

## A Survey of Various Bilateral Filtering Techniques

Sweety Deswal<sup>1</sup>, Shailender Gupta<sup>2</sup> and Bharat Bhushan<sup>3</sup>

YMCA University of science and technology,  
Faridabad, India

*sweetydeswal92@gmail.com*<sup>1</sup> *Shailender81@gmail.com*<sup>2</sup>, *bhrts@yahoo.com*<sup>3</sup>

### Abstract

Noise not only deteriorates image quality but also may result in loss of important information hidden in images (i.e. medical applications). Various types of noises are available in literature such as Gaussian noise, impulse noise, mixed noise etc. In order to remove these noises from images, bilateral filters and its variants are used. This paper surveys the impact of these techniques using several performance metrics such as Peak Signal to Noise Ratio (PSNR), Mean Square Error (MSE), Mean Absolute Error (MAE) and Time Complexity. For this purpose a simulator is designed in MATLAB to implement these techniques. The results show that Joint Bilateral Filter (JBF) technique is the best technique for removing Gaussian noise, Modified Double Bilateral Filter (MDBF) technique provides good results for removing impulse noise and Switching Bilateral Filter (SBF) technique work well for mixed noise as seen from results.

**Keywords:** Gaussian noise, bilateral filter, PSNR, MSE, MAE, image de-noising

### 1. Introduction

Noise [1] affects the pixel values in the image, which can cause random variation of brightness or color information. It may occur during image capturing, acquisition process, transmission through channel etc. Various types of noises are available in literature that affects the image in a different manner as explained below:

- Gaussian noise: It mainly occurs in images during acquisition process. It is additive in nature and randomly adds any number in pixel values [1].
- Impulse noise: It is also called as salt & pepper noise [1]. This type of noise contains white pixels in darker regions and black pixels in brighter regions. This noise mainly occurs during transmission process as well during conversion of signal from analog to digital form.
- Poisson noise: It's also called as shot noise [1]. It originates from the discrete nature of the electric charge. It is caused by the random fluctuations in number of photons changed at a particular exposure level. Shot noise can also be caused by dark current flowing in electronic circuits. In case of images, it can be caused during capturing of images through camera.
- Speckle noise: This type of noise mainly occurs during transmission and reception process of images. It is basically a multiplicative noise [1] as can be seen from the equation given below:

$$J=I + n*I$$

- Where n is the uniformly distributed random noise with mean 0 and variance v. Default value of v is 0.04.

In the Figure 1 the affect of various noises on perceptible image quality is shown. It can be easily observed from Figure, that all kind of noises degrade image quality. Therefore, to remove these noises various de-noising mechanism have been proposed in literature. In this paper we analyzed the performance of Bilateral Filter [2] and its variant [5, 7, 10, 12]. The rest of the paper is organized as follows: section 2 provides the literature survey of various bilateral filter techniques. Section 3 contains experimental set up parameters and section 4 contains the results of filtering techniques followed by conclusion and references.



Original image



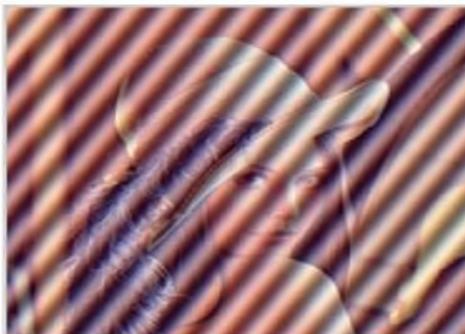
10% impulse noise corrupted image



Gaussian noise corrupted image with sigma=20



Mixed of both Gaussian and impulse



Periodic Noise corrupted image



Speckle noise corrupted image

**Figure 1. Effect of Various Noises to Image**

## 2. Literature Survey of Various Bilateral Filter Techniques

This section gives a brief introduction about the various techniques used for de-noising images.

### 2.1 Bilateral Filter

Bilateral Filter [2] was first proposed by C. Tomasi, R. Manduchi in the year 1998. It is basically a non-linear, edge-preserving and Gaussian noise reducing filter used for gray and color images. It tries to smoothens the image and at the same time preserves edges of the image. As we know that color images have three bands *i.e.* red, green and blue. If these three bands are filtered separately, the smoothness of the image at the edges will be different. Separate smoothing disturbs the balance of colors and unexpected color combinations may appear. Therefore, bilateral filters operate on three bands of color at once and can tell which colors are similar so that they can be averaged together.

