



Deep dilated CNN based image denoising

Rashmi Chaurasiya¹ · Dinesh Ganotra¹

Received: 24 May 2022 / Accepted: 14 October 2022 / Published online: 31 October 2022

© The Author(s), under exclusive licence to Bharati Vidyapeeth's Institute of Computer Applications and Management 2022

Abstract Over the past few years various convolutional neural network (CNN) based approaches have been applied in image denoising. Acknowledging the size of the receptive field in an image denoising problem can increase the network's performance. The present research has altered the receptive field and studied its effect on image denoising. Three networks have been designed and compared– (1) CNN with dilated kernels (2) CNN without dilation but with increased kernel size with similar receptive field and (3) CNN without any dilation and without increased kernel size. After analyzing the results of three cases an additional fourth case is added with an optimum receptive field, which improves upon the state-of-the-art results. Most methods are based on some assumptions on natural images, here a method relying on a purely learning based approach is presented. The simulation has been conducted with randomized noise levels. A single network is able to handle a wide range of noise levels (σ). For 24-bit images with 256 gray scales for each color, it can reduce noise for standard deviation (σ) in the range of [0–75]. Comparable (with contemporary research) results have been presented in terms of qualitative measurements and visual perspectives. Performance of our approach can be validated with various test sets used in our study. We have achieved comparable PSNR values on existing standard test sets.

Keywords Image denoising · Convolutional neural networks · Deep learning

✉ Rashmi Chaurasiya
rashmichaurasiya783@gmail.com

Dinesh Ganotra
dinesh_ganotra@hotmail.com

¹ IGDTUW, Kashmere Gate, New Delhi 110006, India

1 Introduction

Image acquired by the camera is always degraded by noise. Often the scene is not illuminated properly, forcing the camera to increase the sensitivity of the sensor degrades the image. Removal of noise is an essential step in various image restoration [1, 2] tasks. Considering noise as independent of the image gives us a simple model $I = S + \eta$, where I is a noisy image, S is a noiseless ground truth image and η is noise with standard deviation σ . Noisy images with additive white Gaussian noise (AWGN) [3] can be modeled as the sum of ground truth image and noise (η) (Eq. 1)

$$I(x, y) = S(x, y) + \eta(x, y) \quad (1)$$

where $I(x, y)$ is resultant noisy image pixel when $S(x, y)$ is corrupted image with noise (η).

Image denoising is often ill-posed. This ill-posedness can be solved with maximum-a-posteriori (MAP) principle [4]. Solving this problem with MAP requires modeling the image with random variables that follow a prior distribution. The corrupted image is then reconstructed with the priors using maximum-a-posteriori principle. The objective is to maximize the conditional probability of the reconstructed image when a corrupted image is given.

Image denoising methods can be broadly classified into two categories, model-based methods and discriminative methods. Model based methods use a generic prior model and an optimization algorithm. Model based methods include Block Matching and 3D Filtering (BM3D) [5], Expected Patch Log Likelihood (EPLL) [6] and Weighted Nuclear Norm Minimization (WNNM) [7]. These methods are computationally expensive, time consuming and unable to resolve the problem of spatially variant noise. Discriminative methods model an appropriate image prior, this

approach of learning the prior employ Multi-Layer Perceptron (MLP) [8], DnCNN [2], and Trainable nonlinear reaction diffusion (TNRD) [9]. The major difference between model based and discriminative methods is that the model-based methods have flexibility to handle several tasks. Whereas the discriminative methods are solely dependent on the type of dataset used and thus are very specific in nature and can only solve problems they are designed for. For example, a single model-based method NCSR [10] can solve for image denoising, deblurring and super-resolution, whereas three different discriminative methods MLP [8], SRCNN [11], and DCNN [12] are designed for image deblurring, super-resolution and denoising.

Despite having the flexibility of handling multiple tasks, model-based methods are time consuming and need to be optimized with appropriate priors. On the other hand, discriminative methods offer fast speed and promising performance. So far, the most promising results have been provided by discriminative methods. Discriminative methods include usage of convolutional neural network (CNN) [11] and MLP [8] based deep learning techniques [13–15]. The aim of this paper is to obtain an estimate of the ground truth image and a thorough evaluation of different receptive field sizes. Mapping function can be referred by $F(I) = \hat{S}$, for this mapping a deep learning approach is employed.

A major part of deep learning is based on CNNs. These networks perform very well as compared to traditional algorithms and produce state of the art results. CNNs were originally developed for image recognition [14] and classification [16, 17] tasks. Using a CNN for these types of tasks progressively reduces the image resolution. A straightforward approach to increase resolution would be to remove subsampling or strides from the layers. This does increase the resolution but at the same time severely affects the receptive field. So, removing subsampling improves the loss in resolution but on the other hand, it reduces the receptive field [18] in the same proportion as it had improved resolution. Reduction in the receptive field cannot be compromised in any way. A brief introduction of receptive field is given below that explains the variation in the size of the receptive field with a set of equations.

1.1 Receptive field

In a Neural Network each node of a layer is connected with node of the next layer, this way of connection to transfer information requires an extremely large number of parameters. CNN uses a slightly different approach where only a few nodes participate in the connection to the next layer. Since this transfer of the information resembles the response of neurons to stimuli only in certain regions of the visual field, the region is called the receptive field in the visual system. With this analogy it can be stated that

CNN uses a receptive field like layout [19], where the subset of the nodes of the previous layer connected to the next layer is the receptive field of the next layer. Figure 1 shows the receptive field in neural network layers and CNN.

1.2 Size of receptive field

A large receptive field means the network can perceive more information to predict an accurate image. So, the receptive field has to be enlarged to broaden the view of the input to capture wider contextual information. The receptive field can be enlarged either by increasing the size of the kernel or by increasing the depth of the network. Inflating the size of the kernel increases the number of parameters thus makes this approach a computational burden. A larger number of layers makes the network architecture deep thus introduces more operations. A solution to this problem is to replace convolutional layers with dilation layers. It helps in increasing the size of the receptive field.

The dilation layer method increases the effective kernel size by inserting blank spaces between them. Dilated kernel reduces the computation as it uses a smaller number of parameters. Also, it helps in detection of minute details with improved resolution. Dilated convolution is applied in various domains like image super-resolution [11, 21] Text-to-speech [22] solution and language translation [23].

