# DCANet: Dual Convolutional Neural Network with Attention for Image Blind Denoising

Wencong Wu[a], Guannan Lv[a], Yingying Duan[a], Peng Liang[a], Yungang Zhang[a,*], Yuelong Xia[a]

[a]*School of Information Science and Technology, Yunnan Normal University, Kunming 650500, Yunnan Province, China*

## Abstract

Noise removal of images is an essential preprocessing procedure for many computer vision tasks. Currently, many denoising models based on deep neural networks can perform well in removing the noise with known distributions (i.e. the additive Gaussian white noise). However eliminating real noise is still a very challenging task, since real-world noise often does not simply follow one single type of distribution, and the noise may spatially vary. In this paper, we present a new dual convolutional neural network (CNN) with attention for image blind denoising, named as the DCANet. To the best of our knowledge, the proposed DCANet is the first work that integrates both the dual CNN and attention mechanism for image denoising. The DCANet is composed of a noise estimation network, a spatial and channel attention module (SCAM), and a CNN with a dual structure. The noise estimation network is utilized to estimate the spatial distribution and the noise level in an image. The noisy image and its estimated noise are combined as the input of the SCAM, and a dual CNN contains two different branches is designed to learn the complementary features to obtain the denoised image. The experimental results have verified that the proposed DCANet can suppress both synthetic and real noise effectively. The code of DCANet is available at https://github.com/WenCongWu/DCANet.

*Keywords:* Image blind denoising, dual convolutional neural network, attention mechanism, noise estimation.

## 1. Introduction

As one of the most significant research areas of low-level visual tasks, image denoising aims to restore clean images from noisy ones. During the past decades, many researchers have presented a number of denoising methods. Before the wide application of deep neural networks (DNNs), filtering techniques and sparse learning are widely used denoising methods. For instance, in the NLM [1], the weighted average of all pixels within the search window in an image is applied to achieve noise removal. The BM3D [2] improves the sparse representation by collaborative alteration. The trilateral weighted sparse coding (TWSC) [3] accomplishes real image denoising by employing image priors. The weighted nuclear norm minimization (WNNM) [4] and the multi-channel WNNM (MCWNNM) for color images [5] employ the low rank approach and prior knowledge to enhance denoising performance. These denoising methods can obtain favorable denoising performance, however most of them have to contain a complex and time-consuming optimization algorithm. Meanwhile, many manually adjusted parameters are also usually required for these models to perform well, this may lead to uncertainty of their denoising performance. Therefore these models can hardly be applied in practical denoising scenes.

From the early successful deep neural networks (DNNs) based image denoising model DnCNN [6] to the present, the DNNs based denoising models have received much attention due to their superior denoising effect. However, many denoising models based on DNN are designed for the additive white Gaussian noise (AWGN) only. Moreover, in early DNN-based models, usually one model is trained only for one specific noise level, therefore it is very hard to generalize these models for other noise levels, or other types of noise [7]. To make the DNN based denoising models more flexible, many techniques have been proposed. For instance, a tunable noise level map is exploited as the input

---

of the FFDNet [8]. As the map is variable or non-uniform, the FFDNet is able to handle different noise levels and the spatially varying noise can be handled as well. Nevertheless, the map of the FFDNet has to be set manually by human experience, which makes the model still far from the real denoising tasks.

For real denoising scenes, noise removal has to be completed with the unknown noise level or distribution, namely, blind denoising is required. To this end, many methods have been proposed. As an extension of the FFDNet, the CBDNet [9] introduces a noise estimation sub-network, which makes the CBDNet as a fully blind denoising model. An optimal fusion function derived from the Sigal-to-Noise Ratio (SNR) was proposed in [10], and a corresponding BUIFD was then designed for image blind denoising. The AirNet model was proposed in [11], which can restore low quality images corrupted by multiple degradation models by using only one single network. However, many existing blind denoising models have massive parameters to be learned, and their performance still can be further improved, especially in the removal of real noise.

Recently, some researchers have investigated different network structures to improve the learning ability of the denoising models. For instance, the BRDNet [12] contains two different branches (the upper sub-network and the lower one), which enables the model to extract complementary features and therefore enhance the denoising performance. Pan et al. [13] presented a DualCNN containing two sub-networks of different depths, where the shallow sub-network is utilized to extract the image structures, and the deeper one aims to learn the image details. Both the BRDNet and the DualCNN achieve competitive denoising performance, demonstrating that using the dual structure CNN to learn complementary features can be helpful for image denoising.

The attention mechanism can be a helpful tool for image denoising has also been verified by many researchers. For example, Anwar et al. [14] proposed the RIDNet model containing feature attention, which mainly includes four enhancement attention modules (EAM) with short skip connection and local connection. The ADNet model was developed in [15], in which an essential attention block is utilized to filter the noise information. Although both the RIDNet and ADNet achieve noticeable denoising performance, only one type of attention mechanism is used in their models, this may lead to the ignoring of the other inter-relationships of convolutional features.

Motivated by the success of the dual network structure and the attention mechanism in image denoising, in this paper, we present an effective dual CNN with attention mechanism for image blind denoising, named as the DCANet, which can achieve a competitive blind denoising performance. In our DCANet, the noise level map of a noisy image is first estimated by the noise estimation network, the noise information and the noisy image are then concatenated together as the input of the spatial and channel attention module (SCAM). The SCAM makes the model focus on the features that contain more useful contextual information. The dual CNN in our DCANet is composed of two different sub-networks, where the downsampling and dilated convolutions are used to enlarge the receptive field. The skip connections are also used to improve the training of the network. The DCANet owns the following characteristics:

(1) As far as we know, the proposed DCANet is the first model that investigates the integration of dual CNN and attention mechanism for image blind denoising.

(2) A novel dual CNN denoising network and a new noise estimator are designed in our denoising model.

(3) The DCANet is capable of obtaining competitive denoising results compared with other state-of-the-art denoising models. Moreover, the DCANet has a simpler model structure compared with many existing blind denoising models.

The remainder of this paper is organized as follows. Section 2 reviews the related denoising methods. Our proposed DCANet model is introduced in Section 3. Section 4 reports the experimental results. The paper is summarized in Section 5.


## 2. Related work

### 2.1. Image blind denoising models

Some early DNN based denoising models have already made the effort to blind denoising. For example, the famous DnCNN [6] can be used for real denoising, however its performance is far from satisfaction. Zhang et al. [16] proposed the IRCNN model by learning deep denoiser prior to realize blind denoising, and promising results were reported. Peng et al. [17] designed a dilated residual network for blind denoising, where the authors claimed that dilated convolution and skip connection can be helpful for noise removal.

To capture the noise feature better, many researchers have tried to utilize a noise estimation network to estimate the noise distributions and levels in an image. Based on the DnCNN, the BUIFD model with a noise estimator was proposed in [10]. The BUIFD can handle unknown and universal noise by a single model. Soh et al. [18] presented a deep universal blind denoising (DUBD) network, where a conditional estimation network (CENet) is introduced for incorporating human prior knowledge, and a tunable denoising sub-network is designed for noise removal. Guo et al. [9] designed the CBDNet model for blind denoising, where a noise estimation sub-network is used to estimate the noise level, and asymmetric learning is adopted to restrain the possible under-estimation of the noise level. Yue et al. [19] presented a novel variational denoising network (VDN) for image blind denoising. The VDN contains a denoising network and a Sigma network, the Sigma network is applied to estimate the noise distribution in a noisy image. Kim et al. [20] proposed an adaptive instance normalization denoising network (AINDNet), and the adaptive instance normalization (AIN) is developed to regularize the network, so that the model will avoid overfitting to the synthetical noisy images, and a noise level estimator is applied to estimate the noise level.

It can be found that currently most of the blind denoising models utilize a noise estimation sub-network as the solution for achieving blind denoising, and the effectiveness of this solution has been verified. In our proposed denoising model, a noise estimation block is also developed. Compared with the noise estimation networks of the CBDNet [9], VDN [19], and DUBD [18], our noise estimator is designed to contain more convolutional layers, which we think can help to extract more complex noise information.

### 2.2. Attention-guided denoising models

Extracting and selecting appropriate features is extremely important for image processing [21, 22, 23]. However, for images with complex textures, it is very difficult to obtain features with high discriminative ability [24]. To this end, an attention mechanism is proposed in [25], which can focus on the salient regions in images and can capture the useful features better. Since then the attention mechanisms have been heavily utilized in image analysis tasks, including image denoising.

A residual dilated attention network (RDAN) was designed in [26] for blind denoising, which contains two attention blocks, the attention mechanism in RDAN enhances the restoration of texture details. A multi-stage denoising architecture MPRNet [27] was developed for image restoration, where the supervised attention modules and channel attention blocks were introduced and used to promote the restoration effect. A very deep residual non-local attention network (RNAN) was proposed in [28] for image restoration, the attention-based RNAN achieves competitive performances on different image restoration tasks, including image denoising. The channel and space attention neural network (CSANN) was presented in [29], the CSANN can predict the residual image to eliminate noise. An adaptive consistency prior (ACP) was presented in DeamNet [30], in which the dual element-wise attention mechanism (DEAM) modules were designed and utilized to enhance its learning ability.

