Adaptive Non-local Means Using Weight Thresholding

Asif Khan and Mahmoud R. El-Sakka(✉)

Computer Science Department, The University of Western Ontario, London, Canada
akhan644@uwo.ca, elsakka@csd.uwo.ca

Abstract. Non-local means (NLM) is a popular image denoising scheme for reducing additive Gaussian noise. It uses a patch-based approach to find similar regions within a search neighborhood and estimates the denoised pixel based on the weighted average of all pixels in the neighborhood. All weights are considered for averaging, irrespective of the value of the weights. This paper proposes an improved variant of the original NLM scheme by thresholding the weights of the pixels within the search neighborhood, where the *thresholded weights* are used in the averaging step. The threshold value is adapted based on the noise level of a given image. The proposed method is used as a two-step approach for image denoising. In the first step the proposed method is applied to generate a basic estimate of the denoised image. The second step applies the proposed method once more but with different smoothing strength. Experiments show that the denoising performance of the proposed method is better than that of the original NLM scheme, and its variants. It also outperforms the state-of-the-art image denoising scheme, BM3D, but only at low noise levels ($\sigma \leq 80$).

1 Introduction

Image denoising is the process of reducing noise artifacts from a digital image and it is one of the most fundamental problems in image processing. Noise is a random signal which affects the signal from the actual source by adding unwanted information to the signal. In digital image, noise causes random variation of brightness or color. It is usually produced during the image acquisition phase, caused by the sensors of digital cameras or scanners. Modern digital cameras have come a long way in using high quality sensors which have significantly reduced the presence of noise during image acquisition, but still noise can affect an image especially in low light conditions.

Noise in digital images can be categorized either as *additive* or *multiplicative* noise. Additive noise gets added with the image signal. It is modeled as:

$$v(i) = u(i) + \eta(i) \tag{1}$$

where $v(i)$ is the observed intensity value at pixel i, $u(i)$ is the actual raw intensity value and $\eta(i)$ is the random noise affecting pixel i. Multiplicative noise signal gets multiplied in the original image source. It is modeled as:

J. Braz et al. (Eds.): VISIGRAPP 2016, CCIS 693, pp. 493–514, 2017.
DOI: 10.1007/978-3-319-64870-5_24

$$v(i) = u(i) \times \eta(i) \tag{2}$$

The main challenge of a denoising model is to reduce noise while preserving the texture, fine details and edges of an image. A model which is able to reduce significant noise artifacts but completely blurs the entire image, to a point where only minimal visual information can be extracted, is not ideal. Similarly, a denoising method which preserves the textures in the image but fails to reduce the noise to a satisfactory level is not an effective model as well.

The denoising methods can be generally categorized as either *spatial domain approaches* or *transform domain approaches*. The term spatial domain refers to the image plane itself [8] and the methods under this domain uses the raw intensity of the pixels to generate a denoised image. In transform domain approaches, the image is transformed to another domain, e.g., frequency domain using, for example, Fourier transform or wavelet transform. The transform domain decomposes smooth regions in an image into low frequencies, while edges and subtle information into high frequencies, thus making it easier to target and enhance certain regions in an image.

Among the various noise types, additive white Gaussian noise has attracted significant interest among researchers in the past few decades. Our work will focus only on this type of noise reduction. Additive white Gaussian noise is referred to noise signals with a zero-mean Gaussian distribution, having uniform power across the frequency band. Initial approaches to reduce the additive Gaussian noise included the use of basic linear filters, namely mean filter, median filter and Gaussian smoothing [8]. These filtering approaches use only the raw pixel values in a small local neighborhood around each pixels to determine the denoised image. These methods does not take into account the extent to which the neighborhood overlaps with smooth or textured regions. Thus the use of such linear filters are detrimental for edge and texture preservation, resulting in blurry denoised images. To address this problem, Perona and Malik proposed an iterative edge preserving method called Anisotropic Diffusion [16]. It attempts to determine whether a pixel is part of a smooth or a textured region and applies different degree of smoothing based on the characteristics of its locality.

Most of the earlier spatial domain denoising methods used pixel intensities within a defined local neighborhood around each pixel for estimating a denoised version of a noisy image. In recent years, Buades et al. proposed a non-local, patch based approach called *Non-local Means* (NLM) [3,4]. It takes advantage of the fact that similar local regions can be spread through out the entire image. Each of the pixels are denoised using a weighted average of all the pixels within a defined search area. The weights are assigned based on the local characteristics of the pixels used in the weighted averaging step. It uses weighted euclidean distance of the local region around the pixel being denoised, also referred to as the reference patch, and the local regions around each of the pixels within the search area. The patches with smaller euclidean distance, i.e., patches similar to the reference patch are assigned higher weights.

The concept of non-local based approach has also been applied to denoising methods in frequency domain. Dabov et al. proposed *Block Matching and 3D*

Filtering (BM3D) [6,7], using patch based concept for image denoising. It is a two-step process, where the first step groups similar patches into blocks, followed by a transform operation and hard thresholding of the transform coefficients to generate a basic estimate of the denoised image. The basic estimate is used in the second step to generate the actual denoised image. BM3D is one of the state-of-the-art approaches for denoising additive Gaussian noise.

In the field of spatial domain denoising, non-local means demonstrated significant improvement in denoising images affected with additive Gaussian noise and researchers have continued further work on the method and have proposed improvements for it. The exhaustive search nature of non-local means makes it computationally expensive. To improve the computation cost, several methods have been proposed. Tasdizen used principal component analysis (PCA) in conjunction with non-local means [19]. The image neighborhoods are projects to a lower dimension space using PCA and the reduced subspace is used for computing similarities. A similar dimension reduction approach has also been proposed by Maruf and El-Sakka [13], where the image neighborhood are projected to a lower dimension by using *t-test*.

Along with the research focused on improving the computation performance of non-local means, work has also been done on improving the denoising performance as well. Rehman and Wang proposed SSIM-based non-local means [17], utilizing structural similarity instead of euclidean distance when comparing local characteristics between patches. Chaudhury and Singer proposed *Non-Local Euclidean Medians* [5], replacing the use of mean with median. Zhu et al. proposed a two-stage non-local means approach with adaptive smoothing parameters [25]. It generates a basic denoised image by applying NLM in the first stage and the basic image is refined one more time in the second stage by using NLM but with smaller smoothing strength.

