Dense in Dense Network with Attention Module for Blind Image Denoising

Jing-Ming Guo, *Senior Member, IEEE*, and Della Fitrayani Budiono

***Abstract*—Denoising is essential in the field of image restoration, and further, computer vision-related tasks. Most CNN-based denoisers were pair-based, meaning a pair of clean and noisy images are needed to train the networks. However, clean images may not always be available in real-world applications. Recently, it has shown that the training network without clean images is possible, such as Noise2Void (N2V) and self-supervised Gaussian denoising with blind-spot network. These training methods that do not require clean images representations are called blind denoising. The goal of this study is to further improve the denoising quality; thus, a novel method to perform blind denoising using Dense in Dense Network with Attention module, abbreviated as DiDNA, is proposed. The experimental results show that our proposed method outperforms the previous blind denoising works, and achieves the same capabilities to paired denoisers, including CNN-based and non-CNN-based approaches.**

***Index Terms*—Image denoising, blind denoising, dense connectivity, attention module.**

I. Introduction

Denoising is one of the essential problems in terms of image restoration tasks. The pre-processing of denoising is necessary for images, otherwise it may become problematic for further steps. There has been a lot of attempts to perform denoising tasks. There were many approaches proposed for image noise reduction using conventional mathematical models, such as Random Markov Fields [1]-[3], collaborative filtering approaches [4][5], Anscombe transformation for Poisson noise [6], internal statistics methods [7][8], Bayesian minimum mean square error [9], nuclear norm minimization [10], and image diffusion [11]. Recently, the advanced approaches in image denoising have focused on convolutional neural networks (CNN).

Since CNN has been widely adopted in many tasks of image processing, the denoising approach using CNN is also considered. Moreover, the CNN model has shown its impressive performance on other tasks in the field of image restoration, such as image super-resolution and image deblurring, which made CNN-based denoisers become a promising solution. The first CNN-based denoising method was proposed by Jain *et al*. [12], in which they presented an approach that combines two main ideas: The use of convolutional networks as an image processing architecture and an unsupervised learning procedure that synthesizes training samples from a specific noise model. Their results were comparable and, for some of the cases, superior to the state-of-the-art methods using Wavelet and Markov Random Field. Since then, CNN-based denoising methods have been further explored. With the advancement of CNN frameworks in classification, more complicated CNN-based denoisers were well-developed as well. The results show that the CNN-based denoisers outperform the state-of-the-art conventional approaches in terms of the image quality.

Some of the CNN-based denoisers were proposed to solve image restoration problems in general, including image denoising. One of the famous methods is Deep Image Prior (DIP) [13], in which the structure of a generator network was sufficient to capture low-level image statistics prior to any learning. This method was applied in several image restoration tasks such as image denoising, super-resolution, and inpainting. One of the key points in DIP was its ability to bridge the gap between two popular image restoration methods: Learning-based methods using deep convolutional networks and learning-free methods based on handcrafted image priors such as self-similarity.

On the other hand, some of the CNN-based methods tried to solve the denoising problem based on the known synthetic noise models, such as additive Gaussian, Poisson, and Bernoulli noise. One of the popular methods from this aspect is FFDNet [14]. This method had several key properties, including the ability to handle a wide range of noise levels, the ability to remove spatially variant noise by specifying a non-uniform noise level-map, and faster speed in execution than the benchmark of traditional image denoiser, i.e., BM3D [5].

Meanwhile, some studies focused on how to tackle real-world problems when the noise distribution is completely unknown, making the tasks even more challenging in the denoising domain. Two of the latest methods proposed to solve this problem were CBDNet [15] and RIDNet [16], in which they came up with different point of views. The CBDNet suggested that training a convolutional blind denoising network with more realistic noise models and real-world noisy-clean image pairs to enhance the generalization ability of CNN-based denoisers. Signal-dependent noise and in-camera signal processing pipeline to produce synthesized realistic noisy images were considered in CBDNet. On the other hand, RIDNet

Jing-Ming Guo and Della Fitrayani Budiono are with the Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan (e-mail: jmguo@seed.net.tw, m10707811@mail.ntust.edu.tw).

proposed a single-stage blind real image denoising network by employing a modular architecture. A residual on the residual structure was adopted in the model to ease the flow of low-frequency information, and the feature attention was further applied to explain the channel dependencies. Both the results from CBDNet and RIDNet showed impressive performance on several real noisy datasets.

Even though the most state-of-the-art CNN-based denoiser can achieve excellent results, most of them still rely on the availability of the clean target images. Since CNN is basically an input-target structure, it makes sense that the most CNN-based denoisers require the noisy images as the inputs and the corresponding clean images as the targets. However, this is one of the biggest problems in image denoising tasks since the clean images are most likely unavailable in real situations. In this study, we address such issues to tackle the real-world problems in image denoising.

