Unsupervised model-driven neural network based image denoising for transmission line monitoring*

YAO Nan[1], WANG Zhen[1], ZHANG Jun[2], ZHU Xueqiong[1], and XUE Hai[1]**
1. Research Institute, State Grid Jiangsu Electric Power Co., Ltd., Nanjing 210000, China
2. State Grid Jiangsu Electric Power Co., Ltd., Nanjing 210000, China

(Received 9 October 2022; Revised 26 November 2022)

With the expansion of smart grid and Internet of things (IoT) technology, edge computing has a wide variety of applications in these domains. The criteria for real-time monitoring and accuracy are particularly high in the field of online real-time monitoring of electricity lines. Based on edge technology, high-quality real-time monitoring can be performed for transmission lines using image processing techniques. Therefore, we propose an image denoising method, which can learn clean images using a stream-based generative model. The stream model uses a two-stage approach in the network to handle the different training periods of denoising separately. Experimental results show that the proposed method has good denoising performance.
Document code: A **Article ID:** 1673-1905(2023)04-0248-4
DOI https://doi.org/10.1007/s11801-023-2168-0

To meet the challenges posed by massive amounts of data and computation, edge computing technologies are being applied to power systems. Edge computing[1-3] is particularly suitable for application scenarios in power systems, such as smart terminal devices that monitor grid operation in real-time and complete data analysis, then process through local edge computing. The data generated by devices such as node sensors are processed first and then the processed information is fed back to the central network. This greatly reduces the amount of data and the burden on network bandwidth[4,5]. In the transmission line monitoring system, the video image analysis capability is integrated into the edge devices through a gateway with edge computing capability, and only the processing results are fed back to the cloud, effectively reducing the load on the cloud[6,7]. The edge computing nodes can independently identify the safety hazards used in transmission lines and the surrounding environment, as well as localize the images for analysis and processing. However, real-time surveillance images are inevitably affected by noise as well as bad weather. In order not to affect the subsequent image processing tasks, the transmission line monitoring image denoising is a key operation. Edge node devices are basically implemented with dedicated graphics processing unit (GPU) chips, and the data to be denoised is sent to the edge node processor through the network to improve processing efficiency.

The continuous development of deep learning techniques gives excellent ideas for the computer vision domain field of image processing for power grids[8], infrared detection[9], and super-resolution techniques[10] to provide excellent ideas for their applications. Deep learning methods[11] have been proposed for image denoising and significant progress has been made. Although deep convolutional neural network (DCNN)[12] has achieved great results in image denoising, such methods lack flexibility in adapting to different image recovery tasks, because the data likelihood term is not explicitly exploited[13]. Recently proposed deep learning methods, such as Noise2Noise[14] and Noise2Void[15], exploit the statistical properties of noisy image patches to denoise them. The unsupervised image denoising method bridges the technical gap between traditional and supervised image denoising methods. A convolutional neural network's structure alone is able to be utilized as just a prior for natural pictures in the robust noise-suppression approach known as deep image prior (DIP)[16]. An unsupervised three-dimensional (3D) positron emission tomography (PET) image denoising method, called magnetic resonance-guided depth decoder (MR-GDD)[17], can effectively prevent guided image features by integrating anatomical information into the DIP structure through an attention mechanism of leakage. In this paper, we use a class of stream-based generative models[18] for denoising, which can be successfully used to generate realistic images by learning reversible transformations from complex distributions of

* This work has been supported by the Science and Technology Project of State Grid Jiangsu Electric Power Co., Ltd: Research on Early Warning Technology of Overhead Transmission Channel External Invasion Risk Based on Layered Calculation (No.J2021064).
** E-mail: yannanyn@aliyun.com

images to simple distributions like Gaussian ones. Unlike generative adversarial network (GAN)[19], the probability function of clear pictures may be clearly and precisely captured by stream-based models. Additionally, the image edge detection computation drastically reduces the amount of data and eliminates information that can be considered irrelevant, preserving the important structural attributes of the image. Our stream model is inspired by these and uses a two-stage approach in the network to handle the different training periods of denoising separately. In summary, we propose a method for image denoising in this context. First, a stream-based generative model is used to learn a prior from clean images. Then, we use it to train a denoising network without any clean target.