1.3 Equation of receptive field

Convolution and dilated convolution-based model [20] can be defined with Eqs. (2) and (3) respectively–

Let $F : Z^2 \rightarrow R$ be a discrete function. Let $\Omega_r = [-r, r]^2 \cap Z^2$ and let $k : \Omega_r \rightarrow R$ be a discrete filter of size $(2r + 1)^2$. The discrete convolution operator $*$ is defined as (Eqs. 2 and 3)

$$(F * k)(p) = \sum_{s+t=p} F(s)k(t) \quad (2)$$

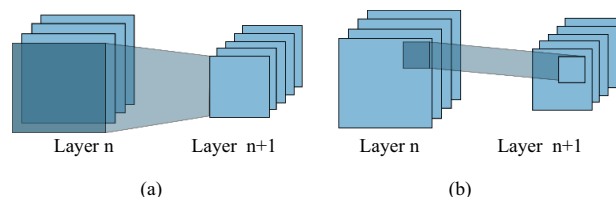


Fig. 1 a Feature transfer in traditional neural network in a multi-channel input layer n and n+1 b. Feature transfer in a CNN. Only a certain region participates i.e. the receptive field

Let l be a dilation factor and let $*_l$ be defined as

$$(F*_l k)(p) = \sum_{s+l=p} F(s)k(t) \tag{3}$$

When $l=1$ refers discrete convolution and $l>1$ refers dilated convolution. Same can be applied on 2-Dimensional dilated convolution and given by Eq. 4:

$$y(m, n) = \sum_{i=1}^m \sum_{j=1}^n x(m + si, n + sj)w(i, j) \tag{4}$$

where $y(m, n)$ is the output of dilated convolution for input $x(m, n)$ and a filter $w(i, j)$ with length and the width of m and n respectively.

The dilation layer can be naively called convolution over input with a sparsely populated filter which expands the size of the convolving filter. The expansion rate is controlled by a hyper-parameter d , where $(d - 1)$ blank spaces are inserted in the kernel. For dilation rate equal to 1, zero space will be added. The effective kernel size under the influence of dilation d with kernel size k is given by [24] as

$$k' = k + (k - 1)(d - 1) \tag{5}$$

where k' represents the effective kernel size. Effective receptive field from Eq. 5, for a kernel ($k=3$) and dilation rate $d=1, 2$ and 3 is $3 \times 3, 5 \times 5$ and 7×7 (shown in Fig. 2a–c).

In addition to this, the receptive field for the depth n and kernel size 3×3 (throughout the network) can be given by $(2d + 1)(2d + 1)$. The relationship between the dilation rate and output size o as in [24] is given by

$$o = \left\lceil \frac{i + 2p - k - (k - 1)(d - 1)}{s} \right\rceil + 1 \tag{6}$$

For the input size i , padding p and stride s . Convolution of 3×3 kernel over an input of size 9×9 , padding zero and dilation rate 2 (i.e., $i=9, k=3, d=2, s=1$ and $p=0$) produces output of dimension 5×5 . Figure 3 shows the output size 3×3 for $i=7$. That concludes the introduction on receptive field.

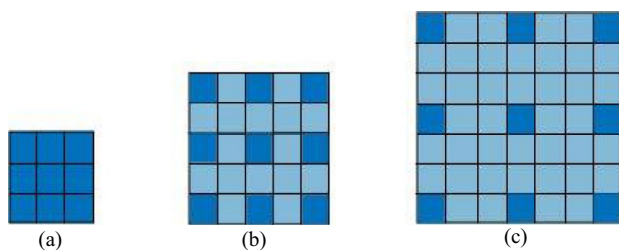


Fig. 2 Effective receptive field: (a) 3×3 kernel for $d=1$ (b) 5×5 for dilation rate 2 ($d=2$), c 7×7 for dilation rate 3($d=3$)

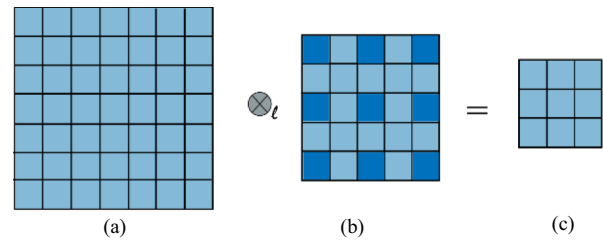


Fig. 3 Estimation of the dimension of the output layer: where a 7×7 input size is used ($i=7$) with kernel size $3 \times 3(k=3)$ convolves with a filter of dilation 2($d=2$) gives the output of 3×3

In this paper we have studied the significance of receptive field in image denoising in 4 study cases. Our work can be summarized with Fig. 4, where images of compared cases are shown. Cases 1–4 are described in Sect. 3. Case 4 showcases our best result in terms of PSNR (Peak Signal to Noise Ratio) comparison. The rest of the paper is organized as follows. Section 2 contains a brief study on related methods. Section 3 explains our study cases. Sections 4 and 5 describe the details of dataset and network structure. In Sect. 6 experimental results and analysis based on compared PSNR values and corresponding images are shown. Based on the analysis in Sect. 6 another network mentioned as case 4 is introduced. Discussion on the results of test sets is in Sect. 7. Lastly, we wrap up our work with the conclusion mentioned in Sect. 8 and future work in 9.

2 Related methods

Filter based techniques are one of the initially proposed methods for AWGN denoising, these filters are further divided into spatial domain filter and transform domain filter. Mean filtering [25], denoising with local statistics [26], Weiner filter [27] and Bilateral Filtering [28] are some of the most prevalent techniques. These techniques were not sufficient to produce a good quality image.

The image prior is an important property in image denoising. In the past decade a lot of methods were proposed based on image priors. Some of them are Markov Random Field (MRF) [20, 29], BM3D [5], NCSR [10], nonlocal self-similarity (NSS) [31] and WNNM [7]. It is convenient to learn the prior model on small image patches. In EPLL [6] optimization is made on an entire image and image prior is given by the product of all patch priors. Non-local self-similarity [6, 7, 31] based methods exploit the property of repetitive patterns in natural images. These similar patches are grouped to collaboratively estimate the final image. Out of the prior based methods mentioned above, BM3D [5] and WNNM [7] are the popular ones. They are capable of

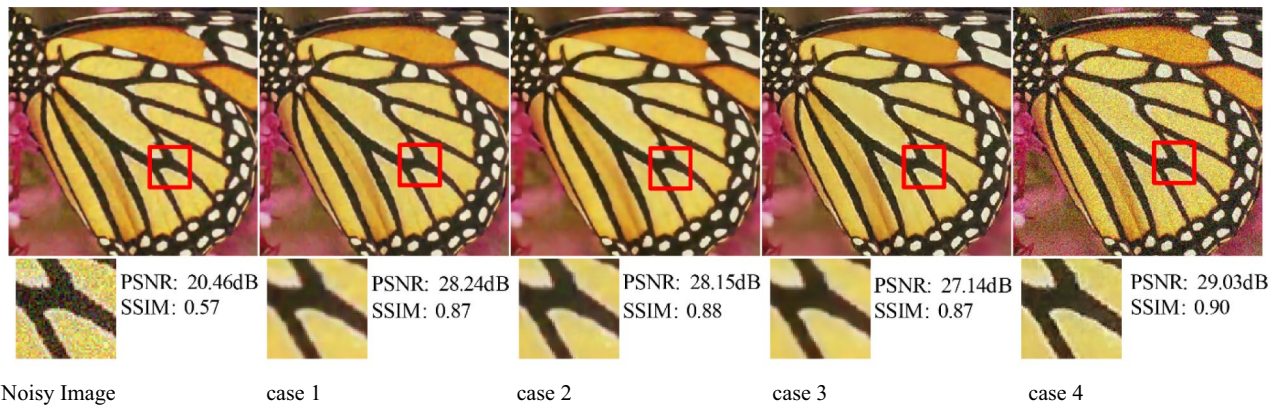


Fig. 4 Predicted denoised images were compared with their PSNR and SSIM, inset image is also shown to compare the results visually. Here a input noisy image $\sigma=25$. **b** Result with case 1. **c** Result with case 2. **d** Result with case 3 and **e** Result with case 4

