International Journal of Control Theory and Applications

ISSN : 0974-5572

Volume 9 • Number 46 • 2016

Analysis on Image Denoising Algorithms (Block Matching 3D, K-Singular Value Decompostion and Non-Local Means)

Gurwinder Singh[a] and Rosepreet Kaur Bhogal[b]

[a-b]*School of Electronic and Electrical Engineering, Lovely Professional University, Phagwara, Punjab, India. Email: [a]gsingh.1655@gmail.com, [b]rosepreetkaur12@gmail.com*

Abstract: Image denoising is the main research area among various researchers. It is becoming popular among various fields like education, medical, astronomical, etc. There have been several algorithms for Image denoising and each algorithm has their own advantages. This review paper is about the NLM (Non-Local Means), BM3D (Block Matching 3D) and K-SVD (K-Means Singular Value Decomposition) algorithms. These are the state-of-the-art algorithms for denoising the images. This paper gives a brief introduction, then an overview of these algorithms and analysis is provided with comparison made on parameter PSNR. It is shown that the BM3D is the best algorithm in terms of PSNR.

Keywords: Image Denoising, Block Matching 3D (BM3D), Non-Local Means (NLM), K-Singular Value Decomposition (K-SVD), Sparsity.

1. INTRODUCTION

One of the central difficulties in the field of image processing and PC vision is image denoising, where the basic objective is to estimate the original picture by suppressing noise disturbance from a noise-contaminated variant of the picture. Image noise may be brought about by distinctive inborn (i.e., sensor) and outward (i.e., environment) conditions which are frequently unrealistic to maintain a strategic distance from in practical circumstances. In this way, picture denoising assumes an imperative part in an extensive variety of utilizations, for example, image restoration, visual following, image enrollment, image segmentation, and image grouping, where getting the original image substance is essential for solid execution. While numerous algorithms have been proposed with the end goal of image denoising, the issue of image noise suppression remains an open test, particularly in circumstances where the images are obtained under poor conditions where the noise disturbance level is high.

The issue of recovering a basic image, or whatever other information, from estimations contaminated with noise is one of the most studied problems in signal processing. In the customary set-up, a signal $y \in R^n$ $y \in R^n$ is contaminated by additive noise η such that $y = x + \eta$. The goal is then to recover the original signal by expelling the noise from the corrupted data y. Sparsity-based models have had a developing significance in signal processing in general, and have prompted proficient algorithms in image denoising, specifically [1]. This class of techniques expect that a natural signal can be expressed as a linear combination of just a couple of atoms from a redundant dictionary $D \in R^{n \times m}$, $n < m$. Searching for such a sparse representation accounts for taking care of the accompanying problem:

$$\min_x \|x\|_0 \; s.t \; \|y - Dx\|_2^2 \leq \varepsilon \tag{1}$$

where, $x \in R^m$ is the sparse representation vector for y inside and $\|x\|_0$ checks the number of non-zeros in x. Getting such a sparse representation is NP-hard in general, yet a few greedy algorithms and different relaxations methods are available to us to handle this issue under certain conditions. Method, for example, the OMP [1] and others enable us to approximate the solution to the sparse coding issue.

A typical feature in most best in class denoising algorithms is a patch-based idea: when dealing with high dimensional data, the intention is to work on overlapping patches of size $\sqrt{n} \times \sqrt{n}$, and afterward stack and average the outcomes. Some of these include the NLM [2], [3], [4], BM3D [5] and the K-SVD [6].

2. DENOISING ALGORITHMS

2.1. Non Local Means

In any digital image, the estimation of the observed image qualities at every pixel depends on the projection of the image on the sensor. The guideline of the first denoising method was very basic: Replacing the value of a center pixel with an average value of the shades of close-by pixels. The most comparative pixels to a given pixel have no motivation to be close by any means. Think about the occasional examples, or the lengthened edges which show up in many pictures. It is subsequently useless to filter a limitless segment of the image looking for every one of the pixels that truly take after the pixel, one need to denoise. Computing so as to denoising is then done the average shade of these most taking similar pixels. Similar pixels are evaluated by looking at an entire window around every pixel, and not only the shading. This new filter is called non-local means and it composes.

