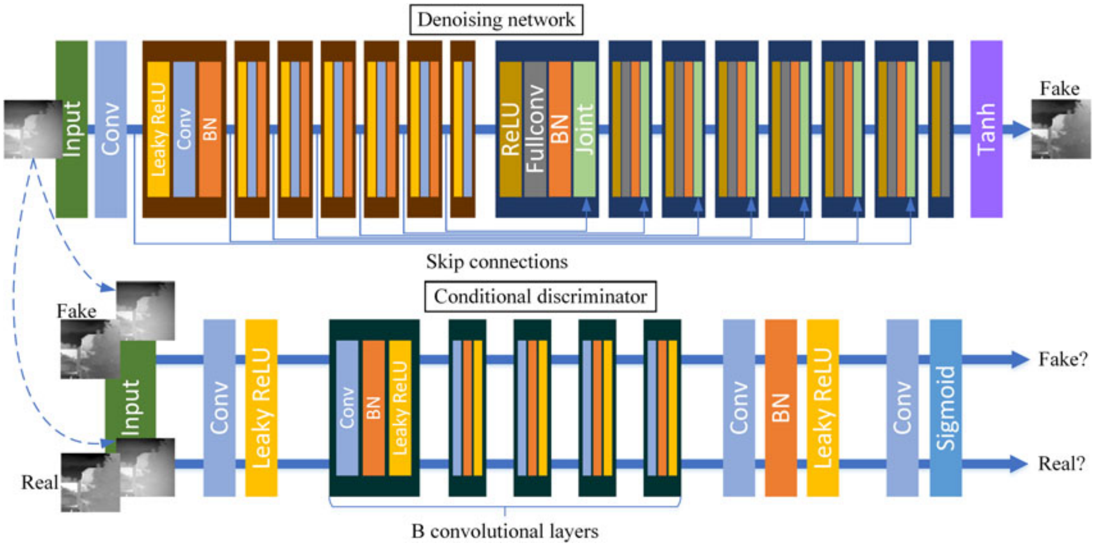
IEEE Photonics Journal
An IEEE Photonics Society Publication

Open Access

Single Infrared Image Optical Noise Removal Using a Deep Convolutional Neural Network

Volume 10, Number 2, April 2018

Xiaodong Kuang
Xiubao Sui
Yuan Liu
Qian Chen
Guohua GU



DOI: 10.1109/JPHOT.2017.2779149
1943-0655 © 2017 IEEE

Single Infrared Image Optical Noise Removal Using a Deep Convolutional Neural Network

Xiaodong Kuang, Xiubao Sui, Yuan Liu, Qian Chen, and Guohua GU

Jiangsu Key Laboratory of Spectral Imaging and Intelligence Sense, Nanjing University of Science and Technology, Jiangsu 210094, China

DOI:10.1109/JPHOT.2017.2779149

Manuscript received October 29, 2017; revised November 22, 2017; accepted November 28, 2017. Date of publication December 4, 2017; date of current version April 5, 2018. This work was supported in part by the National Natural Science Foundation of China under Grants 11503010 and 11773018, in part by the Fundamental Research Funds for the Central Universities under Grant 30916015103, in part by the Qing Lan Project and Open Research Fund of Jiangsu Key Laboratory of Spectral Imaging and Intelligence Sense under Grant 3091601410405. Corresponding author: X. Sui (e-mail: sxbhandsome@163.com).

Abstract: In this paper, we propose a deep learning method for single infrared image optical noise removal. With a fully convolutional neural network, it is able to eliminate the optical noise in single infrared image. Our architecture consists of two networks: a denoising network and a conditional discriminator. The denoising network takes a noise image as input and outputs a denoising result, while the discriminator tries to make the output look more like the target. Actually, only in the testing phase, this method is feed-forward. Significant image quality in experiments is achieved compared with the existing method.

Index Terms: Deep learning, infrared imaging, optical noise.

1. Introduction

Originally developed for military use, infrared imaging has slowly migrated into other fields as varied as medicine and archeology. More recently, the lowing of prices accelerates the development of uncooled infrared viewing technology. However, a variety of noise restricts the use of uncooled infrared camera including electronic noise and non-uniformity noise. The optical noise, which is a typical non-uniformity noise, often occurs mainly due to the inconsistency of the detector unit response. The response of individual detectors is usually influenced by temperature fluctuations of focal plane array detectors and camera lens. When we use a focal plane array detector that is calibrated at room temperature, to observe the high or low temperature target, the detector always outputs a noisy image where the intensity of the noise is proportional to the temperature difference. In addition, with the high-speed flight of the aircraft in aerial photography, the optical lens of the infrared camera mounted on the aircraft is affected by the speed and the temperature rises rapidly, which produces a strong optical noise in the output image.

To compensate the optical noise in the infrared camera, Cao *et al.* proposed a single-image-based method for optical temperature-dependent non-uniformity correction [1]. It built a bivariate polynomial model for the optical noise and exploited the gradient relationship between the optical noise model and the target image. After solving the optical noise model, the estimated optical noise was subtracted from the input infrared image. The method exhibits excellent performance in dealing with the weak optical noise. However, for the strong non-uniformity caused by the high-speed flight, it is difficult to remove the optical noise by using Cao's method.

Recently, deep learning makes major advances in many fields, such as image recognition [2]–[5], speech recognition [6]–[8], natural language understanding [9], object detection [10] and super resolution [11]–[13], image-to-image translation [14]. Based on these observations, we adopt a deep learning approach for infrared image optical noise removal.

Our method is based on an encoder-decoder network, which uses a Convolutional Neural Network that is trained with an adversarial loss. The encoder-decoder network is motivated by feature representation and is composed of a downsampling compression (encoder) and a upsampling restoration (decoder). The encoder-decoder network has achieved great success in image completion, but it is unable to get high-quality results for infrared image optical noise removal. The main reason is that the network will drop many scene information in the input by a downsampling operation, being unable to restore the expected target by a following upsampling operation. Our proposed method considers this point and presents a solution in the following section.

We leverage skip connections to make the scene information avoid being dropped, and present an architecture that achieves high-quality results for infrared image optical noise removal. This architecture consists of two networks: a denoising network, and a conditional discriminator. The denoising network employs an encoder-decoder architecture with skip connections to eliminate the optical noise, while the conditional discriminator is an auxiliary network used exclusively for training. The objective of the conditional discriminator is to distinguish whether the output looks like the expected target. Our algorithm differs fundamentally from traditional non-uniformity correction approaches, in that ours does not need to study and build the noise model beforehand. This model is automatically learned via hidden layers and parameter optimization only exists in the training phase. Once the training is done, we can eliminate the optical noise for infrared images by forward propagation without solving optimization problem.

Our paper is organized as follows. In Section 2, we introduce the background, and related works of traditional optical noise removal methods and convolutional neural networks. Section 3 details the design and the architecture of the proposed method. The experimental results are presented in Section 4, and the network characteristics are also studied. Finally, Section 5 concludes our paper.