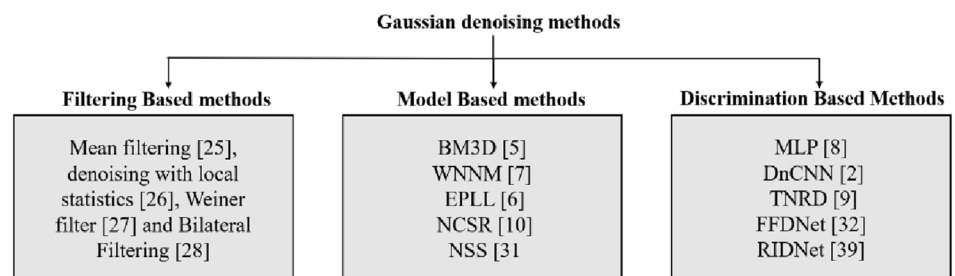
handling various noise levels but they cannot be directly used for spatially variant noise [32].

CNNs are widely used in various image processing tasks due to their excellent performance. Though CNN based methods have also been challenged against prior based methods. Jain et al. [33] compared the performance of Markov random field (MRF) with convolutional neural networks. Another comparison with BM3D is done by Burger et al. [8].

Prior based methods perform well but with some drawbacks. They need to be optimized well, thus there is an increase in the computation cost, in addition to that they rely on manual settings and tuning. To address these problems discriminative approaches were proposed where there is a direct mapping from the noisy image to the ground truth image. In the discriminative learning approach DnCNN [2] is the most popular one. Here a single method aims to solve various image restoration (IR) tasks i.e., it can provide solutions for blind Gaussian denoising, single image super-resolution and JPEG deblocking [34]. They produced promising results for all the three problems. Another similar method [1] used HQS (Half Quadratic Splitting), a variable splitting technique to solve for image denoising, image deblurring and single image super resolution utilizing deep CNN denoiser prior. Chuah et al. [35]

provided a straightforward strategy of estimating noise level before removing noise. Wang et al. [36] achieved comparable results with reduced computational cost, less complicated network structure, using a larger receptive field than DnCNN. A combination of a dilated layer with residual learning is a popular technique in resolving the AWG noise problem [37, 38 and 2]. All the methods described above employ a one-to-one mapping i.e., they require a single image as input to produce a denoised image. Zhang et al. [32] proposed a novel network design FFDNet (Fast and Flexible Denoising Convolutional Neural Network) that takes sampled input images with their noise level maps to produce a denoised image. In addition to that, recent approaches aim to deal with spatially variant and invariant noises, these methods worked on real world noisy images. Anwar et al. [38] incorporated feature attention for image denoising. A benchmark dataset of denoised images is created by Romano et al. [39], these images are captured with different cameras and under different camera settings. Guo et al. [40] uses the same strategy of estimating noise first before noise removal like [35]. Their work is denoted by its network architecture name CBDNet [40], which has two sub-networks one for noise estimation and other non-blind denoising estimation. Some of the methods used for comparison are categorized in Fig. 5.

Fig. 5 Categorization of image denoising methods



3 Our method

The present research has studied three cases, that are as follows:

Case 1: A dilated convolutional network is utilized here. The network structure is similar to Zhang et al. [1] and Peng et al. [37], here a 7 layered dilated network with a receptive field of the network of size 33×33 is used. Zhang et al. [1] and [37] used a residual learning formulation i.e., $F(I) = \eta$. i.e., noise is separated by subtracting the predicted noise from noisy input image. We have used $F(I) = \hat{S}$, a direct mapping from noisy image to noiseless image. No intermediate step is required.

Case 2: In this case plain CNN network with the same configuration and same receptive field of size 33×33 is used. Same receptive field can be ensured either by using larger sized filters or increasing the depth of the network. We have used larger sized filters.

Case 3: To verify the efficacy of dilated layers in image denoising process, receptive field size is reduced. All dilated layers are replaced with plain CNN layers that reduces the size of the receptive field to 15×15 . This way performance can be compared with the above two cases. The figure below represents the three cases (Fig. 6).

Receptive field of each layer for all three cases are shown in Table 1, all the cases have seven layers. We have used the term rcp_i which is the receptive field of the network up to i layers, where $i = 1, 2, \dots, 7$. All these cases mentioned above are essential steps in verifying the effectiveness of receptive field size in image denoising. Three cases are compared qualitatively and visually.

4 Datasets

We have used COCO dataset [41], which is available as an open-source online dataset and contains 5000 images in its val2016 set. After preprocessing this data, an augmented

set of 10,000 images was developed. It is observed that increasing the size of the dataset beyond this does not lead to significant improvement. Images are then separated into training and validation sets in the ratio of 7:3. These images are then cropped to 256×256 pixels. To produce synthetic noisy images, an additive white Gaussian noise (AWGN) is added to the images.

For blind denoising, noise levels (σ) are randomly selected from the range [0, 75] to create the dataset. Test sets BSD68 [30], Set 12 [2], RNI15 [42], NC12 [42] and Nam [43] datasets are used. BSD68 [30] and Set 12 [2] contain classic images in the field of image processing i.e., these images have been extensively used for the evaluation of numerous methods. RNI15 [42] set is a real-world noisy image set having 15 images, these images contain spatially variant noise too. NC12 is a set of 12 noisy images, there are no ground truth images in RNI15 [42], NC12 [42] so the images will be compared visually for this set.

5 Network structure

The architecture of the network used to remove Gaussian noise is shown in Fig. 7, where it takes an image degraded with a certain level of AWGN as input. This image is convolved with dilated kernels. Each layer has these kinds of filters with dimension defined by their dilation rate, which will then be trained to appropriate values during the back-propagation algorithm. No pooling is used here due to the requirement of same dimensions of input and output images. Zero padding is used to avoid boundary artifacts. Filters of dimension $3 \times 3 \times 32$ are used where the third channel refers to the number of filters. ReLu (Rectified Linear Unit) [44] is also placed between two consecutive convolution layers to introduce non-linearity in the network. For adaptive learning Adam [45] is used as an optimizer with learning rate 0.001. Loss function here is MSE (mean squared Error) shown in Eq. (7).