Non-local means and its variants have been used in various imaging applications such as medical imaging, including MRI brain images [10], CT scan imaging [11] and 3D ultrasound imaging [9]. It is also used in video denoising [1,22], surface salinity detection [24] and metal artifact detection [14].

Although much work has been done to improve non-local means, there are still possibilities for further improvements. In the weighted averaging step, non-local means considers all the pixels within a defined search area. The pixel patches having significantly different details than the patch of the reference pixel being denoised are likely to deviate the estimated denoised value of the reference pixel from its true noise-free pixel intensity, even with their smaller weights. In our proposed method we have thresholded the pixel weights and only the pixels with weight higher than the cut-off weight are considered for weighted averaging. The threshold is adapted based on the noise level of the given noisy image. The proposed method is applied in a two-step approach, where the first step applies the proposed method to generate a basic denoised image and in the second step the image generated from the first step is again denoised, using a smaller smoothing parameter. Experiments have illustrated better denoising performance of the proposed method compared to existing methods, e.g., the original NLM, a variant of NLM and BM3D, both in terms of objective measurements and visual image quality.

2 Background

2.1 Non-local Means (NLM)

Buades et al. proposed a non-local based approach for image denoising [3,4]. Images have redundant or similar patterns in them and the *Non-Local Means* (NLM) approach attempts to take advantage of such self-similarities to estimate the denoised gray level value of each pixel. Instead of using only a local region around each pixel for estimating the actual intensity of the pixel, NLM uses a non-local approach by searching for similar patches, within a certain search-bound, in the image. The center pixel of each patch contributes to a weighted averaging based on the similarity between the reference and search patches.

When comparing the reference patch to a search patch, a variation of the euclidean distance is measured. The euclidean distance measures the sum of squared difference between each pixel in a patch. To give more importance to pixels near the center of the patch, a Gaussian weight distribution is used, thus resulting in the final measurement being the weighted euclidean distance, $\|N(i)-N(j)\|^2_{2,a}$, where a is the standard deviation of the Gaussian kernel and $N(i)$ and $N(j)$ are the patches around pixel i and j, respectively. The weight associated with each of the search patches is based on the similarity with the reference patch. After calculating the euclidean distance between the patches, the weight is assigned using Eq. (3),

$$w(i,j) = \frac{1}{Z(i)} e^{-\frac{\|v(N_i)-v(N_j)\|^2_{2,a}}{h^2}}, \tag{3}$$

where $v(N_i)$ and $v(N_j)$ are the gray values of the pixels in the patch centered on i and j respectively. $Z(i)$ is the normalizing constant as defined in Eq. (4),

$$Z(i) = \sum_j e^{-\frac{\|v(N_i)-v(N_j)\|^2_{2,a}}{h^2}} \tag{4}$$

The constant, h, controls the decay rate of the exponential weight function. Given a noisy image, the estimated value $NL[v](i)$, for pixel i, is computed as a weighted average of the center pixels of the patches in a certain search area, see Eq. (5),

$$NL[v](i) = \sum_{j \in I} w(i,j)v(j), \tag{5}$$

where $w(i,j)$ is the weight calculated based on the similarity of neighborhood around pixel i and j.

2.2 Two-Stage Non-local Means with Adaptive Smoothing Parameters

Zhu et al. [25] proposed a two-stage non-local means method with adaptive smoothing parameters. Based on the noise estimation of a given noisy image,

smoothing parameter h_{basic} for the first stage is selected automatically and the basic denoised image is computed, as shown:

$$\hat{y}_{i,basic} = \sum_j w_{ij,basic} y_j, \tag{6}$$

where $w_{ij,basic}$ is the weight depending on the similarity between patches i and j and satisfies the usual conditions $0 \leq w_{ij,basic} \leq 1$ and $\sum_j w_{ij,basic} = 1$. The weight is calculated as show in Eq. (7)

$$w_{ij,basic} = exp(-\frac{1}{h^2_{basic}} \|P_i - P_j\|^2) \tag{7}$$

where P_i and P_j are the patches centered on pixel i and j and h_{basic} is the smoothing parameter which controls the decay rate of the exponential function. For the first stage the smoothing parameter is set as $h_{ij,basic} = 0.75 \times \sigma$.

Most of the image noise is removed after the first stage but for high noise levels some noise artifacts still remain in the image, thus the basic image is refined one more time. The resulting $\hat{y}_{basic}$ image of the first stage is again denoised using the non-local means method but using different smoothing parameters h_{final}. The final image is computed as shown in Eq. (8)

$$\hat{y}_{i,final} = \sum_j w_{ij,final} \hat{y}_{i,basic} \tag{8}$$

Similar to the first stage, the weights between patch i and j are calculated as:

$$w_{ij,final} = exp(-\frac{1}{h^2_{final}} \|P_i - P_j\|^2) \tag{9}$$

In the second step of the process, the smoothing parameter is defined as:

$$h_{final} = \begin{cases} \frac{\sigma^2}{100}, & \sigma < 30 \\ 0.5\sigma, & \sigma \geq 30 \end{cases}.$$

2.3 Non-local Euclidean Median

Chaudhury and Singer [5] proposed the Non-Local Euclidean Median, extending the concept of the original Non-Local Means scheme. The method is derived from the observation that the median is more robust to outliers than the mean. In the presence of noise in the image, the weights averaged over all possible patches, especially in a search-bound defined around image edges and lines, will move the resulting mean towards the outliers. The mean is the minimizer of $\sum_j w_j \|P - P_j\|^2$ over all patches P. Non-Local Euclidean Median proposed the select the patch, P, which minimizes $\sum_j w_j \|P - P_j\|$ and replace the noisy pixel value at position (x, y) in the image with the pixel value of the center of patch P.

2.4 Iterative Non-local Means

Brox and Cremers [2] proposed an iterative non-local means approach. The non-local means method is applied on an image in iterative mode. In each iteration, the similarity between patches are calculated based on the result of the previous iteration. After calculating the weights, the weighted averaging is done by multiplying the weights of a patch with it's center gray value in the noisy version of the image. The proposed method did well on regular textured image but in non-regular textured image the resulting denoising image lost texture and significant blurring is observed.