It can be observed that the MPRNet [27], RNAN [28], and CSANN [29] all use multiple attention mechanisms to exploit different dependencies in convolutional features, and remarkable denoising results were produced by these models. The success of these attention-guided models reveals that the attention mechanism can be an effective tool for image denoising.

### 2.3. Dual convolutional neural network for image denoising

In general, very deep networks may cause the vanishing or exploding of gradients, which affects the performance of the network. To address this issue, some researchers have tried to promote the learning ability of the denoising model by increasing the width of the network instead of the depth. The dual CNN adopts such network composition strategy, where two parallel branches (sub-networks) are contained in a dual CNN, and usually different network structures are designed for the two branches to boost the learning ability.

The dual CNN for image denoising tasks has also been investigated by some researchers. Tian et al. [12] presented a BRDNet model, which consists of a dual CNN containing two different sub-networks. As different network architectures can capture different image features [31], the BRDNet therefore can extract complementary features to improve its denoising performance, impressive denoising results were reported by BRDNet. Later, Tian et al. [32] designed another dual denoising network (DudeNet) with two different branches to achieve effective noise removal, and the model also obtains favorable results. For general low-level vision tasks, Pan et al. [33] presented a dual CNN (DualCNN). The DualCNN is composed of two different parallel branches, where one branch is used for capturing the image structures, and another branch is applied to extract the image details.

## 3. The proposed model

### 3.1. Network architecture

In this section, we introduce the proposed DCANet. Fig. 1 shows the architecture of the DCANet. The model consists of a noise estimation network, a spatial and channel attention module (SCAM), and a dual CNN. It should be noted that there is a convolution layer between the noise estimation network and the SCAM module. The convolution layer here takes the concatenation of the noisy image and the estimated noise level map as its input, then a feature map with 64 channels is produced, which is sent to the SCAM for feature filtering. Moreover, for all the convolutions in the model, the convolutional filter size is set to $3 \times 3 \times 64$, which can balance network complexity and denoising performance. It is worth noting that we do not draw BN and ReLU layers in each layer of convolution or dilated convolution. Although there are many existing methods utilize the dual CNN structures or incorporate attention mechanisms for image denoising, as far as we know, no previous works have been reported to study the combination of the dual CNN and attention mechanism for image blind denoising.
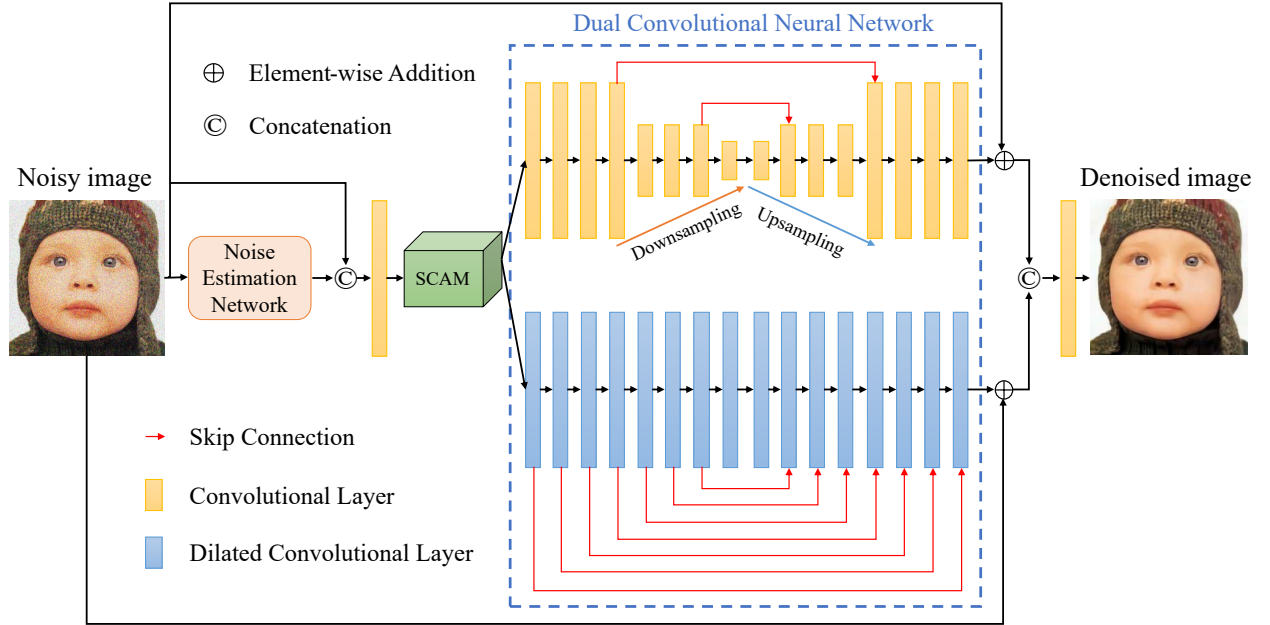


Figure 1: The network structure of the proposed model.

### 3.1.1. Noise estimation network

To accurately estimate the noise in an image, unlike the noise estimation network of CBDNet [9] and VDN [19], our noise estimation network contains more convolutional layers, which can make the network extract complex noise information. The architectures of noise estimation sub-networks of the proposed DCANet, CBDNet [9], AINDNet [20], and VDN [19] can be seen in Fig. 2. Our estimator has 7 convolutional layers, where the Convolution (Conv), Rectified Linear Units (ReLU) [34], Batch Normalization (BN) [35] and Tanh [36] are used in the network. Specifically, "Conv+ReLU", "Conv+BN+ReLU" and "Conv+Tanh" are used in the first, the middle, and the last layers respectively. The zero padding is used in our noise estimator to keep the feature map size constant.

It should be noted that both our noise estimation network and the estimator of AINDNet [20] have seven convolutional layers. Different from ours, the noise estimation network of the AINDNet has two downsampling operations. Although a larger receptive field size is beneficial to extract more image information, the downsampling operations yet will bring information loss. In particular, compared with the noise estimation sub-networks of the CBDNet, VDN, and AINDNet, our estimation network incorporates multiple BN layers, which can not only accelerate network training, but also enhance network performance.

(a) The noise estimation network of our DCANet.

(b) The noise estimation network of CBDNet [9].

Downsampling → Upsampling →

(c) The noise estimation network of AINDNet [20].
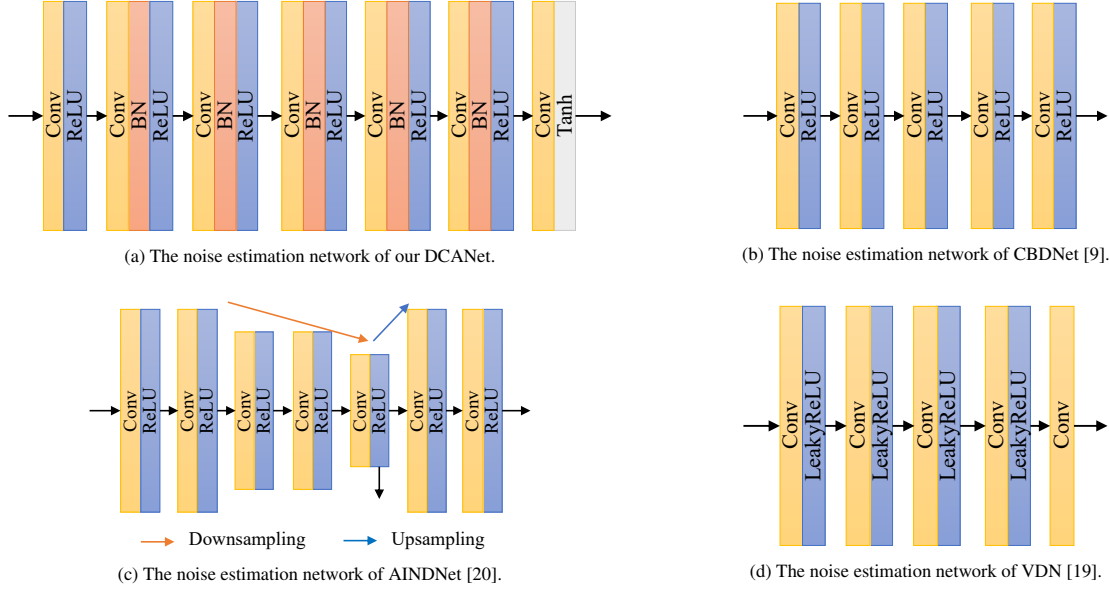
(d) The noise estimation network of VDN [19].

Figure 2: The noise estimation network architectures of our DCANet (a), CBDNet (b), AINDNet (c), and VDN (d).