There have been many attempts to solve this denoising problem without clean target references, as known as blind denoising, i.e., N2N [17], N2V [18], and self-supervised Bayesian denoising with the blind-spot network [19]. The N2N was the first one to address this issue, as it is proposed that the reconstructed clean image can be obtained simply using a pair of noisy images, instead of having clean target references in training. These pairs of noisy images would work as the input-target replacements for the CNN models. It turned out that N2N's backbone network based on RED30 [20] can still learn the real clean representations of the image by utilizing its loss functions. That is, the L2 loss between a pair of noisy images can be used to retain the mean of the image, or the L1 loss between a pair of noisy images can be used to retain the median of the image. It can achieve satisfactory results indeed, as it was comparable against Noise2Clean (N2C). However, a pair of noisy images is still necessary for N2N, and it is very likely that only one noisy sample is available for an image in the dataset in reality. To deal with this problem, the N2V was proposed to achieve the reconstruction of a clean image only with one noisy sample. In the training phase, they 'blind' some of the pixels in the noisy images during the process, called blind-spot masking operation. It can produce masked noisy images that were used as the inputs to the CNN models, meanwhile, the original noisy images became the targets for reconstruction. Using such unique input-target configuration, N2V's backbone network, which was based on CSBDeep's [21] and U-Net [22], showed satisfactory results. Although its performance did not surpass the state-of-the-art denoisers, the results have proven that it is possible to perform image denoising with only one sample for an image. Inspired by N2N and N2V, the self-supervised Bayesian denoising method with the blind-spot network was proposed for image denoising without accessing to clean reference images. This method consists of 2 stages. The first stage is to produce the 'prior' results using the blind-spot network, limiting the network's perceptive field. The second stage is to adopt Bayesian inference as post-processing, leading to 'posterior' results in a Bayesian sense.

Inspired by the self-supervised Bayesian denoising with the blind-spot network, we realized the importance of constructing a robust blind-spot network. This network would produce 'prior' images, which determines the quality of the 'posterior' results. Thus, it is crucial to establish solid prior results first. The main purpose of this study is to build a better blind-spot network that yields reliable prior representations of the images.

In this paper, a novel CNN-based blind denoising method is proposed using a basic idea of dense connectivity, namely Dense in Dense Network with Attention module (DiDNA). Leveraged by the dense connectivity, it allows the network to be trained using deeper layers. Consequently, it enables the network to learn more features for blind denoising. In addition, the attention mechanism is embedded into the proposed network to highlight some of the most informative feature maps.

The paper is organized as follows. First, an introduction to blind denoising as well as its development is explained in Section II. The proposed dense in dense (DiD) network with attention module is described in Section III. Then, the thorough experimental results and discussion are in Section IV, where the proposed network is applied for denoising across three different types of noises: Additive white Gaussian noise (AWGN), Poisson noise, and Impulse noise. Lastly, Section V concludes the paper and narrates the future work of this study.

II. Blind Denoising

Blind denoising in this study means image denoising without knowing the noise distribution. From another aspect, blind denoising can also be interpreted as image denoising without true clean image representations as the target image. Either way, the task of blind denoising is definitely challenging for each scenario. As for this study, the main focus of blind denoising is to achieve clean image reconstruction without a clean target image provided.

A. Noise2Noise (N2N)

To train the N2N network, a pair of noisy images is required as the input-target images. Thus, the risk minimization task can be written as

$$\underset{\theta}{\operatorname{argmin}} \sum_i L(f_\theta(a_i), b_i), \tag{1}$$

where the input a_i and the target b_i were drawn from the same corrupted distribution. The f_θ is the reconstruction network to train, making the input image close to the target image. The network is able to recover the clean version of the images because of the loss function, for example, the *L2* loss can recover the mean of the images, or *L1* loss can recover the median of the image.

B. Noise2Void (N2V)

For N2V, a pair of noisy images becomes unnecessary; instead, only one noisy sample for an image is sufficient to perform the training. A masking operation was performed to facilitate the training process, and it is called blind-spot masking operation. First, select N random pixels which are hidden from the network. Next, choose random pixels which are in the range of selected hidden pixels' radius, and replace the selected hidden pixel values with their corresponding neighbors' values. Then, two kinds of images are considered: The original noisy images and the masked noisy images. The masked noisy images are the inputs fed into the network, while the original noisy images become the target. Finally, the gradients are computed only for the selected pixels, ignoring the rest of the predicted pixels. This modification can be done by the standard *Keras* pipeline with the self-defined mean squared error (MSE) loss function that sets zero for all but the selected pixels.

This means that the risk minimization task is the same as N2N, and the only difference is that the input a_i and target $\hat{a}_i$ are actually from the same noisy sample for an image but with modifications of inputs made by the blind-spot masking operation. This can be formulated as

$$\underset{\theta}{\operatorname{argmin}} \sum_i L(f_\theta(a_i), \hat{a}_i). \qquad (2)$$

C. Self-supervised Bayesian denoising with blind-spot network

This approach for blind denoising is divided into two parts. The first part is to construct a convolutional blind-spot network, which would limit its receptive field, as it is suggested by N2V's approach, to produce the 'prior' results. Subsequently, the second part is to compute the final 'posterior' results using Bayesian inference. Details of each part are described as follows.

First, inspired by the idea of N2V that limits the network's receptive field, the convolutional blind-spot networks are designed by combining multiple branches where each of them has their receptive field restricted to a half-plane that does not contain the center pixel. Subseqeuntly, combining the four branches with a series of 1×1 convolution to obtain a receptive field that could extend in every direction without considering the center pixel. Compared to N2V, the advantage of this architecture is that all output pixels can contribute to the loss function when training. To create a network with a restricted receptive field, some modifications are made to let the receptive field fully containing within one half-plane, including the center row/column. Since the receptive field of the network includes the center pixel, an offset by one pixel is needed between feature maps before combinations. Layers that do not extend the receptive field, i.e., concatenation, summation, 1×1 convolution, etc., can be used without modifications.