2.5 Block Matching and 3D Filtering (BM3D)

In the field of image denoising, specifically for white Gaussian noise, a popular method has emerged for tackling the problem of image noise called Block Matching and 3D Filter (BM3D). It was proposed by Dabov et al. [6,7]. BM3D demonstrated superior denoising performance compared to the existing methods and have attracted the attention of researchers working in the image of image enhancement and restoration.

BM3D is a 2-step process and has been inspired by the non-local concept first introduced in Non-Local Means (NLM). In the first step, the process starts by defining a local neighborhood, also referred to as the reference patch and searches for similar patches inside a search window, usually defined as a bounded area around the reference patch. The similarity between the search patch and the reference patch is decided based on a certain threshold. If the similarity is above the threshold, the search patch is marked as one of the similar patches. After calculating the similarity between the patches inside the bounded search area, all the similar patches are stacked together, building a 3D block. BM3D applies some 3D transformations on the block to transform from spatial domain to frequency domain. After the 3D transformation, the resulting coefficients are thresholded, called *hard thresholding*, where coefficients below a certain threshold are reduced to zero. The block coefficients are transformed back to the spatial domain, using inverse 3D transform. Next, BM3D generates a basic estimate of the denoised image from the block which has been inverse transformed. For estimating the values of the reference patch, all the patches in the 3D block are aggregated and it works by assigning different weights while estimating the pixel values in the reference patch.

In the second step of BM3D another grouping of patches are carried out, similar to that of the first step. This time however the grouping of patches into 3D blocks are done based on the basic estimate obtained from the first step. After 3D blocks are generated by selecting similar patches around the reference patch, the block is transformed using a 3D transformation. Based on the transformed coefficients the restored image coefficients are estimated using the restoration concept used in Wiener filter [21]. The Wiener shrinkage coefficients are calculated for the 3D blocks, followed by inverse transform to revert back to the spatial representation of the 3D block. Finally the Wiener shrinkage coefficients are multiplied by the 3D block to get the denoised representation of the block.

3 Non-local Means Using Adaptive Weight Thresholding

Non-local means method defines a search area of size $S \times S$ centered on the pixel, i, being denoised. The similarity of all the patches defined around each of the pixels within the search area is considered during the weighted averaging process, where higher weights are assigned to patches which are more similar, as determined by lower euclidean distance to the reference patch. The goal of the weighted averaging process is to estimate the true noise-free intensity value of pixel i, based on the similarity of the patches within the defined search area of the given noisy image. The inclusion of the center pixels of patches which are not very similar to the reference patch is likely to move the resulting estimate further from the true pixel intensity value of the noise-free image.

In our proposed method, only a subset of the available patch centers are considered for the final estimation of the denoised pixel. The patches are selected based on the similarity measure compared to the reference patch. Effectively, a cut-off weight, w_{thresh} is selected using a defined percentile position, $w_{percentile}$ among the available patch weights within the bounded search area and the weights of the patches are thresholded against w_{thresh}. All weights above w_{thresh} are unchanged and weights below w_{thresh} are reduced to zero, thus removing their pixel centers from the weighted averaging process. The selected percentile position is determined based on the noise level in a given image. In real systems, the actual amount of noise in a noisy image cannot be known beforehand. The noise can be estimated in digital image using fuzzy processing [18], image filters [15] and local variance estimate [12] methods.

For low noise levels, a higher cut-off weight, w_{thresh}, is selected for thresholding the patch weights and as the noise level of a given image increases, w_{thresh} is lowered to include more patch centers for averaging. For lower noise levels, only the patches with high similarity measure to a reference patch can be used to estimate a denoised image. The remaining patches can be considered as outliers. So, a higher cut-off threshold is selected for low noise levels. In high noise, the euclidean distance measurement may not give a true measure of patch similarity as it will end up comparing, to some extent, the noise between patches along with the structures of the patches. So, considering only the higher weighted patch centers, by keeping the threshold value high, can in fact deviate the denoised estimation from the true value. To mitigate this effect, the threshold value is lowered so that more pixels are averaged for attenuating the noise. The denoised image is calculated as shown in Eq. (10),

$$NL[v](i) = \sum_{jEI} \hat{w}(i,j)v(j), \tag{10}$$

where $\hat{w}(i,j)$ is the thresholded weight between patch at pixel i and patch at pixel j as shown in Eq. (11),

$$\hat{w}(i,j) = \begin{cases} w(i,j), \text{if } w(i,j) > w_{thresh} \\ 0, \qquad\qquad\qquad \text{otherwise} \end{cases} \tag{11}$$

The proposed method is applied in two-step approach. In the first step, proposed method is used to generate a basic estimate of the denoised image. In the basic estimate, most of the noise is reduced but still some visible noise artifacts remain, especially for stronger noise levels and it is necessary to further denoise the basic image for better denoising [23]. As most of the noise is reduced in the basic image, similar regions can be identified more easily which helps to generate better denoised images in the second step. In the second step, the basic image is denoised using similar method used in the first step, but with a smaller smoothing parameters. To verify that the two-step approach is good enough, we conducted experiments to measure the improvement in the denoising performance with further steps and found the amount of improvements to be negligible, and even less in some cases.

Non-local means has two key parameters, namely the patch size and the search size. In our proposed method we have attempted to select the optimal patch and search window sizes based on the noise level in the image. We have empirically defined a model for selecting the patch size and the corresponding search window size for a noise level, σ, see Sect. 4.1.

4 Experimental Results

In this section we will report the experimental results of our proposed method. All the experiments were carried out on the standard Kodak gray-scale image set. It comprises of 24 gray-scale images of dimensions 768×512 and 512×768. The Kodak image set is shown in Fig. 1. For the purpose of our experimentation, the standard noise free image were contaminated by additive Gaussian white noise, randomly distributed throughout the image. The final intensity values were kept within the maximum intensity value of gray-scale images. The noise levels, determined by σ, ranges from 10 to 100, with a step size equals 10. The performance of our proposed method is compared with the original non-local means (NLM) [3,4], the two-stage non-local means (TS-NLM) [25] and the Block Matching and 3D Filtering (BM3D) [6,7] methods.