### 3.1.2. Spatial and channel attention module

To leverage the useful features in images, many denoising models employ attention mechanisms to improve their learning ability, such as the ADNet [15], RDAN [26], RNAN [28], CSANN [29], and DeamNet [30]. Following the attention network introduced in CSANN [29], our proposed DCANet also utilizes the spatial and channel attention for feature selection, which is presented in Fig. 3. Our spatial and channel attention module (SCAM) contains the following operations: Conv, PReLU [37], Global Average Pooling [38], Global Max Pooling [39], BN, ReLU, Sigmoid [40]. The SCAM is employed to exploit the inter-spatial and inter-channel relationships of the convolutional features. The more informative features are retained by the SCAM, and the unimportant ones are suppressed.
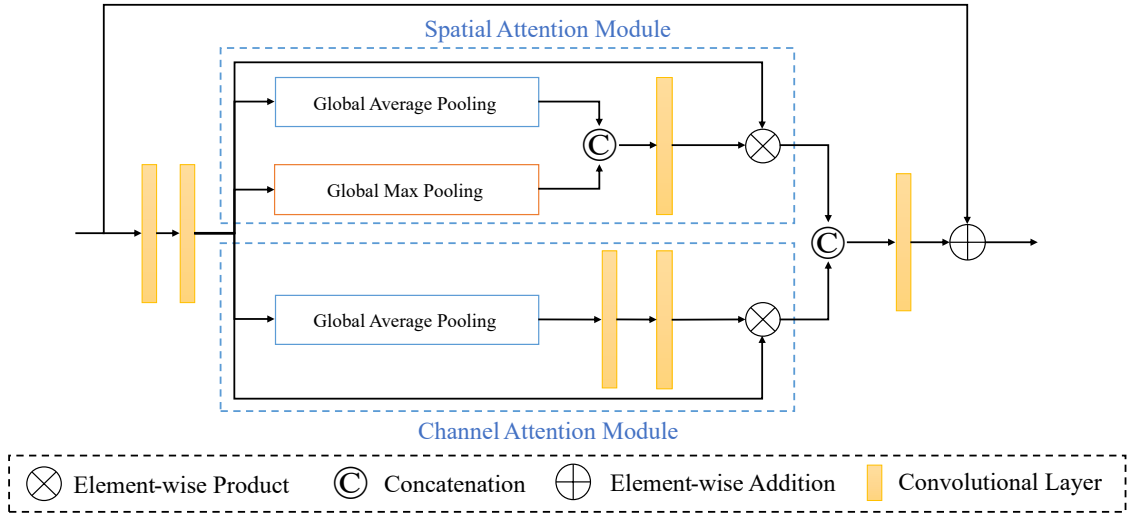


Figure 3: The structure of the SCAM.

5

### 3.1.3. Dual convolutional neural network

It has been verified that expanding the width of the denoising network is also a useful way to improve network performance [41]. In this work, we develop a dual CNN with two different sub-networks for image denoising, which can be seen in the blue dotted box in Fig. 1, where the upper sub-network is an u-shaped network, and the lower branch is a dilated convolution network. It has been verified that using different sub-networks in dual CNN can enhance the performance of the whole network, as the complementary features can be learned by the two different branches [12, 31, 32]. The upper branch of our dual CNN involves the following operations: Conv, ReLU, BN, max-pooling (Downsampling) [42], bilinear interpolation (Upsampling) [43], and skip connections [44, 45]. The lower branch includes Dilated Convolution (DConv) [46], ReLU, BN, and the skip connections as well. The outputs of two different branches are concatenated through a concatenation operation.

Compared with the existing dual CNN based denoising models such as the BRDNet [12] and DualCNN [33], our proposed dual network is designed to own a larger receptive field, which allows the network to capture more context information. Specifically, in our dual network, the upper branch employs two times of downsampling to increase the receptive field. However, the downsampling operation may cause the loss of image information. Therefore, two skip connections are applied in the upper sub-networks to suppress the loss. The upsampling operations are employed to recover the size of the feature map. The lower sub-network applies the dilated convolution to increase the size of the receptive field, and the symmetric skip connections are utilized to speed up network training. Particularly, we adopt the hybrid dilated convolution (HDC) in the lower branch, it has been verified that the HDC can remove the gridding phenomenon effectively and improve network performance [47, 48]. The dilated rate of each dilated convolution layer in the lower branch is shown in Table 1.

Table 1: The dilated rates of different layers in the lower sub-network.

| Layer | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Dilated rate | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 1 |

### 3.2. Loss functions

Mean squared error (MSE) is a widely used loss function for optimizing neural networks [49, 50]. For the AWGN removal, we optimize our DCANet by the following loss function, which is expressed as:

$$\mathcal{L} = \frac{1}{2N} \sum_{i=1}^{N} \|f(y_i, \theta) - x_i\|^2$$
$$= \frac{1}{2N} \sum_{i=1}^{N} \|\hat{x}_i - x_i\|^2,$$

(1)

where $x_i$, $y_i$, and $\hat{x}_i$ are the clean, noisy, and denoised images, respectively. $N$ is the number of clean-noisy image patches, and $\theta$ is the parameters of the DCANet.

For removing the unknown real noise, as the high-frequency textures will be lost due to the square penalty, then the MSE often produces a blurred and overly smoothed visual effect. Therefore, we select the Charbonnier loss [51] as the reconstruction loss to optimize our model for real noise removal. Moreover, to preserve the fidelity and authenticity of the high-frequency details, we utilize the edge loss proposed in [52] to constrain the loss of the high-frequency components between the ground-truth image $x$ and the denoised image $\hat{x}$.

Our proposed DCANet contains a noise estimator to predict the noise level $\sigma(y)$ in a noisy image $y$. For real noise removal, the estimated noise level should be avoided to be too smooth, especially for the spatially variant noise, which may change dramatically throughout the whole noisy image. Therefore, we employ a total variation (TV) regularizer [9] to restrict the smoothness of the estimated noise level.

The total loss of our DCANet is expressed as:

$$\mathcal{L} = \mathcal{L}_{char}(\hat{x}, x) + \lambda_{edge}\mathcal{L}_{edge}(\hat{x}, x) + \lambda_{TV}\mathcal{L}_{TV}(\sigma(y)),$$

(2)

6

where we empirically set $\lambda_{edge}$ and $\lambda_{TV}$ to 0.1 and 0.05 respectively, and $\mathcal{L}_{char}$ is the Charbonnier loss, which is represented as:

$$\mathcal{L}_{char} = \sqrt{\|\hat{x} - x\|^2 + \epsilon^2},  \tag{3}$$

where the constant $\epsilon$ is set as $10^{-3}$. $\mathcal{L}_{edge}$ is the edge loss, which can be expressed as:

$$\mathcal{L}_{edge} = \sqrt{\|\triangle(\hat{x}) - \triangle(x)\|^2 + \epsilon^2},  \tag{4}$$

where $\triangle$ denotes the Laplacian operator [53]. $\mathcal{L}_{TV}$ is defined as:

$$\mathcal{L}_{TV} = \|\nabla_h \sigma(y)\|_2^2 + \|\nabla_v \sigma(y)\|_2^2,  \tag{5}$$

where $\nabla_h(\nabla_v)$ is the gradient operator along the horizontal (vertical) direction.

## 4. Experiments and results

### 4.1. Datasets

The DIV2K dataset [54] containing 800 high-resolution color images was used to train the proposed DCANet for the AWGN removal, and we resized these high-resolution images into the images of the size $512 \times 512$. Moreover, the images were grayscaled to train the DCANet for grayscale image denoising. According to the size of the receptive field of the DCANet, the training images were randomly cropped into patches of the size $140 \times 140$. The AWGN within a noise level range of $[0, 75]$ is added to the clean image patches to generate their noisy counterparts. To augment the training samples, rotation and flipping were utilized. We tested our model on five public commonly used datasets, including Set12 [55], BSD68 [55], CBSD68 [55], Kodak24 [56] and McMaster [57].

For real noise removal, we select the SIDD training data [58] and RENOIR dataset [59] for model training. The SIDD training data and the RENOIR dataset consist of 320 and 240 pairs of noisy images and the near noise-free counterparts, respectively. We randomly cut the images in these datasets into image patches of size $140 \times 140$, and applied rotation and flipping operations for data augmentation. The SIDD validation set [58] and DND sRGB dataset [60] were used for model evaluation.

### 4.2. Experimental settings

All our experiments were performed on a PC equipped with a CPU of Intel(R) Core(TM) i7-11700KF, 32 GB RAM, and a GPU of NVIDIA GeForce RTX 3080Ti. The proposed DCANet was trained for grayscale and color images, respectively. The training of the DCANet for the synthetic noise cost about 48 hours. For real image denoising, it took about 45 hours to train the model.

The network parameters of the DCANet were optimized by the Adam optimizer [61]. For the AWGN removal, the DCANet was trained for 600,000 iterations, during which the initial learning rate is $10^{-4}$ and then decreases by half every 100,000 iterations, and the batch size was set to 24. For real image denoising, we used 120 epochs to train the model, during which the initial learning rate was $2 \times 10^{-4}$ and steadily reduced to $10^{-6}$ using cosine annealing strategy [62], the batch size was set to 16. For other hyper-parameters of the Adam algorithm, we used the default settings.

### 4.3. Ablation study

To verify the effectiveness of the proposed denoising architecture, especially the effect of SCAM and skip connections, we trained four different networks for grayscale image denoising. Models are trained with/without SCAM and with/without skip connection. The four different models and their corresponding performance are shown in Table 2. The Set12 [55] dataset was utilized for our ablation evaluation, and the noise level was set to 35.

In Table 2, one can find that the denoising network with SCAM and skip connection achieved better denoising performance than the other three different networks. The results validate that using SCAM and skip connections in the denoising networks can improve their performance.