Considering the prediction of the clean value x for a noisy pixel y, it should be noted that the clean value x is not simply influenced by y but also its context of neighboring (noisy) pixels, i.e., Ω_y. A standard supervised regression model trained with corrupted-clean pairs, and loss function *L2* loss returns an estimation of $\mathbb{E}_x[p(x|y, \Omega_y)]$, which is the mean over all possible clean pixel values given the noisy pixel and its context. With the assumption that noise is independent between pixels and independent of the context, N2V trains its regressor to estimate $\mathbb{E}_x[p(x|\Omega_y)]$, i.e., the mean of all potential clean values that are consistent with the context.

If extra information is brought in the form of an explicit model of the corruption, this should be able to connect the observed marginal distribution of the noisy training data to the unobserved distribution of clean data:

$$\underbrace{p(y|\Omega_y)}_{\text{Training data}} = \int \underbrace{p(y|x)}_{\text{Noise model}} \underbrace{p(x|\Omega_y)}_{\text{Unobserved}} \, \mathrm{d}x \qquad (3)$$

It indicates that the known noise model can help to predict a parametric model for distribution $p(x|\Omega_y)$ by only observing the corrupted training data. In this case, it is modeled as a multivariate Gaussian $\mathcal{N}(\mu_x, \Sigma_x)$ over the color components. Subsequently, the marginal likelihood $p(y|\Omega_y)$ can be computed in a closed form, allowing the neural network to map the context Ω_y to the mean μ_x and covariance Σ_x by maximizing the likelihood of the data under Eq. 3.

With the approximate distribution $p(x|\Omega_y)$, Bayesian reasoning can now be applied to include information from y for inference. The posterior probability of the clean value x given observations of both the noisy pixel y and its context is given by

$$\underbrace{p(x|y, \Omega_y)}_{\text{Posterior}} \propto \underbrace{p(y|x)}_{\text{Noise model}} \underbrace{p(x|\Omega_y)}_{\text{Prior}}. \qquad (4)$$

When the posterior is obtained, standard Bayesian inference tools can be used. Maximum a posteriori (MAP) estimate would pick x that maximizes the posterior; and in this case, posterior mean $\mathbb{E}_x[p(x|y, \Omega_y)]$ is used to obtain the final denoising results as it minimizes MSE, giving rise to a higher PSNR.

Thus, this approach consists of:
1) Train the neural network to map the context Ω_y to the mean μ_x and variance Σ_x of a Gaussian approximation to the prior $p(x|\Omega_y)$.
2) During inference, first, feed context Ω_y to the neural network to obtain μ_x and Σ_x; then compute posterior mean $\mathbb{E}_x[p(x|y, \Omega_y)]$ by the closed-form analytic integration.

III. Network Architecture

The network architecture plays an important role, as it can determine the 'prior' image, which is very crucial. Since the goal is to restrict the receptive field of the restoration network, several points should be taken into consideration in the design of neural networks.

Convolution layers: To limit the receptive field of a zero-padding convolution layer to extend, in this example, upwards, the viable solution is to offset the feature-maps downwards when performing the convolution operation. For an $h \times w$ kernel size, a downwards offset of $k = [h/2]$ pixels are equal to using a kernel that is shifted upwards so that all weights below the center row are zero. For implementation, k rows of zeros are appended to the input before applying the convolution, and then k bottom rows are cropped out from the output. However, there are some exceptions; for example, the convolution layers in the attention module would normally perform without any modifications. The reason why the convolution layers in attention module are not modified is to maintain the same receptive field while adjusting the ‘weight’ for each feature map.

Residual skip connection: Residual skip connection was first introduced by He *et al*. [23], and it eases up training a network with very deep layers. Since all the convolution layers are being restricted, residual skip connection based on the identity mapping cannot be used inside the restoration network since it would definitely interfere with the feature maps when the element-wise summation is being performed.

Dense connectivity: Following the success of residual skip connection, Huang *et al.* [24] proposed dense connectivity, which further enhances network performance when training a network with very deep layers. Due to the drawback of skip connection mentioned above, dense connectivity can be the substitute to improve the information flow between layers using direct connections from any layer to all its subsequent layers. Consequently, the l^{th} layer receives the feature-maps of all preceding layers, $x_0, \cdots, x_{l-1}$, as the input:

$$x_l = H_l([x_0, \cdots, x_{l-1}]), \tag{5}$$

where $[x_0, \cdots, x_{l-1}]$ is corresponding to the concatenation of the feature maps produced in layers $0, \dots, l-1$.

The overall main architecture of the proposed network is shown in Fig 1. It consists of Dense in Dense (DiD) network with attention module, including the follow-up steps like inverse rotation before the concatenation, followed by 1×1 convolution layer to combine each feature map from different directions. Moreover, the proposed DiD network with attention module comprises Dense Group (DG) and Dense Attention Module (DAM), as illustrated in Fig.2.

A. Dense in Dense (DiD)

Our proposed DiD structure was inspired by the design of DenseNet. Since residual skip connection may not be suited here because the convolution layer has been modified, therefore, doing element-wise summation would only corrupt the entire feature maps. Thus, the dense connection is used to replace the residual skip connection to ease the information flow between layers. This DiD structure comprises J Dense Groups (DG), equipped with Long Dense Connection (LDC). Each DG further comprises K DAM alongside Short Dense Connection (SDC). The DG can be used to enhance the network capability when more layers are stacked. The RG in the j-th group can be expressed as

$$I_j = F_j(I_{j-1}) = F_j\left(F_{j-1}(\cdots F_1(I_0))\right), \tag{6}$$

where F_j is the function of j-th DG, I_{j-1} and I_j are the input and output of j-th DG, respectively. As mentioned before, several K DAM are in each DG to further exploit the dense connectivity. Thus, the k-th DAM in j-th DG can be expressed as:

$$I_{j-1,k} = F_{j,k}(I_{j-1,k-1}) = F_{j,k}\left(F_{j,k-1}(\cdots F_{j,1}(I_{j-1})\,)\right) \tag{7}$$

