missing parts, meanwhile it's hard for the discriminator to distinguish. In order to improve the stability of learning and get rid of mode collapse, we adopt the Wasserstein GAN [1] instead of traditional GAN. The structure of Inpainting GAN is shown in Fig. 3.

Following the network architectures in [29], the generator is a simple encoder-decoder pipeline, which consists of convolution layers and deconvolution layers. The generator extracts feature through first five convolution layers, and recovers the details of image contents through five deconvolution layers. Batch normalization layer is used after every convolution layer and adopt leaky ReLU as the activation function. The decoder uses ReLU as activation function which is different from encoder. In order to make the training more effective, we adopt the mirrored skip connections between the first convolution layers and after their corresponding deconvolution layers. The skip connection simply element-wise add the input image to the generator's output. The size of both ends of skip connections should keep the same.

The discriminator is similar to VGG-16 network, which is proposed by K. Simonyan and A. Zisserman [36]. We use five groups of convolution blocks, and remove max-pooling layers in VGG-16 network. In order to reduce the size of the feature maps, the stride of last convolutional layer in each block is 2, and others are 1. And the number of 3*3 filter kernels increase by a factor of 2 from 64 to 512 as in the VGG-16 network. The last convolution layer is followed by two full connection layers. Since the original GAN training process is unstable at the risk of model collapsing, we use the Wasserstein GAN instead of original GAN. It is important to note that we followed the advice of [1], by removing the sigmoid layer in the output layer of the discriminative network, and using the RMSProp as the optimizer.

The loss of the network consists of three parts: MSE loss, perceptual loss and generative adversarial loss. The MSE loss is calculated through pixel wise mean squared error. In many cases, peak signal-to-noise ratio (PSNR) is an approximation to the human perception

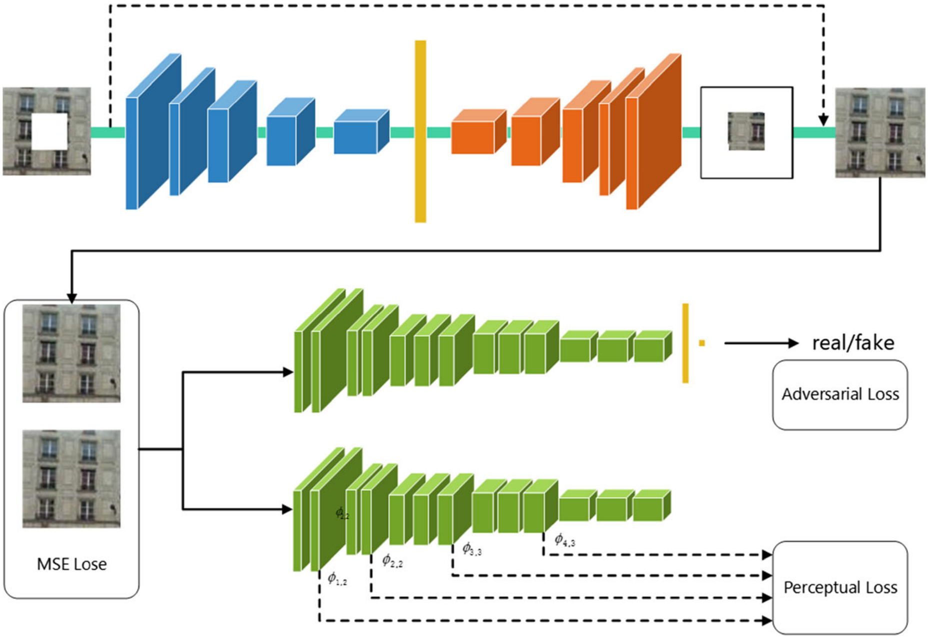


Fig. 3 Detail of the inpainting GAN. The generator contains seven residual blocks. The residual network building block is stacked by two convolutional layers followed by batch normalization layer. ReLU activation layer following the first batch normalization layer. An identity mapping shortcut connects the input of the residual block to the output of last layer in the residual block. Mirrored skip connections are adopted between corresponding convolution layers. The structure of discriminator is similar to VGG networks. We replace the max-pooling layers by adjust the stride of convolution layer to 2

of reconstruction quality, and the lower the MSE will result in the higher the PSNR. Therefore, the MSE loss is the most widely used optimization target for image inpainting task. Johnson et al. [22] proposed the perceptual loss functions based on high-level features extracted from pertained networks. And their experiments also demonstrate that the perceptual loss produces more realistic results in the style transfer and super-resolution tasks. We adopt the Wasserstein GAN loss as the generative adversarial loss. Unlike traditional GAN, Wasserstein loss is differentiable almost everywhere. This nature results in a better discriminator. On the other hand, Wasserstein distance provides a metric that correlates well with training progress.