Fig. 6 Illustration of the three cases in the proposed work

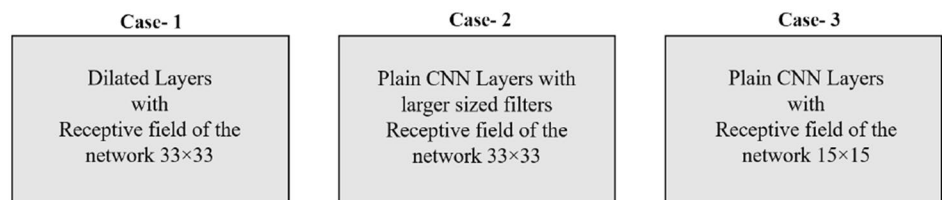


Table 1 Receptive field of the network used in case 1, 3 and case 3

	rcp_1	rcp_2	rcp_3	rcp_4	rcp_5	rcp_6	rcp_7
Case1	3×3	7×7	13×13	21×21	27×27	31×31	33×33
Case2	3×3	7×7	13×13	21×21	27×27	31×31	33×33
Case3	3×3	5×5	7×7	9×9	11×11	13×13	15×15

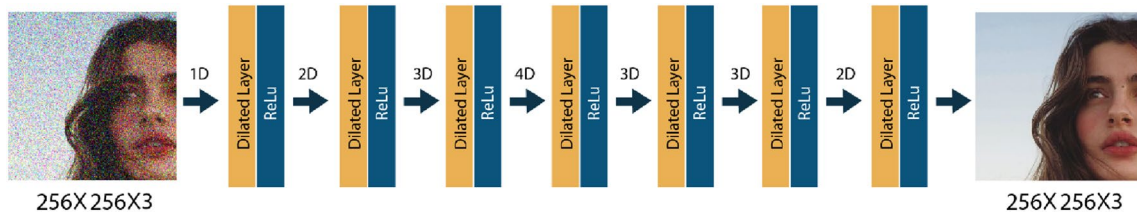


Fig. 7 Network architecture used in simulation. Where 1D, 2D... refers to the dilation rate

Table 2 Network structure used in of case 1

Layer	Operation	Dilation rate	Receptive field of layers
Input layer (350×350×3)			
1	Dilation layer, ReLu	1	$r_1 = 3$
2	Dilation layer, ReLu	2	$r_2 = 5$
3	Dilation layer, ReLu	3	$r_3 = 7$
4	Dilation layer, ReLu	4	$r_4 = 9$
5	Dilation layer, ReLu	3	$r_5 = 7$
6	Dilation layer, ReLu	2	$r_6 = 5$
7	Dilation layer, ReLu	1	$r_7 = 3$
Output layer (350×350×3)			

$$MSE = \frac{1}{n} \sum_{i=1}^n (S_i - \hat{S}_i)^2 \tag{7}$$

where S and \hat{S} are ground truth and predicted denoised images. The final layer gives out the output i.e., an image with reduced noise.

Table 2 shows the parameters and the size of the effective kernel in each layer. Here the receptive field (effective kernel size) of each layer denoted by r_1, r_2, \dots, r_7 is 3, 5, 7, 9, 7, 5, 3 in 1–7 layers respectively assuming $r_0 = 1$. In calculating the receptive field of a network.

6 Experimental results and analysis

Proposed method is a plain discriminative method, since it is feasible to train a deep network with minimal number of layers. Though residual learning framework is not utilized here, equivalent results are achieved. All our models are trained on Nvidia Tesla K80 GPU.

PSNR and SSIM measurements are utilized to compare the performance of the cases mentioned in Sect. 3. Including SSIM for comparison ensures the quality of image for human perception. It estimates the correlation between two normalized images.

$$PSNR = 10 \frac{(255)^2}{MSE} \tag{8}$$

$$SSIM = \frac{(2\mu_S\mu_{\hat{S}} + C_1)(2\sigma_{S\hat{S}} + C_2)}{(\mu_S^2 + \mu_{\hat{S}}^2 + C_1)(\sigma_S^2 + \sigma_{\hat{S}}^2 + C_2)} \tag{9}$$

where μ_S, σ_S^2 and $\mu_{\hat{S}}, \sigma_{\hat{S}}^2$ the local mean and variance of the ground truth image and predicted image. $\sigma_{S\hat{S}}$ denotes the local covariance of ground truth and predicted image.

6.1 Comparison of case 1, case 2, case 3 and case 4

Our cases are compared with three prior based methods BM3D [5], WNNM [8], EPLL [7] and two discriminative methods TNRD [9] and DnCNN [2] as shown in Table 3. Comparison is based on the PSNR values obtained for Set 12 Images [2].

All the simulations are done for a sigma set in the range of $\sigma \in [0-75]$, where a single model is used to denoise an image. Images have a range of [0–255]. These models are referred to as case 1, case 2 and case 3. This type of denoising may also be referred to as blind denoising. Here a single model can handle a series of $\sigma \in [0-75]$. Comparison of each is shown below:

- Case 1 and case 2

It is observed from Table 3 that these two cases produce approximately the same result for both $\sigma = 15$ and 25. Since these cases have the same receptive field, they produce similar results. Also, these results verify the significance of the receptive field. Case 1 and 2 differ in terms of convergence rate, since case 2 has CNNs with large sized filters; its convergence rate is slower than case 1.

- Case 1 and case 3

Case 3 shows slightly less PSNR values than case 1. The difference increases when noise level (σ) increases from 15 to 25. This difference proves the need for a larger receptive field in image denoising. Case 1 has fastest convergence compared to the rest of the cases.

Table 3 PSNR results of various methods on Set 12 dataset, maximum PSNR values are highlighted as bold characters