$$\mathrm{NL}u(p) = \frac{1}{C(p)} \int f(d(B(p), B(q))u(q))\,dq \tag{2}$$

where, $d(B(p), B(q))$ is an Euclidean distance between imagepatches focused individually at p and q, f is a diminishing capacity and $C(p)$ is the normalizing factor. The denoising of a shading picture $u = (u_1, u_2, u_3)$ and a certain patch $B = B(p, f)$ (focused at p and size $(2f + 1 \times 2f + 1)$) composes

$$\hat{B}_i = \frac{1}{C} \sum_{Q = Q(q, f) \in B(p, r)} u_i(Q) w(B \,.\, Q) \tag{3}$$

$$c = \sum_{Q(q, f) \in B(p, r)} w(B \,.\, Q) \tag{4}$$

where, $i = 1, 2, 3$, $B(p, r)$ shows a neighborhood focused at p and size $2r + 1 \times 2r + 1$ pixels and $w(B(p,f), B(q,f))$ has the same formulation than in the pixelwise usage. Thus, by applying the system for all patches in the image, we might discard $N^2 = (2f + 1)^2$ conceivable estimates for every pixel. These estimates can be at long last found the average value at every pixel location while keeping in mind the end goal to get the denoised image.

$$\hat{u}_i(p) = \frac{1}{N^2} \sum_{Q = Q(q, f),\, q \in B(p, f)} \hat{Q}_i(p) \tag{5}$$

2.2 BM3D

BM3D is a latest denoising algorithm taking into account the way that a image has a locally sparse representation in transform domain. This sparsity is upgraded by gathering comparable 2D image patches into 3D bunches.

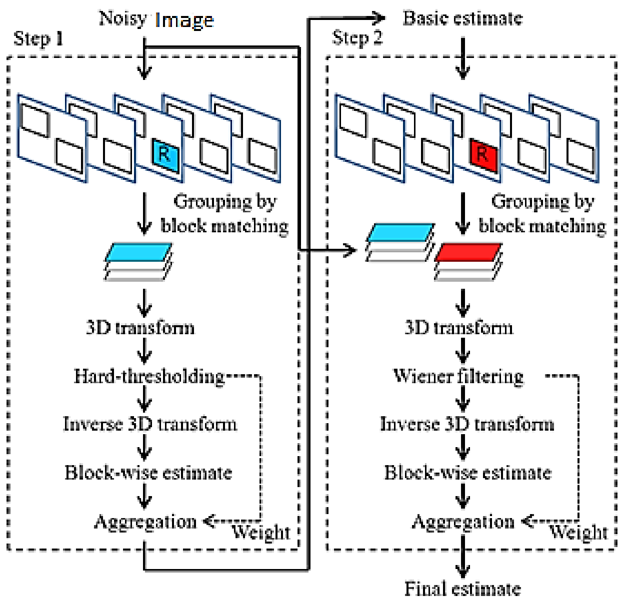


Figure 1: Block Diagram of BM3D

Step 1: Let P be the reference current patch having size $k \times k$ of the image.

Grouping: Similar patch Q is searched in the noisy image centered at $n \times n$. Let $\rho(\mathrm{P}) = \{\mathrm{Q} : d(\mathrm{P}, \mathrm{Q}) \le \tau\}$ denotes the set of similar patches. τ is the distance threshold for d under which two patches are similar. The 3D group is built by stacking up the similar patches $\rho(\mathrm{P})$. It is denoted by $\theta(P)$. Patches are sorted according to their distances and order of the patches is not important.

Collaborative Filtering: First, a 3D transform is applied to the group, followed by the shrinkage of the transform spectrum. At last, the inverse linear transform is applied to estimate for each patch

$$\theta(\mathrm{P})^{\mathrm{hard}} = \tau^{\mathrm{hard}^{-1}}(\gamma(\tau^{\mathrm{hard}}(\theta(\mathrm{P})))) \tag{6}$$

where, γ is hard thresholding operator with threshold $\lambda\sigma$.

$$\gamma(x) = \begin{Bmatrix} 0 & \text{if } |x| \le \lambda^{\mathrm{hard}}\sigma \\ x & \text{otherwise} \end{Bmatrix} \tag{7}$$