4.1 Parameter Selection

The primary parameters in the proposed method are:

1. Cut-off weight for thresholding
2. Patch size
3. Search window size

The parameters are determined empirically, based on experiments conducted on a set of test images. The performance measure of tuning each of the parameters are used to define the models for the parameters. The set of test of images is shown in Fig. 2. The training images are selected to address common image characteristics, including smooth regions, textured regions and fine details. The *Lena* and *Peppers* images have smooth regions, while *Barbara*, *Boats* and *Baboon* images has lot more texture and fine details. All the PSNR values reported are averaged after repeating each of the experiments 10 times.


(a) Boat2


(b) Lighthouse


(c) Woman


(d) Sail


(e) Statue


(f) Model


(g) Beach


(h) Bike


(i) Bridge


(j) Cottage


(k) Door


(l) Flower


(m) Raft


(n) Girl


(o) Hats


(p) House2


(q) Houses


(r) Windows


(s) Island


(t) Lake


(u) Landscape


(v) Lighthouse2


(w) Parrot


(x) Plane

Fig. 1. Test image set (Kodak image set).

(a) Lena (b) Barbara (c) Baboon

(d) Boats (e) Peppers

Fig. 2. Train image set.

4.2 Cut-Off Weight for Thresholding

The patch size and search window size are fixed and the weights are thresholded based on the cut-off weight percentile, $w_{percentile}$, determined by the model. For the purpose of determining the suitable thresholding model, the patch size is fixed at 7×7 and the search window size fixed at 11×11. Two different models are analyzed, the *linear model* and the *exponential model*.

Linear Model. The linear model for determining the cut-off weight percentile is defined as:

$$w_{percentile} = ceil(100 - a\sigma), \tag{12}$$

where $ceil()$ rounds a decimal value to the smallest following integer, coefficient a is a constant and σ is the standard deviation of Gaussian noise.

Different linear models are defined by changing the value of coefficient a. The PSNR comparison of using different linear models for weight thresholding is tabulated in Table 1. For noise levels $\sigma \leq 50$, a coefficient value of $a = 1$ generate better results and for $\sigma > 50$, the coefficient value $a = 0.5$ demonstrate better performance.

Exponential Model. The exponential model for determining the cut-off weight percentile is defined as:

$$w_{percentile} = ceil(100 \times e^{-0.01a\sigma}), \tag{13}$$

Table 1. PSNR comparison of linear models for different coefficient a.

Noise (σ)	$a = 0.25$	$a = 0.5$	$a = 1$
10	33.43	33.75	**33.87**
20	31.85	32.12	**32.25**
30	28.74	29.06	**29.15**
40	26.63	26.91	**26.98**
50	24.71	24.96	**25.07**
60	24.00	**24.28**	23.95
70	23.08	**23.33**	22.75
80	22.75	**23.14**	22.57
90	21.87	**22.28**	21.64
100	21.48	**21.71**	21.22
Average	25.85	**26.15**	25.94

where $ceil()$ rounds a decimal value to the smallest following integer, coefficient a is a constant and σ is the standard deviation of Gaussian noise.

The result of using the exponential model for determining the cut-off weight is shown is Table 2. For the exponential model, the coefficient value of $a = 1$ demonstrate better results for all noise levels. The average performance, as well as the noise-wise performance, of the exponential model, with coefficient $a = 1$, has better denoising performance compared to the linear models. So, the exponential model, with coefficient $a = 1$ is selected for determining the cut-off weight percentile for any given noise level, σ.

Table 2. PSNR comparison between exponential models for different coefficient a.

Noise (σ)	$a = 0.5$	$a = 1$	$a = 2$	$a = 4$
10	33.77	**33.89**	33.68	33.34
20	32.09	**32.22**	31.96	31.62
30	29.13	**29.36**	29.09	28.8
40	26.94	**27.11**	26.86	26.57
50	24.90	**25.13**	24.79	24.68
60	24.16	**24.35**	24.06	23.85
70	23.28	**23.48**	23.15	23.01
80	23.06	**23.24**	22.87	22.66
90	22.21	**22.34**	21.92	21.74
100	21.89	**22.05**	21.68	21.42
Average	26.14	**26.32**	26.01	25.77

4.3 Patch Size and Search Window Size

For varying the search window size, the patch size is fixed at 7×7. The weights are thresholded using the exponential model, selected in previous section. The result of varying the search window is shown in Table 3. The table shows that for $\sigma < 50$ the search window size 11×11 perform optimally and for $\sigma \geq 50$ the search size 21×21 performs best. To help determine a model for selecting the search window size, the denoising performance on the test images are again measured by varying the search window size, but this time using odd-integer window sizes only between 11×11 and 21×21, while using more noise levels.

Table 3. PSNR comparison by changing the search window size.

Noise (σ)	5×5	11×11	21×21	35×35
10	33.26	**34.19**	33.94	33.40
20	31.44	**32.48**	31.66	31.28
30	28.83	**29.35**	28.68	28.27
40	27.44	**28.12**	28.06	27.44
50	23.79	24.56	**25.05**	24.40
60	22.84	23.70	**24.27**	23.91
70	22.15	23.09	**23.73**	23.48
80	21.66	22.56	**23.41**	23.19
90	20.73	21.57	**22.88**	22.42
100	20.35	21.06	**22.52**	22.15
Average	25.25	26.07	**26.42**	25.99

The patch size and search window size for a given noise level are determined empirically, using an iterative learning approach on a training image set. At first, the patch size is fixed and the search window size is varied, for each noise levels, to select the best search window size. The noise levels, σ, ranges from 10 to 100, with a step size equals 5. Next, the patch size is varied for each noise levels, while using the best search window size for each noise as determined in the previous step. The best patch size for each noise level is used to find the corresponding optimal search window sizes one more time. This process is repeated until an iteration is reached where updating the optimal search window size for a noise level did not change the corresponding best patch size and vice versa.