2. Background and Related Work

2.1: Traditional Optical Noise Removal Methods

In the past, many successful methods, such as calibration-based [15], [16] and scene-based [17]–[22] approaches, have been presented for infrared optical noise removal. However, as the internal temperature of the infrared detector increases and the ambient temperature changes, new optical noise will appear on the calibrated detector. Although many scene-based methods can solve optical noise that appears over time, they will not work when the scene is stationary. Recently, Cao *et al.* proposed a single-image-based method for optical temperature-dependent non-uniformity correction [1]. It can effectively remove the spatially continues and low-magnitude optical noise but the performance is not satisfactory when the optical noise is too strong. Therefore, it is necessary to propose a new method that can remove optical noise with different intensities in single infrared image.

2.2 Convolutional Neural Networks

Since the beginning 1990s, Convolutional Neural Networks (CNNs) have been applied to speech recognition [23] and document reading [24]. In the early 2000s, CNNS show great success in detection, separation and identification of the object in images. One of the most typical applications is face recognition [25]. Despite these huge successful applications, CNNs have not been the focus of the machine-learning communities. However, in the ImageNet competition of 2012, deep CNNs achieved striking performance, nearly halving the error rates compared with the state-of-the-art methods [2]. Four factors contribute in the spectacular results: (i) the efficient use of GPUs, (ii) the propose of the Rectified Linear Unit (ReLU), (iii) an effective regularization method named dropout [26], (iv) a big data set (like ImageNet [27]) to generate large training images. These achievements have sparked a revolution in computer vision. CNNs now show an overwhelming advantage in all object recognition and object detection [28]–[31].

2.3 Convolutional Neural Network Architectures

A large number of computer vision tasks are addressed by designing deep CNN architectures. Usually increasing the depth of CNN architectures can significantly improve performance as it allows modeling a highly complex mapping, but the network parameters are difficult to converge by training. To design a very deep CNN architecture, batch normalization [32] is usually employed to normalize the data in each mini-batch, making the data avoid being amplified and shrunken by the weights. In addition, residual blocks [33] and skip connections [12] are also used to ease the difficulty of training a deeper CNN architecture, as they help in handling the vanishing gradient problem in very deep networks.

Residual CNNs and encoder-decoder CNNs are two typical CNN architectures. By adding residual blocks repeatedly and employing batch normalization, a residual CNN architecture can be designed very deep to achieve higher resolution for image super-resolution [13]. An encoder-decoder CNN architecture is often used for image inpainting [34], [35]. Recently an encoder-decoder CNN with skip-connections [14] is presented to solve a variety of image-to-image translation problems, such as photo generation and semantic segmentation.

Generative Adversarial Networks (GANs) [36] are now the most popular CNN architectures that produce encouraging results. GANs employ two CNNs, one called generator and the other called discriminator, both of which are trained in a competing way. The most typical application is natural image generation by designing special GANs [37]. Conditional GANs use a conditional generative network, which conditions on input images and produces corresponding output images. Conditional GANs have achieve better quality results than GANs, and many state-of-the-art approaches have employed the special architecture, such as conditional random fields [38] and feature matching [39].

3. Method

Our work builds upon deep CNNs trained to solve the problem of optical noise removal. A single denoising network is employed for optical noise removal. A conditional discriminator is employed to train the denoising network to accurately eliminate optical noise. In the training phase, the discriminator is trained to be able to accurately distinguish real images from denoising results, while the denoising network tries to fool it. An overview of this architecture is diagrammed in Fig. 1.

3.1. Denoising Network

Our denoising network follows a fully convolutional network. Fig. 2 shows an architecture of the denoising network for every feature map, and Table 1 shows parameter settings of each convolution layer. The denoising network takes a grayscale image with optical noise as input and outputs a denoising result. The general architecture is based on an encoder-decoder model with skip connections [14], [40].

This encoder-decoder model is composed of an encoder and a decoder. In the encoding phase, the input is passed through a series of convolutional layers that gradually halve the input resolution to 1 × 1 and double channels of output feature maps from 64 to 512. The decoder then performs the reverse process of the encoder using deconvolution layers.

Each skip connection is used to concatenate every feature map in the encoder with the corresponding one in the decoder, except the middle feature map. It is very critical for optical noise removal, as the middle feature map have dropped much scene information and are unable to be used to reconstruct the target by the decoder. By using skip connections, feature maps in the

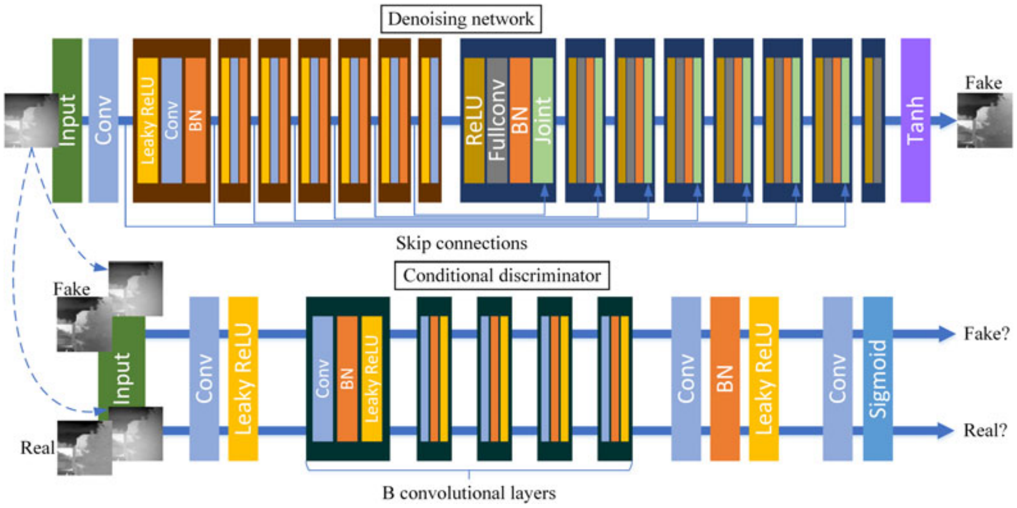


Fig. 1. Framework of our approach showed for each convolutional layer. The slope of the leaky ReLU is 0.2. B is 2 by default.

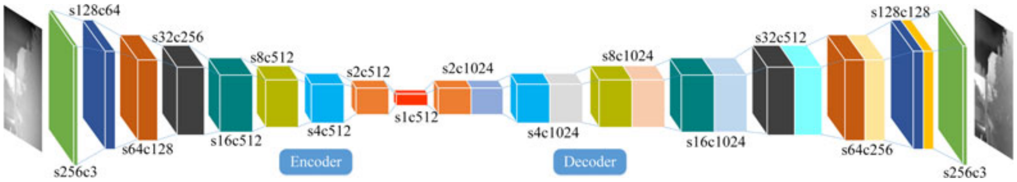


Fig. 2. Architecture of the denoising network for every feature map. The red block represents the middle feature map without skip connection, s represents the size of the feature map, and c represents the channel number of the feature map.

encoder are able to bypass the middle feature map and can be passed directly to the decoder. Without using skip connections, the network leads to extremely poor performance, outputting nearly identical results regardless of inputs.

