CAAI Transactions on Intelligence Technology

Special issue Call for Papers

**Be Seen. Be Cited.
Submit your work to a new IET special issue**

Connect with researchers and experts in your field and share knowledge.

Be part of the latest research trends, faster.

Read more

IET The Institution of Engineering and Technology

CAAI Transactions on Intelligence Technology

ORIGINAL RESEARCH

A robust deformed convolutional neural network (CNN) for image denoising

Qi Zhang[1] | Jingyu Xiao[2] | Chunwei Tian[3,4,5] (iD) | Jerry Chun-Wei Lin[6] |
Shichao Zhang[2]

[1]School of Economics and Management, Harbin Institute of Technology at Weihai, Weihai, China

[2]School of Computer Science, Central South University, Changsha, China

[3]School of Software, Northwestern Polytechnical University, Xi'an, Shaanxi, China

[4]Research & Development Institute, Northwestern Polytechnical University, Shenzhen, China

[5]Yangtze River Delta Research Institute, Northwestern Polytechnical University, Taicang, China

[6]Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, Bergen, Norway

Correspondence

Chunwei Tian, School of Software, Northwestern Polytechnical University, Xi'an, 710129, SN, China.; Research & Development Institute, Northwestern Polytechnical University, Shenzhen, 518057, China.; Yangtze River Delta Research Institute, Northwestern Polytechnical University, Taicang, 21540, China.
Email: chunweitian@163.com

Funding information

Guangdong Basic and Applied Basic Research Foundation, Grant/Award Number: 2021A1515110079; Fundamental Research Funds for the Central Universities, Grant/Award Number: D5000210966; Basic Research Plan in Taicang, Grant/Award Number: TC2021JC23; Key Project of NSFC, Grant/Award Number: 61836016

Abstract

Due to strong learning ability, convolutional neural networks (CNNs) have been developed in image denoising. However, convolutional operations may change original distributions of noise in corrupted images, which may increase training difficulty in image denoising. Using relations of surrounding pixels can effectively resolve this problem. Inspired by that, we propose a robust deformed denoising CNN (RDDCNN) in this paper. The proposed RDDCNN contains three blocks: a deformable block (DB), an enhanced block (EB) and a residual block (RB). The DB can extract more representative noise features via a deformable learnable kernel and stacked convolutional architecture, according to relations of surrounding pixels. The EB can facilitate contextual interaction through a dilated convolution and a novel combination of convolutional layers, batch normalisation (BN) and ReLU, which can enhance the learning ability of the proposed RDDCNN. To address long-term dependency problem, the RB is used to enhance the memory ability of shallow layer on deep layers and construct a clean image. Besides, we implement a blind denoising model. Experimental results demonstrate that our denoising model outperforms popular denoising methods in terms of qualitative and quantitative analysis. Codes can be obtained at https://github.com/hellloxiaotian/RDDCNN.

KEYWORDS

Blind denoising, CNN, Deformed block, Enhanced block

1 | INTRODUCTION

Due to complex collection processing (i.e., capturing, transmitting and compressing images) via digital devices, obtained images may be corrupted by light conditions and sensors in noise manner [1]. To deal with this problem, image denoising techniques were presented. Specifically, image denoising techniques aim to remove the noise to store the given clean image, according to a degradation model of $y = c + n$, where c denotes a potential clean image, n is noise and y expresses a noisy image [2]. Inspired by that, filter methods have been widely used to suppress the noise of a noisy image [3]. Spatial domain filters can average pixels of selected areas to improve the quality of a predicted image [4]. Also, the simultaneous use

of a compensation function and bilateral filtering can enhance the edges to obtain more detail information so that improve denoising efficiency [5]. Besides, transforming images into transform domain was applied to eliminate noises according to their characteristics [6, 7]. The block-matching 3D (BM3D) exploited the judgement of patch comparability to achieve transform domain filtering in image denoising [8]. Specifically, referred pipeline can be obtained via the following three steps: grouping similar patches, shrinking of 3D transform domain for collaborative filtering, and obtaining original positions through aggregating these obtained patches. Due to excellent performance of BM3D in image denoising, its varieties were also developed for image denoising [9]. Alternatively, perception ideas were used to extract noise information from given noisy images [10]. Total variation [11] is introduced into image denoising and minimising total variation of an image is proved to be effective for image denoising tasks. Besides, partial differential equations (PDE) algorithm treated a denoising process as a solving equation through multiple iterations, where the referred solution is close to a latent clean image [12]. Due to attributes of the PDE, its variations inherited merits of maintaining both edge and texture information in image denoising [13]. Besides, a weighted nuclear norm minimisation (WNNM) used low rank clustering correlation idea to filter the noise for restoring high-quality images [14]. There are other popular denoising methods, such as Markov random field [15] and gradient methods [16].

Although the mentioned methods have obtained excellent performance in image denoising, they were faced with the following cons. (1) They need complex optimization methods to obtain optimal solutions for image denoising, which need more computational cost. (2) These method with one noise level can deal with noisy images of certain situations, which may be limited by applications in the real world.

To resolve the referred problems, deep learning techniques are developed. Due to strong learning ability and fast execution speed, deep learning techniques, especially convolutional neural networks are widely applied in image denoising [17]. For instance, Zhang et al. [18] presented a denoiser by the combination of some plug-ins, that is, batch normalisation (BN) [19] and residual learning (RL) [20]. The BN can accelerate training efficiency of denoising network by normalising data. Also, the RL can be used to construct a latent clean image. To reduce the training cost, a fast and flexible denoising network (FFDNet) used a varying tunable noise level map and noisy image in a CNN to quickly train a denoising model [21]. To address long-term dependency problem of deep network, residual learning and concatenation operations are widely applied in image restoration [22]. Using a recursive and unit and a gate unit can mine multi-level features to enhance the memory ability of a deep network for image restoration [23]. Although these methods can obtain excellent performance for image denoising, they may suffer from risk of training difficulty from changed distributions of training data via convolutional operations.

In this paper, we propose a robust deformed denoising CNN (RDDCNN) to deal with the mentioned problem, according to relations of surrounding pixels. The proposed RDDCNN contains three blocks: a deformable block (DB), an enhanced block (EB) and a residual block (RB). The DB can extract more representative noise features via a deformable learnable kernel, according to relation of surrounding pixels. The EB can efficiently facilitate contextual interaction through some common components, that is, a dilated convolution [24] and a novel combination of a single BN and ReLU, which can enhance the learning ability of the proposed RDDCNN. To address long-term dependency problem, the RB is used to enhance the memory ability of shallow layer on deep layers and construct a clean image. Besides, the RDDCNN can deal with a blind denoising.

Our main contributions can be summarised as follows.

(1) A deformable convolution with a deformable learnable kernel is fused in a CNN to address offset pixels of feature mapping from a noisy image by convolution operations, according to relations of surrounding pixels.
(2) A novel architecture is designed via some common components, that is, a dilated convolution and a novel of combination of a single BN and ReLU to efficiently facilitate contextual interaction for improving the denoising performance.
(3) The proposed denoiser can deal with blind denoising.