To further alleviate the missing information problem from the previous $j-1$-th DG layer, the Short Dense Connection (SDC) is used to bypass the information, which can be formulated as

$$\begin{gathered} I_j = [I_{j-1}, W_j I_{j-1,K}] = \\ [I_{j-1}, W_j F_{j,K}\left(F_{j,K-1}\left(\cdots F_{j,1}(I_{j-1})\right)\right)], \end{gathered} \tag{8}$$

where W_j is the weight set of convolution layers at the end of j-th DG. Finally, to stabilize the network since stacking several DGs can degrade the network’s performance, Long Dense Connection (LDC) is used. This can help the flow of information, as the lower frequency information is being passed directly, making it possible to train the earlier layers. The final output can be formulated as

$$I_{FO} = [I_0, W_{LDC} I_J] = [I_0, W_{LDC} F_J\left(F_{J-1}(\cdots F_1(I_0))\right)], \tag{9}$$

where W_{LDC} is the weight set of convolution layers at the end of DiD.

B. Attention Module

The attention mechanism was firstly applied to the image field by Xu *et al.* [25], specifically in image captioning. This idea was further developed by Chen *et al.* [26] into spatial and channel attention for image captioning. Since the attention mechanism had brought significant improvement for the network performance, therefore, attention module is also exploited inside the proposed network to further enhance the network capability. Inspired by the work from Hu *et al.* [27], in this research, the attention mechanism is applied to highlight the most important feature maps or channels. The attention module works because each channel in the feature maps should have different scores according to its informativeness. As a result, in this case, the attention module guides the network to pay more attention to certain informative channels. This can certainly bring benefits to our denoising tasks. High-frequency parts are often corrupted due to the noise; therefore, using this channel attention can definitely help to retain the true informative high-frequency parts in the image. To apply the attention module, the priority is to capture the global information. This information can be captured using global average pooling, which is formulated as

$$s_c = H_{GP}(a_c) = \frac{1}{H \times W}\sum_{x=1}^{H}\sum_{y=1}^{W} a_c(x, y), \tag{10}$$

where H and W are the height and width of the feature maps respectively, and $a_c(x, y)$ stands for the value at the position (x, y) of c-th feature a_c. The H_{GP} corresponds to the global average pooling.

Subsequently, the next step is to obtain the channel-dependent features. As it was mentioned in [27], to create channel-dependent features, there are two requirements: First, it should be easy to adapt, meaning it should be able to learn a nonlinear correlation between channels. Second, because multiple channels are emphasized rather than one-hot-activation, it must be able to learn a non-mutually-exclusive relation. To meet these requirements, a simple gating operation using sigmoid activation is adopted as

$$z = \delta\left(F_U \sigma\left(F_D(s_c)\right)\right), \tag{11}$$

where δ represents the sigmoid activation function, and σ stands for the ReLU activation function. The F_D and F_U are the weight sets of convolution layers for channel-downscaling and channel-upscaling layers. After obtaining the final channel statistics, the final output of the attention module can be computed by

$$\hat{a}_c = z_c \cdot a_c, \tag{12}$$

where the z_c is the scaling factor, and a_c is the c-th channel feature map. Owing to the enhanced feature maps combining with the dense connectivity in DAM, it would be beneficial to the network since the feature maps have been tuned well through the optimization.

IV. Experimental Results

The experiments are conducted under three types of artificial noise, i.e., additive Gaussian noise, Poisson noise, and Impulse noise. For each type of noise, the posterior processing of the images follows the same procedures in [19]. All experiments were run on one NVIDIA Quadro RTX 8000, and it took about 4 to 6 days to perform the training and testing for an experiment on a specific dataset. As for the datasets, around 50,000 images in the ILSVRC2012 (ImageNet) validation set are used as the training dataset, and KODAK, BSD300 validation set, and SET14 datasets are used as the testing dataset. All experiments were performed on the same training and testing datasets for fair comparisons.

A. Denoising on Additive Gaussian Noise

The parameters of a multivariate Gaussian $\mathcal{N}(\mu_x, \Sigma_x) = p(x|\Omega_y)$ stand for the distribution of the signal, and were derived from our restoration network. The covariance matrix can be decomposed to $\Sigma_x = \mathrm{A}_x^{\mathrm{T}}\mathrm{A}_x$, and A_x is an upper triangular matrix so that Σ_x is a valid covariance matrix which complies with symmetric and positive semidefinite properties.

To model the corruption process with the zero-mean Gaussian noise, since a convolution of two mutually independent Gaussians was performed as it is shown in Equation 3, the covariance after the process is simply the sum of the constituents. That is,

$$\mu_y = \mu_x \; and \; \Sigma_y = \Sigma_x + \sigma^2 I, \tag{13}$$

where σ is the standard deviation of the Gaussian noise, and I is an identity matrix.