3.2 Conditional Discriminator

The conditional discriminator is employed to try to distinguish whether the input images in the discriminator are real or fake. It employs CNNs that compress the input into a series of low resolution feature maps. Fig. 3 shows an architecture of the conditional discriminator for every feature map and Table 2 shows the parameter settings of every convolution layer. Our conditional discriminator is based on GANs.

Unlike the normal discriminator of which the input only contains fake and real images, our conditional discriminator introduces the noise image as part of the input to increase the stability in learning and improve the performance of the denoising network [41]. At this point, the input of our conditional discriminator consists of two parts, one part is the fake image concatenated with the noise image, the other part is the real image concatenated with the noise image. The size of all input images is 256 × 256. This conditional discriminator consists of five convolution layers that uses a stride of 2 for the first three layers while 1 for last two ones, and finally produces a smaller image with a size of 30 × 30. It is different from regular discriminator in GANs that regular discriminator in GANs maps from an image to a single scalar output, which signifies real or fake, whereas our discriminator outputs a small image [14], where each pixel signifies whether the patch in the input is real or fake. To compute the patch size, each pixel in the output can be considered as a neuron in a convent that we can trace back its receptive field to see which input pixels it is sensitive to. In our discriminator, the receptive fields turn out to be 70 × 70 patches in the input image. This is all

TABLE 1
Parameter Settings of Every Convolution Layer in the Denoising Network

Type	Kernel	Stride	Padding	Channels
conv.	4 × 4	2 × 2	1 × 1	3, 64
conv.	4 × 4	2 × 2	1 × 1	64, 128
conv.	4 × 4	2 × 2	1 × 1	128, 256
conv.	4 × 4	2 × 2	1 × 1	256, 512
conv.	4 × 4	2 × 2	1 × 1	512, 512
conv.	4 × 4	2 × 2	1 × 1	512, 512
conv.	4 × 4	2 × 2	1 × 1	512, 512
conv.	4 × 4	2 × 2	1 × 1	512, 512
deconv.	4 × 4	2 × 2	1 × 1	1024, 512
deconv.	4 × 4	2 × 2	1 × 1	1024, 512
deconv.	4 × 4	2 × 2	1 × 1	1024, 512
deconv.	4 × 4	2 × 2	1 × 1	1024, 512
deconv.	4 × 4	2 × 2	1 × 1	1024, 256
deconv.	4 × 4	2 × 2	1 × 1	512, 128
deconv.	4 × 4	2 × 2	1 × 1	256, 64
deconv.	4 × 4	2 × 2	1 × 1	128, 3

The first channel number represents the number of expected input planes in the image given into, and the second channel number represents the number of output planes the convolution layer will produce.

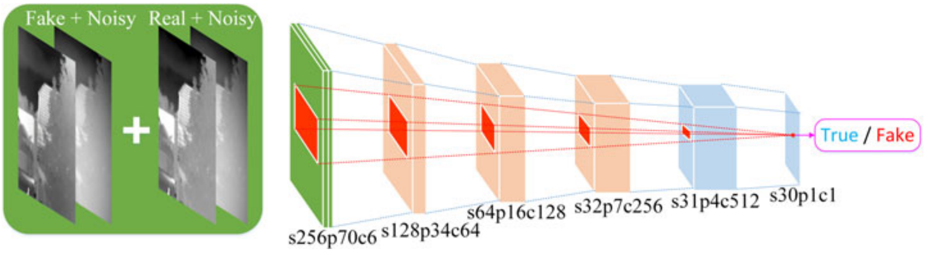


Fig. 3. Architecture of the discriminator for every feature map. The green blocks represent the input images. The bright orange blocks are the feature maps obtained by using convolution layers with a stride of 2, and the bright blue blocks are the feature maps obtained by using convolution layers with a stride of 1. Note that the input of the discriminator is a 6-channel image that concatenates the fake or real image with the noise image. The discriminator eventually outputs a single channel image, where each pixel is sensitive to a receptive field with 70 × 70 red patches in the input image. S represents the size of the feature map, p represents the patch size, and c represents the channel number of the feature map.

mathematically equivalent to if we had manually chopped up the image into 70 × 70 overlapping patches, run a regular discriminator over each patch, and averaged the results.

3.3 Training

Let $\{x_l\}_{l=1}^{N}$ denote an image dataset, with x_l a 8-bit image with a resolution of 256 × 256 and N the number of images. We first set up a training dataset $\{x_l, y_l\}_{l=1}^{N}$, where y_l is the corresponding noise

TABLE 2
Parameter Settings of Every Convolution Layer in the Conditional Discriminator

Type	Kernel	Stride	Padding	Channels
conv.	4 × 4	2 × 2	1 × 1	6, 64
conv.	4 × 4	2 × 2	1 × 1	64, 128
conv.	4 × 4	2 × 2	1 × 1	128, 256
conv.	4×4	1×1	1×1	256, 512
conv.	4×4	1×1	1×1	512, 1

The first channel number represents the number of expected input planes in the image given into, and the second channel number represents the number of output planes the convolution layer will produce.

image that can be obtained according to the following simple formula:

$$\begin{cases} y(i,j) = x(i,j) - \delta[(i-c_i)^2 + (j-c_j)^2] \\ c_i = randperm(256,1),\, c_j = randperm(256,1) \end{cases}, \tag{1}$$

where (i, j) is the coordinates of the pixels, (c_i, c_j) represents the center point of the optical noise and δ controls the intensity of the optical noise. Note that the center point of the optical noise is random in each image. Finally, all images in the training dataset are normalized to the range [0, 1].

To effectively remove the optical noise by training, we consider a joint loss function that consists of denoising loss and adversarial loss. The denoising loss employs L1 distance to capture the general scene structure, while the adversarial loss follows a conditional GAN (cGAN) to concern texture details. This joint loss function encourages the network to produce high-quality images, and has achieved great success in many computer vision tasks, such as image super-resolution and image inpainting. Training is done with backpropagation.

L1 distance is used to encourage the denoising result to near the ground truth image. The L1 loss function is expressed as:

$$L1(x, y) = \left\| x - R(y) \right\|_1, \tag{2}$$

where R is the denoising network, x is the target image that can also be named as a real image, y is the correspond noise image, and $R(y)$ is a fake image.

The conditional discriminator network works as an adversarial loss, which is derived from GANs [36]. A GAN is composed of two models. A generative model G, which is trained to output a plausible image that can confuse the discriminator given a noise image$y$: $G : y \Rightarrow G(y)$. A discriminative model D, which on the other hand is one that discriminates between two (or more) different images - for example a convolutional neural network that is trained to output 1 given a real image x and 0 given a generative image $G(y)$ otherwise: $D : D(x) \Rightarrow 1, D(G(y)) \Rightarrow 0$. In this work, we use the denoising network to replace the generative model and introducing the input image y as a conditioning on the discriminator: $D : D(x, y) \Rightarrow 1, D(R(y), y) \Rightarrow 0$. Therefore the adversarial loss for the conditional discriminator network, $L_{adv}(x, y)$, is:

$$L_{adv}(x, y) = \min_R \max_D E[\log(D(x, y)) + \log(1 - D(R(y), y))]. \tag{3}$$

We combine these two loss functions as:

$$L = \alpha L1 + L_{adv}, \tag{4}$$

where α is 100 in our implementation.