Given a paired image $(x, y) \in (I_{input}, I_{groundtruth})$ The MSE loss is defined as:

$$L_{MSE} = \frac{1}{WH} \sum_{i=1}^{W} \sum_{j=1}^{H} \left[y_{\{i,j\}} - G(x)_{\{i,j\}} \right]^2 \tag{1}$$

The W is the width of image, H is the height of image, G(x) is the image generated by the generator.

We define the perceptual loss on the activation layers of VGG-16 [36]. Denoting $\phi_{i,j}(x)$ as the feature map obtained after ReLU activation of the j-th convolutional layer and before the i-th polling layer in VGG-16, if the shape of feature map is $(H_{i,j} \times W_{i,j} \times C_{i,j})$, the mean Euclidean distance between feature representations is denoted as $L_{i,j}$, and the perceptual loss is the mean value of specified $L_{i,j}$, in this case we use the feature maps: $\phi_{1,2}$, $\phi_{2,2}$, $\phi_{3,3}$, $\phi_{4,3}$(all are the feature maps before polling layer). The perceptual loss is finally given by:

$$L_{i,j} = \frac{1}{H_{i,j}W_{i,j}C_{i,j}} \left\|\phi_{i,j}(y) - \phi_{i,j}(G(x))\right\|_2^2 \tag{2}$$

$$L_{per} = \frac{1}{4}\left(L_{1,2} + L_{2,2} + L_{3,3} + L_{4,3}\right) \tag{3}$$

Arjovsky and collaborators [1] theoretically analyzed the drawback of original GAN, and advice on using Wasserstein distance $W\ (f, g)$ to measure the difference between input data distribution and generator's distribution. The Wasserstein GAN is to solve the adversarial min-max problem:

$$\underset{G}{min}\ \underset{D}{max} \underset{x\sim\mathbb{P}_r}{\mathbb{E}}[D(x)] - \underset{\tilde{x}\sim\mathbb{P}_g}{\mathbb{E}}\left[D\left(\tilde{x}\right)\right] \tag{4}$$

The discriminator loss is:

$$L_D = \underset{\tilde{x}\sim\mathbb{P}_g}{\mathbb{E}}\left[D\left(\tilde{x}\right)\right] - \underset{\tilde{x}\sim\mathbb{P}_d}{\mathbb{E}}[D(x)] \tag{5}$$

The generator loss is:

$$L_G = -\underset{\tilde{x}\sim\mathbb{P}_g}{\mathbb{E}}\left[D\left(\tilde{x}\right)\right] \tag{6}$$

we define the overall loss function as:

$$L = \lambda_{MSE}L_{MSE} + \lambda_{per}L_{per} + \lambda_D L_D \tag{7}$$

3.3 Deblurring-denoising network

The deblurring-denoising network is connected to the generator of inpainting GAN. It takes the generator's output image as input and estimates corresponding clean image. The structure of the deblurring-denoising network is a deep convolutional Auto-Encoder with skip connections. The optimize objective is minimizing the mean squared error of estimating image and the ground-truth image. Instead of learning a mapping $\mathrm{y} = \mathscr{F}(\mathrm{x})$ to directly get clean image from noisy input image, we learn the residual between clean image and noisy observation through the skip connection between input and output. The residual learning method solved the vanishing gradients problem through learning a mapping between noisy input and noise or blur. We get the clean image $\mathrm{y} = \mathrm{x} - \mathrm{v}$, where $\mathrm{v} = \mathscr{R}(\mathrm{x})$.