The degradation caused by noise can usually be described by the equation of $\boldsymbol{Y}$=$\boldsymbol{X}$+$\boldsymbol{N}$, where $\boldsymbol{X}$ is a clean image, $\boldsymbol{N}$ is noise, and $\boldsymbol{Y}$ is the noisy version of $\boldsymbol{X}$. Models with a stream of data study the bijective transformation using a high-dimensional method, complex random variable $\boldsymbol{X}$ to a latent arbitrary variable $\boldsymbol{Z}$. Typically, the image within the data would be symbolized as $\boldsymbol{X}$, while $\boldsymbol{Z}$ is taken to be a typical random vector as

$$\boldsymbol{Z} \approx \boldsymbol{N}(0, \boldsymbol{I}), \tag{1}$$

$$\boldsymbol{X}=h(\boldsymbol{Z}). \tag{2}$$

The unbiased estimate of the negative log likelihood of minimizing $\boldsymbol{X}$ is given below to learn the transformation h as

$$\frac{1}{n}\sum_{i=1}^{N} -\log P(x_i), \tag{3}$$

where x_i is the sample in the data set. In accordance with the fundamental principles of random variable transformation, $\log P(\boldsymbol{X})$ could be written as

$$\log P(\boldsymbol{X}) = \log P(\boldsymbol{Z}) - \log\left|\frac{\mathrm{d}h}{\mathrm{d}x}\right|. \tag{4}$$

If h combines numerous more functions that are typical of deep neural networks, this term can be further dissected.

$$\boldsymbol{X} = \boldsymbol{Z}_0 \xrightarrow{h_1} \boldsymbol{Z}_1 \xrightarrow{h_2} \boldsymbol{Z}_2 \cdots \xrightarrow{h_n} \boldsymbol{Z}_n = \boldsymbol{Z}, \tag{5}$$

$$\log P(\boldsymbol{X}) = \log P(\boldsymbol{Z}) - \sum_{i=1}^{n}\log\left|\frac{\mathrm{d}\boldsymbol{Z}_i}{\mathrm{d}\boldsymbol{Z}_{i-1}}\right|. \tag{6}$$

The flow-based model limits the types of transformations to those where the Jacobi matrix is a triangular (or even diagonal) matrix in order to make the calculation on the right-hand side of Eq.(6) reasonable. A straightforward instance is the coupling layer added below

$$\boldsymbol{y}_{p1} = \boldsymbol{x}_{p1}, \tag{7}$$

$$\boldsymbol{y}_{p2} = \boldsymbol{x}_{p2} + m(\boldsymbol{x}_{pi}), \tag{8}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are the input and output of the layer, respectively, features along the channel dimension are separated into $p1$ and $p2$, and m is an arbitrary transformation. It is simple to determine that the Jacobi matrix for this layer is

$$\boldsymbol{J} = \begin{bmatrix} \boldsymbol{I}_{p1} & 0 \\ \dfrac{\mathrm{d}m(\boldsymbol{x}_{p1})}{\mathrm{d}(\boldsymbol{x}_{p1})} & \boldsymbol{I}_{p2} \end{bmatrix}, \tag{9}$$

where $\boldsymbol{I}_{p1}$ and $\boldsymbol{I}_{p2}$ are unit matrices which are the same size as the partitions $p1$ and $p2$. The determinants and permanents of the matrix in Eq.(9) are just 1, so it is very suitable for stream-based models. Unlike non-linear independent components estimation (NICE)[18] and density estimation using real nvp[19], we do not need invertible transformations because when we learn only the prior, no sampling is needed. However, in our work, we use the hierarchies and formulas of the flow-based models proposed by Glow[20].