The basic idea of bilateral filtering [3] is that it makes a nonlinear combination of similar pixel values. It filters the image using range and domain filter. For domain filtering, values chosen show the desired amount of combination of pixels, while the range filtering chooses values based on the desired amount of low pass filtering. A low pass domain filter on image  $I(x)$  can be defined as

$$h(x) = \frac{1}{k_r(x)} \iint_{-\infty}^{\infty} I(y) * c(x,y) dy$$

Where  $c(x,y)$  measures the geometric closeness between the neighborhood center  $x$  and a nearby point  $y$ ,  $h(x)$  is the output image and  $k_r(x)$  is the normalized constant which is calculated as

$$k_r(x) = \iint_{-\infty}^{\infty} c(x,y) dy$$

Similarly range filter is defined as

$$h(x) = \frac{1}{k_s(x)} \iint_{-\infty}^{\infty} I(y) * s(f(x),f(y)) dy$$

now  $s(f(x),f(y))$  measures the photometric similarity between the pixel at the neighborhood center  $x$  and that of a nearby point  $y$ . The normalized constant ( $k_r$ ) is replaced by

$$k_s(x) = \iint_{-\infty}^{\infty} s(f(x),f(y)) dy$$

Both geometric and photometric similarity is combined to obtain bilateral filter as follows:

$$h(x) = \frac{1}{k_r(x)k_s(x)} \iint_{-\infty}^{\infty} I(y) * c(x,y) s(f(x),f(y)) dy$$

The normalization term  $k_r(x)$  shows that the weights for all the pixels. The output image can be redefined as:

$$c(x,y) = \frac{\exp(-\|x-y\|^2 / 2\sigma_s^2)}{\exp(-\|x-y\|^2 / 2\sigma_s^2) + \exp(-\|I(x)-I(y)\|^2 / 2\sigma_r^2)}$$

Thus, mathematically at a pixel location  $(x,y)$  the output  $I(x,y)$  of the bilateral filter is calculated as follows

$$I'(x,y) = \sum_{y \in N(x)} \frac{\exp(-\|x-y\|^2 / 2\sigma_s^2) \exp(-\|I(x)-I(y)\|^2 / 2\sigma_r^2)}{\sum_{y \in N(x)} \exp(-\|x-y\|^2 / 2\sigma_s^2) \exp(-\|I(x)-I(y)\|^2 / 2\sigma_r^2)} I(x,y)$$

Where  $\sigma_s$  and  $\sigma_r$  are parameters controlling the fall-off of weights in spatial and intensity domains, respectively.  $N(x)$  is a spatial neighborhood of pixel  $I(x)$ .  $\sigma_s$  is the geometric spread parameter that is to be chosen based on the amount of the low pass filtering required. A large value of  $\sigma_s$  means it combines values from more distance in an image. Similarly  $\sigma_r$  is the parametric spread that is set to achieve the desired amount of combination of pixel

values. Now let us discuss the pros and cons of bilateral filter which will help us in understanding its variants.

#### **Advantages of Bilateral Filter:**

- It preserves edges while removing noise and causes smoothening of images.
- While working with color images, it reduces color blurriness.
- It is simple, local and non-iterative scheme to implement.

#### **Disadvantages of Bilateral Filter:**

- It cannot be used for removing of impulse noise or mixed noise.
- It replaces each pixel with filtered value whether it is noisy or not.
- It only smoothen image, doesn't sharpens.

### **2.2 Adaptive Bilateral Filter**

As discussed above, bilateral filter is a smoothening filter and it doesn't sharpen the edges of image. To overcome this limitation, an Adaptive Bilateral filter (ABF) [5] is proposed by Zhang and Allebach in the year 2008 which not only smoothes the image but also sharpens the image by increasing the slope of the edges. ABF adds an offset ( $\phi$ ) to the existing bilateral filter in order to sharpen the edges. The resultant normalization factor ( $\omega_0$ ) is calculated as:

$$\omega_0 = \sum_{i=0}^{255} \exp\left(-\frac{(\phi)^2 + (i - \mu)^2}{2\sigma^2}\right) \times \exp\left(-\frac{(\mu - \mu_0)^2}{2\sigma_0^2}\right)$$

ABF differs from its previous version in two ways. Firstly an offset  $\phi$  is introduced to the range filter so as to shift the range filter on the histogram and this causes sharpening of edges. The other modification done is making the width of the range filter, adaptive. It helps in identifying which pixel values are similar and needs to be averaged. Now let us discuss merit and demerits of this filter.