Aggregation: The basic estimate of the first step is given by

$$u^{\mathrm{basic}}(x) = \frac{\sum_{\mathrm{P}} w_{\mathrm{P}}^{\mathrm{hard}} \sum_{\mathrm{Q} \in \rho(\mathrm{P})} \mathrm{X}_{\mathrm{Q}}(x) u_{\mathrm{Q,P}}^{\mathrm{hard}}(x)}{\sum_{\mathrm{P}} w_{\mathrm{P}}^{\mathrm{hard}} \sum_{\mathrm{Q} \in \rho(\mathrm{P})} \mathrm{X}_{\mathrm{Q}}(x)} \tag{8}$$

where, $\mathrm{X}_{\mathrm{Q}}(x) = 1$ if $x \in \mathrm{Q}$, 0 otherwise

$$w_{\mathrm{P}}^{\mathrm{hard}} = \begin{Bmatrix} \left(\mathrm{N}_{\mathrm{P}}^{\mathrm{hard}}\right)^{-1} & \text{if } \mathrm{N}_{\mathrm{P}}^{\mathrm{hard}} \ge 1 \\ 1 & \text{otherwise} \end{Bmatrix} \tag{9}$$

N_{P} is the number of non-zero coefficients in the 3D block.

$u^{\text{hard}}_{\text{Q, P}}(x)$ is the estimate value of the pixel x of the patch Q found during the collaborative filtering of the reference patch P.

Step 2: Grouping: Two 3D groups are formed when a set of like patches has been obtained.

$$\varphi^{\text{basic}}(\text{P}) = \{\text{Q} : d(\text{P, Q}) \le \tau^{\text{wien}}\} \tag{10}$$

$\varphi^{\text{basic}}_{3\text{D}}(\text{P})$ by stacking up the patches from the estimation of $u(x)$ and; $\varphi_{3\text{D}}(\text{P})$ is obtained by stacking up the patches in the same order from the original noisy image u.

Collaborative Filtering: The Wiener coefficients are given by

$$w^{\text{P}}(\xi) = \frac{\left|\tau^{\text{wien}}(\varphi^{\text{basic}}(p))(\xi)\right|^2}{\left|\tau^{\text{wien}}(\varphi^{\text{basic}}(\text{P}))(\xi)\right|^2 + \sigma^2} \tag{11}$$

Aggregation: The final estimate of the second step is given by

$$u^{\text{final}}(x) = \frac{\sum_{\text{P}} w_{\text{P}}^{\text{wien}} \sum_{\text{Q} \in \rho(\text{P})} \text{X}_{\text{Q}}(x) u^{\text{wien}}_{\text{Q, P}}(x)}{\sum_{\text{P}} w_{\text{P}}^{\text{wien}} \sum_{\text{Q} \in \rho(\text{P})} \text{X}_{\text{Q}}(x)} \tag{12}$$

BM3D architecture on hardware has been discussed in [7].

2.3. K-SVD

K-SVD is a signal representation technique which, from an arrangement of signals, can determine a dictionary ready to approximate every signal with a sparse combinations of the atoms. In [8] denoising of images using statistical parameters based on patches has been discussed.

Task: Denoise a given image Y from white and additive Gaussian white noise with standard deviation σ.

Algorithm Parameters: n-block size, k-dictionary size, J-number of training iterations, λ-Lagrange multiplier, and C-noise gain.

$$\min_{\text{X, D, A}} \left\{\lambda \|\text{Y} - \text{X}\| + \sum_{ij} \mu_{ij} \|\alpha_{ij}\|_0 + \sum_{ij} \|\text{D}\alpha_{ij} - \text{R}_{ij}x\|_2^2\right\} \tag{13}$$

- **Initialization:** Set X = Y, D = overcomplete DCT dictionary.
- **Repeat J times:**
- **Sparse Coding Stage:** Use any pursuit algorithm to compute the representation vectors a_{ij} for each patch $\text{R}_{ij}x$ by approximating the solution of

$$\forall_{ij} \min_{\alpha_{ij}} \|\alpha_{ij}\|_0 \; s.t \|\text{R}_{ij}x - \text{D}\alpha_{ij}\|_2^2 \le (\text{C}\sigma)^2 \tag{14}$$

- **Dictionary:**
- **Update Stage:** For each column $l = 1, 2, \ldots, k$ in D, update it by

  Find the set of patches that use this atom,

$$w_l = \{(i, j) \,|\, \alpha_{ij}(l) \ne 0\} \tag{15}$$

- For each index $(i, j) \in w_l$, compute its representation error

$$e^l_{ij} = \text{R}_{ij}\text{X}_{ij} - \sum_{m \ne l} d_m \alpha_{ij}(m) \tag{16}$$

- set E_l as the matrix whose columns are $\{e_{ij}^{l}\}_{(i,j)\in w_l}$
- Apply SVD decomposition, $E_l = U\Delta V^T$. Choose the updated dictionary column $\widetilde{d_l}$ to be the first column of U. Update the coefficient values $\{\alpha_{ij}(l)\}_{(i,j)\in w_l}$ to be the entries of V multiplied by $\Delta(1, 1)$.
- Set

$$X = \left(\lambda I + \sum_{ij} R_{ij}^{T} R_{ij}\right)^{-1} \left(\lambda Y + \sum_{ij} R_{ij}^{T} \alpha_{ij}\right)^{-1} \tag{17}$$

Both the algorithms K-SVD and NLM are combined together to get better results has been discussed in [9]. Another improvement in K-SVD denoising algorithm is done in [10] and [11] using patch-disagreement and steepest descent OMP algorithm for better convergence.