Remainder of this paper is arranged as follows. Section 2 illustrates related work of the proposed method. Section 3 reveals the main network architecture of the proposed RDDCNN. Section 4 gives experiments. Section 5 is conclusion of this paper.

2 | RELATED WORK

The section provides basis of the proposed method, including CNNs for image denoising and CNNs based flexible kernels in image applications.

2.1 | CNNs for image denoising

Due to strong expressive abilities, CNNs have obtained remarked results in low-level vision tasks, such as image denoising [17]. These methods can be divided into two kinds in general, improving the network architecture and fusing new components into CNNs.

In terms of the first method, Zhang et al. [18] integrated batch normalisation and residual learning into a CNN to filter the noise. To reduce the computational cost, a fast and flexible convolutional neural network used a noise mapping and a noisy image patch to accelerate training speed for achieving blind denoising [21]. To deal with noisy images from complex screens, Tian et al. [25] used a dual network to extract complementary features to enhance the robustness of a denoiser. To extract salient features, Anwar et al. [26] merged a channel attention block to enhance the relationship of different

channels to improve the denoising effect. Besides, to remove noise of unknown distributions, Guo et al. [27] proposed a two-phased method for blind denoising. The first phase used a sub-network to estimate the noise. The second utilised method was used to learn a blind denoiser.

In terms of the second method, optimised methods embedded into a CNN are very popular for image denoising. For instance, Alawode and Alfarraj [28] combined a meta-optimiser and CNN to make a tradeoff between denoising performance and efficiency. Besides, utilising Bregman iteration algorithm to transfer a depth image inpainting into image denoising is very effective [29]. The mentioned methods based CNNs show that CNNs are very effective tools for image denoising. Inspired by that, we also employ a CNN to address image denoising problem in this paper.

2.2 | CNNs based flexible kernels for image applications

It is known that CNNs tended to improve the performance in image applications through increasing network depth or width [30]. However, that may cause training costs. To resolve this phenomenon, lightweight CNNs use smaller convolutions to reduce computational costs. For instance, Tian et al. [31] utilised heterogenous convolutions of 3×3 and 1×1 rather than 3×3 to extract representative features to improve pixels of predicted images. Although these lightweight CNNs enjoy fast processing fast in SR, they may reduce performance of image applications by limiting depth and width of deep networks, and sizes of convolutional kernels [32]. To handle this question, dynamic convolution techniques are presented [32]. This method fuses dynamically parallel convolutional kernels by an attention mechanism rather than a certain kernel to extract salient features. That is, an attention mechanism can dynamically adjust weight of each kernel to enhance expressive abilities of obtained features, according to different inputs [32]. Inspired by that, many variants of dynamic convolution methods have been applied in many image applications [33–35]. For instance, Wang et al. [36] used two dynamic parts to restore images. The first part used dynamically select spatial sampling locations to achieve a flexible extraction feature process. The second part utilised different normalisation methods to normalise observation features for image inpainting. Sun et al. [37] sampled different offset values to

dynamically adjust receptive field to address image segmentation task. Although these methods can make a tradeoff between performance and computational coats, they may have challenges in deformed objects.

Alternatively, deformed convolution techniques are proposed [38]. Chen et al. [39] used interdependency of different attention mechanisms to define deformable mechanism to improve the ability of single generalisation convolution in image segmentation. Yu et al. [40] proposed a deformable attention to enhance contextual information and interdependencies between target template and search image for object tracking. Inspired by these, a deformable convolution is used into a CNN to improve the clarity of predicted denoising images.

3 | THE PROPOSED METHOD

We train a deep denoising model RDDCNN via breaking the following rules: (1) designing a denoising architecture. (2) Training a denoising model according to deep learning knowledge. In this section, we only depict the whole architecture of the designed RDDCNN and implementational details of important components.

3.1 | Network architecture

The paper presents a robust deformed denoising CNN (RDDCNN), which is composed of a DB, an enhanced block (EB) and a RB as shown in Figure 1. Because convolutional operations may change original distributions of noise in corrupted images, which may increase the training difficulty in image denoising. The 12-layer DB is proposed to extract more representative noise features via a deformable and stacked convolutional architecture, according to relations of surrounding pixels. To mine more struct information at a less cost, a 5-layer EB can facilitate contextual interaction via a dilated convolution to extract more robust noisy information for enhancing learning ability of the proposed RDDCNN. To address long-term dependency problem, the RB is used to enhance the memory ability of a shallow layer on deep layers and construct a latent clean image. To visually express the process, we conduct the following Equation (1).
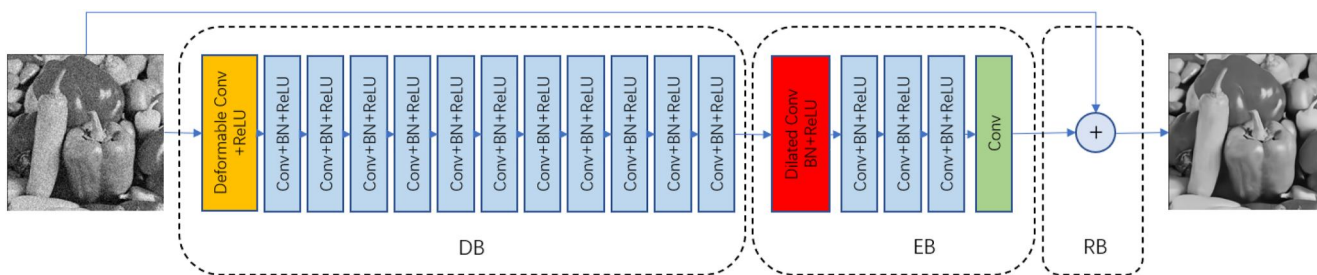

FIGURE 1 Architecture of RDDCNN

$$I_c = RDDCNN(I_n) = RB(EB(DB(I_n))) \qquad (1)$$

where I_n expresses a given noisy image. $RDDCNN$ denotes function of the RDDCNN. Also, DB, EB and RB are functions of DB, EB and RB, respectively. Also, I_c is symbolled as a predicted clean image. Besides, parameters of the RDDCNN model can be obtained by the following loss function.

3.2 | Loss function

To make obtained denoising model fairer and more robust, we choose public mean square error (MSE) [41] as a loss function to update parameters. The following loss function can be represented as

$$L(\theta) = {}^1\!/_{2N} \sum_{i=1}^{N} \left\| RDDCNN(I_n^i) - I_{gc}^i \right\|^2, \qquad (2)$$

where I_n^i and I_{gc}^i stand for the $i-th$ noisy image and given clean image, respectively. Also, N expresses the total of noisy images. Besides, obtained parameters can be optimised via Adam [42].