Table 2: Quantitative comparison results of the four models containing different components on the Set12 dataset. The noise level is set to 35. The best results are bolded.

| Methods | PSNR | SSIM |
|---|---|---|
| model without SCAM and with skip connection | 28.95 | 0.8308 |
| model with SCAM and without skip connection | 29.00 | 0.8326 |
| model without SCAM and skip connection | 28.98 | 0.8314 |
| model with SCAM and skip connection | **29.08** | **0.8353** |

## 4.4. The additive white Gaussian noise removal

In this subsection, we show the results of our DCANet on grayscale and color images corrupted by the AWGN. The BM3D [2], WNNM [4], MCWNNM [5], TNRD [63], DnCNN [6], BUIFD [10], IRCNN [16], FFDNet [8], BRDNet [12], ADNet [15], DudeNet [32], DSNetB [17], RIDNet [14], AINDNet [20] and AirNet [11] were used to compare with our DCANet.

Table 3 lists the PSNR values at different noise levels of the compared denoising models on the Set12 dataset. One can see that the denoising performances of the DCANet are slightly behind the BRDNet and ADNet at noise levels 15 and 25, however our DACNet beats other methods at noise level 50. It also should be noted that for the image "Barbara", the traditional methods BM3D and WNNM obtain significant denoising results, we think the reason is that this image contains rich repetitive structures, and the non-local self-similarity learning based methods can capture such structures better.

Table 3: Quantitative comparison results on the Set12 dataset. The three best results are respectively emphasized in red, blue and green.

| Noise levels | Models | C.man | House | Peppers | Starfish | Monar. | Airpl. | Parrot | Lena | Barbara | Boat | Man | Couple | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\sigma$=15 | BM3D [2] | 31.91 | 34.93 | 32.69 | 31.14 | 31.85 | 31.07 | 31.37 | 34.26 | 33.10 | 32.13 | 31.92 | 31.10 | 32.37 |
| | WNNM [4] | 32.17 | 35.13 | 32.99 | 31.82 | 32.71 | 31.39 | 31.62 | 34.27 | 33.60 | 32.27 | 32.11 | 32.17 | 32.70 |
| | TNRD [63] | 32.19 | 34.53 | 33.04 | 31.75 | 32.56 | 31.46 | 31.63 | 34.24 | 32.13 | 32.14 | 32.23 | 32.11 | 32.50 |
| | DnCNN-S [6] | 32.61 | 34.97 | 33.30 | 32.20 | 33.09 | 31.70 | 31.83 | 34.62 | 32.64 | 32.42 | 32.46 | 32.47 | 32.86 |
| | IRCNN [16] | 32.55 | 34.89 | 33.31 | 32.02 | 32.82 | 31.70 | 31.84 | 34.53 | 32.43 | 32.34 | 32.40 | 32.40 | 32.77 |
| | BUIFD [10] | 31.74 | 34.78 | 32.80 | 31.92 | 32.77 | 31.34 | 31.39 | 34.38 | 31.68 | 32.18 | 32.25 | 32.22 | 32.46 |
| | FFDNet [8] | 32.43 | 35.07 | 33.25 | 31.99 | 32.66 | 31.57 | 31.81 | 34.62 | 32.54 | 32.38 | 32.41 | 32.46 | 32.77 |
| | BRDNet [12] | 32.80 | 35.27 | 33.47 | 32.24 | 33.35 | 31.85 | 32.00 | 34.75 | 32.93 | 32.55 | 32.50 | 32.62 | 33.03 |
| | ADNet [15] | 32.81 | 35.22 | 33.49 | 32.17 | 33.17 | 31.86 | 31.96 | 34.71 | 32.80 | 32.57 | 32.47 | 32.58 | 32.98 |
| | DudeNet [32] | 32.71 | 35.13 | 33.38 | 32.29 | 33.28 | 31.78 | 31.93 | 34.66 | 32.73 | 32.46 | 32.46 | 32.49 | 32.94 |
| | DCANet | 32.26 | 35.18 | 33.05 | 32.11 | 33.18 | 31.53 | 31.81 | 34.73 | 32.56 | 32.49 | 32.42 | 32.50 | 32.82 |
| $\sigma$=25 | BM3D [2] | 29.45 | 32.85 | 30.16 | 28.56 | 29.25 | 28.42 | 28.93 | 32.07 | 30.71 | 29.90 | 29.61 | 29.71 | 29.97 |
| | WNNM [4] | 29.64 | 33.22 | 30.42 | 29.03 | 29.84 | 28.69 | 29.15 | 32.24 | 31.24 | 30.03 | 29.76 | 29.82 | 30.26 |
| | TNRD [63] | 29.72 | 32.53 | 30.57 | 29.02 | 29.85 | 28.88 | 29.18 | 32.00 | 29.41 | 29.91 | 29.87 | 29.71 | 30.06 |
| | DnCNN-S [6] | 30.18 | 33.06 | 30.87 | 29.41 | 30.28 | 29.13 | 29.43 | 32.44 | 30.00 | 30.21 | 30.10 | 30.12 | 30.43 |
| | IRCNN [16] | 30.08 | 33.06 | 30.88 | 29.27 | 30.09 | 29.12 | 29.47 | 32.43 | 29.92 | 30.17 | 30.04 | 30.08 | 30.38 |
| | BUIFD [10] | 29.42 | 33.03 | 30.48 | 29.21 | 30.20 | 28.99 | 28.94 | 32.20 | 29.18 | 29.97 | 29.88 | 29.90 | 30.12 |
| | FFDNet [8] | 30.10 | 33.28 | 30.93 | 29.32 | 30.08 | 29.04 | 29.44 | 32.57 | 30.01 | 30.25 | 30.11 | 30.20 | 30.44 |
| | BRDNet [12] | 31.39 | 33.41 | 31.04 | 29.46 | 30.50 | 29.20 | 29.55 | 32.65 | 30.34 | 30.33 | 30.14 | 30.28 | 30.61 |
| | ADNet [15] | 30.34 | 33.41 | 31.14 | 29.41 | 30.39 | 29.17 | 29.49 | 32.61 | 30.25 | 30.37 | 30.08 | 30.24 | 30.58 |
| | DudeNet [32] | 30.23 | 33.24 | 30.98 | 29.53 | 30.44 | 29.14 | 29.48 | 32.52 | 30.15 | 30.24 | 30.08 | 30.15 | 30.52 |
| | DCANet | 30.11 | 33.38 | 30.79 | 29.49 | 30.50 | 29.07 | 29.49 | 32.73 | 30.28 | 30.35 | 30.13 | 30.25 | 30.55 |
| $\sigma$=50 | BM3D [2] | 26.13 | 29.69 | 26.68 | 25.04 | 25.82 | 25.10 | 25.90 | 29.05 | 27.22 | 26.78 | 26.81 | 26.46 | 26.72 |
| | WNNM [4] | 26.45 | 30.33 | 26.95 | 25.44 | 26.32 | 25.42 | 26.14 | 29.25 | 27.79 | 26.97 | 26.94 | 26.64 | 27.05 |
| | TNRD [63] | 26.62 | 29.48 | 27.10 | 25.42 | 26.31 | 25.59 | 26.16 | 28.93 | 25.70 | 26.94 | 26.98 | 26.50 | 26.81 |
| | DnCNN-S [6] | 27.03 | 30.00 | 27.32 | 25.70 | 26.78 | 25.87 | 26.48 | 29.39 | 26.22 | 27.20 | 27.24 | 26.90 | 27.18 |
| | IRCNN [16] | 26.88 | 29.96 | 27.33 | 25.57 | 26.61 | 25.89 | 26.55 | 29.40 | 26.24 | 27.17 | 27.17 | 26.88 | 27.14 |
| | BUIFD [10] | 25.44 | 29.76 | 26.50 | 24.87 | 26.49 | 25.34 | 25.07 | 28.81 | 25.49 | 26.59 | 26.87 | 26.34 | 26.46 |
| | FFDNet [8] | 27.05 | 30.37 | 27.54 | 25.75 | 26.81 | 25.89 | 26.57 | 29.66 | 26.45 | 27.33 | 27.29 | 27.08 | 27.32 |
| | BRDNet [12] | 27.44 | 30.53 | 27.67 | 25.77 | 26.97 | 25.93 | 26.66 | 25.93 | 26.66 | 27.38 | 27.27 | 27.17 | 27.45 |
| | ADNet [15] | 27.31 | 30.59 | 27.69 | 25.70 | 26.90 | 25.88 | 26.56 | 29.59 | 26.64 | 27.35 | 27.17 | 27.07 | 27.37 |
| | DudeNet [32] | 27.22 | 30.27 | 27.51 | 25.88 | 26.93 | 25.88 | 26.50 | 29.45 | 26.49 | 27.26 | 27.19 | 26.97 | 27.30 |
| | DCANet | 27.42 | 30.63 | 27.54 | 25.90 | 27.06 | 25.96 | 26.66 | 29.83 | 27.16 | 27.44 | 27.30 | 27.20 | 27.51 |

Table 4 presents the average SSIM results of the compared models on the Set12 dataset. As can be seen from the table, our DCANet also produces competitive results.