Table 1
Comparison between the denoising results (PSNR) of above algorithms NLM, K-SVD and BM3D. The best results for each denoising algorithm, image and noise level are highlighted

σ	*5*	*10*	*15*	*20*	*25*	*30*	*35*	*40*
Methods				Baboon 512 x 512				
NLM	34.355	30.193	27.793	26.294	25.069	24.09	23.384	22.699
K-SVD	35.231	30.392	27.902	26.373	25.204	24.308	23.568	22.919
BM3D	35.231	30.578	28.155	26.585	25.42	24.554	23.814	23.084

σ	*5*	*10*	*15*	*20*	*25*	*30*	*35*	*40*
Methods				Boat 512 x 512				
NLM	36.603	32.946	30.738	29.721	28.608	27.659	26.87	26.238
K-SVD	37.14	33.661	31.736	30.361	29.351	28.442	27.652	26.979
BM3D	37.239	33.879	32.063	30.788	29.863	29.058	28.369	27.632

σ	*5*	*10*	*15*	*20*	*25*	*30*	*35*	*40*
Methods				Cameraman 256 x 256				
NLM	37.501	33.493	31.133	29.866	28.821	27.913	27.145	26.447
K-SVD	37.97	33.679	31.531	29.987	28.974	28.117	27.454	26.86
BM3D	38.227	34.056	31.864	30.418	29.445	28.538	28	27.213

σ	*5*	*10*	*15*	*20*	*25*	*30*	*35*	*40*
Methods				Fingerprint 512 x 512				
NLM	35.16	31.047	28.734	27.301	26.165	25.299	24.774	24.126
K-SVD	36.603	32.373	30.038	28.434	27.261	26.352	25.456	24.683
BM3D	36.494	32.457	30.283	28.781	27.697	26.855	26.094	25.299

σ	*5*	*10*	*15*	*20*	*25*	*30*	*35*	*40*
Methods				Hill 512 x 512				
NLM	36.621	32.881	30.709	29.735	28.641	27.759	27.074	26.425
K-SVD	36.966	33.368	31.456	30.178	29.181	28.376	27.717	27.144
BM3D	37.118	33.597	31.823	30.702	29.78	29.096	28.465	27.914

σ	*5*	*10*	*15*	*20*	*25*	*30*	*35*	*40*
Methods				Man 512 x 512				
NLM	37.038	33.068	30.786	29.68	28.579	27.673	26.966	26.289
K-SVD	37.495	33.619	31.578	30.127	29.075	28.315	27.496	27.017
BM3D	37.759	33.955	31.89	30.554	29.543	28.785	28.111	27.612