3.3 | Deformable block

The 12-layer deformable block utilises a deformable learnable kernel and stacked convolutions to obtain more representative noise features. That is the first layer is deformable Conv + ReLU, which denote a deformable convolution and an activation function of ReLU. A deformable convolution [38] uses relations of surrounding pixels to restore the position of the original pixel to enhance clarity of predicted image. Also, ReLU is used to convert obtained linear features into non-linear features. Input and output channels of this layer are 3 and 64. Specifically, if given images is grey, its input channel is 1. Also, its kernel size is 3×3. The 2nd layer - 11th layer are composed of Conv + BN + ReLU, which denotes the combination of a convolutional layer, BN and ReLU. Their input and output channels are 64. Also, their kernel sizes are 3×3. BN is used to normalise data to accelerate network speed. To visually express the mentioned process, we define some symbols as follows. DC and R are used to represent a deformable convolution and ReLU, respectively. C and B express functions of a convolution and BN, respectively. Also, RBC denotes a Conv + BN + ReLU. 11 RBC is used to stand for eleven stacked conv + BN + ReLU.

$$O_{DB} = DB(I_n) = 11RBC(R(DC(I_n))) \qquad (3)$$

where O_{DB} is the output of DB, which is followed by an enhanced block. Also, the mentioned deformable convolution [38] can be given as follows.

Obtaining more accurate contextual information can improve clarities of damaged images in image denoising [43]. Deformable convolution has strong geometric reshaping capability, which is used in our network for image denoising in this paper. It can learn from input and adjust position of each sampling point in a kernel rather than rigid rectangle kernel in a standard convolution. And its detailed information is as follows.

A standard convolution operation can be expressed as Equation (4).

$$Y(p_x, p_y) = \sum_{k=1}^{N} X(p_x, p_y, k) W(p_x + \Delta x_k, p_y + \Delta y), \quad (4)$$

where (p_x, p_y) represents location of centre point in a given kernel W. Its size is set to 3×3. Also, X denotes obtained feature map. N is total of pixels from obtained features in X. Besides, Δx_k denotes offset of p_x in horizontal and Δy_k stands for offset of p_y in vertical. Also, $\Delta x_k \in \{-1, 0, 1\}$ and $\Delta y_k \in \{-1, 0, 1\}$.

But for a deformable convolution with 3×3 kernel, it is as shown below:

$$\{(\Delta x_k, \Delta y_k)\} = \{(-1 + L_{x1}, -1 + L_{y1}), .., (0 + L_{x5}, 0 + L_{y5}), .., (1 + L_{x9}, 1 + L_{y9})\} \qquad (5)$$

where (L_{xk}, L_{yk}) are the learnt offset for the kth point.

According to analysis of mentioned formulae, we can see that deformable convolution has good performance to obtain more contextual information. Thus, we put a deformable convolutional layer in the front of the whole denoising as Figure 1, where its parameters are input channel of 3, output channel of 64, kernel size of 3×3. Subsequently, a ReLU can convert obtained linear features into non-linear features. To further learn accurate features, a 12-layer the combination of convolutional layer, BN and ReLU (i.e., Conv + BN + ReLU) are embedded in the DB, where their parameters are input channel and output channel of 64, kernel size of 3×3. BN is used to normalise obtained features and ReLU is exploited to convert obtained linear features into non-linearity. The output of the last Conv + BN + ReLU acts the EB.

3.4 | Enhanced block

It is known that deep architectures can extract more accurate to enhance performance of image applications. Also, contextual interaction can enhance the learning ability of deep network. Inspired by these, a 5-layer EB is designed. EB uses a dilated convolution to obtain more contextual information for enhancing SR performance. Also, three stacked convolutional layers are used to further learn these obtained features for

image denoising. More information of EB can be shown as follows.

The 1st layer is Dilated Conv + BN + ReLU, which represents the combination of a dilated convolution [24], BN and ReLU. Also, its input channel and output channel are 64. Also, convolutional kernel is 3×3. The 2nd, 3rd and 4th layers are composed of Conv + BN + ReLU, where their input and output channels are 64, and convolutional kernel is 3×3. The mentioned illustrations can be presented via Equation (6).

$$\begin{aligned} O_{EB} &= EB(O_{DB}) \\ &= C(RBC(RBC(RBC(R(B(DC(O_{DB}))))))) \end{aligned} \tag{6}$$

TABLE 1 Denoising results of different methods on BSD68 for noise level of 25

Methods	PSNR (dB)
RDDCNN without dilated convolution, deformable convolution and BN	29.219
RDDCNN without dilated convolution and Conv + BN + ReLU in DB	28.952
RDDCNN without convolution and deformable convolution	29.258
RDDCNN without dilated convolution and three Conv + BN + ReLU in EB	29.226
RDDCNN without dilated convolution	29.263
RDDCNN	29.270

TABLE 2 Comparisons of deformable convolution and common convolution

Metrics	RDDCNN	RDDCNN repacling deformable convolution with common convolution
Denoising result (PSNR/dB) on BSD68 for noise level of 25	29.270	29.262
Parameters	556,302	556,096
Running time on a grayscale image of 256×256	0.0119 s	0.0079 s

where DC denotes function of a dilated convolution. Also, C and B express functions of a convolution and BN, respectively. O_{EB} stands for output of EB, which is input of RB.

3.5 | Residual block

The residual block is used to construct a latent clean image. That includes a single convolutional layer, which is expressed as Conv in Figure 1. The residual block utilises a residual learning operation to remove the noise from the given image as follows.

$$I_c = RB(O_{EB}) = I_n - O_{EB} \tag{7}$$

where $-$ presents a residual operation, which is as well as $\oplus$ in Figure 1.

4 | EVALUATION

4.1 | Training and test datasets

Training datasets: To fairly evaluate denoising performance of the proposed method, we choose public dataset the same as a denoising CNN (DnCNN) [18] as a training dataset to train a grey synthetic noisy image denoiser. Specifically, the mentioned training dataset has 400 Gy images with sizes of 180×180. To further verify denoising effect of our model on devices in the real world, we choose real noisy images from the PolyU [44] to conduct experiments, where they contain 80 natural images with size of 2784×1856. To increase diversity of training samples, we use the following two kinds to solve this problem [45]. Firstly, we scale every training image via one of bicubic interpolations from 1, 0.9, 0.8 and 0.7 to increase the number of training samples. Secondly, we randomly choose one of eight modes (i.e., no manipulation, rotating by 90° counter clockwise, rotating by 180° counter clockwise, rotating by 270° counter clockwise, horizontal flip, rotating by 90° counter clockwise together with horizontal flip, rotating by 180° counter clockwise together with horizontal flip, and rotating by 270° counter clockwise together with horizontal flip) to deal with noisy training images for increasing diversities of noisy images.