The network structure is shown in Fig. 4. We use 4 convolutional layers encoding the input image to features, and 4 convolutional layers decoding these features to restore a full-detail image. Every convolutional layer is followed by a batch normalization layer except the first and the last. LeakyReLU activation with negative slope parameter set to 0.001 is applied after batch normalization. We set the size of all convolutional filter to be 3*3, and the number of

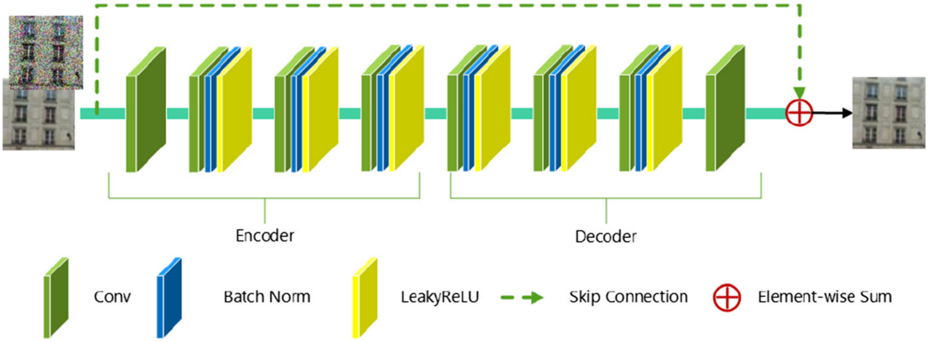


Fig. 4 Detail of the deblurrring-denoising network. Four convolution layers are used to get the encoded presentation, and 4 convolution layers are used to decode these features to reconstruct clean image. The size of all convolutional filters is set to 3*3 and stride to 1. All feature maps keep the same size with input image

channels is 96. To preserve the dimension of feature map, every convolutional layer is given zero-padding.

3.4 Gradual training strategy

The model is composed by two parts: inpainting GAN and deblurring-denoising network. The deblurring-denoising networks are directly connected to the generator of inpainting GAN. Because the network structure is complex and handles multiple tasks at the same time, so end-to-end training is difficult to make the network converge. Therefore, we designed a gradual training strategy in order to acquire better results.

Firstly, we pre-train the generator of inpainting GAN by only using MSE-loss. The process is similar to training an auto-encoder, enabling the generator has the basic ability to extract useful features and reconstruction. Secondly, we add the discriminator and VGG-loss into the training process. Next, pre-train the deblurring-denoising network independently. The image pairs are used to pre-train the deblurring-denoising network which is generated by adding noise and blur to a clean image. This step also can be parallelized with training inpainting GAN, because we can regard them as separate networks at this point. After the two networks being well trained, we combine the generator of inpainting GAN with deblurring-denoising network, and jointly training them. The finally training step is conforms to an end-to-end mode. The image firstly enters into the generator and the output comes from deblurring-denoising network; and the gradient is calculated and update weight propagated backwards, starting from the output until the generator's input.

4 Experiments and results

In this section, for evaluation purpose, extensive experimental results are introduced to evaluate the performance of the deep cascade of neural networks. We firstly present the details of datasets and experiments settings. Next, we conduct a series of experiments to evaluate the comprehensive effectiveness of the learned model. We also compare our model with the recent state-of-art approaches systematically to show the difference and advantage of our model. Finally, we analyze the architecture of our model.

4.1 Data sets and evaluation metrics

We evaluate the proposed approach on ImageNet [23], BSD300 [28]. ImageNet, which contains 1000 categories and 1.2 million images, is the authoritative dataset to evaluate the classification task. The BSD300 is widely used for segmentation task. The BSD300 dataset only contains 300 images, so BSD300 is only used for testing. Testing set is randomly picked from validation set. Dataset for deblurring-denoising network pre-training is randomly selected from training sets of ImageNet. The input images are blurred by the random blur kernels followed by adding Gaussian white noise. Generating datasets for inpainting GAN pre-training is simple which is just removing the center part of images.

Following previous works, we adopt the Peak Signal to Noise Ratio (PSNR), and the Structural Similarity Index(SSIM) [40] as the evaluation metrics. The PSNR estimates the absolute errors in pixel values between two images, while SSIM is a perception-based model that estimates the structural similarity of two images.

4.2 Comparisons with state-of-the-art methods

From various classic and recent state-of-the art image inpainting approaches, five representative methods are selected as the comparison baselines, including k-Nearest Neighbor, PatchMatch [2], Context Encoders [29], Neural Patch Synthesis(NPS) [42]. The resolution of test images for comparison is 128 * 128. NN is implemented by ourselves. The algorithms of PatchMatch have been provided by the author. The results of Context Encoders are provided by the author. The results of Neural Patch Synthesis are generated through running the author’s model. We use the same test dataset provided by the author of Context Encoder to compare the effect of these methods.