To determine the patch and search window size models, the patch size is initially fixed at 7×7 and the search window size is varied. The average PSNR comparison of the various search window size is shown in Table 4.

For the next step of determining the patch size to be used in our proposed method, the patch size is varied while using the optimal search window for a given noise level, as shown in Table 3. The result of changing the patch size is

Table 4. PSNR comparison for fine tuning the search window size.

Noise (σ)	11×11	15×15	19×19	21×21
10	**34.16**	33.84	33.61	33.43
20	**32.50**	32.23	31.88	31.65
30	29.32	**29.38**	28.87	28.73
40	27.12	**27.79**	27.15	27.07
50	24.52	25.03	**25.15**	25.02
60	23.78	24.28	**24.39**	24.27
70	23.12	23.77	**23.88**	23.65
80	22.55	23.00	**23.62**	23.45
90	21.23	21.75	22.27	**22.56**
100	20.64	21.15	21.66	**22.02**
Average	25.89	26.21	**26.25**	26.19

shown in Table 5. The results indicated that the patch size 7×7 works optimally for noise level $\sigma < 85$ and for noise level $\sigma \geq 85$ the patch size 9×9 is optimal.

From our experiments, we have selected a patch size of 7×7 when the noise strength is, $\sigma \leq 80$ and for $\sigma > 80$ the patch size is increased to 9×9. For high noise levels, the larger patch size is needed to reduce the effect of noise in patch similarity measurement.

From our experiments, we also determined a model for selecting the search window size for a noise level, σ. The model used to select the search size $S \times S$ for a given noise, σ, is shown in Eq. (14),

Table 5. PSNR comparison by changing the patch size.

Noise (σ)	3×3	5×5	7×7	9×9
10	33.64	33.86	**34.12**	33.96
20	31.92	32.22	**32.45**	32.36
30	29.07	29.43	**29.61**	29.47
40	27.21	27.50	**27.74**	27.59
50	24.99	25.29	**25.46**	25.35
60	24.13	24.42	**24.60**	24.46
70	23.48	23.81	**24.02**	23.95
80	23.09	23.37	**23.55**	23.52
90	22.08	22.45	22.63	**22.82**
100	21.35	21.78	21.97	**22.41**
Average	26.10	26.41	**26.61**	26.59

$$S = round_{odd}(0.117\sigma + 9.758), \tag{14}$$

where, $round_{odd}()$ rounds a decimal value to its nearest odd integer. As the search window is centered on pixel, i, being denoised, the search window size needs to be an odd integer.

4.4 The Two-Step Approach

Using the patch and search window size models along with the exponential weight thresholding model, the proposed method was applied on the training image set as a multiple step approach. In each step, the proposed method was applied on the output image of the previous step. After each iteration, the PSNR of the resulting denoised image was measured. The change in the PSNR measurement, compared to the previous iteration, was used to determine the number of iterations which demonstrated satisfactory performance improvement due to the extra iteration. The PSNR comparison of multiple iterations of the proposed method is shown in Table 6. The 2-step has the optimal performance compared to the other multi-step approaches.

Table 6. PSNR comparison of proposed method for multiple steps for various noise levels.

Noise Level	1 step	2 step	3 step	4 step
10	34.11	**34.16**	33.71	33.56
20	32.43	**32.70**	32.31	32.23
30	29.55	**29.86**	29.47	29.34
40	27.78	**28.29**	27.92	27.83
50	25.44	**26.10**	25.77	25.69
60	24.66	**25.42**	25.18	25.08
70	23.99	**24.88**	24.73	24.66
80	23.61	**24.62**	24.48	24.30
90	22.79	23.96	**23.99**	23.85
100	22.48	23.76	**23.80**	23.68
Average	26.68	**27.37**	27.14	27.02

4.5 Performance Measure

To measure the performance of our proposed method in comparison to other existing denoising methods, we have used the Peak Signal to Noise Ratio (PSNR) and the Mean Structural SIMilarity (MSSIM) measure [20]. These measures are generally used for objective evaluation and measurement of various denoising methods. We also evaluated subject comparison between our proposed method and existing denoising methods.

Peak Signal to Noise Ratio (PSNR). The Peak Signal to Noise Ratio measures the ratio between the maximum possible power of a signal to the power of the noise which affects the quality of the original signal. The PSNR is usually expressed as the logarithmic decibel scale. A higher value in PSNR represents better reconstructed or denoised image. The PSNR is measured using Eq. (15),

$$PSNR = 10\log_{10}(\frac{MAX_I^2}{MSE}), \tag{15}$$

where MAX_I represents the maximum intensity of the image (255, for grayscale image) and MSE measures the mean squared error between the original image and the degraded image, as defined in Eq. (16),

$$MSE = \frac{1}{M \times N}\sum_{i=0}^{M}\sum_{j=0}^{N}(u_{ij} - v_{ij})^2, \tag{16}$$

where u_{ij} is the original image, v_{ij} is the degraded image and the size of the images is $M \times N$.

Mean Structural Similarity (MSSIM). One of the drawbacks of the PSNR measure is that it relies on the mean square error for calculating the ratio. Mean squared error considers only the differences between isolated data points. To evaluate the performance of a denoising method based on the degree of structural similarity between the original and the reconstructed image, the Structural SIMilarity (SSIM) measure is used. The SSIM measure provides a better assessment of an image restoration or denoising method. The SSIM between two blocks is defined in Eq. (17),

$$SSIM = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \tag{17}$$

where, x and y are two identical sized window or patch, μ_x and μ_y are the averages of x and y, σ_x^2 and σ_y^2 are the variance of x and y and σ_{xy} is the covariance. The mean SSIM (MSSIM), averaged over all SSIM, is used as for the quality measurement of a denoising method.

4.6 Performance Evaluation Using PSNR

Table 7 shows the PSNR comparison of the proposed method, the original non-local means, the variant of non-local means and BM3D, on the *Girl* image. Table 8 shows the average PSNR values over all images in the Kodak image set, for various noise levels. The performance of the proposed method is better than the original non-local means method and its variant for all noise levels. Yet, when compared to BM3D, our proposed method managed to produce better results only when $\sigma \leq 80$. The proposed method also demonstrated better performance than existing methods on the average of all the noise levels used in our experiments (Tables 7 and 8).