TABLE 3 Peak Signal to Noise Ratio (PSNR) (dB) results of several networks on BSD68 for noise level of 15, 25, and 50

Methods	BM3D [8]	WNNM [14]	EPLL [52]	TNRD [51]	CSF [54]	MLP [53]
$\sigma = 15$	31.07	31.37	31.21	31.42	31.24	–
$\sigma = 25$	28.57	28.83	28.68	28.92	28.74	28.74
$\sigma = 50$	25.62	25.87	25.67	25.97	–	26.03
DnCNN [18]	**IRCNN [50]**	**ECNDNet [31]**	**FFDNet [21]**	**ADNet [43]**	**RDDCNN**	**RDDCNN-B**
31.72	31.63	31.71	31.63	31.74	31.76	31.62
29.23	29.15	29.22	29.19	29.25	29.27	29.16
26.23	26.19	26.23	26.29	26.29	26.30	26.23

Test datasets: we choose public BSD68 [46] and Set12 [47] as test datasets to verify grey noisy image denoising performance of our method. Also, CC [48] is used to evaluate the denoising performance of the proposed RDDCNN on real noisy image denoising.

4.2 | Implementation details

All the experiments are conducted on a GPU of Nvidia TiTAN XP, a CPU of Intel Xeon Gold 6140, a RAM with 51G. Also, CUDA is 10.1 and cuDNN is 7.0. All codes are

TABLE 4 Average Peak Signal to Noise Ratio (PSNR) (dB) results of different methods on Set12 with noise levels of 15, 25 and 50

Images	C.man	House	Peppers	Starfish	Monarch	Airplane	Parrot	Lena	Barbara	Boat	Man	Couple	Average
Noise level							15						
BM3D [8]	31.91	34.93	32.69	31.14	31.85	31.07	31.37	34.26	33.10	32.13	31.92	32.10	32.37
WNNM [14]	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
EPLL [52]	31.85	34.17	32.64	31.13	32.10	31.19	31.42	33.92	31.38	31.93	32.00	31.93	32.14
CSF [54]	31.95	34.39	32.85	31.55	32.33	31.33	31.37	34.06	31.92	32.01	32.08	31.98	32.32
TNRD [51]	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 [18]	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 [50]	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
FFDNet [21]	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
ECNDNet [31]	32.56	34.97	33.25	32.17	33.11	31.70	31.82	34.52	32.41	32.37	32.39	32.39	32.81
RDDCNN	32.61	35.01	33.31	32.13	33.13	31.67	31.93	34.57	32.62	32.42	32.38	32.46	32.85
RDDCNN-B	32.20	34.88	33.14	32.05	32.87	31.57	31.73	34.44	32.22	32.28	32.28	32.34	32.67
Noise level							25						
BM3D [8]	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 [14]	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
EPLL [52]	29.26	32.17	30.17	28.51	29.39	28.61	28.95	31.73	28.61	29.74	29.66	29.53	29.69
CSF [54]	29.48	32.39	30.32	28.80	29.62	28.72	28.90	31.79	29.03	29.76	29.71	29.53	29.84
TNRD [51]	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 [18]	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 [50]	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
FFDNet [21]	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
ECNDNet [31]	30.11	33.08	30.85	29.43	30.30	29.07	29.38	32.38	29.84	30.14	30.03	30.03	30.39
RDDCNN	30.20	33.13	30.82	29.38	30.36	29.05	29.53	32.40	30.03	30.19	30.05	30.10	30.44
RDDCNN-B	29.95	33.03	30.70	29.29	30.25	28.99	29.40	32.30	29.62	30.09	29.97	30.00	30.30
Noise level							50						
BM3D [8]	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 [14]	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
EPLL [52]	26.10	29.12	26.80	25.12	25.94	25.31	25.95	28.68	24.83	26.74	26.79	26.30	26.47
CSF [54]	26.37	29.64	26.68	25.43	26.26	25.56	26.12	29.32	25.24	27.03	27.06	26.67	26.78
TNRD [51]	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 [18]	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 [50]	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
FFDNet [21]	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
ECNDNet [31]	27.07	30.12	27.30	25.72	26.82	25.79	26.32	29.29	26.26	27.16	27.11	26.84	27.15
RDDCNN	27.16	30.21	27.38	25.72	26.84	25.88	26.53	29.32	26.36	27.23	27.22	26.88	27.23
RDDCNN-B	27.07	30.07	27.27	25.62	26.78	25.81	26.45	29.19	26.20	27.16	27.16	26.78	27.13

conducted by PyTorch of 1.1.0 and Python of 3.7.10. Besides, initial learning rate is 1e-3 and it reduces to 0.2 times each 30 epochs. Parameter epsilon and momentum are set to 1e-4 and 0.95, respectively.

4.3 | Network analysis

The proposed RDDCNN is composed of DB, EB and RB. Their rationality and validity are shown as follows.

DB: It is known that convolutional operations can extract effective information to enhance performance in image applications. However, convolutional operations may change original distributions of noise in given corrupted images, which may increase training difficulty for image denoising. To address this problem, we design a DB. DB uses a deformable Conv + ReLU as the first layer to restore distribution of noise, according to surrounding pixels. Also, ReLU is used to covert linear features to non-linearity. Its effectiveness is verified as shown Table 1, where 'RDDCNN without dilated convolution' outperforms 'RDDCNN without dilated convolution and deformable convolution' in Peak Signal to Noise Ratio (PSNR) [49]. To prevent overcorrection of deformable convolution, we use 11-layer combination of a convolutional layer, BN and ReLU in the DB to refine obtained features as shown in Figure 1. Its effectiveness is verified in Table 1. That is,
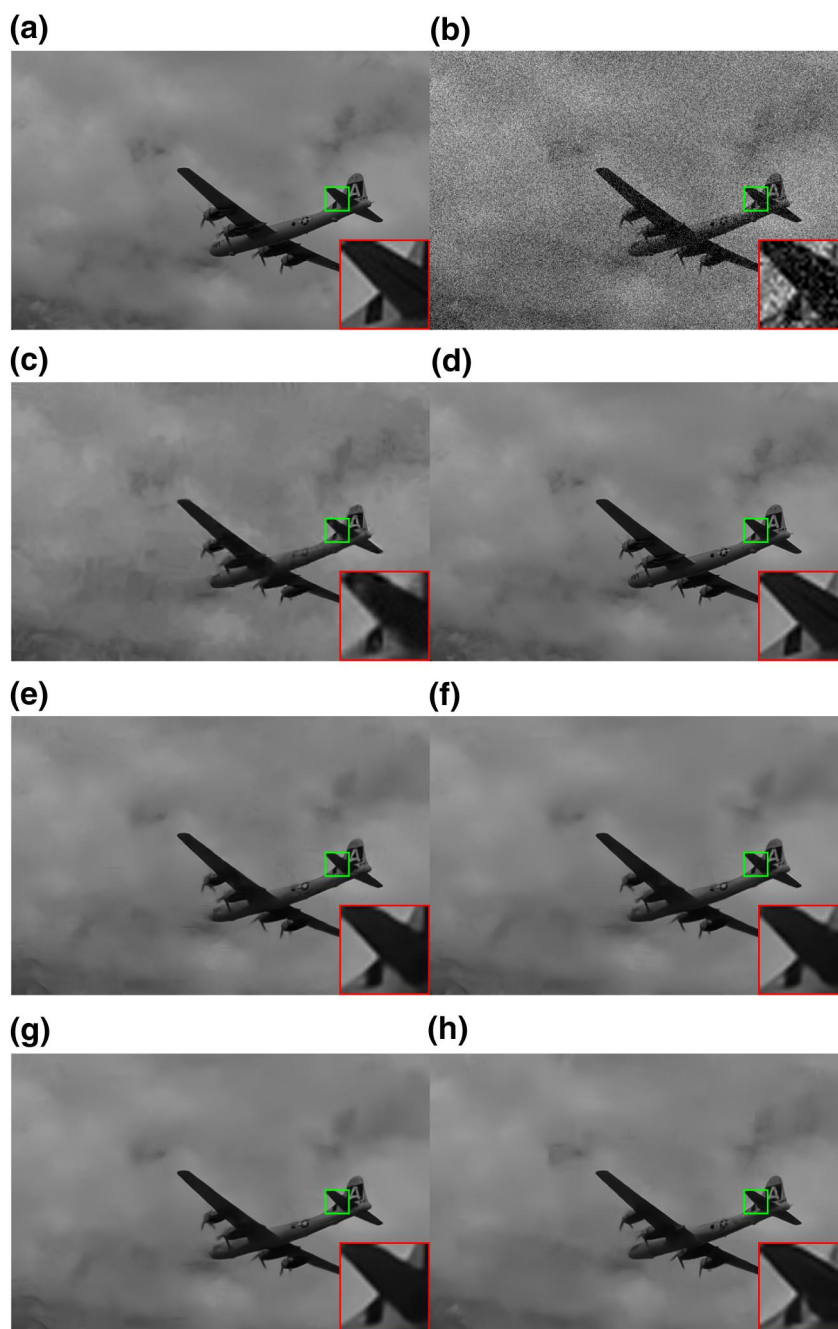