Table 5 reports the quantitative results on the BSD68 dataset of the compared denoising models. It can be found that the RIDNet and BRDNet obtain leading performance at noise levels 15 and 25. Our proposed model can obtain

Table 4: Quantitative comparison results on the Set12 dataset. The three best results are respectively emphasized in red, blue and green.

| Models | $\sigma$=15 | $\sigma$=25 | $\sigma$=50 |
|---|---|---|---|
| BM3D [2] | 0.896 | 0.851 | 0.766 |
| WNNM [4] | 0.894 | 0.846 | 0.756 |
| TNRD [63] | 0.896 | 0.851 | 0.768 |
| DnCNN-S [6] | 0.903 | 0.862 | 0.783 |
| IRCNN [16] | 0.901 | 0.860 | 0.780 |
| BUIFD [10] | 0.899 | 0.855 | 0.755 |
| FFDNet [8] | 0.903 | 0.864 | 0.791 |
| BRDNet [12] | 0.906 | 0.866 | 0.794 |
| ADNet [15] | 0.905 | 0.865 | 0.791 |
| RIDNet [14] | 0.906 | 0.867 | 0.793 |
| DCANet | 0.903 | 0.866 | 0.797 |

competitive denoising results at all tested noise levels, and especially our model outperforms other denoising methods at noise level 50.

Fig. 4 shows the visual comparison results on the "test003" image at noise level 50 of different denoising models. One can find that compared with other models, the DCANet obtains better denoising performance, both quantitatively and qualitatively.

It can be seen from our experimental results on the grayscale images that our DCANet can achieve competitive denoising performance compared with other state-of-the-art methods. Especially, a trend can be discovered from Table 3, Table 4, and Table 5, that for grayscale image denoising, our model becomes more effective as the noise power increases.

For the noise removal evaluation on color images, the CBSD68, Kodak24, and McMaster datasets were applied. Table 6 lists the average PSNR values at different noise levels of many models on three datasets. It can be seen that our model obtains leading denoising results compared to other state-of-the-art denoising models.

Table 7 reports the average SSIM values at multiple noise levels of the compared denoising methods on three color image datasets. One can find that the DCANet also generates better results.

Fig. 5 presents the visual comparison results on the "kodim23" image from the Kodak24 dataset of different denoising models. One can see that our model obtains a visual appealing result, and a favorable trade-off between image detail preservation and noise reduction is obtained.

### 4.5. Real noise removal

For real noise removal, we evaluated the denoising methods on two commonly used public datasets: the SIDD validation set and the DND sRGB images. The two datasets contain real noisy images and near noise-free counterparts, and the counterparts can be provided as the ground truth, which can be utilized to obtain the PSNR and SSIM values.

Table 8 lists the average PSNR and SSIM values of the compared models on the SIDD validation set and the DND sRGB images. One can see that the DCANet model obtains effective denoising performance.

Fig. 6 shows the visual results of the compared models for real noise removal. It can be found the MCWNNM, TWSC, and CDnCNN-B generated unsatisfactory visual effects. In contrast, the AINDNet, VDN, and the proposed DCANet achieved much better visual quality and PSNR/SSIM values.

### 4.6. Model complexity analysis

We evaluated the network complexity of the proposed DCANet from the perspectives of the running time, the floating point operations per second (FLOPs), and the number of network parameters. The codes of all the compared denoising models are from the original authors. The BM3D [2], WNNM [4] and MCWNNM [5] were run in Matlab (R2020a) environment. The DnCNN-B [6], BUIFD [10], IRCNN [16], FFDNet [8], BRDNet [12], CBDNet [9], RIDNet [14], DudeNet [32], AINDNet [20], VDN [19], ADNet [15], AirNet [11], and the proposed DCANet were implemented in PyCharm (2021) environment.

Table 5: Quantitative comparison results on the BSD68 dataset. The three best results are respectively emphasized in red, blue and green.

| Metrics | Models | $\sigma$=15 | $\sigma$=25 | $\sigma$=50 |
|---------|--------|------|------|------|
| PSNR | BM3D [2] | 31.07 | 28.57 | 25.62 |
| | WNNM [4] | 31.37 | 28.83 | 25.87 |
| | DnCNN-S [6] | 31.72 | 29.23 | 26.23 |
| | TNRD [63] | 31.42 | 28.92 | 25.97 |
| | IRCNN [16] | 31.63 | 29.15 | 26.19 |
| | BUIFD [10] | 31.35 | 28.75 | 25.11 |
| | FFDNet [8] | 31.63 | 29.19 | 26.29 |
| | DSNetB [17] | 31.69 | 29.22 | 26.29 |
| | RIDNet [14] | 31.81 | 29.34 | 26.40 |
| | BRDNet [12] | 31.79 | 29.29 | 26.36 |
| | ADNet [15] | 31.74 | 29.25 | 26.29 |
| | AINDNet [20] | 31.69 | 29.26 | 26.32 |
| | DudeNet [32] | 31.78 | 29.29 | 26.31 |
| | DCANet | 31.70 | 29.29 | 26.42 |
| SSIM | BM3D [2] | 0.872 | 0.802 | 0.687 |
| | WNNM [4] | 0.878 | 0.810 | 0.698 |
| | TNRD [63] | 0.883 | 0.816 | 0.703 |
| | DnCNN-S [6] | 0.891 | 0.828 | 0.719 |
| | IRCNN [16] | 0.888 | 0.825 | 0.717 |
| | BUIFD [10] | 0.886 | 0.819 | 0.682 |
| | FFDNet [8] | 0.890 | 0.830 | 0.726 |
| | BRDNet [12] | 0.893 | 0.831 | 0.727 |
| | ADNet [15] | 0.892 | 0.829 | 0.722 |
| | RIDNet [14] | 0.893 | 0.833 | 0.727 |
| | DCANet | 0.890 | 0.831 | 0.730 |
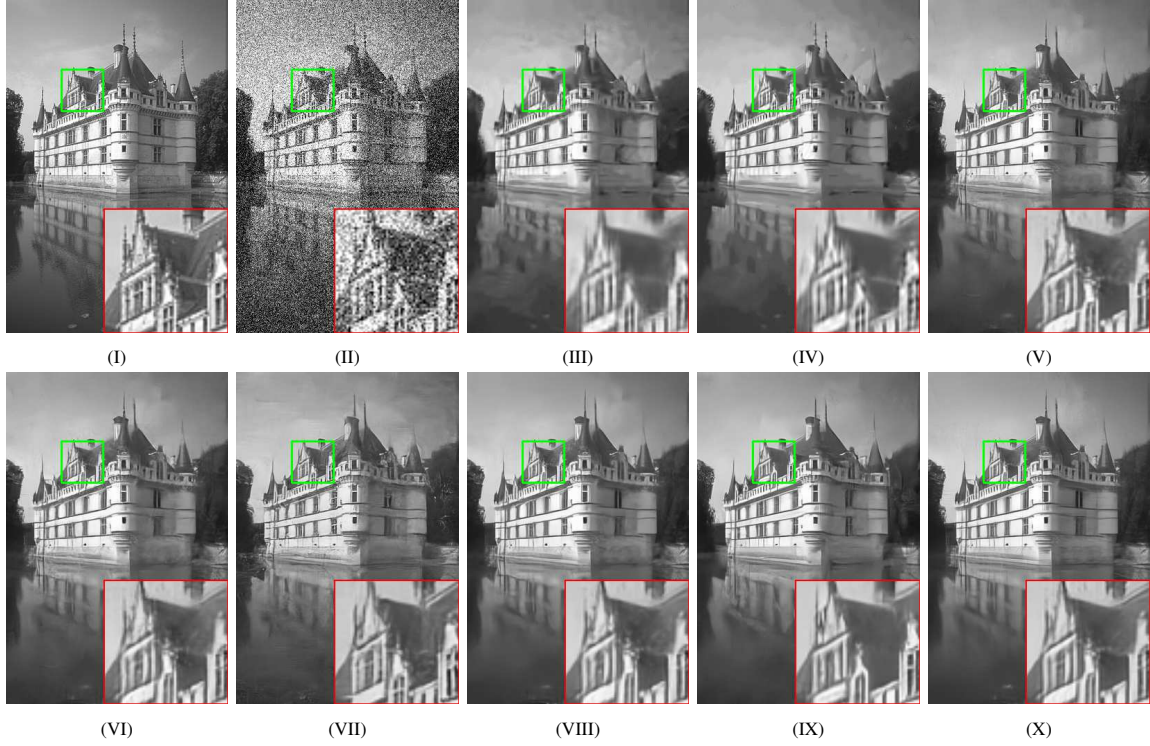
Figure 4: Visual comparison results on the "test003" image from the BSD68 dataset at the noise level 50. (I) Ground-truth image, (II) Noisy image / 14.15dB, (III) BM3D / 26.21dB, (IV) WNNM / 26.50dB, (V) DnCNN-S / 26.90dB, (VI) IRCNN / 26.85dB, (VII) BUIFD / 26.32dB, (VIII) FFDNet / 26.92dB, (IX) ADNet / 27.06dB, (X) DCANet / 27.13dB.