Image	Monarch	Lena	Barbara	Parrot	House	Couple	Pepper	Airplane	C.man	Boat	Man	Starfish
$\sigma = 15$												
BM3D [5]	31.85	34.26	33.10	31.37	34.93	32.10	32.69	31.07	31.91	32.13	31.92	31.14
WNNM [8]	32.71	34.27	33.60	31.37	35.13	32.10	32.99	31.39	32.17	32.27	32.11	31.82
EPLL [7]	32.10	33.92	31.38	31.42	34.17	31.93	32.64	31.19	31.85	31.93	32.00	31.13
TNRD [9]	32.56	34.24	32.13	31.63	34.53	31.11	33.04	31.46	32.19	32.14	32.23	31.75
DnCNN [2]	33.09	34.62	32.64	31.83	34.97	32.47	33.30	31.70	32.61	32.42	32.46	32.20
DnCNNB [2]	32.94	34.56	32.09	31.63	34.93	32.41	33.15	31.56	32.10	32.35	32.41	32.02
FFDNet [32]	32.77	34.63	32.50	31.77	35.01	32.45	33.10	31.58	32.42	32.35	32.40	32.02
Case 1	34.10	33.02	27.46	31.75	36.51	30.61	36.51	32.48	31.59	30.74	30.32	34.65
Case 2	34.10	32.66	27.15	31.04	35.72	30.13	34.36	31.70	31.25	30.19	30.19	32.04
Case 3	33.2	31.68	27.02	31.57	35.61	29.34	33.66	32.14	31.64	29.71	29.46	31.82
Case 4	32.08	34.61	30.93	30.98	34.52	32.72	31.34	30.44	30.8	33.01	33.08	31.96
$\sigma = 25$												
BM3D [5]	29.25	32.07	30.71	28.93	32.85	29.71	30.16	28.42	29.45	29.90	29.61	28.56
WNNM [7]	29.84	32.24	31.24	29.15	33.22	29.82	30.42	28.69	29.64	30.03	29.76	29.03
EPLL [6]	29.39	31.73	28.61	28.95	32.17	29.53	30.17	28.61	29.26	29.74	29.66	28.51
MLP [8]	29.61	32.25	29.54	29.25	32.56	29.73	30.30	28.82	29.61	29.97	29.88	28.82
TNRD [9]	29.85	32.00	29.41	29.18	32.53	29.71	30.57	28.88	29.72	29.91	29.87	29.02
DnCNN [2]	30.28	32.44	30.00	29.43	33.06	30.12	30.87	29.13	30.18	30.21	30.10	29.41
DnCNNB [2]	30.25	32.42	29.69	29.35	33.05	30.10	30.84	29.09	29.94	30.20	30.09	29.34
FFDNet [32]	30.14	32.59	29.98	29.43	33.27	30.18	30.79	29.05	30.06	30.23	30.10	29.33
Case 1	30.3	29.71	25.69	27.45	32.77	27.77	30.64	27.79	29.08	27.79	27.79	28.47
Case 2	30.98	30.3	26.24	29.25	32.29	28.3	31.35	28.49	29.89	28.39	28.51	29.59
Case 3	28.2	27.49	24.61	26.77	29.54	25.88	28.51	25.72	28.14	26.17	25.97	26.36
Case 4	30.67	32.67	29.46	29.65	33.16	30.73	30.31	29.38	29.61	31.02	31.03	30.25
$\sigma = 35$												
BM3D [5]	27.58	30.56	28.98	27.40	31.36	28.15	28.51	26.83	27.92	28.43	28.22	26.86
MLP [8]	27.97	30.82	27.62	27.72	31.18	28.24	28.54	27.22	28.08	28.53	28.47	27.12
DnCNN [2]	28.51	30.91	28.09	27.94	31.61	28.52	29.14	27.52	28.61	28.72	28.66	27.53
FFDNet [32]	28.54	31.20	28.29	28.02	31.99	28.68	29.18	27.47	28.54	28.82	28.70	27.58
Case 4	29.46	31.24	28.23	28.34	32.08	29.30	29.27	28.27	28.36	29.55	29.62	28.69
$\sigma = 50$												
BM3D [5]	25.82	29.05	27.22	25.90	29.69	26.46	26.68	25.10	26.13	26.78	26.81	25.04
MLP [8]	26.26	29.32	25.24	26.12	29.64	26.67	26.68	25.56	26.37	27.03	27.06	25.43
DnCNN [2]	26.78	29.39	26.22	26.48	30.00	26.90	27.32	25.87	27.03	27.20	27.24	25.70
FFDNet [32]	26.88	29.68	27.32	26.58	30.43	27.32	27.43	25.90	27.03	27.30	27.07	25.77
Case 4	27.92	29.52	26.92	26.60	30.53	27.75	27.87	26.65	26.70	27.86	28.07	27.0
$\sigma = 75$												
BM3D [5]	23.91	27.25	25.12	24.18	27.51	24.70	24.73	23.48	24.32	25.12	25.32	23.27
MLP [8]	24.40	27.68	23.39	24.55	27.78	25.02	24.88	23.87	24.63	25.44	25.59	23.57
DnCNN [2]	24.71	27.54	23.63	24.71	27.85	24.97	25.17	24.03	25.07	25.47	25.64	23.64
FFDNet [32]	24.99	27.97	24.24	24.94	28.43	25.49	25.39	24.18	25.29	25.75	25.29	25.82
Case 4	25.33	26.74	24.75	23.96	27.83	25.40	25.54	23.99	24.21	25.50	25.71	24.36

- Case 2 and case 3

Case 2 outperforms case 3 significantly when noise is high ($\sigma = 25$), although it maintains comparable PSNR values when noise levels are less ($\sigma < 20$). It is visible in Table 3

that as the noise (σ) goes above 30 all these cases fail to beat the state-of-the-art methods.

These comparisons lead this work in the direction of case 1. A slight modification is done in case 1 i.e., use of a dilation layer with a larger receptive field. Further, previous

methods validate the need of batch normalization so batch normalization layers are also added in this case 4.

- Case 4

This case is the same as case 1 with some variations. This network is designed with Batch Normalization (BN) layer [15], where Conv + BN + ReLu series is utilized. Utilization of Batch normalization slows down the process of convergence due to increased number of parameters but it is essential to reduce the problem of covariance shift in a network.

Recalling the inabilities of previous network of case 1 with 7 layers, there is a need of increasing the number of layers. This network with BN has 9 layers. Receptive field of this network comes out to be 51×51 (Table 4). Network structure is shown in Fig. 8. Mean squared logarithmic error (MSLE) is used in this case unlike MSE in previous cases. As name suggests it is the (Mean Squared Error) MSE calculated over logarithmic error values S and \hat{S} (shown in Eq. (10)).

$$L(S, \hat{S}) = \frac{1}{N} \sum_{i=0}^N (\log(S_i + 1) - \log(\hat{S}_i + 1))^2 \quad (10)$$

where S and \hat{S} are ground truth and predicted denoised images respectively and N denotes the number of samples. 1 is added to both S and \hat{S} to validate this equation mathematically when $S, \hat{S} \rightarrow 0$.

Comparison of case 4 with the state of the art is shown in Table 3, where case 4 surpasses BM3D [5], WNNM [8], MLP [8] and DnCNN [2] by a margin of at-least 0.4 dB for a range of noise level beyond ($\sigma > 25$). For noise levels less than ($\sigma < 25$) case 4 produces comparable results, this is because of the larger receptive field. Lower noise levels are removed better by methods having larger modeling capacity e.g. DnCNN [2]. Case 4 worked well for $\sigma = 25, 35$ and 50 , even for $\sigma = 75$, case 4 outperformed FFDNet [33] in 4 out of 12 images.

Table 4 Receptive field of the network of case 4

	rcp_1	rcp_2	rcp_3	rcp_4	rcp_5	rcp_6	rcp_7	rcp_8	rcp_9
Case4	3×3	7×7	13×13	21×21	31×31	39×39	45×45	49×49	51×51