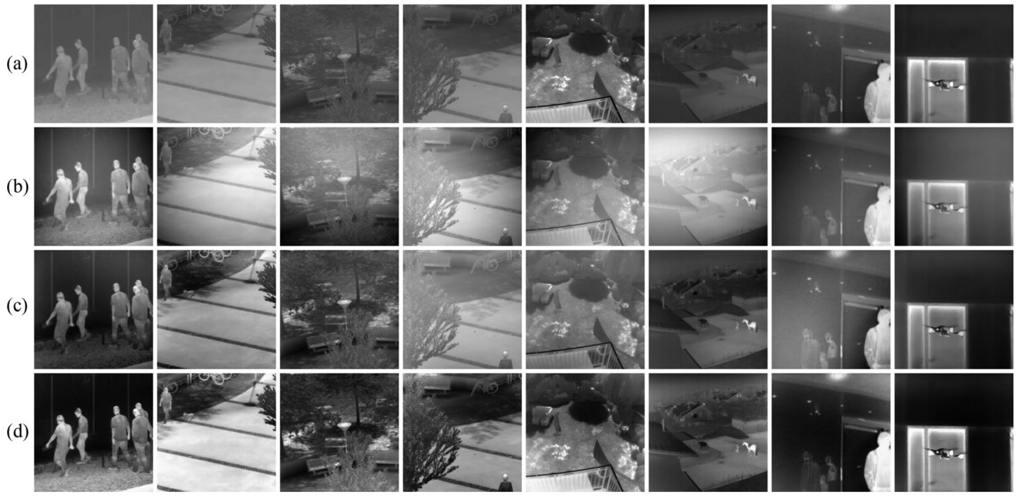


Fig. 4. Comparisons with Cao's method on the infrared dataset. The intensity of the optical noise is set to $\delta = 0.002$. (a) the target images, (b) input images, (c) images corrected by Cao's method, and (d) images corrected by our method.

Using a small batch size 1 could reduce GPU memory usage but will slow down the training speed. During training, the denoising network and the conditional discriminator are optimized using alternating ADAM algorithm [42]. Our model is implemented in Torch with a GTX1080 GPU.

4. Experiments

In theory, we should use infrared images to train a special network for infrared image optical noise removal. However, it is difficult to collect large amounts of infrared images and infrared image usually contain a variety of noise, which will affect the training. A grayscale visible image and an infrared image can both be approximated as grayscale images. In addition, compared with infrared images, grayscale visible images have more texture details and a higher contrast. If we can accurately remove the optical noise on grayscale visible images, we should be able to achieve the same or even better results on infrared images. Around these considerations, we use a great diversity of grayscale visible images from ImageNet [27] to train our network.

We randomly select 10241 images from ImageNet and crop them into 185386 sub-images with a stride of 41. All images are converted to grayscale images, and then we generate corresponding noise images according to (1). The size of sub-images is 256×256 and the intensity of the optical noise is $\delta = 0.01$. We train the network over 4 epochs (741584 iterations). The entire training process takes about 15 hours using a GTX1080 GPU.

In order to quantitatively evaluate the experimental results, peak signal-to-noise ratio (PSNR), structure similarity index (SSIM) [43], noise quality measure (NQM) [44], and measure of enhancement (EME) [45] are employed as the evaluation metrics.

4.1 Comparison With Existing Method

We compare our method with Cao's approach [1] on a dataset consisting of 85 infrared images quantitatively and qualitatively. All infrared images are shown at 256×256 resolution, and the intensity of the optical noise added to the infrared images is $\delta = 0.002$ and $\delta = 0.005$ respectively.

Figs. 4 and 5 show the qualitative results of different methods. We can clearly observe that Cao's method is unable to effectively eliminate the optical noise in all scenes, unlike our method. Especially, if the luminance difference between the object and the background is large, part of the luminance of the object is easily considered as optical noise and then is removed by using Cao's method (see results in the third and sixth columns of Fig. 5). Furthermore, the strong optical noise is able

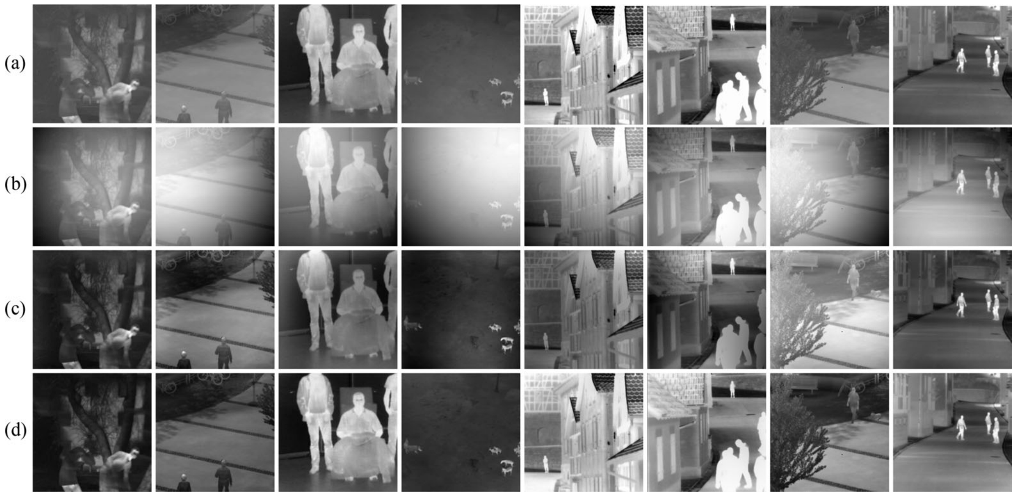


Fig. 5. Comparisons with Cao's method on the infrared dataset. The intensity of the optical noise is set to $\delta = 0.005$. (a) the target images, (b) input images, (c) images corrected by Cao's method, and (d) images corrected by our method.

TABLE 3

Average Results of Evaluation Metrics Using Different Approaches on the Infrared Dataset

Eval. Met	Intensity	Cao's	Ours
PSNR	0.002	**18.1698**	17.4761
	0.005	17.6157	**20.3598**
SSIM	0.002	**0.8586**	0.7698
	0.005	**0.8564**	0.8422
NQM	0.002	**10.0412**	9.8992
	0.005	9.9609	**12.0197**
EME	0.002	7.1406	**10.0314**
	0.005	6.8311	**7.3625**

The average contrast of target images is 4.2403.

to only be reduced rather than being completely eliminated by using Cao's method (see results in the seventh column of Fig. 5). In contrast, our method does not need to consider these problems and produces compelling results. Interestingly, our method can also increase the contrast of infrared images, which is proportional to the intensity difference of the optical noise added on the infrared images and the training images. The main reason is that the network needs to extract quantitative optical noise that has been learned by training. If the intensity of the optical noise added on infrared images is lower than that added on training images, the network will extract the luminance of the infrared image itself as part of the optical noise to improve the contrast. Usually we add a relatively strong optical noise on training images.