FIGURE 2 Denoising results of different methods on one image from BSD68 when noise level 25. (a) Original image (b) Noisy image/ 20.19 dB (c) BM3D [8]/36.59 dB (d) WNNM [14]/ 37.22 dB (e) IRCNN [50]/38.17 dB (f) FFDNet [21]/38.41 dB (g) DnCNN [18]/38.45 dB (h) RDDCNN/38.64 dB

'RDDCNN without dilated convolution' has higher PSNR result than that of 'RDDCNN without dilated convolution and Conv + BN + ReLU in DB'. Besides, the last Conv + BN + ReLU acts EB.

To further validate the superiority of the deformable convolution over common convolution, we provide a comprehensive comparison of the RDDCNN and RDDCNN replacing deformable convolution with common convolution. The deformable convolution with the flexible kernel can model the pixel relations and thus extract more salient features, which makes it effective for image denoising. This can be tested through the comparison of 'RDDCNN' and 'RDDCNN replacing deformable convolution with common convolution'

in PSNR in Table 2. Besides, the deformable convolution only brings about sufferable parameter amount and computational overhead. This can be seen in Table 2 through the comparisons of parameter amount and running time between 'RDDCNN' and 'RDDCNN replacing deformable convolution with common convolution'. Therefore, the deformable convolution is useful in image denoising.

EB: It is known that deep architectures can extract more accurate to enhance performance of image applications. Also, contextual interaction can enhance the learning ability of deep network. Inspired by these, a 5-layer EB is designed. EB uses a dilated convolution to obtain more contextual information for enhancing SR performance. Its effectiveness is tested via
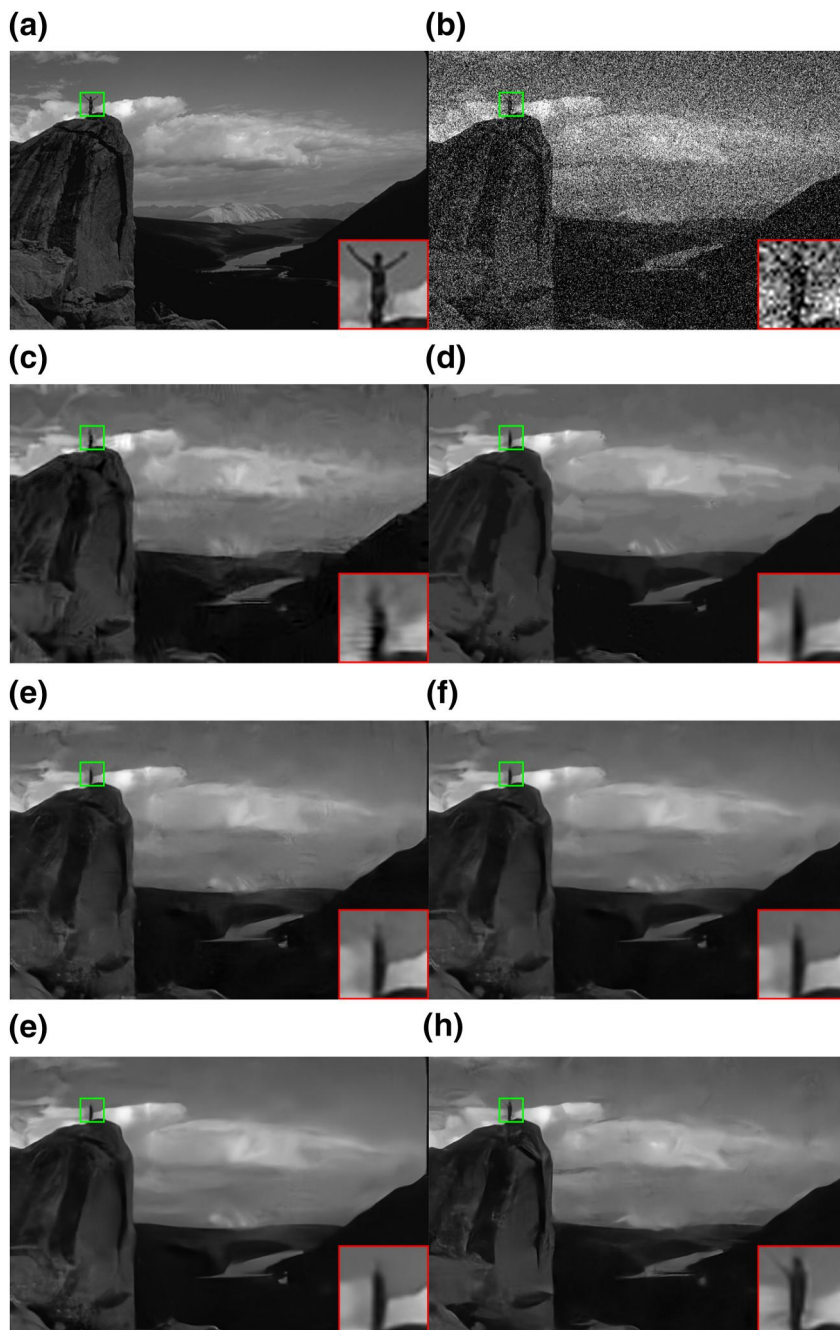

FIGURE 3 Denoising results of different methods on one image from BSD68 when noise level is 50. (a) Original image (b) Noisy image/ 14.66 dB (c) BM3D [8]/29.87 dB (d) WNNM [14]/ 30.07 dB (e) IRCNN [50]/30.33 dB (f) DnCNN [18]/30.48 dB (g) FFDNet [21]/30.56 dB (h) RDDCNN/30.67 dB

'RDDCNN' and 'RDDCNN without dilated convolution' in Table 1. To refine obtained features, three stacked convolutional layers are used to further learn these obtained features for image denoising. Specifically, the last convolutional layer is used to convert obtained features into noisy mapping images. Also, effectiveness of the front two Conv + BN + ReLU is proved via 'RDDCNN without dilated convolution' and 'RDDCNN without dilated convolution and three Conv + BN + ReLU in EB' as illustrated in Table 1.