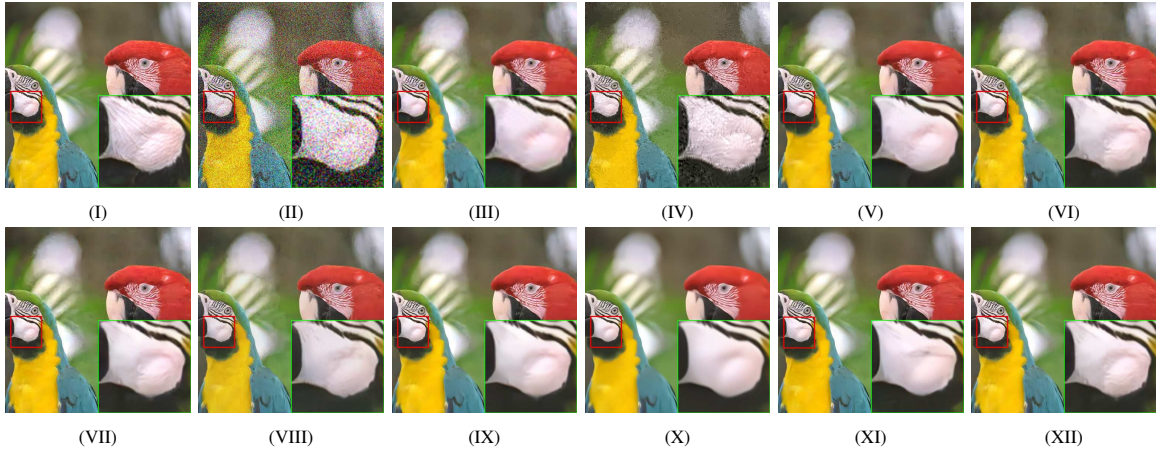

Figure 5: Visual comparison results on the "kodim23" image from the Kodak24 dataset at the noise level 50. (I) Ground-truth image, (II) Noisy image / 14.16dB, (III) CBM3D / 30.95dB, (IV) MCWNNM / 25.25dB, (V) CDnCNN-S / 31.47dB, (VI) CDnCNN-B / 31.12dB, (VII) IRCNN / 31.16dB, (VIII) BUIFD / 29.41dB, (IX) FFDNet / 31.50dB, (X) ADNet / 31.47dB, (XI) AirNet / 31.50dB, (XII) DCANet / 31.86dB.

11

Table 6: Quantitative comparison results on three commonly used color datasets. The three best results are respectively emphasized in red, blue and green.

| Datasets | Models | $\sigma$=15 | $\sigma$=25 | $\sigma$=35 | $\sigma$=50 | $\sigma$=75 |
|---|---|---|---|---|---|---|
| CBSD68 | CBM3D [2] | 33.52 | 30.71 | 28.89 | 27.38 | 25.74 |
| | MCWNNM [5] | 30.91 | 27.61 | 25.88 | 23.18 | 21.21 |
| | CDnCNN-S [6] | 33.89 | 31.23 | 29.58 | 27.92 | 24.47 |
| | IRCNN [16] | 33.86 | 31.16 | 29.50 | 27.86 | - |
| | BUIFD [10] | 33.65 | 30.76 | 28.82 | 26.61 | 23.64 |
| | FFDNet [8] | 33.87 | 31.21 | 29.58 | 27.96 | 26.24 |
| | DSNetB [17] | 33.91 | 31.28 | - | 28.05 | - |
| | RIDNet [14] | 34.01 | 31.37 | - | 28.14 | - |
| | BRDNet [12] | 34.10 | 31.43 | 29.77 | 28.16 | 26.43 |
| | ADNet [15] | 33.99 | 31.31 | 29.66 | 28.04 | 26.33 |
| | DudeNet [32] | 34.01 | 31.34 | 29.71 | 28.09 | 26.40 |
| | AirNet [11] | 33.92 | 31.26 | - | 28.01 | - |
| | DCANet | 34.05 | 31.45 | 29.86 | 28.28 | 26.61 |
| Kodak24 | CBM3D [2] | 34.28 | 31.68 | 29.90 | 28.46 | 26.82 |
| | MCWNNM [5] | 32.00 | 28.76 | 27.02 | 21.18 | 18.06 |
| | CDnCNN-S [6] | 34.48 | 32.03 | 30.46 | 28.85 | 25.04 |
| | IRCNN [16] | 34.56 | 32.03 | 30.43 | 28.81 | - |
| | BUIFD [10] | 34.41 | 31.77 | 29.94 | 27.74 | 24.67 |
| | FFDNet [8] | 34.63 | 32.13 | 30.57 | 28.98 | 27.27 |
| | DSNetB [17] | 34.63 | 32.16 | - | 29.05 | - |
| | BRDNet [12] | 34.88 | 32.41 | 30.80 | 29.22 | 27.49 |
| | ADNet [15] | 34.76 | 32.26 | 30.68 | 29.10 | 27.40 |
| | DudeNet [32] | 34.81 | 32.26 | 30.69 | 29.10 | 27.39 |
| | AirNet [11] | 34.68 | 32.21 | - | 29.06 | - |
| | DCANet | 34.85 | 32.43 | 30.91 | 29.36 | 27.69 |
| McMaster | CBM3D [2] | 34.06 | 31.66 | 29.92 | 28.51 | 26.79 |
| | MCWNNM [5] | 32.75 | 29.39 | 27.44 | 21.37 | 18.16 |
| | CDnCNN-S [6] | 33.44 | 31.51 | 30.14 | 28.61 | 25.10 |
| | IRCNN [16] | 34.58 | 32.18 | 30.59 | 28.91 | - |
| | BUIFD [10] | 33.84 | 31.06 | 28.87 | 26.20 | 22.75 |
| | FFDNet [8] | 34.66 | 32.35 | 30.81 | 29.18 | 27.33 |
| | DSNetB [17] | 34.67 | 32.40 | - | 29.28 | - |
| | BRDNet [12] | 35.08 | 32.75 | 31.15 | 29.52 | 27.72 |
| | ADNet [15] | 34.93 | 32.56 | 31.00 | 29.36 | 27.53 |
| | AirNet [11] | 34.70 | 32.44 | - | 29.26 | - |
| | DCANet | 34.84 | 32.62 | 31.16 | 29.59 | 27.81 |

Table 7: Quantitative comparison results on three commonly used color datasets. The three best results are respectively emphasized in red, blue and green.

| Datasets | Models | $\sigma=15$ | $\sigma=25$ | $\sigma=50$ |
|---|---|---|---|---|
| CBSD68 | CDnCNN-B [6] | 0.929 | 0.883 | 0.790 |
| | IRCNN [16] | 0.929 | 0.882 | 0.790 |
| | BUIFD [10] | 0.930 | 0.882 | 0.777 |
| | FFDNet [8] | 0.929 | 0.882 | 0.789 |
| | ADNet [15] | 0.933 | 0.889 | 0.797 |
| | AirNet [11] | 0.933 | 0.888 | 0.798 |
| | DCANet | 0.932 | 0.888 | 0.804 |
| Kodak24 | CDnCNN-B [6] | 0.920 | 0.876 | 0.791 |
| | IRCNN [16] | 0.920 | 0.877 | 0.793 |
| | BUIFD [10] | 0.923 | 0.879 | 0.786 |
| | FFDNet [8] | 0.922 | 0.878 | 0.794 |
| | ADNet [15] | 0.924 | 0.882 | 0.798 |
| | AirNet [11] | 0.924 | 0.882 | 0.799 |
| | DCANet | 0.925 | 0.885 | 0.811 |
| McMaster | CDnCNN-B [6] | 0.904 | 0.869 | 0.799 |
| | IRCNN [16] | 0.920 | 0.882 | 0.807 |
| | BUIFD [10] | 0.901 | 0.847 | 0.733 |
| | FFDNet [8] | 0.922 | 0.886 | 0.815 |
| | ADNet [15] | 0.927 | 0.894 | 0.825 |
| | AirNet [11] | 0.925 | 0.891 | 0.822 |
| | DCANet | 0.924 | 0.893 | 0.831 |

Table 8: Quantitative comparison results on the SIDD validation set and the DND sRGB images. The three best results are respectively emphasized in red, blue and green.