To understand how to convert a clean image into a typical multivariate Gaussian random variable, we firstly train a stream-based clean image model. We may evaluate Eq.(6) for any given image and determine the likelihood that the image is clean thanks to the structure of the stream-based model as mentioned in Eq.(2) and the processable probability density of a Gaussian random variable. It is independent of training and can therefore be eliminated. Note that once the first stage of training is completed, h is fixed in the second stage.

$$-\log P(\boldsymbol{X}) = -\log P(\boldsymbol{Z}) + \sum_{i=1}^{n}\log\left|\frac{\mathrm{d}\boldsymbol{Z}_i}{\mathrm{d}\boldsymbol{Z}_{i-1}}\right| = \frac{1}{2}\|\boldsymbol{Z}\|_2^2 + \sum_{i=1}^{n}\log\left|\frac{\mathrm{d}\boldsymbol{Z}_i}{\mathrm{d}\boldsymbol{Z}_{i-1}}\right| + C. \tag{10}$$

When a clean image $\boldsymbol{X}$ is compared to a noisy image $\boldsymbol{Y}$, the posterior distribution of the clean image $\boldsymbol{X}$ is

$$P(\boldsymbol{X} \mid \boldsymbol{Y}) = \frac{P(\boldsymbol{Y} \mid \boldsymbol{X})P(\boldsymbol{X})}{P(\boldsymbol{Y})}. \tag{11}$$

The denominator can be disregarded in order to maximize the numerator or its logarithmic value in order to get a maximum a posteriori (MAP) estimate of a clean image. $\log P(\boldsymbol{Y}|\boldsymbol{X})$ is just the squared error between $\boldsymbol{X}$ and $\boldsymbol{Y}$ when additive Gaussian white noise is assumed.

$$\arg\max_{X} \log P(\boldsymbol{X} \mid \boldsymbol{Y}) = \arg\max_{X} \log P(\boldsymbol{Y} \mid \boldsymbol{X}) + \log P(\boldsymbol{X}). \tag{12}$$

Additionally, we are able to determine the prior log-likelihood of $\boldsymbol{X}$ using the stream model h developed in stage 1. According to Eq.(12), we could obtain the loss function for noise reduction d (noting the typical sign change, we desire to reduce this loss as much as possible) as follows

$$f = (\boldsymbol{Y} - \boldsymbol{X})^2 - \lambda \log P(\boldsymbol{X}), \tag{13}$$

where the hyperparameter λ regulates how significant the prior probability distribution and conditions are in relation to one another. Precisely, λ depends on the amount of noise in the image. We also learned from the studies that the choice of the input has a significant impact on the denoiser's performance. We want to train a single denoiser for a variety of noise levels, which is a difficulty. We changed the first term in Eq.(13) to substitute the squared error between the fuzzy versions of the measured $\boldsymbol{X}$ and $\boldsymbol{Y}$ in order to lessen the dependency of the incoming on the noise level. Intuitively, we train the model to repeat only the low-frequency data from the input $\boldsymbol{Y}$ while including additional details to improve the output's appearance, whose details to add to $\boldsymbol{Y}$ are indicated by the stream model h. The loss function's final representation of the denoiser d is as follows

$$f=(B(\boldsymbol{Y})-B(\boldsymbol{X}))^{2}-\lambda\log P(\boldsymbol{X}), \tag{14}$$

where B is a local mean filter, and 3×3 is selected as its size, since it delivers the best results on the validation set.