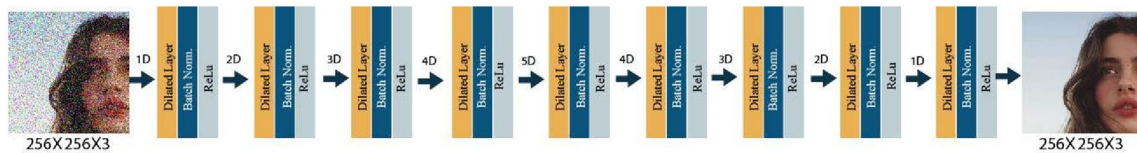


Fig. 8 The network architecture with Batch Normalization

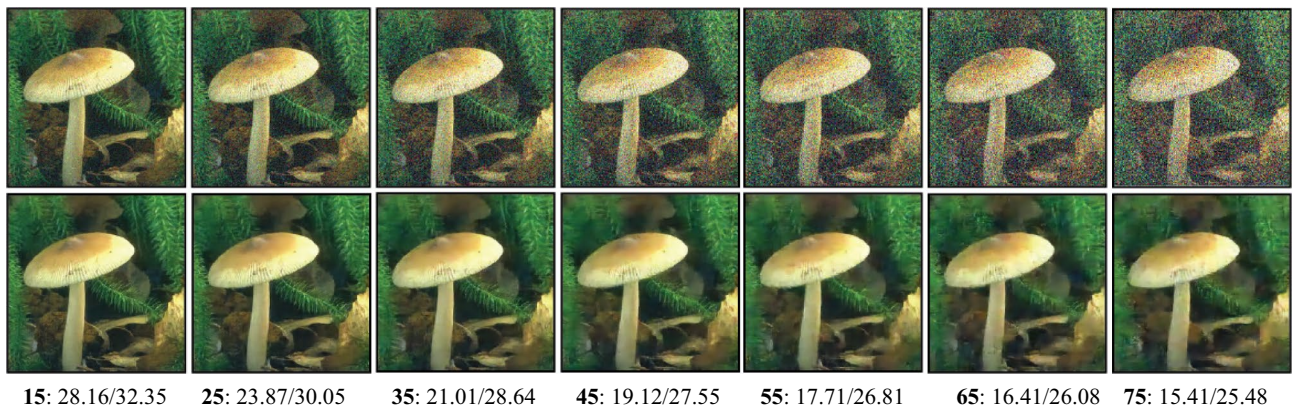


Fig. 9 Image denoising results for noise levels $\sigma = 15, 25, 35, 45, 55, 65$ and 75 (written in bold). PSNR values (Noisy/Ground Truth) of noisy and predicted image are shown, where upper row belongs to the noisy image set and lower row shows predicted image

7 Discussion

Case 4 proved to be our best model so far according to Table 3, thus this will be used as a predicting model for test sets. Figure 9 shows predicted images by case 4 for a series of noise levels ($\sigma = 15, 25, 35, 45, 55, 75$). Where an image from CBSD68 [30] is corrupted with a series of noise levels to create a set of synthetic images.

Figure 10 shows predicted PSNR results on popular test images from BSD68 [30] and CBSD68 [30], used by previous methods. Ours (Case4) performs best on BSD68 [30] than CBSD68 [30]. For color images FFDNet works best among others. Average of the predicted PSNR values computed on 68 images of BSD68 [30] and CBSD68 [30] dataset is shown in Table 5, DnCNN [2] and FFDNet [33] are used for comparison. where bolded values specify the maximum values among the comparing methods, ours work best on BSD68 [30] and RIDNet [39] performs best on CBSD68 [30]. Results on Real world dataset (RNI15) [42] were compared (Fig. 11) with DnCNN [2], FFDNet [33] and RIDNet [39], these images contain spatially variant noise. Given the fact that our method is not specifically designed for this type

of noise, it performs surprisingly well. Here in these images, it can be clearly seen that for the Flower image it outperforms [39]. Result on Pattern 3 image is also shown, no doubt that overall noise reduction is better done by RIDNet but by analyzing these images carefully one can conclude that an important piece of data is missing in RIDNet’s [39] output. The inset image shown in the bottom illustrates the loss in image quality with RIDNet and FFDNet. Quality of the images are better compared with zoomed out versions of resultant images. Overall performance of our model on these datasets validates the broad spectrum of our method.

8 Conclusion

The presented paper incorporates a detailed study on receptive fields. It includes significance of receptive field in image denoising [46, 47] problem and calculation of receptive field of a network as well as its layers. To achieve this goal several comparison techniques were utilized in network design and training. Performances on different sizes of receptive fields were compared. Our work not only provides a comparative

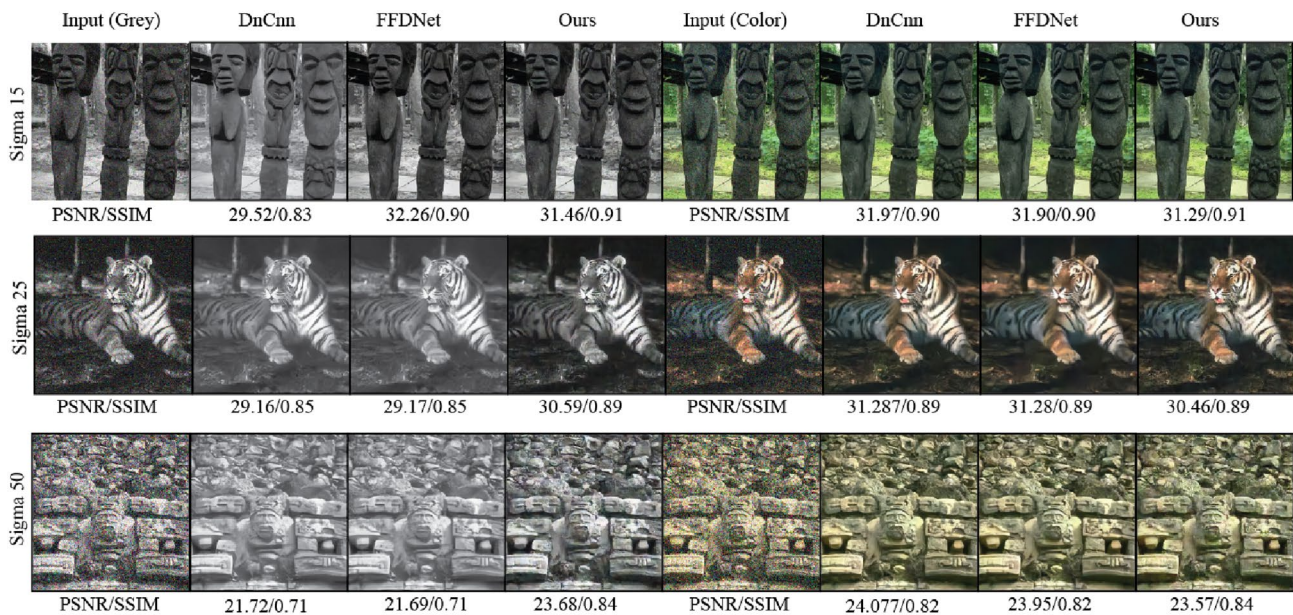


Fig. 10 Image denoising results on BSD68 (grey) and CBSD68 (color) dataset for sigma value 15, 25 and 50

Table 5 Average PSNR result comparison

Method		DnCNN [2]		FFDNet [33]		RIDNet [39]		DSNet [37]		Ours	
		BSD68	CBSD68	BSD68	CBSD68	BSD68	CBSD68	BSD68	CBSD68	BSD68	CBSD68
$\sigma = 15$	PSNR	31.77	33.92	31.57	33.97	31.81	34.01	31.69	33.91	33.21	32.93
$\sigma = 25$	PSNR	29.32	31.29	29.35	31.32	29.34	31.37	29.22	31.28	30.77	30.61
$\sigma = 50$	PSNR	26.33	28.01	26.41	28.07	26.40	28.14	26.29	28.05	27.70	27.61