RB: To construct a latent clean image, we use a residual operation to act output of EB and given noisy image to obtain a clean image as shown in Figure 1 and Section 3.4.

4.4 | Comparisons with the state-of-the-art denoising methods

To test denoising performance of RDDCNN, we conduct experiments on grey synthetic noisy images, real noisy images. Also, we use running time and complexity to test denoising efficiency. To verify good performance of RDDCNN for image denoising, we choose ten popular denoising methods as comparative methods, where they are BM3D [8] and weighted nuclear norm minimisation (WNNM) [14], DnCNN [18], FFDNet [21], enhanced convolutional neural denoising network (ECNDNet) [31], image restoration CNN (IRCNN) [50], one trainable nonlinear reaction diffusion (TNRD) [51], one generative algorithm expected patch log likelihood (EPLL) [52], multi-layer perceptron (MLP) [53] and cascade of shrinkage fields (CSF) [54]. In the synthetic noisy image and image blind denoising experiments as shown in Table 3 and Table 4, we test these

methods for three noise levels (i.e., 15, 25, and 50). In terms of testing denoising efficiency, we choose an image with three sizes (256×256, 512×512 and 1024×1024) to

TABLE 5 Complexity of different denoising methods

Methods	Parameters (G)	Flops (GFLOPs)
DnCNN [18]	0.56	0.891
ADNet [43]	0.52	0.832
DUBD [55]	2.09	–
RED30 [56]	4.13	10.33
ATDNet [57]	9.45	–
RDDCNN	0.56	0.891

TABLE 6 Running time (s) of different methods for 256×256, 512×512, and 1024×1024

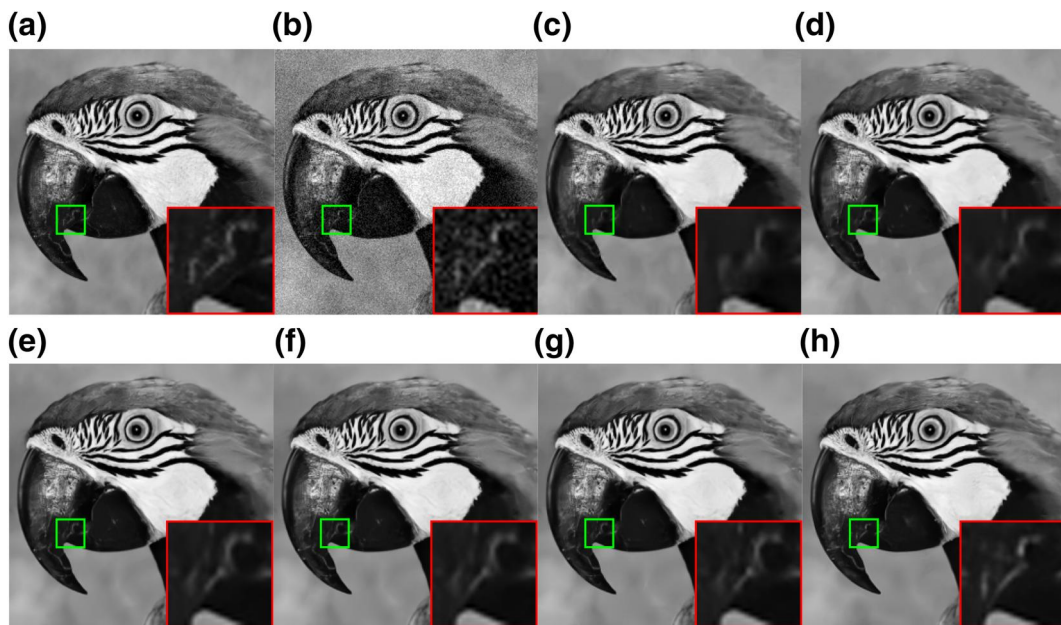| Methods	Device	256×256	512×512	1024×1024
CSF [54]	CPU	–	0.92	1.72
TNRD [51]	CPU	0.45	1.33	4.61
BM3D [8]	CPU	0.59	2.52	10.77
WNNM [14]	CPU	203.1	773.2	2536.4
DnCNN [18]	GPU	0.0344	0.0681	0.1556
ADNet [43]	GPU	0.0467	0.0798	0.2077
CTCNN [58]	GPU	0.068	0.103	0.364
MemNet [59]	GPU	0.8775	3.606	14.69
RED30 [56]	GPU	1.362	4.702	15.77
RDDCNN	GPU	0.0119	0.0437	0.182

FIGURE 4 Denoising results of different methods on one image from Set12 when noise level is 15. (a) Original image (b) Noisy image/24.60 dB (c) BM3D [8]/31.37 dB (d) WNNM [14]/31.62 dB (e) FFDNet [21]/31.81 dB (f) DnCNN [18]/31.83 dB (g) IRCNN [50]/31.84 dB (h) RDDCNN/31.93 dB

Camera settings	GAT-BM3D [61]	DnCNN [18]	TID [60]	RDDCNN
Canon 5D ISO = 3200_1	31.23	37.26	37.22	37.00
Canon 5D ISO = 3200_2	30.55	34.13	34.54	33.88
Canon 5D ISO = 3200_3	27.74	34.09	34.25	33.82
Nikon D600 ISO = 3200_1	28.55	33.62	32.99	33.24
Nikon D600 ISO = 3200_2	32.01	34.48	32.99	33.76
Nikon D600 ISO = 3200_3	39.78	35.41	35.58	34.91
Nikon D800 ISO = 1600_1	32.24	37.95	34.49	35.47
Nikon D800 ISO = 1600_2	33.86	36.08	34.49	34.81
Nikon D800 ISO = 1600_3	33.90	35.48	35.26	35.71
Nikon D800 ISO = 3200_1	36.49	34.08	33.70	33.26
Nikon D800 ISO = 3200_2	32.91	33.70	31.04	32.89
Nikon D800 ISO = 3200_3	40.20	33.31	33.07	32.91
Nikon D800 ISO = 6400_1	29.84	29.83	29.40	29.86
Nikon D800 ISO = 6400_2	27.94	30.55	29.40	29.97
Nikon D800 ISO = 6400_3	29.15	30.09	29.21	29.63
Average	32.45	33.86	33.36	33.41

TABLE 7 Average Peak Signal to Noise Ratio (PSNR) (dB) of different denoising methods on CC

conduct experiments. Besides, we use complexity to test the efficiency of the proposed method in image denoising. The detailed analysis of mentioned illustrations as follows.