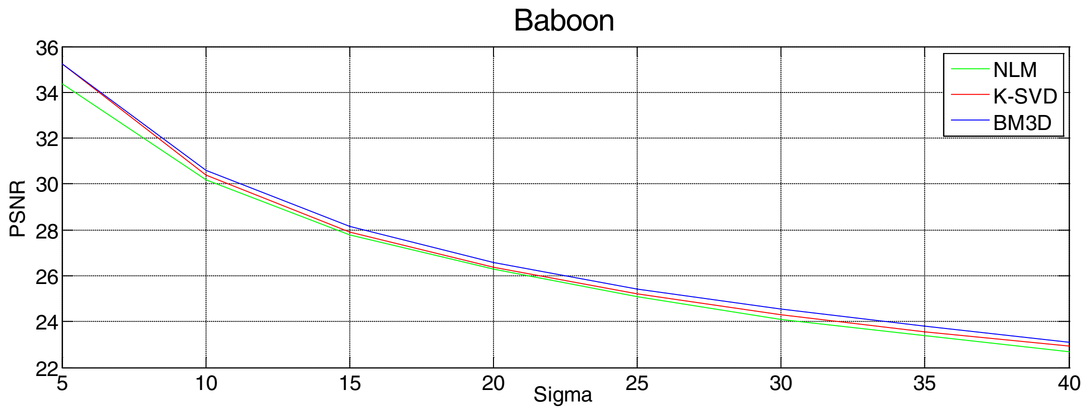


Figure 2: Denoised Baboon Comparison

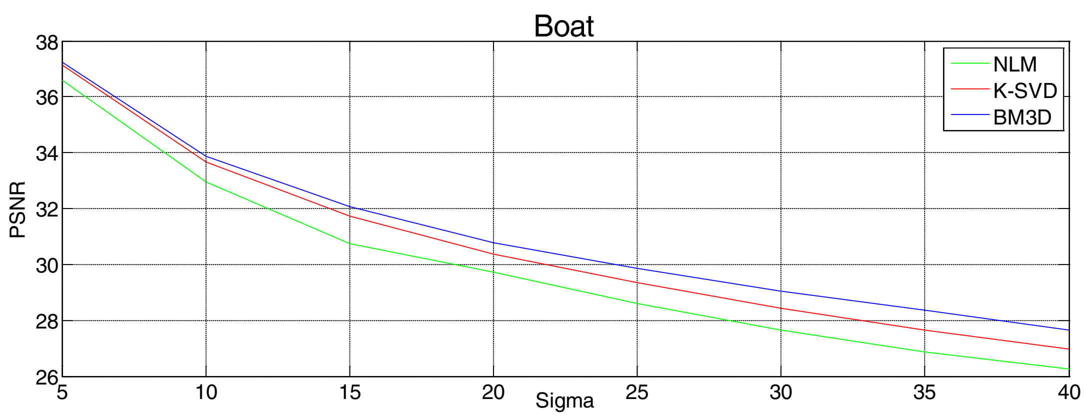


Figure 3: Denoised Boat Comparison

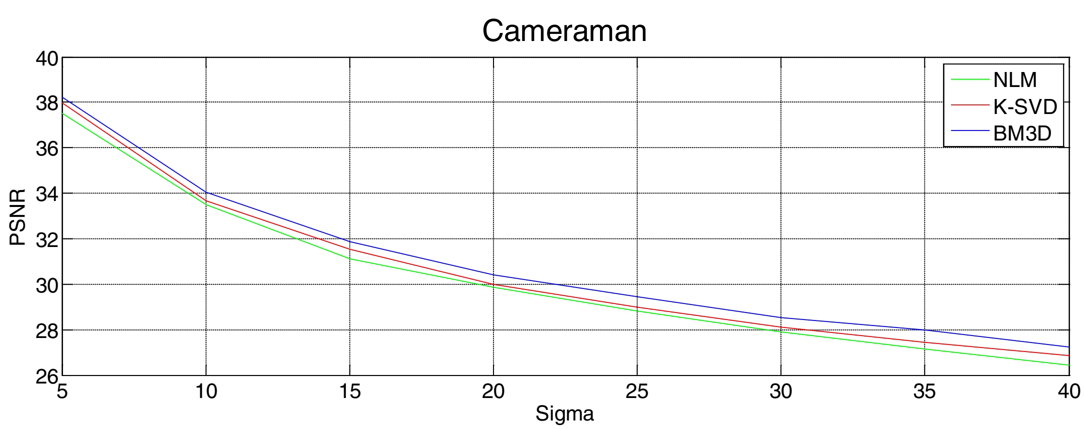


Figure.4: Denoised Cameraman Comparison

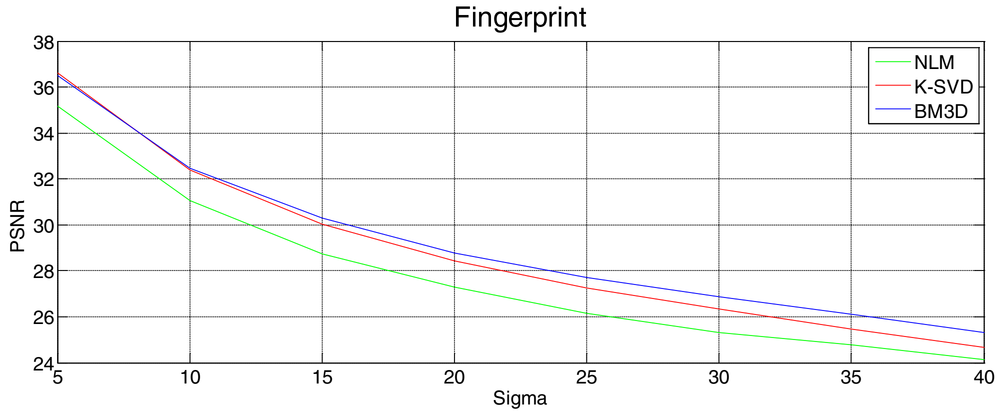


Figure5: Denoised Fingerprint Comparison

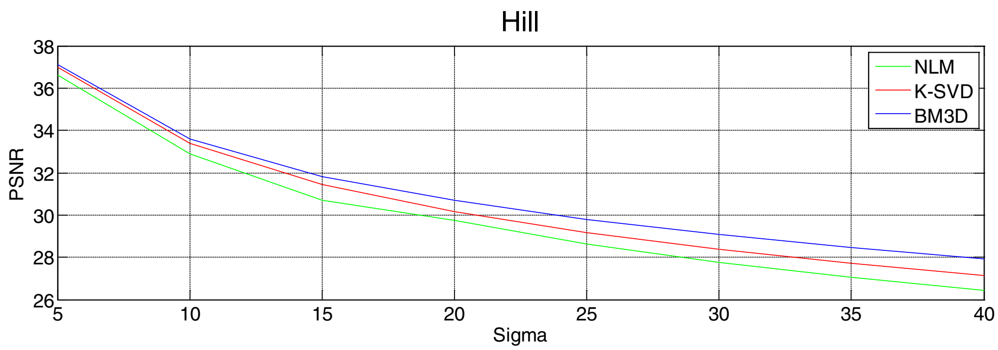


Figure.6: Denoised Hill Comparison

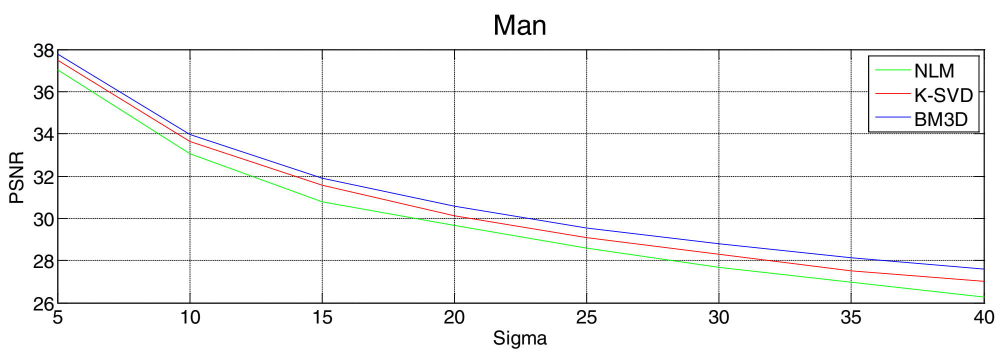


Figure 7: Denoised Man Comparison

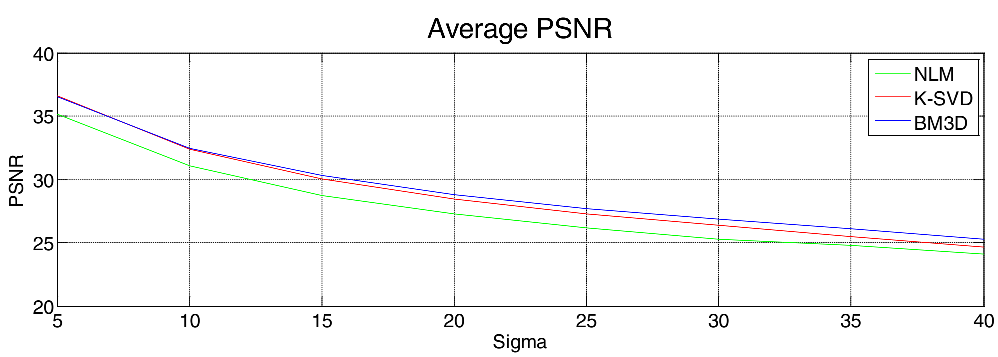


Figure 8: Denoised Average PSNR Comparison

3. EXPERIMENTAL RESULTS

The experiment is performed on the standard images. First, take the original image as input. The images size is referred in the Table I. After that add the noise with different standard deviations σ values. Then the above three algorithms are executed. In this paper results are obtained using σ = 5, 10, 15, 20, 25, 30, 35, 40. In this paper only the visual results are shown for the σ = 5, 10, 15, 20. In BM3D algorithm τ^{hand} is Bior1.5 transform for all values of σ. τ^{wien} is a 2D-DCT transform. Default values of BM3D parameters can be taken from [5]. K-SVD algorithms parameters are taken from [6].