#### **Advantages of ABF:**

- It sharpens images by increasing the slope of the edges over the conventional bilateral filter.
- By combining offset and range of width filter, the technique becomes much more powerful and versatile. As a result overall quality of de-noised is improved.

#### **Disadvantages of ABF:**

- During the process of sharpening of images, ABF tends to posterize the image by pulling up or pushing down pixels along the edge slope.
- ABF does not perform well at corners of an image and does not work for impulse noise and mixed noise

### **2.3 Modified Double Bilateral Filter**

As discussed Bilateral Filter (BF) and Adaptive Bilateral Filter (ABF), both of them work well for Gaussian noise but are not able to de-noise image corrupted with impulse noise. Therefore, a Modified Double Bilateral Filter (MDBF)[7] was proposed in 2010 to remove impulse noise from images. This adds black and white pixels in the image. White dots in the image represent the salt noise and black dots represent the pepper noise. Traditionally we used Standard Median Filter (SMF) to remove impulse noise from images but it has a

drawback in decision which value should be chosen as threshold. In this filter when the window size increases, noise level as well as processing time rises too much and also exhibit blurring of de-noised image.

To overcome these problems, a decision based algorithm is introduced with the objectives of lowering the processing time and removing highly corrupted images with impulse noise, a MDBF technique was proposed. In this method the image is divided into window size of  $3 \times 3$  matrixes and the noisy pixel is figured out in the window. A pixel is said to have noise when its value is either 0 or 255. Then median value for each window is calculated and used to remove noise. Let us see these three cases:

- If pixel value is in between 0 and 255 then nothing is to be done as the pixel is noise free.
- If pixel values and median value is in between 0 and 255 then that pixel value is replaced by the median.
- If both the pixel value and median value is 0 and 255 then that pixel value is replaced by left value.

#### Advantages of Modified Double BF:

- This modified bilateral filter eliminates fair amount of impulse noise while preserving edge boundaries and fine details in image.
- It can reduce high noise densities in less time.

#### Disadvantages of Modified Double BF:

- It can't remove mixed noise from images.
- It causes blurring of images while taking larger window size and poor noise removal while taking smaller window sizes.

### 2.4 Switching Bilateral Filter

All the techniques discussed so far remove only single noise but none of them remove mixed noises *i.e.* image contaminated with two different kinds of noises. To overcome this drawback, a new filtering scheme named as Switching Bilateral filter (SBF) [8] was introduced. It is based upon the “detect and replace” methodology and for detection purpose, we used a noise detector [10] in the switching filtering technique. The absolute difference between current pixel and reference median is calculated. Depending upon the value of absolute difference, we can determine whether the pixel is noisy or not. If the value of absolute difference is large, then the processing pixel is said to have noise otherwise it is noise free. It is very important to choose a proper value of reference median in the given window, otherwise it may lead to false detection of the noisy pixel. To determine the value of reference median, we define an approach called Sorted Quadrant Median Vector (SQMV) [11].

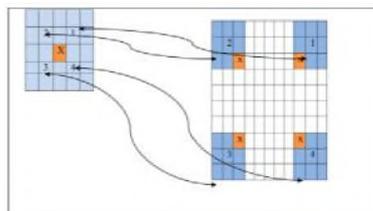


Figure 2. Forming of Four Sub-windows from a Large Window

In SQMV approach, a larger window of 5x5 is selected out of which four sub-windows of size 3x3 are formed. Median value for each window is found and then these median values are sorted. After sorting the medians of four sub-windows, some of the values are very similar while some are different, leading to the formation of different clusters. Following this, we determine the two parameters for cluster distribution  $SQMDB$  as the difference between the two boundary values of the sorted quadrant medians.

$$SQMDB = SQMV4 - SQMV1$$

$SQMD_c$  defined as the difference between the two center values of the sorted quadrant medians.

$$SQMD_c = SQMV3 - SQMV2$$

Both these parameters provide the measure of similarity between the four quadrant blocks and three texture cases i.e. without edge, weak edge and strong edge are obtained. A direction average approach is used to determine which cluster is more similar to the current pixel. The four pixel values are averaged together and represented by  $dav$ .

$$dav = (x1 + x2 + x3 + x4)/4$$

If the value of  $dav$  is close to  $SQM1$  and  $SQM2$  then  $SQM2$  is taken as reference median but if it is close to  $SQM3$  and  $SQM4$  then  $SQM3$  is chosen as reference median. The comparison between reference median and current pixel is used to calculate whether it's noisy or not.