By the image consequence generated by k-NN methods, there is no gainsaying that k-NN methods have the inferior performance. As the input images contain high-frequency scene, the output is entirely unpredictable even emerge radicalized filling part centered by loss region. In low-frequency region, the PatchMatch has exceptional performance. But in the images that contain relatively high-frequency scenes, the PatchMatch failed to fill loss region. We can easily conduct that the generative model based approach has better performance to fill damaged images and generate legible output images. Compared with CE and NPS, the PSNR and SSIM scores of our model is litter higher. We think this improvement benefits from our cascade structure and gradual training strategy. The detail quantitative comparison on ImageNet is listed in Table 1, and examples of visual comparison is presented in Fig. 5.

4.3 Architecture analysis

In order to evaluate the effectiveness of our method, we have done an additional set of experiments. The first one doesn’t use the pre-training strategy, in order to verify the effect of the gradual training strategy. In the second experiment the model with only the inpainting GAN but no deblurring-denoising network to verify the ability of the deblurring-denoising network.

Table 1 Quantitative comparison on ImageNet and BSD300 between different methods

DataSet	Metric	PatchMatch	CE	NPS	Ours
ImageNet	PSNR	16.85	17.88	18.06	**18.08**
	SSIM	0.821	0.845	0.848	**0.851**
BSD300	PSNR	17.12	18.09	18.33	**18.41**
	SSIM	0.828	0.844	0.850	**0.855**

The Bold Fonts entries represent the results of our model. And the Header of table we uesed the blod fonts

We present some of the results in Fig. 6, and detail quantitative comparison of these experiments in Table 2. From the result, we found that the PSNR score will drop about 1.2 dB without the pre-training procedure on ImageNet and 1.4 dB on BSD300. Without the