Subsequently, to fit $\mathcal{N}(\mu_y, \Sigma_y)$ to the observed noisy training data, it can be achieved by minimizing the corresponding negative log-likelihood loss in the training process, which can be written as

$$\begin{aligned} loss(y, \mu_y, \Sigma_y) &= -\log f(y; \mu_y, \Sigma_y) \\ &= \frac{1}{2}\left[(y-\mu_y)^{\mathrm{T}} \Sigma_y^{-1} (y-\mu_y)\right] \\ &+ \frac{1}{2}\log\left|\Sigma_y\right| + C \end{aligned} \tag{14}$$

where C corresponds to an additive constant term, and $f(y; \mu_y, \Sigma_y)$ is the probability density of a multivariate Gaussian distribution $\mathcal{N}(\mu_x, \Sigma_x)$ at pixel value y.

The results and the comparisons of denoising on additive Gaussian noise can be seen in Tables I and II. The results are compared with other methods, including N2C (Noise2Clean), N2N (Noise2Noise), the baseline (Bayesian denoising with the blind-spot network), and CBM3D (non-CNN based method). The performance results of N2C, N2N, and the baseline model followed the records in [19]. The proposed results were obtained after training for 2,000,000 iterations.

Table I
Image Quality Results (PSNR) for Gaussian Noise ($\sigma = 25$)

Method	**Kodak**	**BSD300**	**Set14**	**Average**
N2C [17]	32.46	31.08	31.26	31.60
N2N [17]	32.45	31.07	31.23	31.58
baseline[19]	32.45	31.03	31.25	31.57
CBM3D [5]	31.82	30.40	30.68	30.96
DiDNA (proposed)	**32.52**	**31.45**	**31.35**	**31.77**

Table II
Image Quality Results (PSNR) for Gaussian Noise ($\sigma \in [5, 50]$)

Method	**Kodak**	**BSD300**	**Set14**	**Average**
N2C [17]	32.57	31.29	31.27	31.71
N2N [17]	32.57	31.29	31.26	31.70
baseline[19]	32.47	31.19	31.21	31.62
CBM3D [5]	31.99	30.67	30.78	31.15
DiDNA (proposed)	**32.58**	**31.84**	**31.41**	**31.94**

B. Denoising on Poisson noise.

Poisson noise should be taken into account since it can be used to model the photon noise in imaging sensors. For noise implementation, $x_i \in [0,1]$ is the clean color component for all color channels i, and λ is the maximum event count, the noise can be simulated as

$$y_i = \frac{\mathrm{Poisson}(\lambda x_i)}{\lambda} \tag{15}$$

For the corruption model, signal-dependent Gaussian noise is adopted to approximate the Poisson noise for denoising. The standard deviation is defined as

$$\sigma_i = \sqrt{\frac{x_i}{\lambda}} \tag{16}$$

and the corruption model can be written as

$$\mu_y = \mu_x \text{ and } \Sigma_y = \Sigma_x + \lambda^{-1}\text{diag}(\mu_x) \tag{17}$$

The results and comparisons of denoising on Poisson noise are listed in Tables III and IV. Similar to the previous setting for additive Gaussian noise, the proposed results are compared with several methods, including N2C (Noise2Clean), N2N (Noise2Noise), the baseline (Bayesian denoising with the blind-spot network), and Anscombe (non-CNN based method), and N2C, N2N, and the baseline results followed the records in [19]. The proposed results are obtained after training for 4,000,000 iterations.

TABLE III
IMAGE QUALITY RESULTS (PSNR) FOR POISSON NOISE ($\lambda = 30$)

Method	Kodak	BSD300	Set14	Average
N2C [17]	**31.81**	30.40	**30.45**	30.89
N2N [17]	31.80	30.39	30.44	30.88
baseline [19]	31.65	30.25	30.29	30.73
Anscombe [6]	29.15	27.56	28.36	28.62
DiDNA (proposed)	31.69	**30.67**	30.39	**30.92**

TABLE IV
IMAGE QUALITY RESULTS (PSNR) FOR POISSON NOISE ($\lambda \in [5, 50]$)

Method	Kodak	BSD300	Set14	Average
N2C [17]	**31.33**	**29.91**	**29.96**	**30.40**
N2N [17]	31.32	29.90	**29.96**	30.39
baseline [19]	31.16	29.75	29.82	30.24
DiDNA (proposed)	31.01	29.88	29.66	30.20

C. Denoising on Impulse Noise.

For the implementation of Impulse noise, each pixel was assigned to a uniformly sampled random color in $[0,1]^3$ with probability α. To approximate $p(y|\Omega_y)$ with a Gaussian and match its first and second raw moments to the data during the training process, the resulting mean and covariance are as follows.

$$\begin{aligned} \mu_y &= \frac{\alpha}{2}\begin{bmatrix}1\\1\\1\end{bmatrix} + (1-\alpha)\mu_x \\ \text{and } \Sigma_y &= \frac{\alpha}{12}\begin{bmatrix}4 & 3 & 3\\3 & 4 & 3\\3 & 3 & 4\end{bmatrix} + (1-\alpha)(\Sigma_x + \mu_x\mu_x^{\mathrm{T}}) \\ &\quad - \mu_y\mu_y^{\mathrm{T}} \end{aligned} \tag{18}$$

The results and comparisons of denoising on Poisson noise are listed in Tables V and VI. Similar to the previous types of noises, the proposed results are also compared with several methods, including N2C (Noise2Clean), N2N (Noise2Noise), and the baseline (Bayesian denoising with the blind-spot network), following the records in [19]. The proposed results are obtained after training for 2,000,000 iterations.