Table 7. PSNR comparison of the *Girl* image for the proposed method and existing methods.

Noise	NLM	TS-NLM	BM3D	Proposed
10	33.92	33.93	35.42	**35.61**
20	31.83	32.01	33.46	**33.60**
30	29.43	29.70	31.03	**31.12**
40	28.47	28.96	29.88	**30.04**
50	26.64	27.20	28.21	**28.27**
60	25.12	25.77	26.53	**26.60**
70	24.78	25.24	25.88	**26.03**
80	23.69	24.46	25.33	**25.50**
90	23.15	24.04	**24.95**	24.81
100	22.91	23.88	**24.43**	24.28
Average	26.99	27.52	28.51	**28.59**

Table 8. PSNR comparison of the proposed method with existing methods.

Noise	NLM	TS-NLM	BM3D	Proposed
10	32.61	32.63	34.05	**34.27**
20	30.77	30.94	32.25	**32.43**
30	28.58	28.83	29.80	**29.95**
40	27.02	27.47	28.19	**28.33**
50	24.88	25.54	26.07	**26.09**
60	23.93	24.66	25.38	**25.46**
70	23.24	24.02	24.74	**24.91**
80	22.90	23.56	24.46	**24.65**
90	22.21	23.18	**24.25**	24.13
100	21.98	22.83	**23.97**	23.84
Average	25.81	26.36	27.36	**27.41**

4.7 Performance Evaluation Using MSSIM

Table 9 shows the MSSIM comparison of the proposed method, the original non-local means, the variant of non-local means and BM3D, on the *Girl* image. Table 10 shows the MSSIM comparison over all images in the Kodak image set, for various noise levels. In terms of MSSIM, the performance of the proposed method is consistent with PSNR, which means it is better than the original non-local means and its variant for all noise levels. When compared to BM3D, our proposed method managed to produce better results only when $\sigma \leq 80$. On average across all noise levels, the performance of proposed method has been found to be better than existing methods.

Table 9. MSSIM comparison of the *Girl* image for the proposed methods with existing methods.

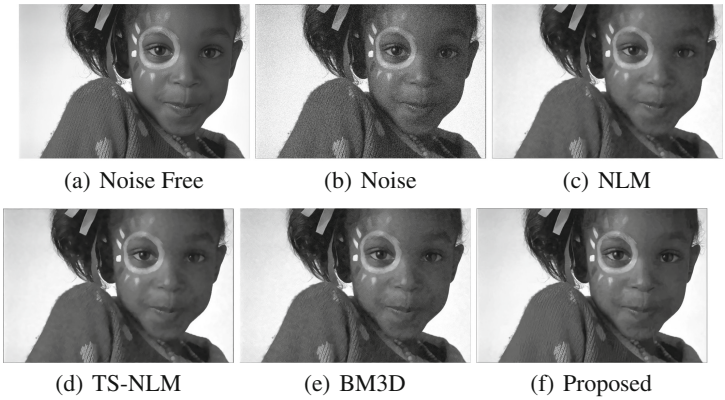
Noise	NLM	TS-NLM	BM3D	Proposed
10	0.919	0.922	0.927	**0.936**
20	0.875	0.882	0.889	**0.897**
30	0.849	0.855	0.857	**0.862**
40	0.818	0.822	0.832	**0.834**
50	0.790	0.797	0.811	**0.813**
60	0.761	0.767	0.780	**0.784**
70	0.728	0.731	0.747	**0.750**
80	0.713	0.717	0.738	**0.741**
90	0.692	0.694	**0.724**	0.720
100	0.678	0.683	**0.713**	0.708
Average	0.782	0.787	0.802	**0.804**

Table 10. MSSIM comparison of the proposed methods with existing methods.

Noise	NLM	TS-NLM	BM3D	Proposed
10	0.916	0.918	0.921	**0.932**
20	0.871	0.876	0.882	**0.891**
30	0.843	0.847	0.851	**0.857**
40	0.815	0.817	0.826	**0.829**
50	0.786	0.792	0.801	**0.806**
60	0.755	0.760	0.772	**0.778**
70	0.724	0.726	0.742	**0.744**
80	0.709	0.714	0.734	**0.735**
90	0.689	0.691	**0.712**	0.709
100	0.672	0.678	**0.708**	0.704
Average	0.777	0.782	0.795	**0.798**

4.8 Visual Quality

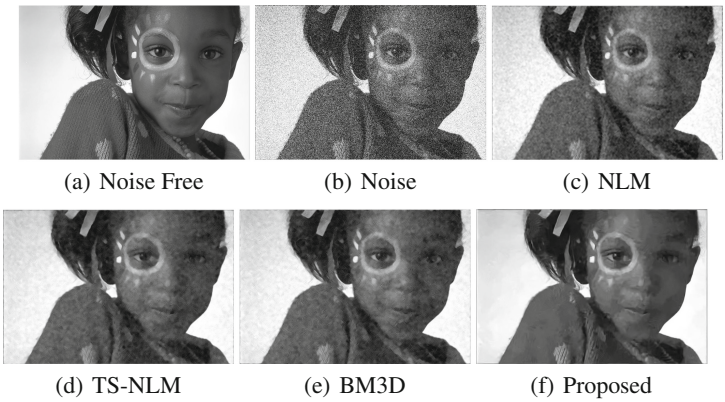
Figures 3 and 4 shows the visual comparison of the proposed method with the original non-local means (NLM), the two-stage non-local means (TS-NLM) and the Block Matching and 3D Filtering (BM3D) methods for noise level, $\sigma = 20$ and $\sigma = 70$ respectively. Figures 5 and 6 shows the visual comparison by zooming in on a particular region, the face. From Fig. 6, it can be noticed that the denoised output from the proposed method has fewer noise artifacts remaining when compared to the other methods. The blurring is also less in the denoised output of the proposed method compared to NLM, TS-NLM and BM3D.