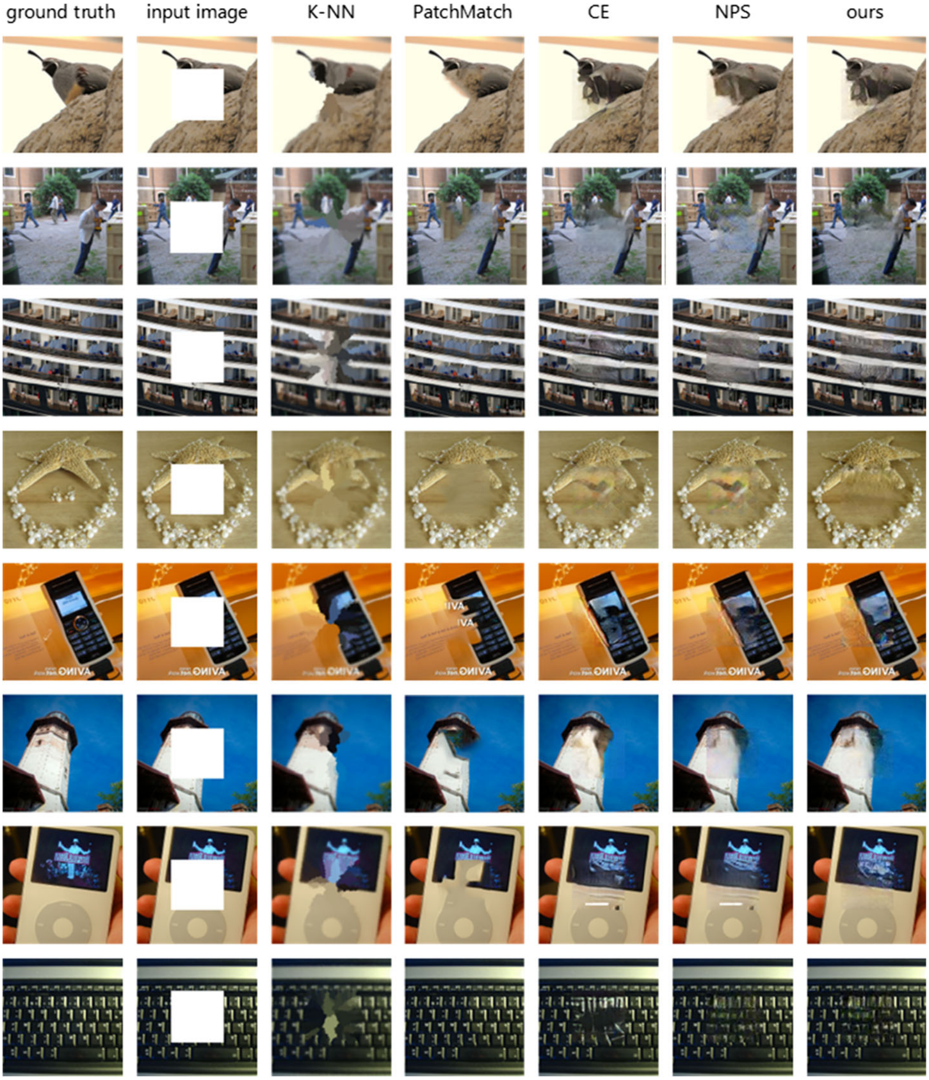


Fig. 5 Visual comparisons of different methods on ImageNet. From left to right: ground truth, input image, K-NN, PatchMatch, Context Encoder, Neural Patch Synthesis, and Ours

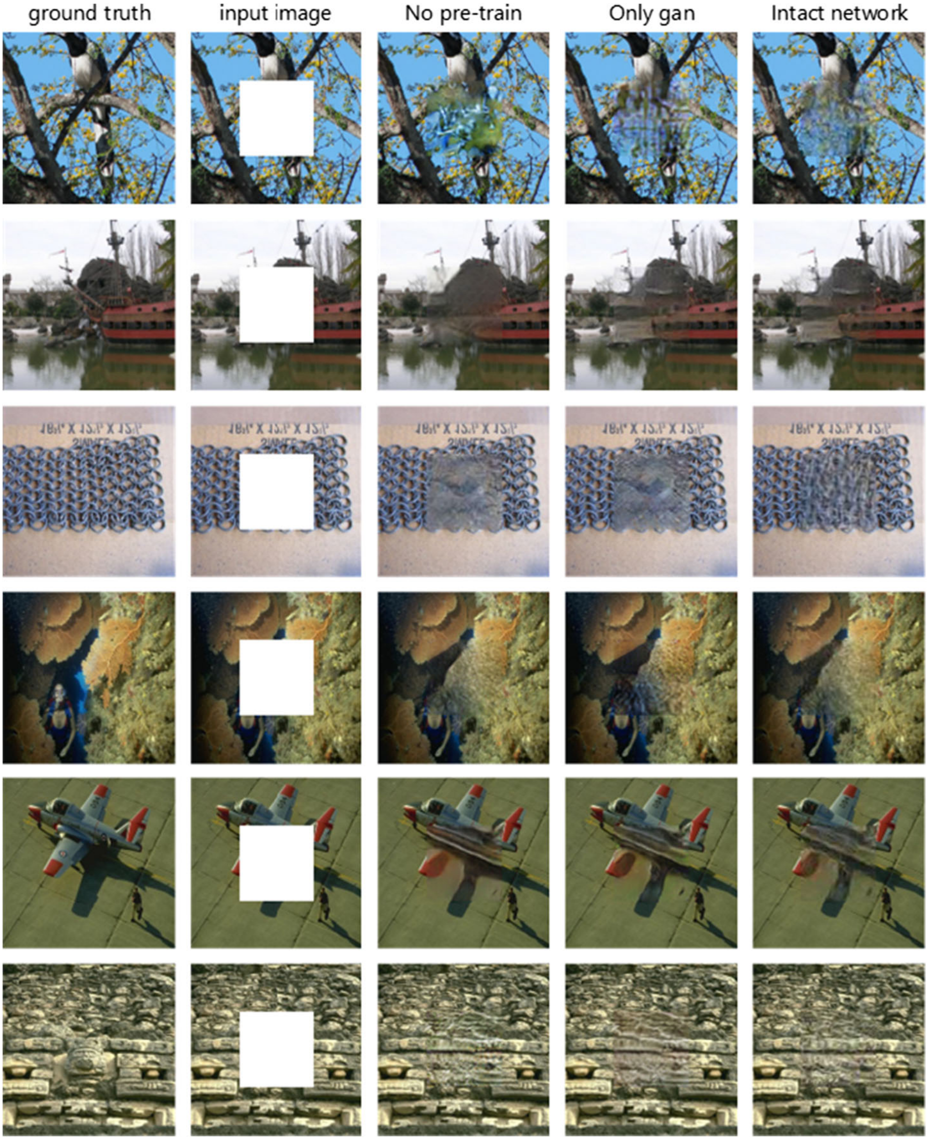


Fig. 6 Visual comparisons of different setting result on ImageNet and BSD300.The image three lines above come from ImageNet. The image three lines below come from BSD300. From left to right: ground truth, input image, model without pre-training, model without deblurring-denoising network and intact model

deblurring-denoising network, the PSNR score will drop about 0.8 dB on ImageNet and 1.1 dB on BSD300. It is obviously that the performance of our intact model surpasses all the others. We demonstrate the role of the pre-training strategy and deblurring-denoising network to enhance the image quality is very conspicuous.

