Volume 6, Issue 5
October 2017

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

Image Denoising using BM3D and Principal Component Analysis

[1] Raghvendra Singh, [2]S. K. Tiwari and [3]Vandana Gupta

[1,2] School of Studies in Mathematics, Vikram University, Ujjain, Madhya Pradesh, INDIA
[3] Department of Mathematics, Govt. Kalidas Girls College, Ujjain, Madhya Pradesh, INDIA

E-mail: [1] raghvendraknp16@gmail.com

ABSTRACT

Image denoising is a process which tries to recover error free image from its noisy counterparts. Most of times it is not possible to restore original image, thus noise suppression methods are adopted. Earlier methods based on linear and non-linear filters are not so effective, thus algorithms based methods based on iterative approach are preferred. However, in case of complex noises, these methods also fail due to under and over fitting. With the latest advancements, an approach that works on the basis of BM3D for removing noise is presented. This approach works with arranging 2D blocks of image segments in 3D arrays. Collaborative filtering is implemented in these arrays. This is accomplished in the following three steps: 3D transformation of a group, shrinkage of transform spectrum, and inverse 3D transformation. Principal component analysis is further helpful in dimension and noise reduction. This paper discusses the BM3DPCA based noise removal process. For the illustration purpose six images: Airplane, Baboon, Barbara, Boat, Lenna and Peppers are considered and results are discussed in terms of Peak Signal to Noise Ratio (PSNR).

Keywords: *PCA, BM3D, filtering*

1. INTRODUCTION

The whole process of digital image processing is the method of processing of an image with the implementation of computer algorithms. As we all know that digital image processing is evolved from processing of digital signal, it contains a lot of merits in the event of making comparison with the analog image processing; it allows a considerably measure of broad extent of algorithms that can be used to apply on the input data and during the processing time, it can overcome the issues such as the noise build-up and signal distortion. The process of configuring the partition of a digital image into several regions (set of pixels) is being referred to the image segmentation [1]. The primary goal of this is of simplifying, rearranging and/or modifying the image representation into that which is of high importance and it is simpler to make it examination. We normally use segmentation of image is to find boundaries and objects (lines, curves, etc.) in an image.

The process of recovering error free image from noisy image is termed as Image denoising. For a majority of time, we are not able to restore original image and because of this, we adopt methods like noise suppression. The previous methods that works on the basis of linear and non-linear filters are not quite effective, so preference is given to the algorithms based methods based on iterative approach. However, these methods are also not very fruitful in the case of complex noises. The reason behind this failure is under and over fitting [2-6].

With the latest advancements, an approach that works on the basis of BM3D for removing noise is presented. This approach works with arranging 2D blocks of image segments in 3D arrays. Collaborative filtering is implemented in these arrays [7-11]. This is accomplished in the following three steps: 3D transformation of a group, shrinkage of transform spectrum, and inverse 3D transformation.

The purpose of denoising is carried out by shrinkage of the spectrum of a 3-D transform which is applied on such sorts of groups. The extent of impact of the shrinkage relies on the capability extent of the transform to sparsely represent the true-image data, hence making its separation from the noise. We can improve the sparsity can be in two aspects. The first one is by employing image patches (neighborhoods) which contain data-adaptive shape. The other one is by making use of PCA on the above mentioned adaptive shape neighborhoods as a part of the employed 3-D transform. We can get the PCA bases are by eigenvalue decomposition of empirical second-moment matrices the estimation of which is carried out from groups of similar adaptive-shape neighborhoods [11-13].

2. BM3D Principal Component Analysis

This section describes the BM3D process in detail, and thereafter use of principal component analysis in noise reduction is detailed.

2.1 Grouping and Matching:

We can define Grouping with the help of numerous techniques such as, Kmeans clustering, self-organizing maps, fuzzy clustering, vector quantization, and many others. Normally, we compute the similarity between signal fragments as the inverse of some distance measure. Therefore, a small measure of distance implies higher similarity. A number of different distance measures could be used to employ such as the p-norm of the difference between two signal fragments. Alongside other examples are the weighted Euclidean distance (p = 2) applied in the case of non-local means estimator [10], and too the normalized distance which is used in the exemplar-based estimator [11].

In the event of carrying out the processing of complex or uncertain (e.g. noisy) data, we must first extract certain features from the signal and after this we take the measurement of the distance for just these features. Some techniques of grouping such as vector quantization or Kmeans clustering are necessarily based on the concept of partitioning. It implies that they develop clusters or groups (classes) which are disjoint in such a manner that each one of the fragment belongs to just one group only. In the process of constructing disjoint groups, the elements of which enjoy great mutual similarity generally needs recursive methods and it could be computationally demanding. In addition, the partitioning results in an unequal treatment of the various fragments due to the reason that the ones which are near to the group's centroid are better represented in a better way in comparison to those lying far from it. This takes place forever without the exception of the uncommon case in which each fragment of the signal is equidistantly placed.

A more straightforward and fruitful grouping of mutually similar signal fragments can be noticed by matching, where in contrast to the above mentioned partitioning procedures, the developed groups are not disjoint in the mandatory basis. The process of matching could be defined as a method for discovering signal fragments just like to a provided reference one.

This could be achieved by pair wise examination of the similarity between the candidate fragments and reference fragment situated at various spatial positions. The fragments with a considerable small distance from the reference one in comparison to that of a given threshold are accepted as mutually similar and are grouped. This similarity is the basis of the role of the membership function for the group which is considered and the reference fragment can be considered as certain kind of centroid for the group.

We can use any signal fragment as a reference one and due to this a group can be developed for it.

Block-matching (BM) is a dedicated matching process that has been applied in a broad scale for the purpose of motion estimation in video compression. As a particular method of grouping, it is applied to hunt similar blocks, which are afterwards grouped in a 3D array (i.e. a group).