#### Advantages of SBF:

- It removes both types of noise i.e. Gaussian noise and impulse noise while preserving edges without adding any weighting function..
- It includes a noise detector to correctly classify the noisy pixels.
- For any level of noise density window size can be fixed.
- It is non-iterative and only few parameters needed to adjust for best results.

#### Disadvantages of SBF:

- SBF removes mixed effectively but it has the disadvantage of color blurriness.

### 2.5 Joint Bilateral Filter

Many techniques have been discussed so far for removal of Gaussian noise. A new technique was proposed by O.U.Nirmal Jith and R. Venkatesh Babu in the year 2014 to achieve high quality image from two images. It performs a Non Local Means (NLM) approach for de-noising of images. It mainly consists of two steps. First step is to de-noise the image using non local means by using Patch Match algorithm for finding matching packets. A non-local means is used to average all pixels in an image. This algorithm uses a window of size  $S \times S$  out of which  $P \times P$  windows have to be made that means to perform NLM, given image of certain size is divided in to  $P \times P$  overlapping patches. Then by making use of the Patch Match,  $k$  nearest patches to these patches has to be obtained.

The joint bilateral filtering technique [12] is firstly used to combine a high quality image from the other two images, one is captured with flash and other is captured without flash [13]. But now we consider two images for joint bilateral filtering taken from first NLM step one is noisy image and other is de-noised image. The output obtained from first step is the de-noised image with over smoothing in edge regions.

Let us consider an image of size  $M \times N$ . The joint bilateral filtered pixel value  $if(xoy0)$  at  $(xoy0)$  from the noisy image  $I_n$  is obtained using the equation given below.

$W(00)$  represents the window in the pixel location  $(00)$  and  $(x,y)$  represents the pixel locations within the window. The joint bilateral filtering process is done in the patches  $p$ 's with patch size of  $(2 * p'+1) \times (2 * p'+1)$ .

$$W(00) = \frac{1}{\sum_{(x,y) \in (00)} \sum_{(x+i,y+j) \in (00)} \exp(-\frac{d(x+i,y+j, x,y)}{\sigma_r})} \sum_{(x+i,y+j) \in (00)} C_r(x+i,y+j) \exp(-\frac{(I(00) - I(x+i,y+j))^2}{\sigma_c})$$

$\sigma_r$  = range filter weight,  $\sigma_c$  is the Euclidean distance between  $(00)$  and  $(x, y)$

$$C_r(x,y) = \exp(-\frac{d(x,y)}{\sigma_r})$$

$C_r$  = range filter weights and given as

$$C_r(x,y) = \exp(-\frac{(I(00) - I(x,y))^2}{\sigma_c})$$

Normalization term  $K$  is given as

$$K(00) = \sum_{(x,y) \in (00)} \sum_{(x+i,y+j) \in (00)} C_r(x+i,y+j) \exp(-\frac{(I(00) - I(x+i,y+j))^2}{\sigma_c})$$

The parameters for window size  $W$ ,  $O_d$ ,  $O_r$  are selected on the basis of noise density. **Advantages of JBF:**

- It provides better perceptual quality as well as high value of PSNR as compared to above discussed Gaussian noise removal techniques.
- It performs well on high value of noise densities.
- In the next section simulation setup and various performance metrics are discussed.

### 3. Simulation set up

To compare the performance of various Bilateral Filter techniques, a simulator is designed in MATLAB - 09. It evaluates the performance of all the techniques using following performance metrics:

#### 3.1 Performance Evaluation Metrics

Following parameters are used to measure the performance of various noise removal techniques.

**3.1.1 Mean Square Error:** It is the cumulative squared error between the final de-noised image and the original image. This enables us to compare mathematically =  $\frac{1}{M \times N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [I_c(x,y) - \tilde{I}_c(x,y)]^2$  as to which method provides better results. For color image it can be expressed as:

$$MSE = \frac{1}{M \times N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [I_c(x,y) - \tilde{I}_c(x,y)]^2$$

Where:

$M \times N$  is the size of image (height and width respectively)

$C=1$  to  $3$  denotes the Red, Green and Blue color plane respectively

$F^c(x, y)$ =value of pixel at position  $(x, y)$  in  $c$  color plane of original image