TABLE V
IMAGE QUALITY RESULTS (PSNR) FOR IMPULSE NOISE ($\alpha = 0.5$)

Method	Kodak	BSD300	Set14	Average
N2C [17]	33.32	31.20	**31.42**	31.98
N2N [17]	32.88	30.85	30.94	31.56
baseline [19]	32.98	30.78	31.06	31.61
DiDNA (proposed)	**33.70**	**31.97**	**31.77**	**32.48**

TABLE VI
IMAGE QUALITY RESULTS (PSNR) FOR IMPULSE NOISE ($\alpha \in [0,1]$)

Method	Kodak	BSD300	Set14	Average
N2C [17]	31.69	30.27	29.77	30.58
N2N [17]	31.53	30.11	29.51	30.38
baseline [19]	31.36	30.00	29.47	30.28
DiDNA (proposed)	**32.63**	**31.57**	**30.52**	**31.57**

D. Discussion

As it can be observed from the experimental results, it shows that the proposed method yields superior performance against the baseline [19], in particular for additive Gaussian noise and Impulse noise. Surprisingly, for many cases, the proposed framework even outperforms the non-blind denoiser (the approach with clean references), like Noise2Clean (N2C) from [19]. It means that if the blind-spot network could produce the prior images with high quality, it would definitely improve the further posterior results (as the final outputs). It also suggests that the proposed DiD network with attention module could indeed enhance the performance of the blind-spot network. The depth of DiD network can help the model to learn more features, and with attention module, it would be more beneficial since the network can focus on those informative features.

For the comparison in perceptual quality, the reconstruction results after denoising from our proposed methods and the baseline [19] with three different types of noise are shown in Fig. 3. From observations, the results from our proposed framework show better restorations, in particular on the details. For example, from the additive Gaussian noise part in Fig. 3, it can be observed that the restored sea background from our proposed method presents more details of waves, compared with the results from the baseline. Moreover, as it is shown in the Poisson noise part of Fig. 3, the proposed method can retain correct color details, highlighted by a blue square. For the denoising of Impulse noise, the details of alphabets, inside the blue bounding box, are clearer from our reconstruction results than the results from the baseline. The experimental results show our proposed denoiser can work for various noise types, achieving better reconstruction results and higher performance in evaluation.

V. CONCLUSION

In this study, a novel method to solve the blind denoising problem is proposed using Dense in Dense (DiD) Network with Attention module (DiDNA). According to the extensive experiments, this method has proven to enhance the image quality, both in terms of visual results and also standard quality assessment, compared to the previous methods of blind denoising and the non-CNN-based denoising methods as well. The key reason why our proposed method has better performance is because the network complexity has been expanded, enabling the network to learn more features from the images. Because of the dense connectivity inside the network, it prevents the network from collapsing due to the number of layers. In addition, the attention module also plays an important role in choosing more informative features. The proposed network can work for various noise types, i.e., additive Gaussian, Poisson, and Impulse noises.

REFERENCES

[1] S. Roth and M. J. Black, “Fields of experts: A framework for learning image priors,” in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05), vol. 2, pp. 860–867, IEEE, 2005.

[2] A. Barbu, “Training an active random field for real-time image denoising,” IEEE Transactions on Image Processing, vol. 18, no. 11, pp. 2451–2462, 2009.

[3] J. Sun and M. F. Tappen, “Learning non-local range markov random field for image restoration,” in CVPR 2011, pp. 2745–2752, IEEE, 2011.

[4] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, “Image denoising with block-matching and 3d filtering,” in Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning, vol. 6064, p. 606414, International Society for Optics and Photonics, 2006.

[5] K.Dabov, A.Foi, V.Katkovnik, and K.Egiazarian, “Image denoising by sparse 3-d transform-domain collaborative filtering, ”IEEE Transactions on image processing, vol.16, no.8, pp.2080–2095, 2007.

[6] M. Makitalo and A. Foi, “Optimal inversion of the anscombe transformation in low-count poisson image denoising,” IEEE transactions on Image Processing, vol. 20, no. 1, pp. 99–109, 2010.

[7] M. Elad and M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” IEEE Transactions on Image processing, vol. 15, no. 12, pp. 3736–3745, 2006.

[8] A. Buades, B. Coll, and J.-M. Morel, “A non-local algorithm for image denoising,” in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05), vol.2, pp.60–65, IEEE, 2005.

[9] A. Levin and B. Nadler, “Natural image denoising: Optimality and inherent bounds,” in CVPR 2011,pp. 2833–2840, IEEE, 2011.

[10] S. Gu, L. Zhang, W. Zuo, and X. Feng, “Weighted nuclear norm minimization with application to image denoising,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2862–2869, 2014.

[11] P. Qiao, Y. Dou, W. Feng, R. Li, and Y. Chen, “Learning non-local image diffusion for image denoising,” in Proceedings of the 25th ACM international conference on Multimedia, pp. 1847–1855,2017.

[12] V. Jain and S. Seung, “Natural image denoising with convolutional networks,” in Advances in neural information processing systems, pp. 769–776, 2009.

[13] D. Ulyanov, A. Vedaldi, and V. Lempitsky,“Deep image prior,”in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 9446–9454, 2018.

[14] K. Zhang, W. Zuo, and L. Zhang, “Ffdnet: Toward a fast and flexible solution for cnn-based image denoising,”IEEE Transactions on Image Processing, vol. 27, no. 9, pp. 4608–4622, 2018.

[15] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang, “Toward convolutional blind denoising of real photographs,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1712–1722, 2019.

[16] S. Anwar and N. Barnes, “Real image denoising with feature attention,” in Proceedings of the IEEE International Conference on Computer Vision, pp. 3155–3164, 2019.

[17] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila, “Noise2noise: Learning image restoration without clean data,” arXiv preprint arXiv:1803.04189, 2018.