To test visual effect of the proposed method, we use BM3D, WNNM, IRCNN, FFDNet and DnCNN as comparative methods on BSD68 and Set12 to conduct experiments. Specifically, we choose an area of predicted images to amplified it as observation areas, where an observation area is clearer, its corresponding method has better denoising performance. As shown in Figure 2 and Figure 3, our method is clearer than DnCNN on BSD68 for noise level of 25 and 50, which shows that our method has better denoising result for noise of medium and high levels. As shown in Figure 4, our method is clearer than other method on Set12 for noise level of 15, which illustrates our method is more effective than other method for noise of low-level. According to mentioned illustrations, our method is very effective for noise of high, medium and low-level.

We use running time and complexity to verify its efficiency as shown in Tables 5 and 6. The following 11 methods are selected for comparison: DnCNN, ADNet, Deep universal blind denoisor (DUBD) [55], Residual encoder decoder network (RED30) [1, 56], Adaptively tuned denoising network (ATDNet) [57], CSF, TNRD, BM3D, WNNM, Countourlet transform based CNN (CTCNN) [58], Memory network (Memnet) [59], Due to the different computational performances on different GPU devices, the running time can be significantly different. Therefore, in the spirit of equality, we directly pick the experiment results from their original paper. Our denoising method is competitive with other denoising methods. That shows the effectiveness of the proposed method in denoising efficiency.

We extend our denoising method on real noisy images (i.e., CC) to test its performance. In Table 7, we can see that our method is competitive with targeted image denoising (TID) [60] and generalised anscombe transformation BM3D (GAT-BM3D) [61], which shows its performance in real noisy image denoising.

According to mentioned illustrations, it is known that our method is effective for image denoising.

5 | CONCLUSION

We propose a robust deformed denoising CNN as well as RDDCNN to enhance the pixels, according to relations of surrounding pixels. RDDCNN is implemented by collaborations of a deformable block, an enhanced block and a residual block. Deformable block uses a deformable learnable kernel and stacked convolutions to obtain more representative noise features. Enhanced block uses a dilated convolution and a novel combination of convolutional layers, BN and ReLU to better facilitate contextual interaction, which can accelerate the convergence speed of training a denoiser. Also, residual block utilises a residual operation to construct a clean image. These components make RDDCNN robust in image denoising.

DATA AVAILABILITY STATEMENT

Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

ORCID

Chunwei Tian https://orcid.org/0000-0002-6058-5077

REFERENCES

1. Healey, G.E., Kondepudy, R.: Radiometric ccd camera calibration and noise estimation. IEEE Trans. Pattern Anal. Mach. Intell. 16(3), 267–276 (1994). https://doi.org/10.1109/34.276126

2. Gurrola-Ramos, J., Dalmau, O., Alarcón, T.E.: A residual dense U-net neural network for image denoising. IEEE Access. 9, 31742–31754 (2021). https://doi.org/10.1109/ACCESS.2021.3061062

3. Lee, J.S.: Digital image enhancement and noise filtering by use of local statistics. IEEE Trans. Pattern Anal. Mach. Intell. (2), 165–168 (1980). https://doi.org/10.1109/TPAMI.1980.4766994

4. Kuan, D.T., et al.: Adaptive noise smoothing filter for images with signal dependent noise. IEEE Trans. Pattern Anal. Mach. Intell. 7(2), 165–177 (1985). https://doi.org/10.1109/tpami.1985.4767641

5. Zhang, B.Y., Allebach, J.P.: Adaptive bilateral filter for sharpness enhancement and noise removal. J IEEE Transactions Image Process. 17(5), 664–678 (2008). https://doi.org/10.1109/tip.2008.919949

6. Katkovnik, V., et al.: From local kernel to nonlocal multiple-model image denoising. Int. J. Comput. 86(1), 1–32 (2010). https://doi.org/10.1007/s11263-009-0272-7

7. Alkinani, M.H., El-Sakka, M.R.: Patch-based models and algorithms for image denoising: a comparative review between patch-based images denoising methods for additive noise reduction. EURASIP J. Image Video Proces., 1–27 (2017). https://doi.org/10.1186/s13640-017-0203-4

8. Dabov, K., et al.: Image denoising by sparse 3-D transform-domain collaborative filtering. IEEE Trans. Image Process. 16(8), 2080–2095 (2007). https://doi.org/10.1109/tip.2007.901238

9. Yang, D., Sun, J.: BM3D-Net: a convolutional neural network for transform-domain collaborative filtering. IEEE Signal Process. Lett. 25(1), 55–59 (2018). https://doi.org/10.1109/lsp.2017.2768660

10. Abbas, J.K., et al.: Visual Perception Method for Medical Image Denoising, pp. 2636–9346 (2022)

11. Chambolle, A.: An algorithm for total variation minimization and applications. J. Math. Imag. Vis. 20(1), 89–97 (2004). https://doi.org/10.1023/b:jmiv.0000011321.19549.88

12. Perona, P., Malik, J.: Scale space and edge detection using anisotropic diffusion. J IEEE Transactions Pattern Analysis Machine Intelligence. 12(7), 629–639 (1990). https://doi.org/10.1109/34.56205

13. Li, Q., Jiang, L.: A novel variable-separation method based on sparse representation for stochastic partial differential equations. SIAM J. Sci. Comput. 39(6), 2879–2910 (2016)

14. Gu, S., et al.: Weighted nuclear norm minimization with application to image denoising. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2862–2869 (2014)

15. Lan, X., et al.: Efficient belief propagation with learned higher-order Markov random fields. In: European Conference on Computer Vision, pp. 269–282 (2006)

16. Li, P., et al.: Joint image denoising with gradient direction and edge-preserving regularization. J Pattern Recognition. 125, 108506 (2022). https://doi.org/10.1016/j.patcog.2021.108506

17. Tian, C., et al.: Deep learning on image denoising: an overview. Neural Network. 131, 251–275 (2020). https://doi.org/10.1016/j.neunet.2020.07.025

18. Zhang, K., et al.: Beyond a Gaussian denoiser: residual learning of deep cnn for image denoising. IEEE Trans. Image Process. 26(7), 3142–3155 (2017). https://doi.org/10.1109/tip.2017.2662206

19. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning, pp. 448–456 (2015)

20. He, K., et al.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)

21. Zhang, K., Zuo, W., FFDNet, L.Z.: Toward a fast and flexible solution for CNN-based image denoising. IEEE Trans. Image Process. 27(9), 4608–4622 (2018). https://doi.org/10.1109/tip.2018.2839891