Quantitative results are shown in Table 3. Surprisingly, our approach yields lower scores than Cao's method on PSNR, SSIM and NQM. It is obvious that the quality of images using our results is the best from the above qualitative observation. The main reason is that the infrared results output using our method have a better quality than the original label images. The contrast of results obtained using our method is higher than that of target images. Specifically, the average

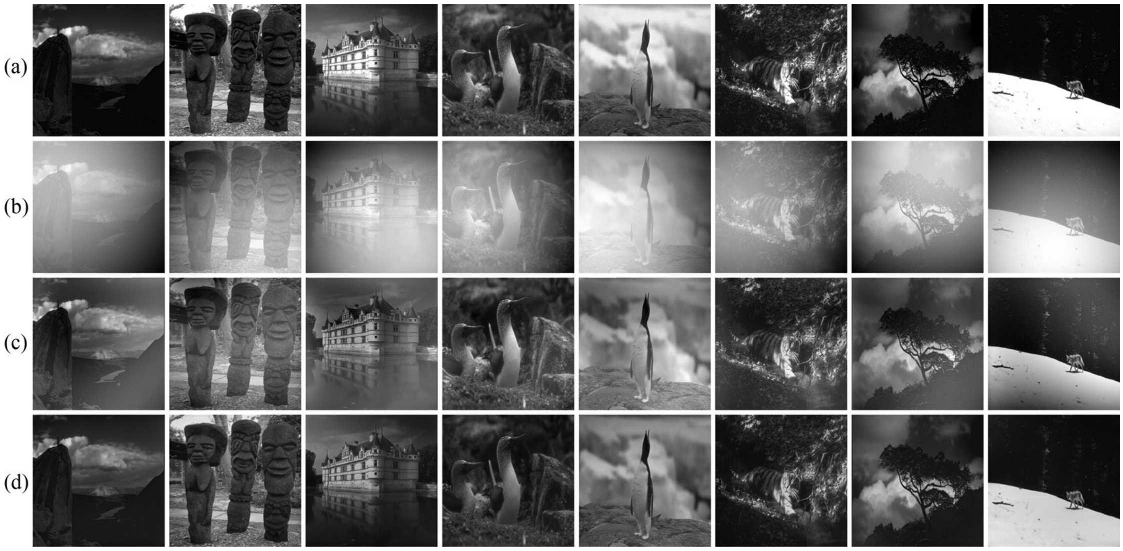


Fig. 6. Comparisons with Cao's method on dataset 'b100'. The intensity of the optical noise is set to $\delta = 0.01$. (a) the target images, (b) input images, (c) images corrected by Cao's method, and (d) images corrected by our method.

TABLE 4
Average Results of Evaluation Metrics Using Different Approaches on the Dataset 'b100'

Eval.Met	Intensity	Cao's	Ours
PSNR	0.01	10.5029	**22.0043**
SSIM		0.1703	**0.9251**
NQM		3.7213	**14.4280**
EME		12.9070	**13.9602**

The average contrast of target images is 19.2648.

contrasts of these two experimental results obtained by our method are 10.0314 and 7.3625, higher than that of target images with 4.2403. This also confirms that the larger the intensity difference of the optical noise added on infrared images and training images is, the higher the contrast of the results obtained by our approach is.

For a fair comparison, a dataset 'b100', which contains 100 high-contrast natural images in Berkeley Segmentation Dataset [46], is used. All images in dataset 'b100' are not used in the training dataset. Note that unless other mentioned, all images used for test are rescaled to the size of 256 × 256. Results in Fig. 6 and Table 4 all demonstrate that our method is superior to Cao's approach.

4.2 Analysis of the CNN Architecture in the Denoising Network

Can other CNN architectures effectively remove optical noise? We employ two typical CNN architectures, a residual CNN architecture (ResCNN) and an encoder-decoder CNN architecture (EDCNN), to replace the proposed architecture in the denoising network. The ResCNN architecture is based on the generative network in [13] and Fig. 7 shows its framework. Note that our ResCNN architecture only use four residual blocks limited by the graphics memory. The EDCNN architecture is

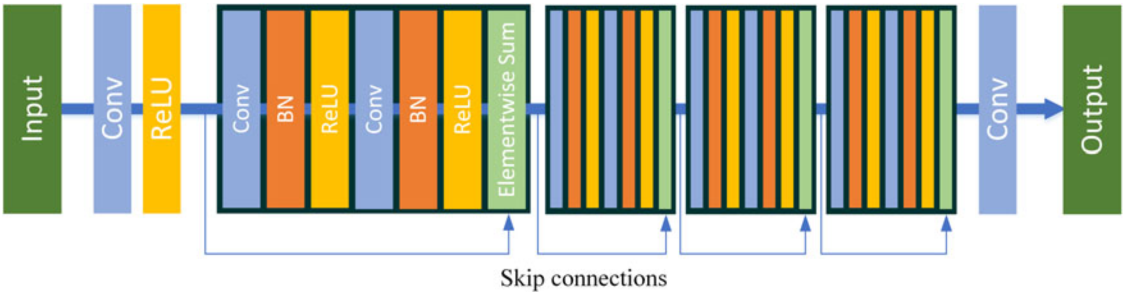


Fig. 7. Framework of the ResCNN showed for each convolutional layer. The kernel size of all convolution is 3 × 3, the stride and the padding size are 1. Except the last convolution outputting 3 feature maps, the number of feature maps of other convolutions are 64.

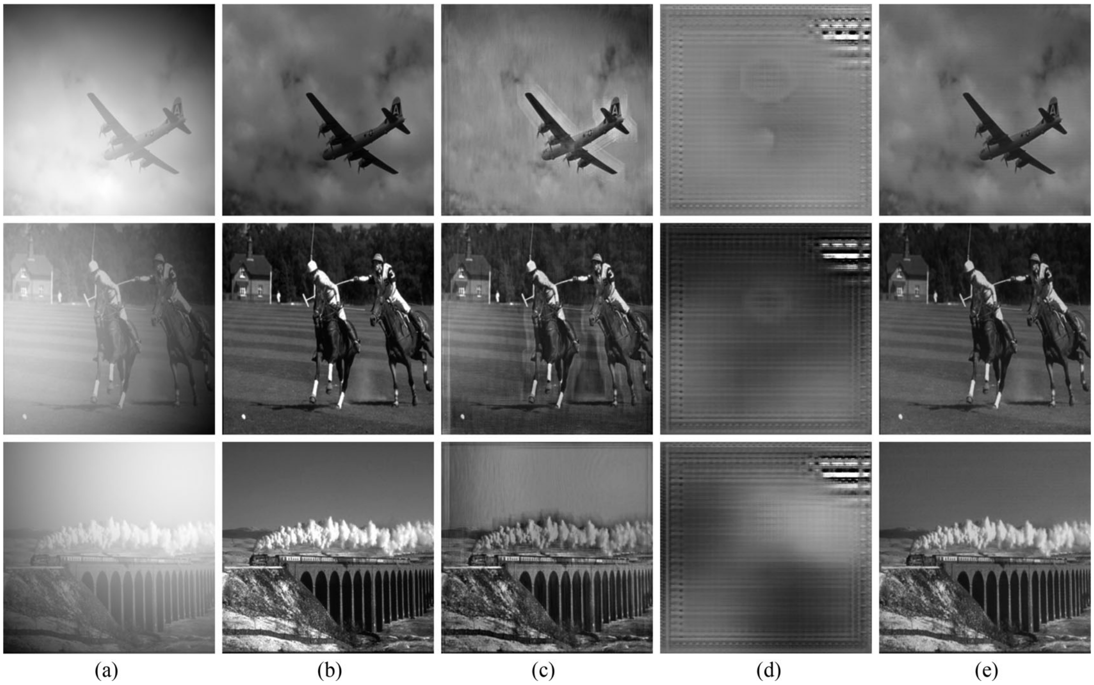


Fig. 8. Different CNN architectures in the denoising network produce results of different quality on the dataset ‘b100’. The intensity of the optical noise is set to $\delta = 0.01$. (a) Input images, (b) target images, (c) images corrected by ResCNN, (d) images corrected by EDCNN, and (e) images corrected by our method.

built simply by severing the skip connections in the proposed architecture. For a fair comparison, dataset ‘b100’ is the primary dataset in the following experiments.