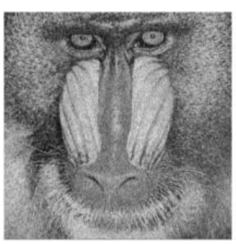
The Peak Signal to Noise Ratio:

$$\mathrm{PSNR} = 20\log_{10}\left(\frac{255}{\mathrm{RMSE}}\right) \tag{18}$$

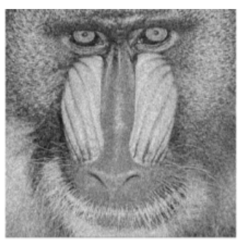
It is the ratio between the maximum possible powers of a signal (Image) to the power of noisy signal that affects the fidelity of its representation. PSNR is usually expressed in logarithmic decibel scale. The larger the PSNR, the better is the denoising.

4. CONCLUSION

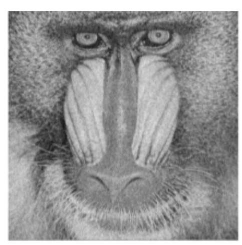
Image noise can degrade the quality of an image. Among various noise models, Gaussian noise is the noise added mostly during the acquisition of an image. Its principle sources are the poor illumination and high temperature during acquisition. These algorithms are the state-of–the-art algorithms for denoising the images. From Table I, Cameraman has high PSNR for σ = 5(PSNR=38.227), 10(PSNR=34.056), Boat for σ = 15(PSNR=32.063), 20(PSNR=30.788), 25(PSNR=29.863) and Hill for σ = 30(PSNR=29.096), 35(PSNR=28.465), 40(PSNR=27.914). From the results BM3D algorithm is the efficient algorithm among the NLM and K-SVD algorithm. Results shows that as the value of σ increases the BM3D algorithm perform better.



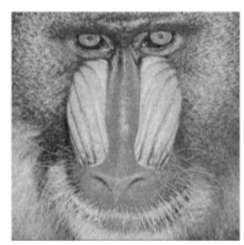
(a)



(b)



(c)



(d)

(e)

(f)

(g)

(h)

(i)

(j)

(k)

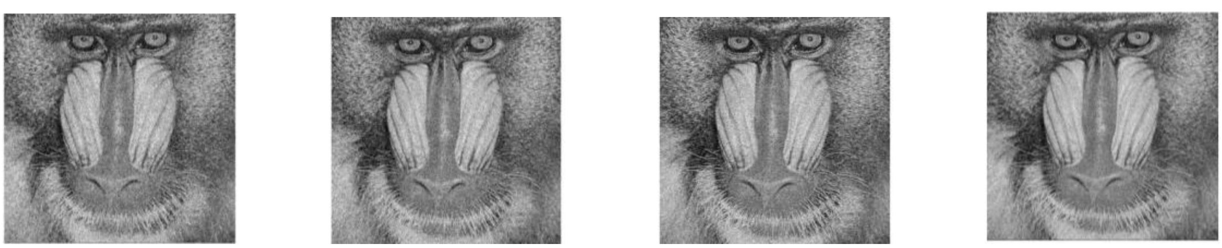
(l)

(m) **(n)** **(o)** **(p)**

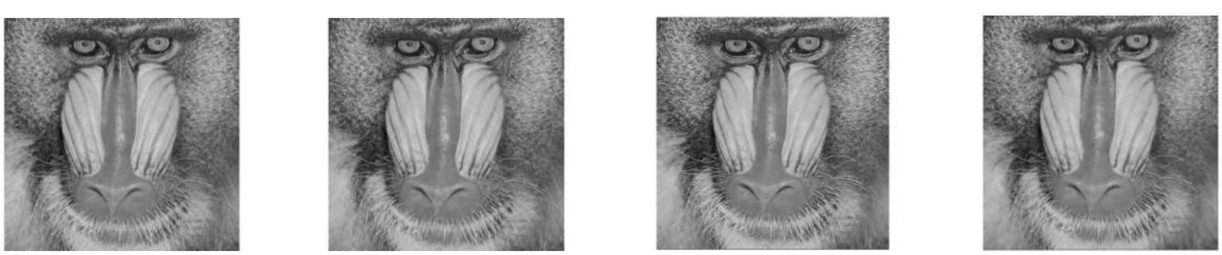
(q) **(r)** **(s)** **(t)**

(u) **(v)** **(w)** **(x)**

Figure 9: Results of NLM Algorithm. (a,b,c,d),(i,j,k,l),(q,r,s,t) are the Noisy Images of Baboon, Boat and Man with σ = 5,10,15 and 20 respectively. (e,f,g,h),(m,n,o,p),(u,v,w,x) are the Denoised Images of Baboon, Boat and Man respectively.