(a) Noise Free (b) Noise (c) NLM

(d) TS-NLM (e) BM3D (f) Proposed

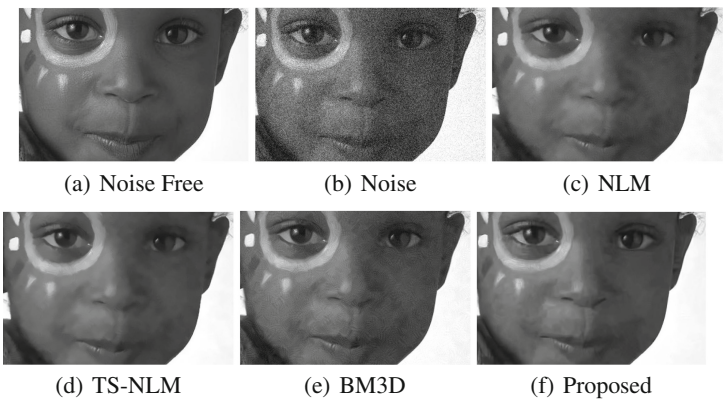
Fig. 3. Visual comparison of proposed method with existing method ($\sigma = 20$).



(a) Noise Free (b) Noise (c) NLM

(d) TS-NLM (e) BM3D (f) Proposed

Fig. 4. Visual comparison of proposed method with existing method ($\sigma = 70$).



(a) Noise Free (b) Noise (c) NLM

(d) TS-NLM (e) BM3D (f) Proposed

Fig. 5. Visual comparison (zoomed) of proposed method with existing method ($\sigma = 20$).

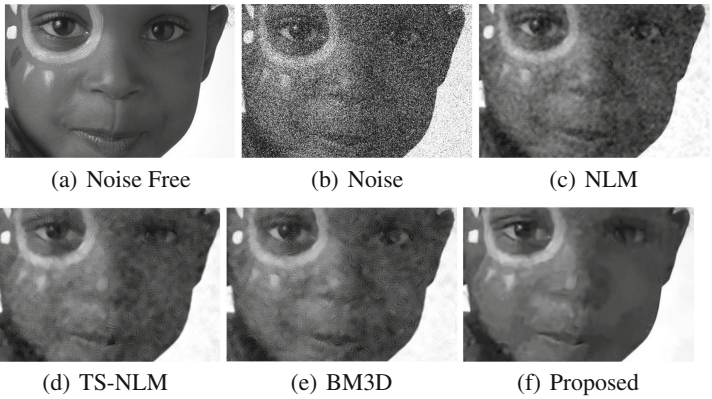


Fig. 6. Visual comparison (zoomed) of proposed method with existing method ($\sigma = 70$).

Fig. 7. Row number 100 of *Girl* image used for generating intensity profile. Scan line shown as a black line.

4.9 Intensity Profile

The image intensity profile can help analyze how similar the profile of a denoised image is to that of the original noise-free image. Figure 7 shows the chosen horizontal scan line 100 from the *Girl* image. Figure 8 shows the intensity profiles of the true image, the noisy image at noise level, $\sigma = 70$ and the profiles of denoised images produced by the original NLM scheme, the variant of NLM, BM3D and the proposed method. The Pearson correlation coefficient between the original intensity profile and the profile of the noisy and each of the denoised images is shown in Table 11 (for $\sigma = 70$).

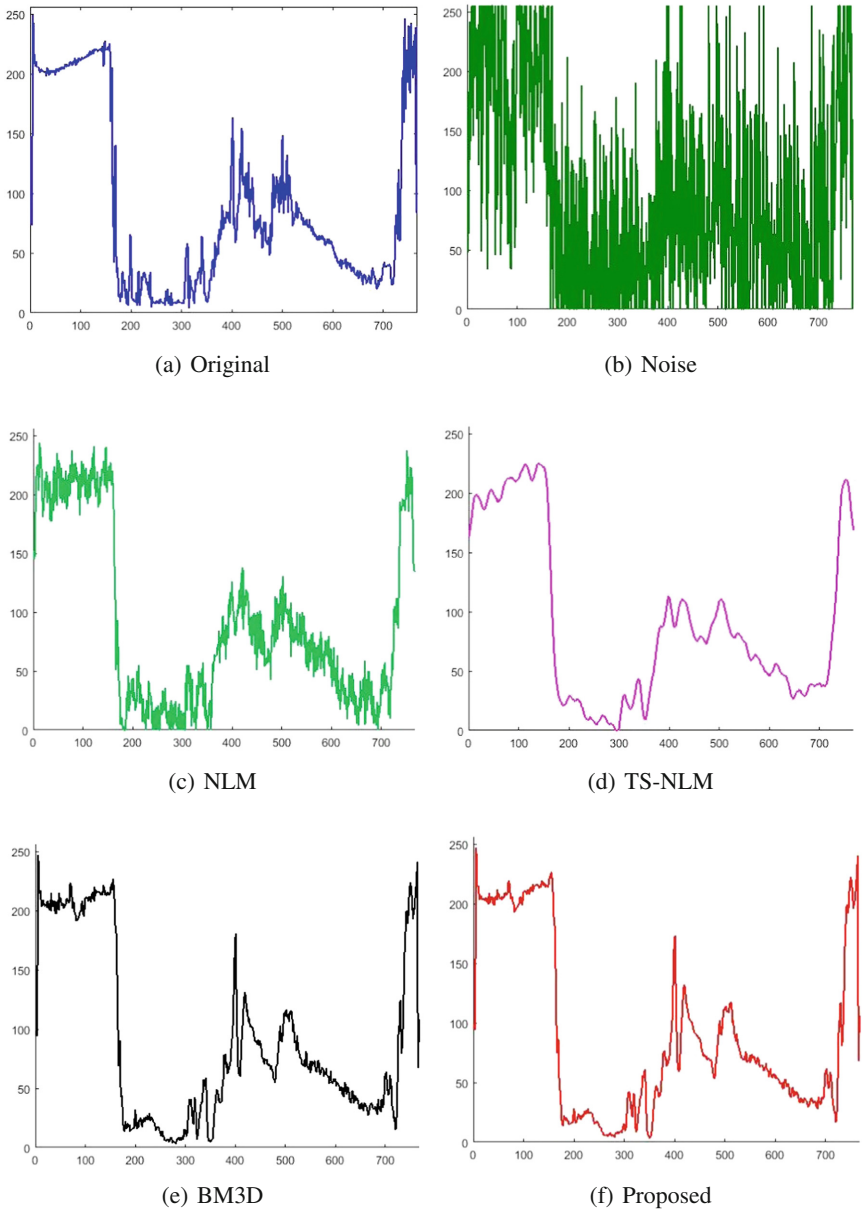


Fig. 8. Intensity profile comparison for the *Girl* image at scan line 100 ($\sigma = 70$).