Table 2 Comparison between different architectures and training strategies

DataSet	Metric	No pre-training	Only GAN	Intact Network
ImageNet	PSNR	16.91	17.22	**18.08**
	SSIM	0.815	0.845	**0.851**
BSD300	PSNR	17.01	17.34	**18.41**
	SSIM	0.819	0.843	**0.855**

The Bold Fonts entries represent the results of our model. And the Header of table we uesed the blod fonts

5 Conclusion

This paper presents a novel cascade of neural networks for multiple low-level vision problems. The model contains two parts: inpainting GAN and deblurring-denoising network. The inpainting GAN adopt three weighted loss functions as training loss. Using an effective joint optimization, the two parts are well trained to generate clean version of the image. Future work will focus on optimizing the structure of generative network, so as to improve the generator's ability to learn and represent.

References

1. Arjovsky M, Chintala S, Bottou L (2017) Wasserstein gan. arXiv preprint arXiv:170107875
2. Barnes C, Shechtman E, Finkelstein A, Goldman DB (2009) PatchMatch: a randomized correspondence algorithm for structural image editing. ACM Trans Graph 28 (3):24:21–24:11
3. Beck A, Teboulle M (2009) Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. IEEE Trans Image Process 18(11):2419–2434
4. Bengio Y, Yao L, Alain G, Vincent P (2013) Generalized denoising auto-encoders as generative models. Adv Neural Inf Proces Syst:899–907
5. Bertalmio M, Sapiro G, Caselles V, Ballester C (2000) Image inpainting. In: Proceedings of the 27th annual conference on computer graphics and interactive techniques. ACM Press/Addison-Wesley Publishing Co., pp 417–424
6. Bertalmio M, Vese L, Sapiro G, Osher S (2003) Simultaneous structure and texture image inpainting. IEEE Trans Image Process 12(8):882–889
7. Burger HC, Schuler CJ, Harmeling S (2012) Image denoising: can plain neural networks compete with BM3D? In: Computer Vision and Pattern Recognition (CVPR), 2012 I.E. Conference on, IEEE, pp 2392–2399
8. Cai J-F, Ji H, Liu C, Shen Z (2009) Blind motion deblurring from a single image using sparse approximation. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, IEEE, pp 104–111
9. Cho S, Lee S (2009) Fast motion deblurring. ACM Trans Graph 28(5):1–8. https://doi.org/10.1145/1618452.1618491
10. Criminisi A, Perez P, Toyama K (2003) Object removal by exemplar-based inpainting. In: Computer vision and pattern recognition, 2003. Proceedings. 2003 I.E. Computer Society Conference on, IEEE, pp II-II
11. Criminisi A, Pérez P, Toyama K (2004) Region filling and object removal by exemplar-based image inpainting. IEEE Trans Image Process 13(9):1200–1212
12. Dabov K, Foi A, Katkovnik V, Egiazarian K (2007) Image denoising by sparse 3-D transform-domain collaborative filtering. IEEE Trans Image Process 16(8):2080–2095
13. Dong W, Zhang L, Shi G, Wu X (2011) Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization. IEEE Trans Image Process 20(7):1838–1857
14. Elad M, Aharon M (2006) Image denoising via sparse and redundant representations over learned dictionaries. IEEE Trans Image Process 15(12):3736–3745