[18] A. Krull, T.-O. Buchholz, and F. Jug, “Noise2void-learning denoising from single noisy images,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2129–2137, 2019.

[19] S. Laine, T. Karras, J. Lehtinen, and T. Aila, “High-quality self-supervised deep image denoising,” in Advances in Neural Information Processing Systems, pp. 6968–6978, 2019.

[20] X. Mao, C. Shen, and Y.-B. Yang, “Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections,” in Advances in neural information processing systems, pp. 2802–2810, 2016.

[21] M. Weigert, U. Schmidt, T. Boothe, A. Müller, A. Dibrov, A. Jain, B. Wilhelm, D. Schmidt, C. Broaddus, S. Culley, et al., “Content-aware image restoration: pushing the limits of fluorescence microscopy,” Nature methods, vol. 15, no. 12, pp. 1090–1097, 2018.

[22] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in International Conference on Medical image computing and computer-assisted intervention, pp. 234–241, Springer, 2015.

[23] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778, 2016.

[24] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional net-works,”in Proceedings of the IEEE conference on computer vision and pattern recognition, pp.4700–4708, 2017.

[25] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, “Show, attend and tell: Neural image caption generation with visual attention,” in International conference on machine learning, pp. 2048–2057, 2015.

[26] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, and T.-S. Chua, “Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning,”in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5659–5667, 2017.

[27] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7132-7141, 2018.

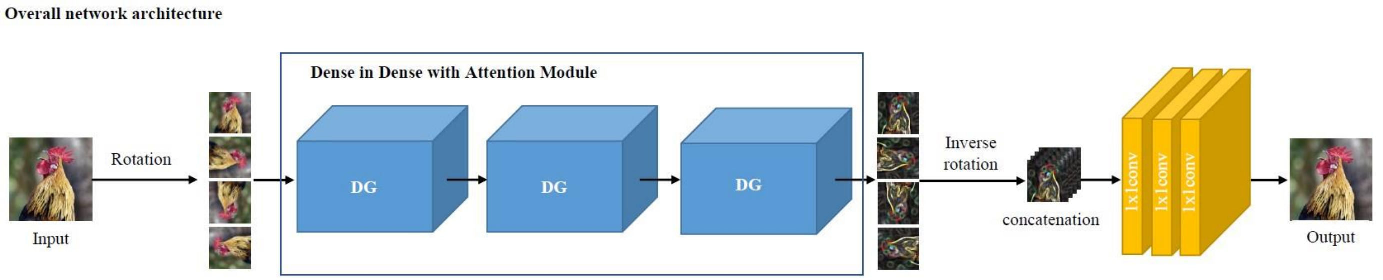

Fig. 1. Overall network architecture. First, the input image was duplicated to four rotating directions (0°,90°,180°,270°) to extend the receptive field. Subsequently, all rotated images were fed into the Dense in Dense (DiD) Network with Attention module (DiDNA). Afterward, all outputs from the network are rotated back to the original orientation and concatenated together along the feature map axis. Finally, a series of 1x1 convolution is applied to combine all the feature maps to derive the outcome of the proposed framework.