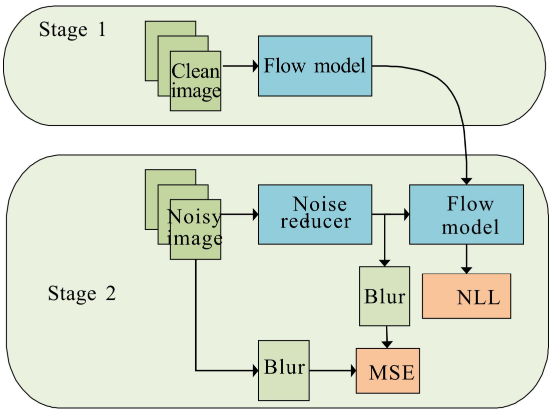

Fig.1 Basic flow of two-stage training

In this paper, our experimental part uses three background datasets with one consisting of the original 2 000 real visible images containing transmission lines, labeled as 2k, to which additional 4 000 pseudo background (raindrops, fog) images generated by PPGAN and 6 000 simulated noises (Gaussian noise, random noise) are added, labeled as 6k and 8k, respectively. We fed this model with patches of size 32 from a clean dataset. Utilizing the losses from Eq.(11), we use the Adam optimizer[21] for 100 stages of training. We use ResNet[22] as a noise reducer.

On the dataset, we tested our approach for various noise levels as well as for image denoising, and the algorithm is an iterative denoising process. The weighted noise image is added to the noisy image for the next denoising process. It is not difficult to conclude that the framework in this paper has significant effectiveness in image denoising operation. We compared the denoising network (denoted as Den network) with that used for image de-rain and de-fog. In this paper, we choose two objective metrics, peak signal-to-noise ratio (*PSNR*) and structural similarity (*SSIM*). *PSNR* is used to measure the difference between two images, and the larger the value, the smaller the difference between the two images. *SSIM* extracts structured information from images, which is more in line with human eye visual perception than the traditional way. The comparison results are shown in Tab.2, and we can see that the proposed method performs much better than the denoising network. The average *PSNR* gains of the image deblurring and SR denoising networks are as high as 0.42 dB and 0.63 dB, respectively, which shows the advantages of the stream-based generation model. It can be seen from Tab.3 that the proposed method outperforms the BM3D method. Compared with other methods, our denoising method obtains higher *PSNR* values.

Fig.2 Schematic diagram of denoising results

Tab.1 Denoising effect of the method in this paper for different types of noises

Datasets	Training datasets				Test datasets			
δ	Gaussian noise		Random noise		Gaussian noise		Random noise	
	PSNR	*SSIM*	*PSNR*	*SSIM*	*PSNR*	*SSIM*	*PSNR*	*SSIM*
15	**32.91**	**0.917**	**32.86**	**0.913**	**32.29**	**0.903**	**32.27**	**0.871**
25	30.54	0.912	30.50	0.906	29.88	0.883	29.86	0.861
50	27.50	0.905	27.53	0.897	27.02	0.875	27.03	0.772

Improving the quality of transmission line monitoring images facilitates the realization of condition monitoring and fault analysis of power equipment. In this paper, a stream-based generative model is proposed for deep learning methods to learn a prior knowledge from clean images. The stream-based model is used as the premise of the image denoising method, and an effective solution algorithm is designed for the proposed model to improve the model denoising ability. The experimental results show that after denoising the qualitative and quantitative experimental datasets in this study, our technique is more competitive. Different prior approaches can be used

Tab.2 Average *PSNR* results for Den networks and stream-based generative models on the dataset

Nuclear	Nuclear1		Nuclear2	
Den	33.04	28.85	32.16	28.20
N2N	32.58	28.96	31.88	28.10
N2V	31.76	27.85	32.31	28.33
DIP	32.24	29.00	32.57	27.18
MR-GDD	33.12	28.36	32.36	28.19
Ours	**33.19**	**29.01**	**32.64**	**28.54**

Tab.3 Test image *PSNR* (dB) after image denoising

Images	1	2	3	4	5
N2N	32.62	35.00	33.29	32.23	33.10
N2V	32.51	35.10	33.31	32.12	33.04
DIP	31.92	35.21	32.79	32.09	33.18
MR-GDD	32.32	35.09	33.17	32.13	33.23
Ours	**32.44**	**35.40**	**33.19**	**32.08**	**33.33**

in denoising networks in the future to enhance the denoising ability of the network.

Statements and Declarations

The authors declare that there are no conflicts of interest related to this article.