Results are shown in Fig. 8. We can observe that the ResCNN architecture can effectively remove the optical noise but produces artifacts at the edge of images. The EDCNN architecture leads to very poor results, generating nearly identical noise outputs regardless of inputs. In contrast, the proposed architecture achieves the superior results.

4.3 Analysis of the Patch Size in the Discriminator

In this section, we explore the relationship between the network performance and the receptive fields of the input image in the discriminator. Our default receptive fields is 70 × 70 patches. To change the patch size $n_p \times n_p$, we modify the number of convolution layers in Table 2. Specifically, we fix the first and the last two convolution layers, and set the number B of the other convolution

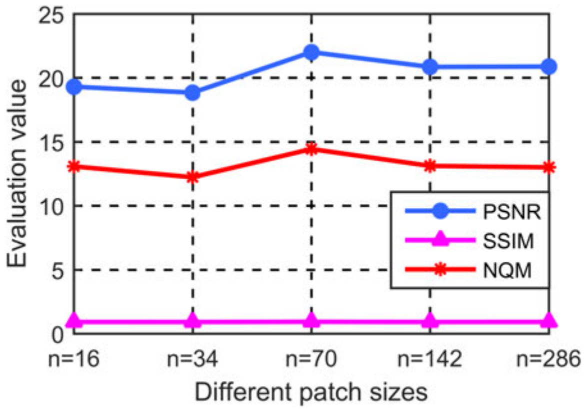


Fig. 9. Comparison of evaluation metrics using different patch sizes on the dataset ‘b100’.

layers from 1 to 5. The number 1 corresponds to the patch size of $n_p = 16$, and the number 5 corresponds to the patch size of $n_p = 286$. Note that the maximum number of output planes the middle convolution layers will produce is 512.

The comparison curves are shown in Fig. 9. We can observe that receptive fields of 70 × 70 patches yield the highest scores in all evaluation metrics. The results suggest that a moderate patch size is beneficial for the network to achieve a better performance.

4.4 Analysis of the Loss Function

Which part of the loss in (4) is important? We conduct five experiments with different combinations of loss functions, i.e., L1, GAN, cGAN, L1+GAN and L1+cGAN.

The results are shown in Fig. 10. L1 alone and cGAN alone result in reasonable images but blurry details. GAN alone leads to poor performance, generating nearly identical noise outputs regardless of inputs. It only cares whether the output is real. It is apparent that conditioning penalize error between input and output, encouraging the output to respect the input. Meanwhile, L1 also measure the quality of the match between output and target, prompting the output to respect the target. They all contribute to remove the optical noise while restoring the general structure. Indeed, L1+cGAN performs much better than L1+GAN, which leads to some artifacts. The comparison curves in Fig. 11 bear out this conclusion: L1+cGAN achieves the highest scores.

4.5 Analysis of Weights for L1 Loss and Adversarial Loss

The default weight in the joint loss function is 100. Can the network performance be improved by increasing or decreasing the weight of L1 loss in the joint loss function? We conduct two controlled experiments, setting $\alpha = 50$ and $\alpha = 200$ respectively. The intensity of the optical noise added on test images is set to $\delta = 0.01$. Quantitative results on the dataset ‘b100’ are shown in Table 5. We can see that, the default weight $\alpha = 100$ achieves the highest scores in all evaluation metrics. It indicates that increasing or decreasing the weight of L1 loss is unable to further improve the network performance and the default weight is the most appropriate.

4.6 Analysis of Computational Time

Computational time of optical noise removal depends on the resolution of the input image. Table 6 shows the computational time for different resolutions. They are all evaluated using an Intel Core i7-7700K CPU @ 4.2 GHz with 8 cores and GTX1080 GPU. We can observe that even larger images are able to be processed in under a second using a GPU.

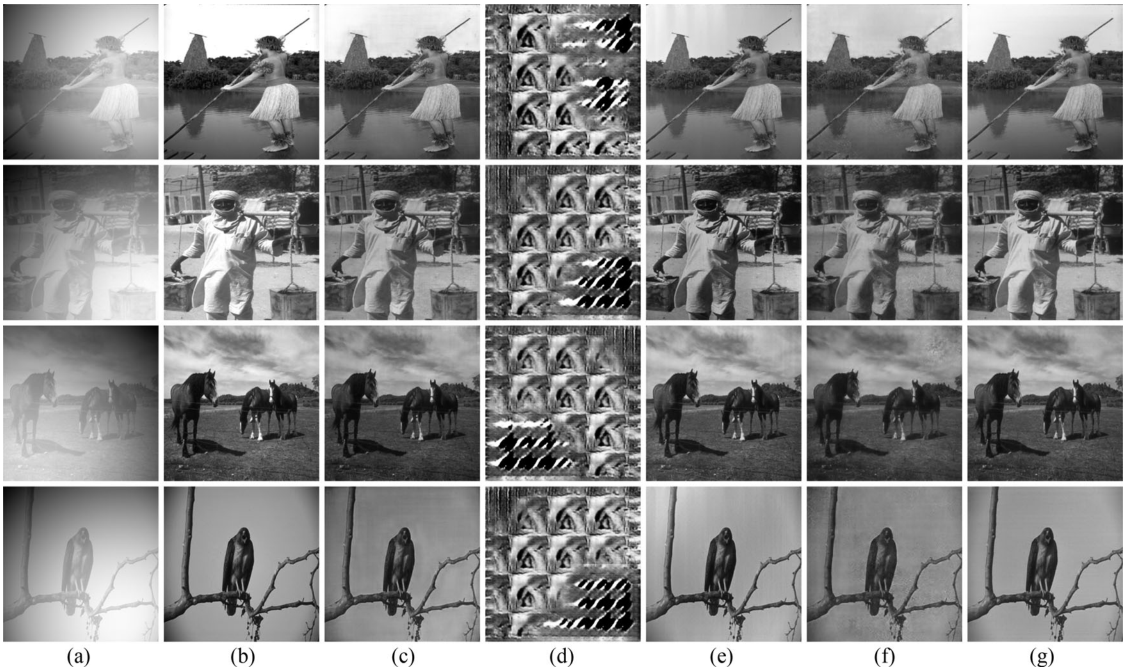


Fig. 10. Different losses produce different quality of images on the dataset ‘b100’. The intensity of the optical noise is set to $\delta = 0.01$. (a) Input images, (b) target images, (c) images corrected by using L1, (d) images corrected by using GAN, (e) images corrected by using cGAN, (f) images corrected by using L1 + GAN, and (g) images corrected by using L1 + cGAN.