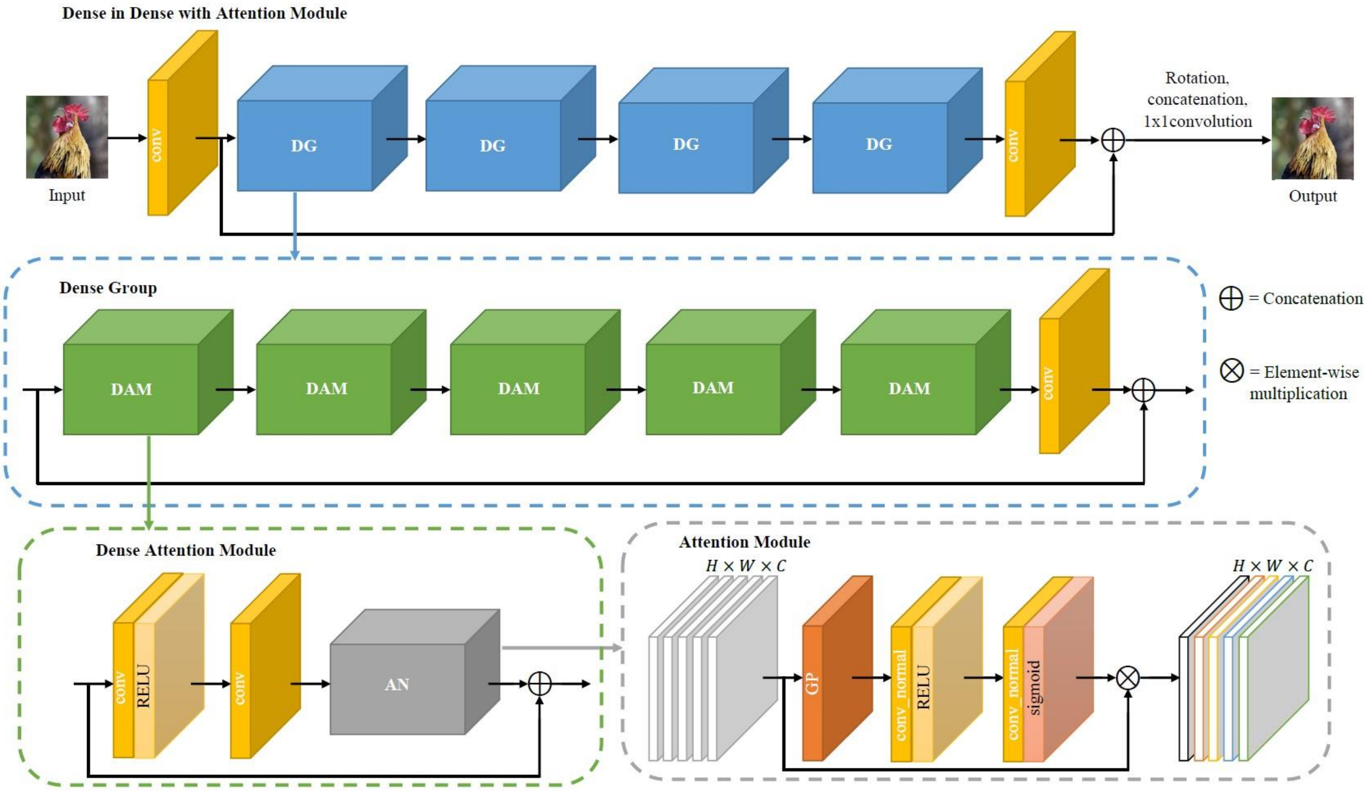

Fig. 2. Details inside the proposed Dense in Dense (DID) network, including Dense Group (DG) and Dense Attention Module (DAM). The module of DG is designed to enhance the network capability when more layers are stacked. In addition, each DG module comprises several DAMs, which are equipped with the Attention Module to highlight the most informative features.

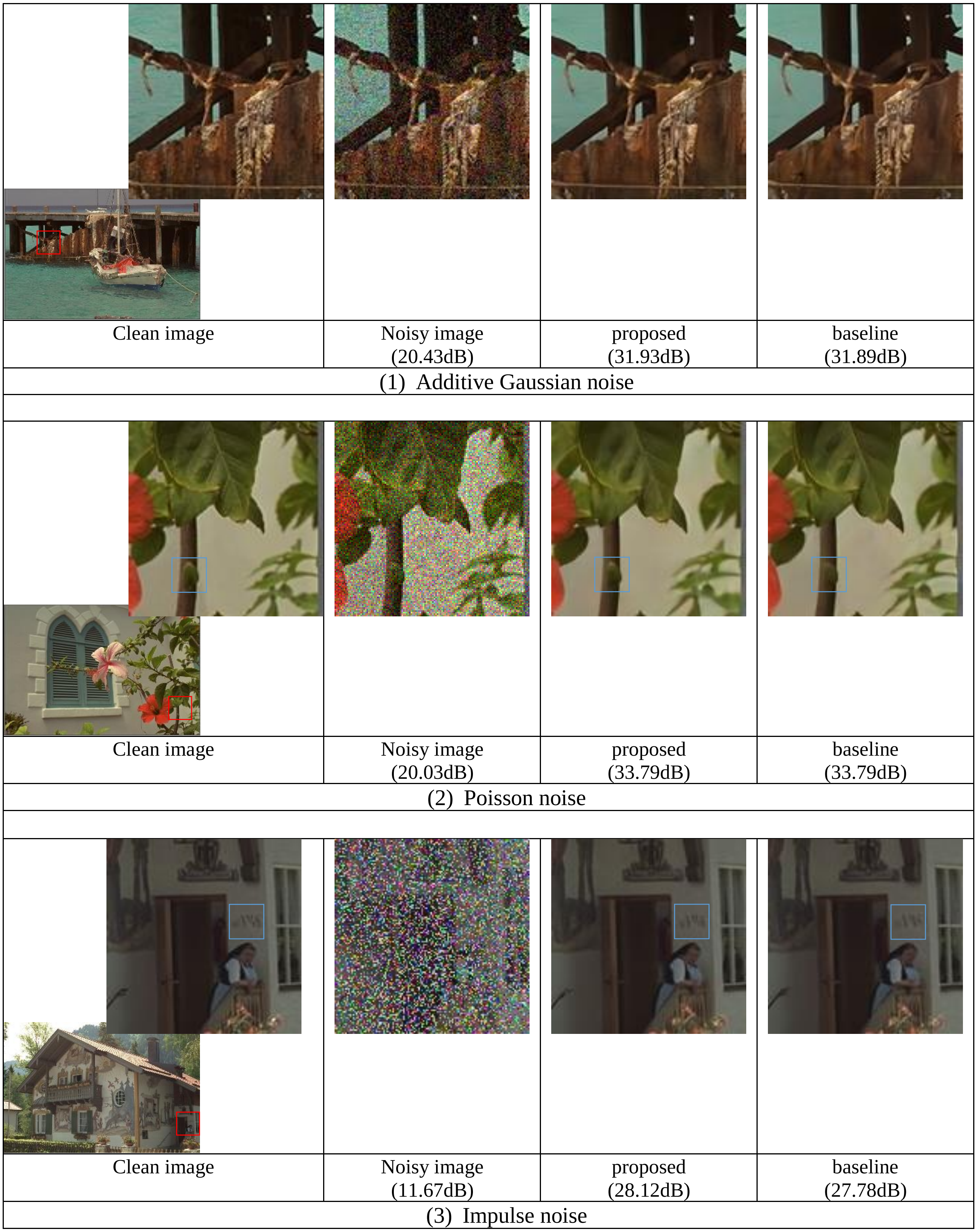

Fig. 3. Performance comparison with perceptual reconstruction results between our proposed method and the baseline [19] on different noisy images. As it can be observed, our proposed method presents more details and precise boundaries, achieving higher PSNR in the evaluation as well.