$F^{c'}(x, y)$ =value of pixel at position  $(x, y)$  in  $c$  color plane of de-noised image

**3.1.2. Mean Absolute Error:** It is the absolute error between the original image and the de-noised image. It represents the average value of introduced deviation per pixel with respect to original image. It is given by

$$MAE = \frac{1}{M \times N \times 3} \sum_{c=1}^3 \sum_{x=1}^M \sum_{y=1}^N |F^c(x, y) - \hat{F}^c(x, y)|^2$$

Where:

M x N is the size of image (height and width respectively)

C=1 to 3 denotes the Red, Green and Blue color plane respectively

F<sup>c</sup>(x, y)=value of pixel at position (x, y) in c color plane of original image

F<sup>c</sup>(x, y)=value of pixel at position (x, y) in c color plane of de-noised image

**3.1.3 Peak Signal to Noise Ratio (PSNR):** It is the measure of quality of the image by comparing de-noised image with original image. It is an expression used to depict the ratio of maximum possible power of image (signal) and the power of the corrupting noise that affects the quality of its representation. It is represented in terms of mean square error as:

$$PSNR = 10 \log_{10} \left( \frac{MAX^2}{MSE} \right)$$

MAX is the maximum possible pixel value of the image. It is equal to 255 for 8 bit represented image.

**3.1.4 Perceptual Quality:** Instead of de-noised image should possess high PSNR and Low MSE, MAE; the de-noised image should be smooth, clean and clear also. De-noised image should be so fine for human observer as if it seems natural image. It should not have color blurriness or any odd looking structure.

**3.1.5 Normalized Color Difference (NCD):** It is used to measure the degradation in color quality in color images since it approaches the human perception. It is defined as:

$$NCD = \frac{\sum_{x=1}^M \sum_{y=1}^N \Delta}{M \times N}$$

where M, N are the image dimensions.

**3.1.6 Time Complexity:** It is the total time taken to execute an algorithm. It is defined in terms of seconds.

**3.2 Set up Parameters:** The simulator is designed for various techniques and it uses various set up parameters values. The Table 1 gives parameters values common to all techniques, whereas parameters value given in Table 2 to Table 6 are used for different techniques.

**Table 1. Simulation Set up Parameters**

Image	256*256 (gray)-Einstein 256*256 (color)-basket
Image type	Png
Impulse noise	5% 10% 15% 20% 25%
Gaussian noise	Standard deviation()=5, 10, 15, 20, 25
Mixed noise	Impulse + Gaussian noise
Simulation tool	MATLAB 2009
Processor	Core-i31.73 GHz RAM 4GB

**Table 2. Bilateral Filter Parameters for Gray Image**

Symbol	Parameter	Value
W	Window size	5
<i>ad</i>	Spatial domain standard deviation	3
<i>a</i>	Intensity domain standard deviation	0.1
N	Gaussian noise intensity	0.03

**Table 3. Bilateral filter Parameters for Color Image**

Symbol	Parameter	Value
W	Window size	5
<i>ad</i>	Spatial domain standard deviation	3
<i>a</i>	Intensity domain standard deviation	10
N	Gaussian noise intensity	0.03

**Table 4. Modified Double Bilateral Filter**

Symbol	Parameter	Value
Ws	Window size	3
N	Salt and pepper noise intensity	0.1
N	Noise components	0 and 255

**Table 5. Switching Bilateral Filter**

Symbol	Parameter	Value
W	Window size	5
	Reference median	40
N	Salt & pepper noise	0.2
Tk1	Threshold 1	[25,30]
Tk2	Threshold 2	[5,10,15]
<i>a</i>		3,1
<i>a</i>		[30,50]