The intensity profile of the proposed method shows better preservation of edges and textures, represented as sharp changes in profile graph. The original non-local means method and its variant have more noise artifacts remaining, as represented by the more jagged lines in the profile graph, closer to the origin.

Table 11. Pearson correlation coefficient comparison of the proposed method, the noisy image, the NLM method, variant of NLM and BM3D denoising scheme for noise $\sigma = 70$.

Noise	NLM	TS-NLM	BM3D	Proposed
0.680	0.975	0.980	0.988	**0.990**

When comparing the Pearson correlation coefficient, the correlation between the intensity profile of the original image and the proposed method is higher compared to those of the other existing methods. It shows that the proposed method has the closest resemblance to the intensity profile of the original image.

5 Conclusion

This paper proposed an improvement over the non-local means method, the patch-based approach for denoising additive Gaussian noise in the spatial domain. The proposed method thresholds the weights of the pixels defined around a search area of the pixel being denoised. The thresholded weights are used for weighted averaging, whereby pixels below a defined cut-off weight are ignored. The cut-off weight is determined based on the noise level estimation of an image. For a noise level, the patch and search window size are determined by a model, which is empirically defined through a learning approach. The proposed method is applied in a two-step approach for image denoising. The proposed method has demonstrated better objective and subjective denoising performance, compared to the original non-local means algorithm and its variant. When compared to BM3D, the state-of-the-art approach for image denoising, the proposed method demonstrated better results when $\sigma \leq 80$.

References

1. Basavaraja, V., Bopardikar, A., Velusamy, S.: Detail warping based video super-resolution using image guides. In: International Conference on Image Processing (2010)
2. Brox, T., Cremers, D.: Iterated nonlocal means for texture restoration. In: Sgallari, F., Murli, A., Paragios, N. (eds.) SSVM 2007. LNCS, vol. 4485, pp. 13–24. Springer, Heidelberg (2007). doi:10.1007/978-3-540-72823-8_2
3. Buades, A., Coll, B., Morel, J.: A non-local algorithm for image denoising. In: Computer Vision and Pattern Recognition (2005)
4. Buades, A., Coll, B., Morel, J.: A review of image denoising algorithms, with a new one. Multiscale Model. Simul.: SIAM Interdisc. J. **4**, 490–530 (2005)
5. Chaudhury, K.N., Singer, A.: Non-local euclidean medians. IEEE Signal Process. Lett. **19**, 745–748 (2012)
6. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising by sparse 3-D transform-domain collaborative filtering. IEEE Trans. Image Process. **16**, 2080–2095 (2007)

7. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising with block-matching and 3D filtering. In: SPIE Electronic Imaging (2006)
8. Gonzalez, R.C., Woods, R.E.: Digital Image Processing. Prentice Hall, New Jersey (2008)
9. Hu, S., Hou, W.: Denosing 3D ultrasound images by non-local means accelerated by GPU. In: International Conference on Intelligent Computation and Bio-Medical Instrumentation (2011)
10. Iftikhar, M., Rathore, S., Jalil, A., Hussain, M.: A novel extension to non-local means algorithm: application to brain MRI de-noising. In: International Multi Topic Conference (2013)
11. Kelm, Z., Blezek, D., Bartholmai, B., Erickson, B.: Optimizing non-local means for denoising low dose CT. In: International Symposium on Biomedical Imaging: From Nano to Macro (2009)
12. Lim, J.S.: Two-Dimensional Signal and Image Processing. Prentice Hall, New Jersey (1990)
13. Maruf, G.M., El-Sakka, M.R.: Improved non-local means algorithm based on dimensionality reduction. In: Kamel, M., Campilho, A. (eds.) ICIAR 2015. LNCS, vol. 9164, pp. 43–50. Springer, Cham (2015). doi:10.1007/978-3-319-20801-5_5
14. Mouton, A., Megherbi, N., Flitton, G.T., Bizot, S., Breckon, T.P.: A novel intensity limiting approach to metal artefact reduction in 3D CT baggage imagery. In: International Conference on Image Processing (2012)
15. Mouton, A., Megherbi, N., Flitton, G.T., Bizot, S., Breckon, T.P.: Gaussian noise estimation in digital images using nonlinear sharpening and genetic optimization. In: Instrumentation and Measurement Technology Conference (2007)
16. Perona, P., Malik, J.: Scale-space and edge detection using anisotropic diffusion. IEEE Trans. Pattern Anal. Mach. Intell. **12**, 629–639 (1990)
17. Rehman, A., Wang, Z.: SSIM-based non-local means image denoising. In: Proceedings of Image Processing (2011)
18. Russo, F.: Noise estimation in digital images using fuzzy processing. In: International Conference on Image Processing (2001)
19. Tasdizen, T.: Principal Components for non-local means image denoising. In: International Conference on Image Processing (2008)
20. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**, 600–612 (2004)
21. Wiener, N.: The Interpolation Extrapolation and Smoothing of Stationary Time Series. MIT Press, New York (1949)
22. Xu, Q., Jiang, H., Scopigno, R., Sbert, M.: A new approach for very dark video denoising and enhancement. In: International Conference on Image Processing (2010)
23. Zhang, L., Dong, W., Zhang, D., Shi, G.: Two-stage image denoising by principal component analysis with local pixel grouping. Pattern Recogn. **43**, 1531–1549 (2010)
24. Zhao, Y., Liu, Y.: Patch based saliency detection method for 3D surface simplification. In: International Conference on Pattern Recognition (2012)
25. Zhu, S., Li, Y., Li, Y.: Two-stage non-local means filtering with adaptive smoothing parameter. Int. J. Light Electron Opt. **125**, 7040–7044 (2014)