15. Gharbi M, Chaurasia G, Paris S, Durand F (2016) Deep joint demosaicking and denoising. ACM Trans Graph (TOG) 35(6):191
16. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 580–587
17. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. International Conference on Neural Information Processing Systems, In, pp 2672–2680
18. Hays J, Efros AA (2007) Scene completion using millions of photographs. In: ACM SIGGRAPH, p 4
19. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
20. Isola P, Zhu J-Y, Zhou T, Efros AA (2016) Image-to-image translation with conditional adversarial networks. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1125–1134
21. Ji H, Wang K (2012) Robust image deblurring with an inaccurate blur kernel. IEEE Trans Image Process 21(4):1624–1634
22. Johnson J, Alahi A, Fei-Fei L (2016) Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision (ECCV). pp 694–711
23. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: International Conference on Neural Information Processing Systems, pp 1097–1105
24. Le Meur O, Ebdelli M, Guillemot C (2013) Hierarchical super-resolution-based inpainting. IEEE Trans Image Process 22(10):3779–3790
25. Liu D, Sun X, Wu F, Li S, Zhang Y-Q (2007) Image compression with edge-based inpainting. IEEE Trans Circuits Syst Video Technol 17(10):1273–1287
26. Liu J, Shang S, Zheng K, Wen J-R (2016) Multi-view ensemble learning for dementia diagnosis from neuroimaging: an artificial neural network approach. Neurocomputing 195:112–116
27. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 3431–3440
28. Martin D, Fowlkes C, Tal D, Malik J (2001) A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Computer Vision, 2001 ICCV 2001 Proceedings Eighth IEEE International Conference on, IEEE, pp 416–423
29. Pathak D, Krahenbuhl P, Donahue J, Darrell T, Efros AA (2016) Context encoders: feature learning by inpainting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 2536–2544
30. Radford A, Metz L, Chintala S (2016) Unsupervised representation learning with deep convolutional generative adversarial networks. In: International Conference on Learning Representation (ICLR)
31. Shan Q, Jia J, Agarwala A (2008) High-quality motion deblurring from a single image. ACM Trans Graph 27(3):1–10
32. Shang S, Liu J, Zhao K, Yang M, Zheng K, Wen J-r (2015) Dimension reduction with meta object-groups for efficient image retrieval. Neurocomputing 169:50–54
33. Shang S, Guo D, Liu J, Zheng K, Wen J-R (2016) Finding regions of interest using location based social media. Neurocomputing 173:118–123
34. Shang S, Guo D, Liu J, Wen J-R (2016) Prediction-based unobstructed route planning. Neurocomputing 213:147–154
35. Shao L, Yan R, Li X, Liu Y (2014) From heuristic optimization to dictionary learning: a review and comprehensive comparison of image denoising algorithms. IEEE Trans Cybern 44(7):1001–1013
36. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representation (ICLR), pp 1–14
37. Sun J, Cao W, Xu Z, Ponce J (2015) Learning a convolutional neural network for non-uniform motion blur removal. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 769–777
38. Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol P-A (2010) Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. J Mach Learn Res 11:3371–3408
39. Vondrick C, Pirsiavash H, Torralba A (2016) Generating videos with scene dynamics. In: International Conference on Neural Information Processing Systems, pp 613–621
40. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process 13(4):600–612
41. Xu L, Ren JS, Liu C, Jia J (2014) Deep convolutional neural network for image deconvolution. In: International Conference on Neural Information Processing Systems, pp 1790–1798
42. Yang C, Lu X, Lin Z, Shechtman E, Wang O, Li H (2017) High-resolution image inpainting using multiscale neural patch synthesis. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 6721–6729

43. 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
44. Zhu S, Wang Y, Shang S, Zhao G, Wang J (2017) Probabilistic routing using multimodal data. Neurocomputing 253:49–55

Guoping Zhao is currently working toward the PhD degree in the School of Information, Renmin University of China. He received the M.S. in 2012 from the Department of Computer Science at the Beihang University, and obtained his B.S. form Northwestern Polytechnical University. He was a research assistant at The Second Research Institute of China Aerospace Science and Industry Corporation, from 2012 to 2015. His research interests include computer vision, deep learning, and multimedia retrieval.

Jiajun Liu is currently an Associate Professor with Renmin University of China. He received his Ph.D. from The University of Queensland, Australia in 2012 and his B.Eng. from Nanjing University, China in 2006. He worked as a Researcher at IBM China Research Labs from 2006 to 2008. His research interests include multimedia retrieval, and management and mining for spatial temporal data. He has published extensively in various venues and has served as invited reviewer for multiple top conferences and journals.

Jiacheng Jiang received his B.S.(2016) from the north china university of technology. He is working toward the graduate degree from the School of Information, Renmin University of China. His research interests include deep learning, computer vision, and data mining.

Weiying Wang is currently a senior student in Miami University of the United States and pursuit the bachelor of computer science. His research interests include machine learning, natural language processing, multimedia retrieval. He worked in the lab of Renmin University of China to do research about the computer vision recognition and video retrieval in 2017 summer.