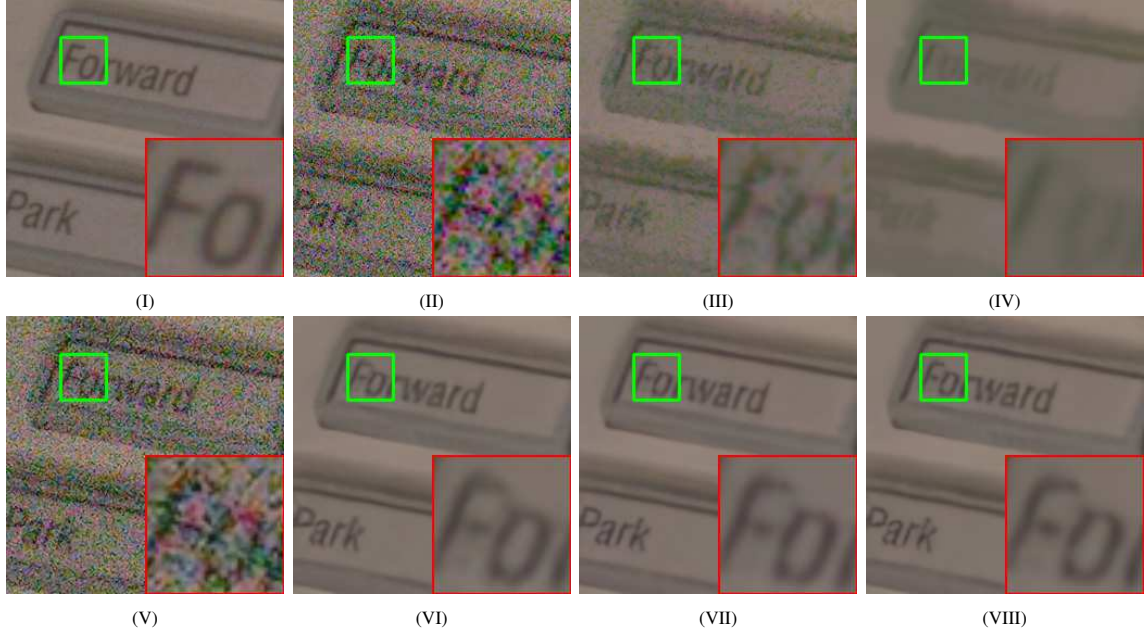| Dataset | Metrics | MCWNNM [5] | TWSC [3] | DnCNN-B [6] | CBDNet [9] | RIDNet [14] | AINDNet [20] | VDN [19] | DCANet |
|---|---|---|---|---|---|---|---|---|---|
| SIDD | PSNR | 33.40 | 35.33 | 23.66 | 30.78 | 38.71 | 38.95 | 39.28 | 39.27 |
| | SSIM | 0.879 | 0.933 | 0.583 | 0.951 | 0.954 | 0.952 | 0.957 | 0.956 |
| DND | PSNR | 37.38 | 37.94 | 37.90 | 38.06 | 39.23 | 39.37 | 39.38 | 39.57 |
| | SSIM | 0.929 | 0.940 | 0.943 | 0.942 | 0.953 | 0.951 | 0.952 | 0.953 |

Figure 6: Visual comparison results on the "11_4" image from the SIDD validation set. (I) Ground-truth image / (PSNR/SSIM), (II) Noisy image / (18.25/0.169), (III) MCWNNM / (28.63/0.725), (IV) TWSC / (30.42/0.837), (V) CDnCNN-B / (20.76/0.231), (VI) AINDNet / (36.24/0.904), (VII) VDN / (36.39/0.907), (VIII) DCANet / (36.36/0.912).

Three randomly chosen grayscale and color noisy images of different sizes were used to evaluate the runtime of different denoising models. The noise level was set to 25. We obtained the runtime of every tested method on each image by averaging the time of 20 implementations. It should be noted that our experiments neglected the memory transfer time between the CPU and GPU. Table 9 lists the runtime of the tested denoising methods. It can be found that our DCANet has a fast speed for both grayscale and color images.

Table 9: Runtime (in seconds) comparison of different denoising models on different images.

| Devices | Models | $256 \times 256$ | | $512 \times 512$ | | $1024 \times 1024$ | |
| | | Gray | Color | Gray | Color | Gray | Color |
| --- | --- | --- | --- | --- | --- | --- | --- |
| CPU | BM3D [2] | 0.458 | 0.593 | 2.354 | 3.771 | 9.782 | 12.818 |
| | WNNM [4] | 63.867 | - | 277.003 | - | 1150.842 | - |
| | MCWNNM [5] | - | 62.777 | - | 277.623 | - | 1120.112 |
| GPU | DnCNN-B [6] | 0.032 | 0.032 | 0.037 | 0.037 | 0.057 | 0.057 |
| | IRCNN [16] | 0.030 | 0.030 | 0.030 | 0.030 | 0.030 | 0.030 |
| | BUIFD [10] | 0.035 | 0.037 | 0.050 | 0.053 | 0.112 | 0.123 |
| | FFDNet [8] | 0.031 | 0.030 | 0.031 | 0.030 | 0.032 | 0.030 |
| | AINDNet [20] | - | 0.531 | - | 2.329 | - | 9.573 |
| | VDN [19] | 0.144 | 0.162 | 0.607 | 0.597 | 2.367 | 2.376 |
| | ADNet [15] | 0.031 | 0.033 | 0.035 | 0.045 | 0.051 | 0.093 |
| | AirNet [11] | - | 0.143 | - | 0.498 | - | 2.501 |
| | DCANet | 0.058 | 0.059 | 0.099 | 0.104 | 0.252 | 0.258 |

Table 10 reports the number of parameters and FLOPs of different models on grayscale and color images, respectively. It can be observed from Table 10 that some state-of-the-art image blind denoising methods have large numbers of parameters, these models are usually equipped with complex model structures, however the proposed DCANet can obtain competitive or even better performance. For those models have the similar number of parameters, like the BUIFD, BRDNet, DudeNet, and RIDNet, our model achieve better results in most of the tested denoising tasks,

especially in color image denoising and real noise removal. Namely, our model obtains a favorable tradeoff between network complexity and denoising performance.

Table 10: The number of parameters (in K) and FLOPs (in G) of different models.

| Models | Model Parameters | | FLOPs | |
| --- | --- | --- | --- | --- |
| | Gray | Color | $256 \times 256 \times 1$ | $256 \times 256 \times 3$ |
| DnCNN-B [6] | 666 | 668 | 21.93 | 22.12 |
| IRCNN [16] | 186 | 188 | 6.08 | 6.15 |
| BUIFD [10] | 1186 | 1196 | 35.31 | 35.65 |
| FFDNet [8] | 485 | 852 | 3.97 | 3.14 |
| BRDNet [12] | 1113 | 1117 | 36.48 | 36.63 |
| CBDNet [9] | - | 4365 | - | 20.14 |
| RIDNet [14] | 1497 | 1499 | 48.90 | 48.98 |
| VDN [19] | 7810 | 7817 | 24.47 | 24.70 |
| ADNet [15] | 519 | 521 | 17.08 | 17.15 |
| AINDNet [20] | - | 13764 | - | - |
| DudeNet [32] | 1077 | 1079 | 35.35 | 35.43 |
| AirNet [11] | - | 8930 | - | 150.64 |
| DCANet | 1382 | 1389 | 37.65 | 37.88 |

## 5. Conclusion

In this paper, we propose a dual convolutional neural network with attention (DCANet) for image blind denoising. The proposed DCANet is composed of a noise estimation network, an attention module, and a dual convolutional denoising network. The noise estimation network is applied to estimate the noise in a noisy image. The attention module consists of spatial and channel attention block, and is utilized to filter unimportant information. The dual convolutional denoising network contains two different branches, which not only widens the network to improve its learning ability, but also can capture the complementary image features to enhance the denoising effect. To the best of our knowledge, the combination of the dual CNN and the attention mechanism for image blind denoising has not been investigated before.

Compared with the state-of-the-art denoising models, the proposed DCANet obtains competitive denoising performance. Our proposed DCANet also obtains a favorable trade-off between model complexity and denoising ability, therefore the model can be an option for the practical image denoising tasks. In the future, we will further investigate the noise estimation network to obtain more accurate noise estimation. The application of the DCANet on other low-level visual tasks such as image deraining and super-resolution will also be addressed.

## 6. Acknowledgements

## References

[1] A. Buades, B. Coll, J. Morel, A non-local algorithm for image denoising, IEEE Conference on Computer Vision and Pattern Recognition (2005) 60–65.

[2] K. Dabov, A. Foi, V. Katkovnik, K. O. Egiazarian, Image denoising by sparse 3-d transform-domain collaborative filtering, IEEE Trans. Image Process. 16 (8) (2007) 2080–2095.

[3] J. Xu, L. Zhang, D. Zhang, A trilateral weighted sparse coding scheme for real-world image denoising, European Conference on Computer Vision 11212 (2018) 21–38.

[4] S. Gu, L. Zhang, W. Zuo, X. Feng, Weighted nuclear norm minimization with application to image denoising, IEEE Conference on Computer Vision and Pattern Recognition (2014) 2862–2869.

[5] J. Xu, L. Zhang, D. Zhang, X. Feng, Multi-channel weighted nuclear norm minimization for real color image denoising, IEEE International Conference on Computer Vision (2017) 1105–1113.

[6] K. Zhang, W. Zuo, Y. Chen, D. Meng, L. Zhang, Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising, IEEE Trans. Image Process. 26 (7) (2017) 3142–3155.

[7] P. Chatterjee, P. Milanfar, Is denoising dead?, IEEE Transactions on Image Processing 19 (4) (2010) 895–911.

[8] K. Zhang, W. Zuo, L. Zhang, Ffdnet: Toward a fast and flexible solution for cnn-based image denoising, IEEE Trans. Image Process. 27 (9) (2018) 4608–4622.

[9] S. Guo, Z. Yan, K. Zhang, W. Zuo, L. Zhang, Toward convolutional blind denoising of real photographs, IEEE Conference on Computer Vision and Pattern Recognition (2019) 1712–1722.

[10] M. E. Helou, S. Süsstrunk, Blind universal bayesian image denoising with gaussian noise level learning, IEEE Trans. Image Process. 29 (2020) 4885–4897.

[11] B. Li, X. Liu, P. Hu, Z. Wu, J. Lv, X. Peng, All-in-one image restoration for unknown corruption, IEEE Conference on Computer Vision and Pattern Recognition (2022) 17431–17441.