**Table 6. Adaptive Bilateral Filter**

Symbol	Parameter	Value
W	Window	3
<i>ad</i>	Spatial domain filter	1.0

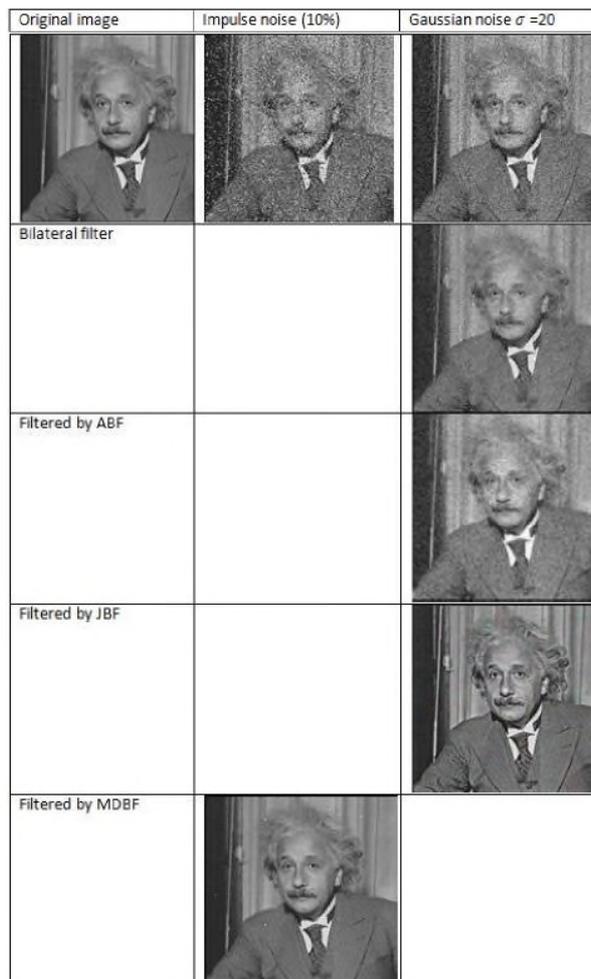


## 4. Results

In this section results of simulation based on performance metrics are discussed as follows:

### 4.1 Perceptual Quality

Figure 3 shows the snapshots of the outputs of applying various techniques mentioned above to grey images. From the results of perceptual quality, it can be observed that best results are obtained in case of images corrupted with Gaussian noise by JBF technique followed by ABF and BF in that order. For images corrupted with Impulse noise, MDBF technique provides excellent result. Figure 4 shows results of perceptual quality for color images corrupted with Gaussian and mixed noise. From the results, it can be observed that the same results are obtained for color images also corrupted with Gaussian noise. However for images corrupted with mixed type of noise the SBF technique gives good results but with slight blurriness in de-noised image.



**Figure 4. Perceptual Quality Results for Grayscale Images**

Original color image	Gaussian noise	Mixed noise
Filtered by BF		
Filtered by ABF		
Filtered by SBF		



**Figure 4. Perceptual Quality Results for Color**

**Images** In the next section the results of simulation are discussed.

#### 4.2 Impact on PSNR, MSE, MAE, NCD

The results of PSNR, MSE, MAE and NCD are shown in Table 7 to Table 10 for all the techniques for different noises on gray and color images for different value of noise densities as discussed below.

##### 4.2.1. Impact of Removal of Gaussian Noise

Impact of removal of Gaussian noise from gray and color images on PSNR, MSE, MAE and NCD as shown in Table 7 and Table 9. From the results it can be seen that the bilateral filter has a good value of PSNR and variant of bilateral filter i.e. ABF have better value of PSNR than BF because of addition of some extra parameters which sharpens the image by increasing the slope of the edges by shifting the range filter on histogram. As compared to BF and ABF, JBF has the highest value of PSNR among these due to the usage of NLM technique used for smoothing of images. It also makes use of patch match algorithm that provides increased search space which causes over smoothing in edge regions. Hence for removal of Gaussian noise, various techniques in decreasing order based PSNR value for all the values of noise density are:

$$\text{JBF} > \text{ABF} > \text{BF}$$

MDBF also provides a high value of PSNR for removing of impulse noise from gray scale images as shown in Table 8. SBF provides a good value of PSNR for color images having mixed noise as shown in Table 10.

**Table 7. Impact of PSNR, MSE, MAE for Gaussian Noise (Gray Images)**

		BF	ABF	JBF
5	PSNR	55.3189	52.6347	55.7775
	MSE	0.1911	0.3545	0.1719
	MAE	0.1911	0.3545	0.1719
10	PSNR	53.0753	51.4394	52.9285
	MSE	0.3203	0.4668	0.3313
	MAE	0.3203	0.4668	0.3313

15	PSNR	51.7623	50.9265	51.8666
	MSE	0.4334	0.5253	0.4231
	MAE	0.4334	0.5253	0.4231
20	PSNR	51.0913	50.5977	51.3672
	MSE	0.5058	0.5667	0.4746
	MAE	0.5058	0.5667	0.4746
25	PSNR	50.6722	50.2270	51.0437
	MSE	0.5570	0.6101	0.5113
	MAE	0.5570	0.6101	0.5113