References

[1] KRESTINSKAYA O, JAMES A P, CHUA L O. Neuromemristive circuits for edge computing：a review[J]. IEEE transactions on neural networks and learning systems, 2019, 31(1)：4-23.

[2] SHI W, CAO J, ZHANG Q, et al. Edge computing：vision and challenges[J]. IEEE internet of things journal, 2016, 3(5)：637-646.

[3] CHEN S, WEN H, WU J, et al. Internet of things based smart grids supported by intelligent edge computing[J]. IEEE access, 2019, 7：74089-74102.

[4] MEHMOOD M Y, OAD A, ABRAR M, et al. Edge computing for IoT-enabled smart grid[J]. Security and communication networks, 2021.

[5] SAMIE F, BAUER L, HENKEL J. Edge computing for smart grid：an overview on architectures and solutions[J]. IoT for smart grids, 2019：21-42.

[6] HUANG Y, LU Y, WANG F, et al. An edge computing framework for real-time monitoring in smart grid[C]//2018 IEEE International Conference on Industrial Internet (ICII), October 21-23, 2018, Seattle, WA, USA. New York：IEEE, 2018：99-108.

[7] LI H, DONG Y, YIN C, et al. A real-time monitoring and warning system for power grids based on edge computing[J]. Mathematical problems in engineering, 2022, 2022：8719227.

[8] KHODAYAR M, LIU G, WANG J, et al. Deep learning in power systems research：a review[J]. CSEE journal of power and energy systems, 2020, 7(2)：209-220.

[9] LI B, XIAO C, WANG L, et al. Dense nested attention network for infrared small target detection[EB/OL]. (2021-06-01) [2022-09-24]. https：//arxiv.org/abs/2106.00487.

[10] HAN M, LIU H. Retraction note：super-resolution restoration of degraded image based on fuzzy enhancement[J]. Arabian journal of geosciences, 2022, 15(9)：1-1.

[11] ZHANG K, ZUO W, GU S, et al. Learning deep CNN denoiser prior for image restoration[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, July 21-26, 2017, Honolulu, HI, USA. New York：IEEE, 2017：3929-3938.

[12] LIU W, LEE J. A 3-D atrous convolution neural network for hyperspectral image denoising[J]. IEEE transactions on geoscience and remote sensing, 2019, 57(8)：5701-5715.

[13] LIU L, OUYANG W, WANG X, et al. Deep learning for generic object detection：a survey[J]. International journal of computer vision, 2020, 128(2)：261-318.

[14] CALVARONS A F. Improved noise2noise denoising with limited data[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 20-25, 2021, Nashville, TN, USA. New York：IEEE, 202：796-805.

[15] KRULL A, BUCHHOLZ T O, JUG F. Noise2void-learning denoising from single noisy images[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 16-22, 2019, Long Beach, USA. New York：IEEE, 2019：2129-2137.

[16] ULYANOV D, VEDALDI A, LEMPITSKY V. Deep image prior[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York：IEEE, 2018：9446-9454.

[17] ONISHI Y, HASHIMOTO F, OTE K, et al. Anatomical-guided attention enhances unsupervised PET image denoising performance[J]. Medical image analysis, 2021, 74：102226.

[18] AGCAOGLU O, SILVA R F, CALHOUN V. Multimodal fusion of brain imaging data with joint non-linear independent component analysis[C]//2022 IEEE 14th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP), June 26-29, 2022, Nafplio, Greece. New York：IEEE, 2022：1-5.

[19] CAI Z, XIONG Z, XU H, et al. Generative adversarial networks：a survey toward private and secure applications[J]. ACM computing surveys, 2021, 54(6)：1-38.

[20] KINGMA D P, DHARIWAL P. Glow：generative flow with invertible 1×1 convolutions[EB/OL]. (2018-07-09) [2022-09-24]. https：//arxiv.org/abs/1807.03039.

[21] KINGMA D P, BA J. Adam：a method for stochastic optimization[J]. Computer Science, 2014.

[22] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, June 27-30, 2016, Las Vegas, NV, USA. New York：IEEE, 2016：770-778.