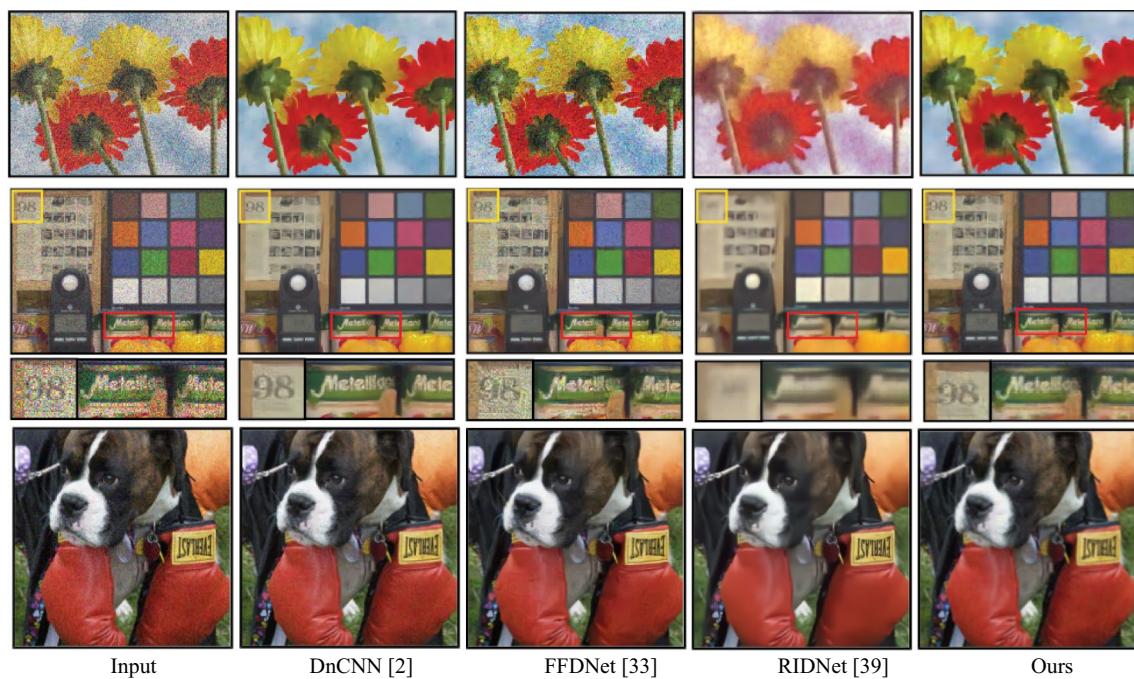


Fig. 11 Test Results on RNI15 and NC12 dataset

study but also solves the problem of image denoising. With these variations we were able to produce results that are competitive to the state of the art. Previous methods articulated the crucial need of residual learning whereas present work is able to perform without residual learning. Proposed work is an end-to-end approach that only requires a noisy image. A single network can work well for a wide range of sigma (σ) values of grayscale and colored images. The results on real noisy images further demonstrated that our work can deliver perceptually appealing denoised results when compared with BM3D [5], WNNM [8], EPLL [7], TNRD [9] and DnCNN [2]. Though this work did not concern spatially variant noise in its methodology, it can compete with the method RIDNet [39] designed specifically for spatially variant noise.

9 Future scope

There is a continuous effort on image enhancement [48, 49] and restoration [50] in various fields, despite that performance on real images is still lacking. This is due the fact that any simulated noise is much simpler than the real noise. In real life components as illumination, camera shaking and sensors are accountable for degrading the image. Thus, a further powerful noise modeling is required that can handle a variety of noise.

Funding The corresponding author has not received any funding.

Data availability All the data used in this work is available online.

Declarations

Ethical approval Not applicable.

Consent to participate Not applicable.

Consent for publication Not applicable.

References

1. Zhang K, Zuo W, Gu S, Zhang L (2017) Learning deep CNN denoiser prior for image restoration. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, pp 2808–2817. <https://doi.org/10.1109/CVPR.2017.300>
2. Zhang K, Zuo W, Chen Y, Meng D, Zhang L (2017) Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising. *IEEE Trans Image Process* 26(7):3142–3155. <https://doi.org/10.1109/TIP.2017.2662206>
3. Liu W, Lin W (2013) Additive white Gaussian noise level estimation in svd domain for images. *IEEE Trans Image Process* 22(3):872–883. <https://doi.org/10.1109/TIP.2012.2219544>
4. Mihcak KM, Kozintsev I, Ramchandran K, Moulin P (1999) Low-complexity image denoising based on statistical modeling of wavelet coefficients. *IEEE Signal Process Lett* 6(12):300–303. <https://doi.org/10.1109/97.803428>
5. Dabov K, Foi A, Katkovnik V, Egiazarian K (2007) Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE*