**Table 8. Impact of PSNR, MSE, MAE for Impulse Noise**

Impulse noise		MDBF
5	PSNR	70.4978
	MSE	0.0058
	MAE	0.0058
10	PSNR	68.3368
	MSE	0.0098
	MAE	0.0098
15	PSNR	66.6342
	MSE	0.0141
	MAE	0.0141
20	PSNR	65.6026
	MSE	0.0179
	MAE	0.0179
25	PSNR	64.2300
	MSE	0.0246

	MAE	0.024 6
--	-----	------------

**Table 9. Impact of PSNR, MSE, MAE for Color Images having Gaussian Noise**

		BF	ABF	SBF	JBF
5	PSNR	59.5904	63.3120	60.9198	61.7473
	MSE	0.0715	0.0382	0.0526	0.0435
	MAE	0.0715	0.0382	0.0526	0.0435
	NCD	0.1721	0.116	0.8591	0.1058
10	PSNR	58.4277	60.9029	62.5772	61.9488
	MSE	0.0934	0.0528	0.0359	0.0415
	MAE	0.0934	0.0528	0.0359	0.0415
	NCD	0.1849	0.1412	0.8504	0.1251
15	PSNR	57.5278	60.0536	64.1323	61.3550
	MSE	0.1149	0.0642	0.0251	0.0476
	MAE	0.1149	0.0642	0.0251	0.0476
	NCD	0.2092	0.1769	0.8428	0.1470
20	PSNR	56.9152	59.5555	65.7437	60.7898
	MSE	0.1323	0.0720	0.0173	0.0542
	MAE	0.1323	0.0720	0.0173	0.0542
	NCD	0.2372	0.2126	0.8371	0.1659
25	PSNR	56.4683	59.2455	67.0845	60.2398
	MSE	0.1466	0.0774	0.0127	0.0615
	MAE	0.1466	0.0774	0.0127	0.0615
	NCD	0.2674	0.2419	0.8332	0.1823

**Table 10. Impact of PSNR, MSE, MAE for Color Images having Mixed Noise**

and impulse noise=10%		SBF
5	PSNR	58.2811
	MSE	0.0966
	MAE	0.0966
	NCD	0.8596
10	PSNR	58.9487
	MSE	0.0828
	MAE	0.0828
	NCD	0.8512
15	PSNR	59.5075
	MSE	0.0728
	MAE	0.0728
	NCD	0.8512
20	PSNR	59.9539
	MSE	0.0657
	MAE	0.0657
	NCD	0.8376

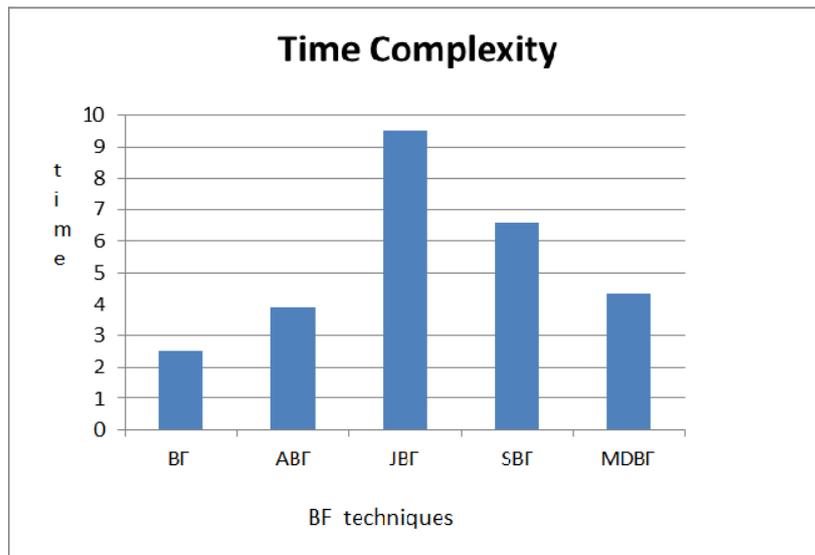
25	PSNR	60.2452
	MSE	0.0645
	MAE	0.0645
	NCD	0.8339

### 4.3 Impact on Time Complexity

The results of Time complexity for image size 256 x 256 for noise density 20% are shown in Figure xx. It is observed from the results that the time complexity of BF is best (least) because it simultaneously operate on the three bands of color at once whereas it is slightly higher for ABF technique by adding some extra parameters which increase the running time. Further the time complexity of JBF is highest because this filter separately operates on three bands of color. Hence for removal of Gaussian noise, various techniques in decreasing order based on time complexity are:

$$JBF > ABF > BF$$

The similar results (order) are obtained for images of different sizes and corrupted with different levels of noise. MDBF technique works only on gray images i.e. work on two planes that is why it takes less time to compute the algorithm. While in case of SBF some time is taken due to the presence of noise detector and remaining time by algorithm.