22. Bae, W., Yoo, J., Ye, J.C.: Beyond deep residual learning for image restoration: persistent homology-guided manifold simplification. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 1141–1149. CVPRW (2017)

23. Tai, Y., et al.: A persistent memory network for image restoration [C]. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4539–4547 (2017)

24. Hassani, I.K., Pellegrini, T., Masquelier, T.: Dilated convolution with learnable spacings. arXiv preprint arXiv:2112.03740 (2021)

25. Tian, C., Xu, Y., Zuo, W.: Image denoising using deep CNN with batch renormalization. Neural Network. 121, 461–473 (2020). https://doi.org/10.1016/j.neunet.2019.08.022

26. Anwar, S., Barnes, N.: Real image denoising with feature attention. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 3155–3164 (2019). https://doi.org/10.1109/ICCV.2019.00325

27. Guo, S., et al.: Toward convolutional blind denoising of real photographs. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1712–1722 (2019). https://doi.org/10.1109/CVPR.2019.00181

28. Alawode, B.O., and Alfarraj, M.: Meta-optimization of deep CNN for image denoising using LSTM at. arXiv:2107.06845, (2021)

29. Li, Z., Wu, J.: Learning deep cnn denoiser priors for depth image inpainting. J. Appl. Sci. 9(6), 1103 (2019). https://doi.org/10.3390/app9061103

30. Wang, T., Sun, M., Hu, K.: Dilated deep residual network for image denoising. In: 2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 1272–1279 (2017). https://doi.org/10.1109/ICTAI.2017.00192

31. Tian, C., et al.: Lightweight image super-resolution with enhanced CNN. In: J Knowledge-Based Systems. 106235 (2020)

32. Chen, Y., et al.: Dynamic convolution: attention over convolution kernels. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 11027–11036 (2020). https://doi.org/10.1109/CVPR42600.2020.01104

33. Chen, J., et al.: Dynamic region-aware convolution. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8060–8069 (2021). https://doi.org/10.1109/CVPR46437.2021.00797

34. Xu, Y.S., et al.: Unified dynamic convolutional network for super-resolution with variational degradations. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 12493–12502 (2020). https://doi.org/10.1109/CVPR42600.2020.01251

35. Li, Y., et al.: Revisiting dynamic convolution via matrix decomposition. arXiv preprint arXiv:2103.08756 (2021)

36. Wang, N., Zhang, Y., Zhang, L.: Dynamic selection network for image inpainting. IEEE Trans. Image Process. 30, 1784–1798 (2021). https://doi.org/10.1109/TIP.2020.3048629

37. Sun, X., et al.: Gaussian dynamic convolution for efficient single-image segmentation. IEEE Trans. Circ. Syst. Video Technol. 32(5), 2937–2948 (2022). https://doi.org/10.1109/TCSVT.2021.3096814

38. Dai, J., et al.: Deformable convolutional networks. CoRR (2017)

39. Chen, F., et al.: Adaptive deformable convolutional network. J Neurocomputing. 453, 853–864 (2021). https://doi.org/10.1016/j.neucom.2020.06.128

40. Yu, Y., et al.: Deformable siamese attention networks for visual object tracking[C]. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6728–6737 (2020)

41. Ephraim, Y., Malah, D.: Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator. IEEE Trans.

Acoust. Speech Signal Process. 32(6), 1109–1121 (1984). https://doi.org/10.1109/tassp.1984.1164453

42. Kingma, D., & Adam, J.B.: A method for stochastic optimization. arXiv Preprint 2014, arXiv:1412.6980 (2014)

43. Tian, C., et al.: Attention-guided CNN for image denoising [J]. Neural Network. 124, 117–129 (2020). https://doi.org/10.1016/j.neunet.2019.12.024

44. Xu, J., et al.: Real-world noisy image denoising: a new benchmark. arXiv preprint arXiv:1804.02603 (2018)

45. Simard, P.Y., et al.: Best practices for convolutional neural networks applied to visual document analysis. In: Icdar, vol. 3 (2003)

46. Roth, S., Black, M.J.: 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 (2005). https://doi.org/10.1109/CVPR.2005.160

47. Mairal, J., et al.: Non-local sparse models for image restoration. In: Proceedings of the International Conference on Computer Vision, pp. 2272–2279. IEEE (2009)

48. Nam, S., et al.: A holistic approach to cross-channel image noise modeling and its application to image denoising. In: Proceedings of the IEEE Conference on Computer Vision Pattern Recognition, pp. 1683–1691 (2016)

49. Huang, D.A., et al.: Self-learning based image decomposition with applications to single image denoising. IEEE Trans. Multimed. 16(1), 83–93 (2013). https://doi.org/10.1109/tmm.2013.2284759

50. Zhang, K., et al.: Learning deep CNN denoiser prior for image restoration. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3929–3938 (2017)

51. Chen, Y., Pock, T.: Trainable nonlinear reaction diffusion: a flexible framework for fast and effective image restoration. IEEE Trans. Pattern Anal. Mach. Intell. 99(6)1 (2016). https://doi.org/10.1109/tpami.2016.2596743

52. Zoran, D., Weiss, Y.: From learning models of natural image patches to whole image restoration. IEEE International Conference on Computer Vision, 479–486 (2011)

53. Burger, H.C., Schuler, C.J., Harmeling, S.: Image denoising: can plain neural networks compete with BM3D? In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2392–2399 (2012)

54. Schmidt, U., Roth, S.: Shrinkage fields for effective image restoration. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2774–2781 (2014)

55. Soh, J.W., Cho, N.I.: Deep Universal Blind Image Denoising. 2020 25th International Conference on Pattern Recognition (ICPR). (2021). https://doi.org/10.1109/icpr48806.2021.9412605

56. Mao, X., Shen, C., Yang, Y.B.: Image restoration using very deep convolutional encoder–decoder networks with symmetric skip connections, in: Proc. Adv. Neural Inf. Process. Syst, pp. 2802–2810 (2016)

57. Kim, Y., Soh, J.W., Cho, N.I.: Adaptively tuning a convolutional neural network by gate process for image denoising. IEEE Access, vol. 7, pp. 63447–63456 (2019)

58. Lyu, Z., Zhang, C., Han, M.: A nonsubsampled countourlet transform based CNN for real image denoising, Signal Process. Image Commun. 82, 115727 (2020)

59. Tai, Y., et al.: Memnet: A persistent memory network for image restoration, in: Proc. IEEE Int. Conf. Comput. Vis, pp. 4539–4547 (2017)

60. Luo, E., Chan, S.H., Nguyen, T.Q.: Adaptive image denoising by targeted databases, IEEE Trans. Image Process. 24 (7), 2167–2181 (2015)

61. Makitalo, M., Foi, A.: Optimal inversion of the generalized anscombe transformation for Poisson–sgaussian noise, IEEE Trans. Image Process. 22(1), pp. 91–103 (2012)