- Trans Image Process 16(8):2080–2095. <https://doi.org/10.1109/TIP.2007.901238>
6. Zoran D, Weiss Y (2011) Learning models of natural image patches to whole image restoration. In: 2011 International Conference on Computer Vision, Barcelona, pp 479–486. <https://doi.org/10.1109/ICCV.2011.6126278>
 7. Gu S, Zhang L, Zuo W, Feng X (2014) Weighted nuclear norm minimization with application to image denoising. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, pp 2862–2869. <https://doi.org/10.1109/CVPR.2014.366>
 8. Burger CH, Schuler JC, Harmeling S (2012) Image denoising: can plain neural networks compete with BM3D? In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, pp 2392–2399. <https://doi.org/10.1109/CVPR.2012.6247952>
 9. Chen Y, Pock T (2017) Trainable nonlinear reaction diffusion: a flexible framework for fast and effective image restoration. IEEE Trans Pattern Anal Mach Intell 39(6):1256–1272. <https://doi.org/10.1109/TPAMI.2016.2596743>
 10. Dong W, Zhang L, Shi G, Li X (2013) Nonlocally centralized sparse representation for image restoration. IEEE Trans Image Process 22:1620–1630. <https://doi.org/10.1109/TIP.2012.2235847>
 11. Dong C, Loy CC, He K, Tang X (2016) Image super-resolution using deep convolutional networks. IEEE Trans Pattern Anal Mach Intell 38(2):295–307. <https://doi.org/10.1109/TPAMI.2015.2439281>
 12. Xu L, Ren JS, Liu C, Jia J (2016) Deep convolutional neural network for image deconvolution. Adv Neural Info Process Syst 1:1790–1798
 13. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 15:1929–1958
 14. Kaiming H, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, pp 770–778. <https://doi.org/10.1109/CVPR.2016.90>
 15. Ioffe S, Szegedy C (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. In: Proc. of the 32nd International Conference on International Conference on Machine Learning, Lille, France, 37:448–456. [arXiv:1502.03167v3](https://arxiv.org/abs/1502.03167v3)
 16. Krizhevsky A, Sutskever I, Hinton EG (2015) ImageNet classification with deep convolutional neural networks. Assoc Comput Mach 60:89–90. <https://doi.org/10.1145/3065386>
 17. Szegedy C et al (2015) Going deeper with convolutions. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, pp 1–9. <https://doi.org/10.1109/CVPR.2015.7298594>
 18. Luo W, Li Y, Urtasun R, Zemel R (2016) Understanding the effective receptive field, in Deep Convolutional Neural Networks. In: Advances in Neural Information Processing Systems, Barcelona, pp 4898–4906. [arXiv:1701.04128](https://arxiv.org/abs/1701.04128)
 19. Yu F, Koltun V (2016), Multi-scale context aggregation by dilated convolutions. In: International Conference on Learning Representations (ICLR), San Juan, Puerto Rico. [arXiv:1511.07122v3](https://arxiv.org/abs/1511.07122v3)
 20. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. In: 3rd International Conference for Learning Representations (ICLR), San Diego, pp 1404–1556. [arXiv:1409.1556v6](https://arxiv.org/abs/1409.1556v6)
 21. Yu F, Koltun V, Funkhouser T (2017) Dilated residual networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CPVR), Honolulu, HI, pp 636–644. [arXiv:1705.09914v1](https://arxiv.org/abs/1705.09914v1)
 22. Oord et al (2016) WaveNet: a generative model for raw audio, pp 1–15. arxiv.org/abs/1609.03499
 23. Kalchbrenner N, Espenholt L, Simonyan K, Oord DVA, Graves A, Kavukcuoglu K (2016) Neural machine translation in linear time. [arXiv:1610.10099v2](https://arxiv.org/abs/1610.10099v2)
 24. Dumoulin V, Visin F (2016) A guide to convolution arithmetic for deep learning, [arXiv:1603.07285v2](https://arxiv.org/abs/1603.07285v2)
 25. Chandel R, Gupta G (2013) Image filtering algorithms and techniques: a review. Int J Adv Res Comput Sci Softw Eng 3:198–202
 26. Sen-Jong L (1960) Digital image enhancement and noise filtering by use of local statistics. IEEE Trans Pattern Analysis Machine Intell 2:165–168. <https://doi.org/10.1109/TPAMI.1980.4766994>
 27. Chen J, Benesty J, Huang Y, Doclo S (2006) New insights into the noise reduction Wiener filter. IEEE Trans Audio Speech Lang Process 14(4):1218–1234. <https://doi.org/10.1109/TSA.2005.860851>
 28. Tomasi C, Manduchi R (1998) Bilateral filtering for gray and color images, In: 6th International Conference on Computer Vision, Bombay, India, pp 839–846. <https://doi.org/10.1109/ICCV.1998.710815>
 29. Lan X, Roth S, Huttenlocher D, Black JM (2006) Efficient belief propagation with learned higher-order Markov random fields. In: Proc. of the European Conference on Computer Vision (ECCV), Springer, LNCS; 3952:269–282. https://doi.org/10.1007/11744047_21
 30. Martin D, Fowlkes C, Tal D, Malik J (2001) A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, Proceedings Eighth IEEE International Conference on Computer Vision, vol 2, pp 416–423. <https://doi.org/10.1109/ICCV.2001.937655>
 31. Buades A, Coll B, Morel JM (2005) A non-local algorithm for image denoising, In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 2:60–65. <https://doi.org/10.1109/CVPR.2005.38>
 32. Zhang K, Zuo W, Zhang L (2018) FFDNet: toward a fast and flexible solution for CNN-based image denoising. IEEE Trans Image Process 27:4608–4622. <https://doi.org/10.1109/TIP.2018.2839891>
 33. Jain V, Seung S (2009) Natural image denoising with convolutional networks. Adv Neural Info Process Syst. <https://doi.org/10.5555/2981780.2981876>
 34. Xiong Z, Orchard TM, Zhang Y (1997) A deblocking algorithm for JPEG compressed images using over complete wavelet representations. IEEE Trans Circuits Syst Video Technol 7:433–437. <https://doi.org/10.1109/76.564123>
 35. Chuah HJ, Khaw YH, Soon CF, Chow C (2017) Detection of Gaussian noise and its level using deep convolutional neural network. In: TENCON IEEE Region 10 Conference, Penang, pp 2447–2450. <https://doi.org/10.1109/TENCON.2017.8228272>
 36. Wang T, Sun M, Hu K (2017) Dilated deep residual network for image denoising. In: IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI), Boston, MA, pp 1272–1279. [arXiv:1708.05473v3](https://arxiv.org/abs/1708.05473v3)
 37. Peng Y et al (2018) Dilated Residual networks with symmetric skip connection for image denoising. Neurocomputing 345:67–76. <https://doi.org/10.1016/j.neucom.2018.12.075>
 38. Anwar S, Barnes N (2019) Real image denoising with feature attention. In: The IEEE International Conference on Computer Vision (ICCV), Seoul, Korea, pp 3155–3164. [arXiv:1904.07396v1](https://arxiv.org/abs/1904.07396v1)
 39. Romano Y, Elad M, Milanfar P (2017) The little engine that could: Regularization by denoising (red). SIAM J Imaging Sci. [arXiv:1611.02862v3](https://arxiv.org/abs/1611.02862v3)
 40. Guo S, Yan Z, Zhang K, Zuo W, Zhang L (2018) Toward convolutional blind denoising of real photographs. [arXiv:1807.04686](https://arxiv.org/abs/1807.04686)

41. Fleet D et al (2014) Microsoft COCO: common objects in context, In: *Computer Vision – European Conference on Computer Vision (ECCV)*, LNCS, Springer, Cham, vol 8693
42. Lebrun M, Colom M, Morel J-M (2015) The noise clinic: a blind image denoising algorithm. *Image Processing On Line* 5:1–54. <https://doi.org/10.5201/ipol.2015.125>
43. Nam S, Hwang Y, Matsushita Y, Kim SJ (2016) A holistic approach to cross-channel image noise modeling and its application to image denoising. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 1683–1691. <https://doi.org/10.1109/CVPR.2016.186>
44. Maas AL, Hannun AY, Ng AY (2013) Rectifier nonlinearities improve neural network acoustic models, In: *Proc. International Conference on Machine Learning (ICML)* 30(1): 3. [arXiv:1804.02763v1](https://arxiv.org/abs/1804.02763v1)
45. Kingma DP, Ba JL (2015) Adam: a method for stochastic optimization, *International Conference. Learning. Representation*, pp 1–41. [arXiv:1412.6980v9](https://arxiv.org/abs/1412.6980v9)
46. Hussain J (2022) Vanlaluata Image denoising to enhance character recognition using deep learning. *Int J Inf Tecnol.* <https://doi.org/10.1007/s41870-022-00931-y>
47. Dhanushree M, Priyadharsini R, Sree Sharmila T (2019) Acoustic image denoising using various spatial filtering techniques. *Int J Inf Tecnol* 11:659–665. <https://doi.org/10.1007/s41870-018-0272-3>
48. Kumar M (2019) Priyanka Various image enhancement and matching techniques used for fingerprint recognition system. *Int J Inf Tecnol* 11:767–772. <https://doi.org/10.1007/s41870-017-0061-4>
49. Gupta S, Gupta R, Singla C (2017) Analysis of image enhancement techniques for astrocytoma MRI images. *Int J Inf Tecnol* 9:311–319. <https://doi.org/10.1007/s41870-017-0033-8>
50. Nair RS, Domnic S (2022) Deep-learning with context sensitive quantization and interpolation for underwater image compression and quality image restoration. *Int J Inf Tecnol.* <https://doi.org/10.1007/s41870-022-01020-w>

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.