### 5. Conclusions

In this paper various bilateral filter and its variants are reviewed and analyzed based on various performance metrics. The following points can be inferred (see Table 11) from the simulation and results.

- For removal of mixed noise only SBF technique is available which gives satisfactory results as image obtained is blurred.
- MDBF is best technique for removal of impulse noise alone based on PSNR, MSE, MAE and time complexity.

□The JBF technique is best for removal of Gaussian noise based on PSNR, MSE, MAE. The order of various techniques is

$$JBF > ABF > BF$$

**Table 11. Comparison between Various Techniques**

Bilateral filter techniques	Type of image	Applicability to noise	Time complexity	Best Recovered
BF	Gray and color	Gaussian noise	Less	
ABF	Gray and color	Gaussian noise	Less	
JBF	Gray and color	Gaussian noise	More	
MDBF	Gray	Impulse noise	Less	
SBF	Color	Mixed noise	Moderate	

## References

- [1] R. Sharma and J. Ali, "A Comparative study various types of noise and efficient noise removal techniques", in IJARCSSE, vol. 3, no. 10, (2013), pp. 617-622.
- [2] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images", in the proceedings of 6th IEEE International conference on computer vision, (1998), pp. 834-846.
- [3] G. Goyal, "Impact and analysis of improved bilateral filter on TEM images" in International journals of science and research, vol. 3, no. 6, (2014).
- [4] N. Himayat and S. A. Kassam, "Approximate performance analysis of edge preserving filters", in the proceedings of IEEE Transaction, (1993), pp. 2764-77.
- [5] B. Zhang and J. P. Allebach, "Adaptive Bilateral Filter for Sharpness Enhancement and Noise Removal", in the proceedings of IEEE transactions, vol. 17, no. 5, (2008), pp. 664-678.
- [6] D. Barash, "A fundamental relationship between bilateral filtering, adaptive smoothing and the nonlinear diffusion equation, in the proceedings of IEEE transactions, vol. 24, no. 6, (2002) June, pp. 844-847.
- [7] A. H. Sable and K. C. Jondhale, "Modified Double Bilateral Filter for Sharpness Enhancement and Noise Removal", in proceedings of International conference, (2010), pp. 295-297.
- [8] R. Garnett, T. Huegerich, C. Chui and W. He, "A universal noise removal algorithm with an impulse detector", in the proceedings of IEEE transactions, vol.17, no. 7, (2008) July, pp. 1109-1120.
- [9] M. Zhang and B. K. Gunturk, "Multiresolution bilateral filtering for image denoising", in the proceedings of IEEE transactions, vol. 17, no. 12, (2008) December, pp. 2324-2333.
- [10] C.-H. Lin, J.-S. Tsai, and C.-T. Chiu, "Switching bilateral filter with a texture/noise detector for universal noise removal", in the proceedings of IEEE transactions on the image processing, vol. 19, no. 19, (2010), pp. 2307-2320.
- [11] A. K. Nain, S. Gupta and B. Bhushan, "An extension to switching bilateral filter for mixed noise removal from color images", in the proceedings of Int. J. Signal and Imaging Systems Engineering, vol. X, no. Y, (2014).
- [12] O. U. N. Jith and R. Venkatesh Babu "Joint bilateral filtering based non local means image de-noising", in the proceedings of IEEE transactions on image processing, (2014).
- [13] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama, "Digital photography with flash and no flash image pairs", in ACM SIGGRAPH, (2004), pp. 664-672.
- [14] K. Malik and B. Smolka, "Improved bilateral filtering scheme for noise removal in color images", in International Conference on Informatics and Applications, (2004), pp. 118-130.
- [15] M. Elad, "On the Origin of bilateral filter and ways to improve it", in the processing of IEEE transactions, vol. 11, no. 10, (2002).
- [16] M. Nagao and T. Matusuyama, "Edge preserving smoothing", in the proceedings of IEEE transactions, (1979), pp. 394-407.