(a) **(b)** **(c)** **(d)**



(e) **(f)** **(g)** **(h)**

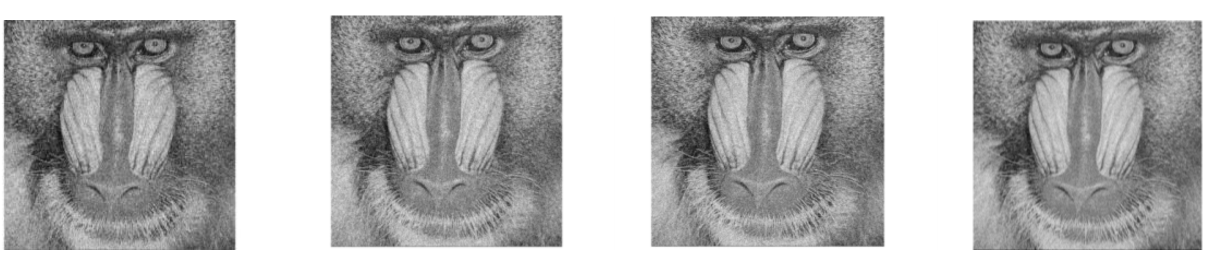
(i) **(j)** **(k)** **(l)**

(m) **(n)** **(o)** **(p)**

(q) **(r)** **(s)** **(t)**

(u) **(v)** **(w)** **(x)**

Figure 10: Results of K-SVD Algorithm. (a,b,c,d),(I,j,k,l),(q,r,s,t) are the Noisy Images of Baboon, Boat and Man with σ = 5,10,15 and 20 respectively. (e,f,g,h),(m,n,o,p),(u,v,w,x) are the Denoised Images of Baboon, Boat and Man respectively



(a) **(b)** **(c)** **(d)**

Figure 11: Results of BM3D Algorithm. (a,b,c,d),(i,j,k,l),(q,r,s,t) are the Noisy Images of Baboon, Boat and Man with σ = 5,10,15 and 20 respectively. (e,f,g,h),(m,n,o,p),(u,v,w,x) are the Denoised Images of Baboon, Boat and Man respectively

REFERENCES

[1] Tropp J.A. Greed is good: algorithmic results for sparse approximation, IEEE Transactions on Information Theory,vol:50 (10),2004,pp:2231-2242.

[2] Baudes A, Coll B, Morel J.M.A review of image denoising methods, with a new one. Multiscale Modeling and Simulation, vol.4 (2),2006, pp:490-530.

[3] Baudes A, Coll B, Morel J.M. A non-local algorithm for image denoising. IEEE Computer Vision and Pattern Recognition, 2005,vol. 2,2005,pp:60-65.

[4] Dixit A.A. Image denoising by non-local means algorithm.International Conference on Signal Processing, Image Proceesing and Pattren Recognition, 2013,pp:275-277.

[5] Lebrun M.An Analysis and Implementation of the BM3D Image Denoising Method. Image Processing On Line, (2), 2012, pp.188-189.

[6] Lebrun M., Leclaire A. An Implementation and Detailed Analysis of the K-SVD Denoising Algorithm. Image Processing On Line, (2),2012,pp.96-133.

[7] Hongming.C. Analysis and architecture design of block matching in BM3D image denoising. International Conference on Electron devices and Solid-State circuits,2011,pp:1-2.

[8] Faktor.T. Denoising of Image patchesvia sparse representationswith learned statistical dependencies. IEEE International Conference on Acoustics, Speech and Signal Processing,2011,pp:5820-5823.

[9] Tang.S. Image denoising using K-SVD and non-local means. IEEE Workshop on Electronics, Computer and Applications, 2014, pp:886-889.

[10] Romano.Y. Patch-disagrrement as a way to improve K-SVD denoising., IEEE International Conference on Acoustics, Speech and Signal Processing,2015,pp:1280-1284.

[11] Deng.X. Image denoising based on steepest descent OMP and K-SVD., IEEE International Conference on Signal Processing, Communications and Computing,2015,pp:1-5.