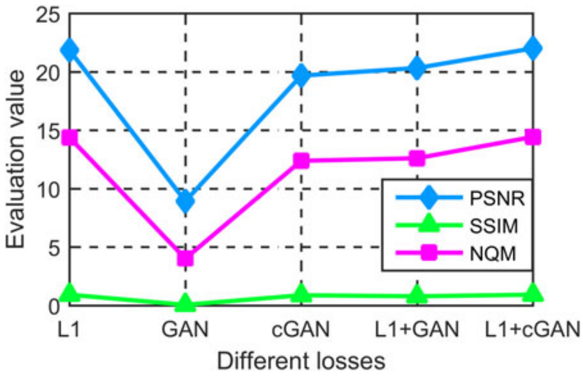


Fig. 11. Comparison of evaluation metrics using different losses on the dataset ‘b100’.

TABLE 5

The Network Performance Does Not be Improved by Trying a Few More Weights

Eval. Met	Intensity	$\alpha = 50$	$\alpha = 100$	$\alpha = 200$
PSNR	0.01	21.5284	**22.0043**	21.2067
SSIM		0.9178	**0.9251**	0.9233
NQM		14.1253	**14.4280**	13.4814

TABLE 6
Analysis of Computational Time for Different Resolutions

Image resolution	Time (s)
256 × 256	0.07
512 × 512	0.24
1024 × 1024	0.92

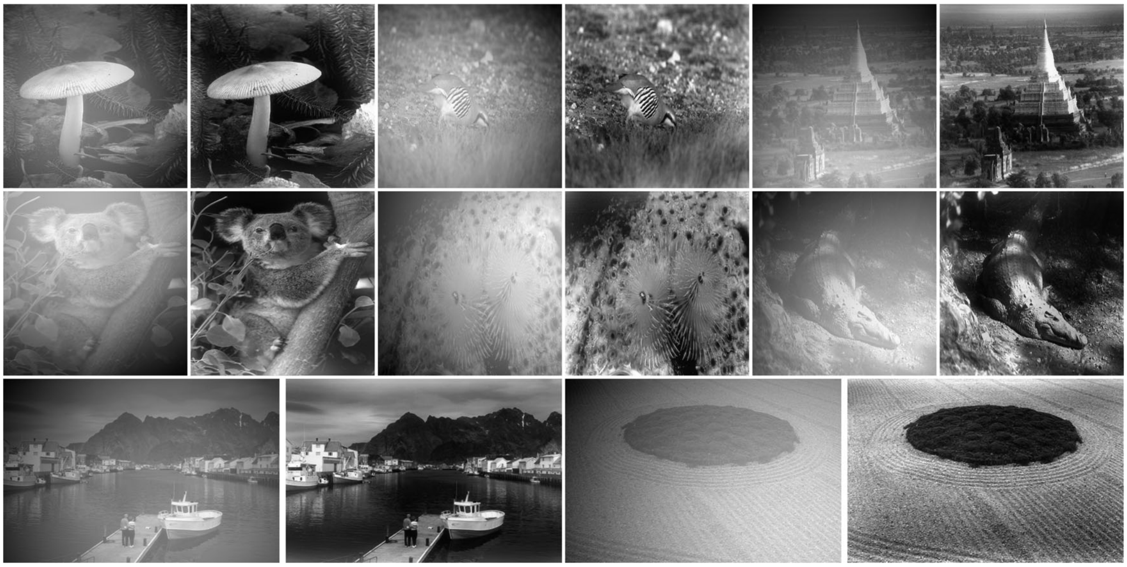


Fig. 12. Results of different size obtained by our method. The size of images are 512 × 512 in the first two rows and are 512 × 768 in the last row.

4.7 Application for Larger Images

All images in the training dataset and the test dataset are presented at 256 × 256 resolution. Can our network be applied to larger images and achieve the same performance? To explore this property, we resize the images in dataset 'b100' to two larger sizes, one size is 512 × 512, and the other size is 512 × 768. The experimental results are shown in Fig. 12. We can observe that for a larger image, our approach also effectively removes the optical noise and achieves satisfactory results.

4.8 Fine-Tune on Infrared Images

Since there are fewer infrared images for training, our network is trained on ImageNet and tested on infrared images. Can the network performance further be improved if we train the network on ImageNet first and fine-tune on a smaller set of infrared images? We collect an infrared dataset containing 276 images downloaded from the Internet.[1] Since the resolution of the original images is 360 × 240, we do not crop them to generate more images, only resize them to 256 × 256. The intensity of the optical noise is $\delta = 0.01$. Once the training on the ImageNet is done, we continue

[1] http://vcipl-okstate.org/pbvs/bench/Data/01/download.html.

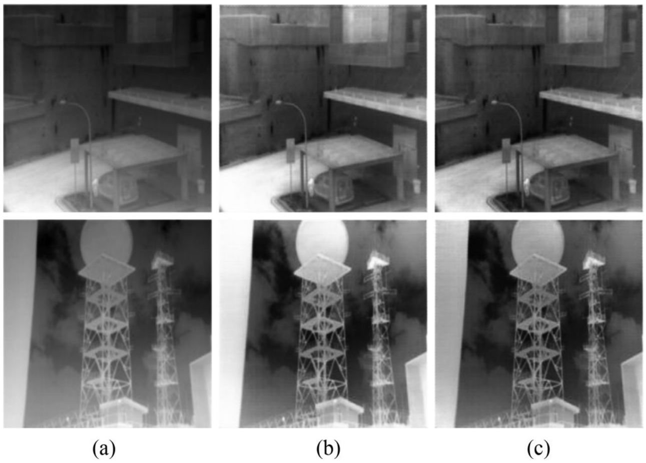


Fig. 13. Fine-tune on an infrared dataset helps to improve the denoising ability of the network. (a) Real-captured noisy infrared images. (b) Results obtained using the network trained on ImageNet. (c) Results obtained using the network trained on ImageNet first and fine-tune on an infrared dataset.

training on the infrared dataset over 20 epochs. Real-captured noisy infrared images obtained from [47] are used to evaluate the network performance. Results in Fig. 13 show that fine-tune on infrared images helps to further remove the optical noise and achieve high-quality images.

5. Conclusion

We have proposed a deep learning method for single infrared image optical noise removal. We have shown that, although grayscale visible images are used as the training images, the network is also able to be trained to solve infrared vision problems, even producing results of a higher quality. Experimental results have certified that deep learning is a promising method to remove the optical noise for infrared images. We have presented in-depth comparisons with Cao's method, achieving superior results in various infrared images.

References

[1] Y. Cao and C. L. Tisse, "Single-image-based solution for optics temperature-dependent nonuniformity correction in an uncooled long-wave infrared camera[J]," *Opt. Lett.*, vol. 39, no. 3, pp. 646–648, 2014.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks[C]," *Adv. Neural Inf. Process. Syst.*, pp. 1097–1105, 2012.