[12] C. Tian, Y. Xu, W. Zuo, Image denoising using deep CNN with batch renormalization, Neural Networks 121 (2020) 461–473.

[13] P. Jinshan, L. Sifei, S. Deqing, Z. Jiawei, L. Yang, R. Jimmy, L. Zechao, T. Jinhui, L. Huchuan, T. Yu-Wing, Y. Ming-Hsuan, Learning dual convolutional neural networks for low-level vision, IEEE Conference on Computer Vision and Pattern Recognition (2018) 3070–3079.

[14] S. Anwar, N. Barnes, Real image denoising with feature attention, International Conference on Computer Vision (2019) 3155–3164.

[15] C. Tian, Y. Xu, Z. Li, W. Zuo, L. Fei, H. Liu, Attention-guided CNN for image denoising, Neural Networks 124 (2020) 117–129.

[16] K. Zhang, W. Zuo, S. Gu, L. Zhang, Learning deep CNN denoiser prior for image restoration, IEEE Conference on Computer Vision and Pattern Recognition (2017) 2808–2817.

[17] Y. Peng, L. Zhang, S. Liu, X. Wu, Y. Zhang, X. Wang, Dilated residual networks with symmetric skip connection for image denoising, Neurocomputing 345 (2019) 67–76.

[18] J. W. Soh, N. I. Cho, Deep universal blind image denoising, International Conference on Pattern Recognition (2020) 747–754.

[19] Z. Yue, H. Yong, Q. Zhao, D. Meng, L. Zhang, Variational denoising network: Toward blind noise modeling and removal, Advances in Neural Information Processing Systems (2019) 1688–1699.

[20] Y. Kim, J. W. Soh, G. Y. Park, N. I. Cho, Transfer learning from synthetic to real-noise denoising with adaptive instance normalization, IEEE Conference on Computer Vision and Pattern Recognition (2020) 3479–3489.

[21] B. Du, Q. Wei, R. Liu, An improved quantum-behaved particle swarm optimization for endmember extraction, IEEE Trans. Geosci. Remote. Sens. 57 (8) (2019) 6003–6017.

[22] J. Li, G. Lu, B. Zhang, J. You, D. Zhang, Shared linear encoder-based multikernel gaussian process latent variable model for visual classification, IEEE Trans. Cybern. 51 (2) (2021) 534–547.

[23] X. Liang, D. Zhang, G. Lu, Z. Guo, N. Luo, A novel multicamera system for high-speed touchless palm recognition, IEEE Trans. Syst. Man Cybern. Syst. 51 (3) (2021) 1534–1548.

[24] Y. Li, X. Chen, Z. Zhu, L. Xie, G. Huang, D. Du, X. Wang, Attention-guided unified network for panoptic segmentation, IEEE Conference on Computer Vision and Pattern Recognition (2019) 7026–7035.

[25] H. Larochelle, G. E. Hinton, Learning to combine foveal glimpses with a third-order boltzmann machine, Advances in Neural Information Processing Systems (2010) 1243–1251.

[26] G. Hou, Y. Yang, J. Xue, Residual dilated network with attention for image blind denoising, IEEE International Conference on Multimedia and Expo (2019) 248–253.

[27] S. W. Zamir, A. Arora, S. H. Khan, M. Hayat, F. S. Khan, M. Yang, L. Shao, Multi-stage progressive image restoration, IEEE Conference on Computer Vision and Pattern Recognition (2021) 14821–14831.

[28] Y. Zhang, K. Li, K. Li, B. Zhong, Y. Fu, Residual non-local attention networks for image restoration, International Conference on Learning Representations.

[29] Y. Wang, X. Song, K. Chen, Channel and space attention neural network for image denoising, IEEE Signal Process. Lett. 28 (2021) 424–428.

[30] C. Ren, X. He, C. Wang, Z. Zhao, Adaptive consistency prior based deep network for image denoising, IEEE Conference on Computer Vision and Pattern Recognition (2021) 8596–8606.

[31] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, Y. Fu, Residual dense network for image super-resolution, IEEE Conference on Computer Vision and Pattern Recognition (2018) 2472–2481.

[32] C. Tian, Y. Xu, W. Zuo, B. Du, C. Lin, D. Zhang, Designing and training of a dual CNN for image denoising, Knowl. Based Syst. 226 (2021) 106949.

[33] J. Pan, D. Sun, J. Zhang, J. Tang, J. Yang, Y. Tai, M. Yang, Dual convolutional neural networks for low-level vision, Int. J. Comput. Vis. 130 (6) (2022) 1440–1458.

[34] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, Advances in Neural Information Processing Systems (2012) 1106–1114.

[35] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, International Conference on Machine Learning 37 (2015) 448–456.

[36] W. Malfliet, W. Hereman, The tanh method: I. exact solutions of nonlinear evolution and wave equations, Physica Scripta 54 (6) (1996) 563.

[37] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, IEEE International Conference on Computer Vision (2015) 1026–1034.

[38] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, IEEE Conference on Computer Vision and Pattern Recognition (2018) 7132–7141.

[39] S. Woo, J. Park, J. Lee, I. S. Kweon, CBAM: convolutional block attention module, European Conference on Computer Vision 11211 (2018) 3–19.

[40] J. Han, C. Moraga, The influence of the sigmoid function parameters on the speed of backpropagation learning, International Workshop on Artificial Neural Networks 930 (1995) 195–201.

[41] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions,

IEEE Conference on Computer Vision and Pattern Recognition (2015) 1–9.

[42] M. Ranzato, Y. Boureau, Y. LeCun, Sparse feature learning for deep belief networks, Advances in Neural Information Processing Systems (2007) 1185–1192.

[43] W. Huang, Y. Xue, L. Hu, H. Liuli, S-eegnet: Electroencephalogram signal classification based on a separable convolution neural network with bilinear interpolation, IEEE Access 8 (2020) 131636–131646.

[44] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, IEEE Conference on Computer Vision and Pattern Recognition (2016) 770–778.

[45] G. Huang, Z. Liu, L. van der Maaten, K. Q. Weinberger, Densely connected convolutional networks, IEEE Conference on Computer Vision and Pattern Recognition (2017) 2261–2269.

[46] F. Yu, V. Koltun, Multi-scale context aggregation by dilated convolutions, International Conference on Learning Representations.

[47] J. Yu, X. Yang, F. Gao, D. Tao, Deep multimodal distance metric learning using click constraints for image ranking, IEEE Trans. Cybern. 47 (12) (2017) 4014–4024.

[48] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, G. W. Cottrell, Understanding convolution for semantic segmentation, IEEE Winter Conference on Applications of Computer Vision (2018) 1451–1460.

[49] W. Yang, J. Liu, S. Yang, Z. Guo, Scale-free single image deraining via visibility-enhanced recurrent wavelet learning, IEEE Trans. Image Process. 28 (6) (2019) 2948–2961.

[50] H. Zhang, V. M. Patel, Density-aware single image de-raining using a multi-stream dense network, IEEE Conference on Computer Vision and Pattern Recognition.

[51] W. Lai, J. Huang, N. Ahuja, M. Yang, Deep laplacian pyramid networks for fast and accurate super-resolution, IEEE Conference on Computer Vision and Pattern Recognition (2017) 5835–5843.

[52] K. Jiang, Z. Wang, P. Yi, C. Chen, B. Huang, Y. Luo, J. Ma, J. Jiang, Multi-scale progressive fusion network for single image deraining, IEEE Conference on Computer Vision and Pattern Recognition (2020) 8343–8352.

[53] B. Kamgar-Parsi, A. Rosenfeld, Optimally isotropic laplacian operator, IEEE Trans. Image Process. 8 (10) (1999) 1467–1472.

[54] E. Agustsson, R. Timofte, NTIRE 2017 challenge on single image super-resolution: Dataset and study, IEEE Conference on Computer Vision and Pattern Recognition Workshops (2017) 1122–1131.

[55] S. Roth, M. J. Black, Fields of experts: A framework for learning image priors, IEEE Computer Society Conference on Computer Vision and Pattern (2005) 860–867.

[56] R. Franzen, Kodak lossless true color image suite, `http://r0k.us/graphics/kodak/` (1999).

[57] L. Zhang, X. Wu, A. Buades, X. Li, Color demosaicking by local directional interpolation and nonlocal adaptive thresholding, Journal of Electronic Imaging 20 (2) (2011) 1–17.

[58] A. Abdelhamed, S. Lin, M. S. Brown, A high-quality denoising dataset for smartphone cameras, IEEE Conference on Computer Vision and Pattern Recognition (2018) 1692–1700.

[59] J. Anaya, A. Barbu, RENOIR - A dataset for real low-light image noise reduction, J. Vis. Commun. Image Represent. 51 (2018) 144–154.

[60] T. Plotz, S. Roth, Benchmarking denoising algorithms with real photographs, IEEE Conference on Computer Vision and Pattern Recognition.

[61] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, International Conference on Learning Representations.

[62] I. Loshchilov, F. Hutter, SGDR: stochastic gradient descent with warm restarts, International Conference on Learning Representations.

[63] Y. Chen, T. Pock, Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration, IEEE Trans. Pattern Anal. Mach. Intell. 39 (6) (2017) 1256–1272.