[3] C. Farabet *et al.*, "Learning hierarchical features for scene labeling[J]," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1915–1929, Aug. 2013.

[4] J. J. Tompson *et al.*, "Joint training of a convolutional network and a graphical model for human pose estimation[C]," *Adv. Neural Inf. Process. Syst.*, pp. 1799–1807, 2014.

[5] C. Szegedy *et al.*, "Going deeper with convolutions[C]," in *Proc. Int. Conf. IEEE Comput. Vis. Pattern Recog.*, 2015.

[6] T. Mikolov *et al.*, "Strategies for training large scale neural network language models[C]," in *Proc. 2011 Int. Workshop IEEE Autom. Speech Recog. Understanding*, 2011, pp. 196–201.

[7] G. Hinton *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups[J]," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.

[8] T. N. Sainath *et al.*, "Deep convolutional neural networks for LVCSR[C]," in *Proc. 2013 IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 8614–8618.

[9] R. Collobert *et al.*, "Natural language processing (almost) from scratch[J]," *J. Mach. Learn. Res.*, vol. 12, pp. 2493–2537, Aug. 2011.

[10] R. Girshick *et al.*, "Rich feature hierarchies for accurate object detection and semantic segmentation[C]," in *Proc. Int. Conf. IEEE Comput. Vis. Pattern Recog.*, 2014, pp. 580–587.

[11] C. Dong *et al.*, "Image super-resolution using deep convolutional networks[J]," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, Feb. 2016.

[12] J. Kim, J. Kwon Lee, and K. Mu Lee, "Deeply-recursive convolutional network for image super-resolution[C]," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 1637–1645.

[13] C. Ledig *et al.*, Photo-realistic single image super-resolution using a generative adversarial network[J]. 2016. [Online]. Available: arXiv:1609.04802.

[14] P. Isola *et al.*, Image-to-image translation with conditional adversarial networks[J]. 2016. [Online]. Available: arXiv:1611.07004.

[15] D. L. Perry and E. L. Dereniak, "Linear theory of nonuniformity correction in infrared staring sensors," *Opt. Eng.*, vol. 32, pp. 1854–1859, 1993.

[16] M. Schulz and L. Caldwell, "Nonuniformity correction and correctability of infrared focal plane arrays," *Proc. SPIE*, vol. 2470, pp. 200–211, 1995.

[17] W. Zhao and C. Zhang, "Scene-based nonuniformity correction and enhancement: Pixel statistics and subpixel motion," *J. Opt. Soc. Amer. A*, vol. 25, pp. 1668–1681, 2008.

[18] J. G. Harris and Y. M. Chiang, "Nonuniformity correction of infrared image sequences using the constant-statistics constraint," *IEEE Trans. Image Process.*, vol. 8, no. 8, pp. 1148–1151, Aug. 1999.

[19] D. A. Scribner, K. A. Sarkady, M. R. Kruer, and J. T. Caulfield, "Adaptive nonuniformity correction for IR focal plane arrays using neural networks," *Proc. SPIE*, vol. 1541, pp. 100–109, 1999.

[20] B. M. Ratliff and M. M. Hayat, "An algebraic algorithm for nonuniformity correction in focal-plane arrays," *J. Opt. Soc. Amer. A*, vol. 19, pp. 1737–1747, 2002.

[21] B. Narayanan, R. C. Hardie, and R. A. Muse, "Scene-based nonuniformity correction technique that exploits knowledge of the focal-plane array readout architecture," *Appl. Opt.*, vol. 44, pp. 3482–3491, 2005.

[22] J. E. Pezoa and M. M. Hayat, "Multimodel Kalman filtering for adaptive nonuniformity correction in infrared sensors," *J. Opt. Soc. Amer. A*, vol. 23, pp. 1282–1291, 2006.

[23] A. Waibel *et al.*, "Phoneme recognition using time-delay neural networks[J]," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 3, pp. 328–339, Mar. 1989.

[24] Y. LeCun *et al.*, "Gradient-based learning applied to document recognition[J]," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[25] S. Lawrence *et al.*, "Face recognition: A convolutional neural-network approach[J]," *IEEE Trans. Neural Netw.*, vol. 8, no. 1, pp. 98–113, Jan. 1997.

[26] N. Srivastava *et al.*, "Dropout: A simple way to prevent neural networks from overfitting[J]," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[27] J. Deng *et al.*, "Imagenet: A large-scale hierarchical image database[C]," in *Proc. 2009 Int. Conf. IEEE Comput. Vis. Pattern Recog.*, 2009, pp. 248–255.

[28] J. Tompson *et al.*, "Efficient object localization using convolutional networks[C]," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 648–656.

[29] Y. Taigman *et al.*, "Deepface: Closing the gap to human-level performance in face verification[C]," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 1701–1708.

[30] P. Sermanet *et al.*, Overfeat: Integrated recognition, localization and detection using convolutional networks[J]. 2013. [Online]. Available: arXiv:1312.6229.

[31] K. Simonyan and A. Zisserman, Very deep convolutional networks for large-scale image recognition[J]. 2014. [Online]. Available: arXiv:1409.1556.

[32] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift[C]," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[33] K. He *et al.*, "Deep residual learning for image recognition[C]," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 770–778.

[34] D. Pathak *et al.*, "Context encoders: Feature learning by inpainting[C]," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 2536–2544.

[35] C. Yang *et al.*, High-resolution image inpainting using multi-scale neural patch synthesis[J]. 2016. [Online]. Available: arXiv:1611.09969.

[36] I. Goodfellow *et al.*, "Generative adversarial nets[C]," *Adv. Neural Inf. Process. Syst.*, pp. 2672–2680, 2014.

[37] A. Radford, L. Metz, and S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks[J]. 2015. [Online]. Available: arXiv:1511.06434.

[38] L. C. Chen *et al.*, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs[J]. 2016. [Online]. Available: arXiv:1606.00915.

[39] A. Dosovitskiy and T. Brox, "Generating images with perceptual similarity metrics based on deep networks[C]," *Adv. Neural Inf. Process. Syst.*, pp. 658–666, 2016.

[40] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation[C]," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assisted Intervention*, 2015, pp. 234–241.

[41] M. Mirza and S. Osindero, Conditional generative adversarial nets. 2014. [Online]. Available: arXiv:1411.1784.

[42] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult[J]," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.

[43] Z. Wang *et al.*, "Image quality assessment: From error visibility to structural similarity[J]," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[44] N. Damera-Venkata *et al.*, "Image quality assessment based on a degradation model[J]," *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 636–650, Apr. 2000.

[45] T. Celik and T. Tjahjadi, "Contextual and variational contrast enhancement[J]," *IEEE Trans. Image Process.*, vol. 20, no. 12, pp. 3431–3441, Dec. 2011.

[46] D. Martin *et al.*, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics[C]," in *Proc. 8th IEEE Int. Conf. Comput. Vis.*, 2001, vol. 2, pp. 416–423.

[47] Y. Cao and C. L. Tisse, "Solid-state temperature-dependent NUC (non-uniformity correction) in uncooled LWIR (long-wave infra-red) imaging system[C]," *Proc. SPIE.*, vol. 8704, 2013.