2.2 Collaborative Filtering:

Provided a collection of n fragments, this filtering of the group develops n estimates. For each of the grouped fragments, it develops one estimate. Generally, the above mentioned estimates are not same. The term collaborative is derived from the sense that all grouped fragments collaborates for making the filtration of all others, and vice versa.

Due to the reason that the corresponding noise-free blocks are considered to be not different, the estimates are unbiased. Consequently, the last estimation error is caused only because of the residual variance. This residual variance is inversely proportional to the blocks number in the group.

Ignoring the complexity of the signal fragments, we can get quite good estimates given that the groups comprise a great measure of fragments.

Though, perfectly indistinguishable blocks are not similar in natural images. In the event of non-identical fragments are permitted inside the same group, the evaluation acquired by element wise averaging get to be biased. The error of bias can represent the biggest part of the whole last error in the appraisals, unless an estimator is used that takes into account for creating a dissimilar estimate of all grouped fragments. In this way, a more successful collaborative filtering methodology than averaging should be utilized.

2.3 Collaborative filtering by shrinkage in transform domain:

A more productive collaborative filtering could be acknowledged as shrinkage in transform domain. Taking in assumption d+1-dimensional groups having similar signal fragments are as of now developed, the collaborative shrinkage comprises of the accompanying strides.

1. A d+1-dimensional linear transform is applied to the group.
2. Shrinking of the (e.g. by soft- and hard-thresholding or Wiener filtering) transform coefficients which leads to the attenuation of noise.
3. In order to develop estimates of all grouped fragments make the inversion of the linear transform.

The above discussed collaborative transform-domain shrinkage can be especially fruitful in the event of applying to groups of fragments of natural image. The groups are featured by both:

1. intra-fragment correlation value which shows up between the pixels of each grouped fragment a peculiarity of natural images;
2. inter-fragment correlation which appears between the corresponding pixels of various fragments - an output of the similarity between grouped fragments.

The 3D transform can count the benefits of both sorts of correlation and therefore create a sparse representation of the genuine group's signal. This sparsity turns the shrinkage exceptionally productive in weakening the noise at the time of safeguarding the elements of the signal.

The complete process is shown in Figure 1.

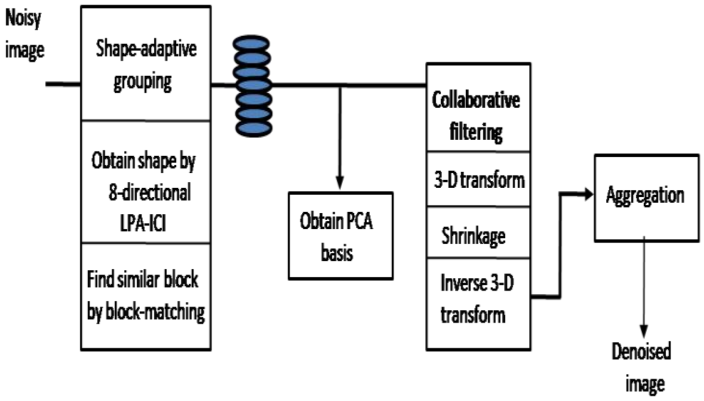


Figure 1: Noise Filtering Process

3. MATHEMAICAL DESCRIPTION

Let us assume a noisy image z of the form

$$z(x) = y(x) + n(x) \qquad x \in X \qquad (1)$$

In this equation, *x* is a 2*D* spatial coordinate that belongs to the image domain *x*, *y* is the genuine image, and *n* is i.i.d. zero implies to the Gaussian noise with variance σ^2, i.e., $N(0,\sigma^2)$. Let us consider, Z_x is of constant size $N_1 \times N_1$ positioned at x in Z . Z_S is a 3D array comprised blocks Z_x situated at x . We make use of '*ht*' for the purpose of hard-thresholding and '*wie*' for Wiener filtering. The primary and last estimates are provided by $\widehat{y}^{basic}$ and $\widehat{y}^{final}$ respectively.

Grouping and collaborative hard-thresholding:

Blocks that have the separation (dissimilarity) in terms of the reference one is not greater than a pre-defined threshold are viewed similar and gathered. Specifically, we apply the L^2-distance as a measure of dissimilarity. Preferably, in the case of the true-image *y* would be accessible, we can figure out the block distance as

$$d^{ideal}(Z_{xR},Z_x)=\frac{\|Y_{xR}-Y_x\|_2^2}{(N_1^{ht})^2} \qquad (2)$$

In this equation, $\|.\|$ represents the L^2-norm and the blocks Y_{xR} and Y_x are respectively situated at xR and $x\in X$ in y. Although, only the noisy image z is accessible and we can calculate the distance with the help of the noisy blocks Z_{xR} and Z_x as

$$d^{ideal}(Z_{xR},Z_x)=\frac{\|Z_{xR}-Z_x\|_2^2}{(N_1^{ht})^2} \qquad (3)$$

Be that as it may, this matching is, to a great extent, computationally complex. In order to overcome this issue, the block-distance utilizing a coarse pre filtering is recommended. The above discussed pre filtering is acknowledged by applying a standardized 2D linear transform on both blocks and afterward hard-thresholding the acquired coefficients, which brings about

$$d(Z_{xR},Z_x)=\frac{\left\|\gamma\left(T_{2D}^{ht}(Z_{xR})\right)-\gamma\left(T_{2D}^{ht}(Z_x)\right)\right\|_2^2}{(N_1^{ht})^2} \qquad (4)$$

In the above equation γ is defined as the hard-thresholding operator with threshold $\lambda_{2D}\sigma$ and T_{2D}^{ht} represents the standardized 2D linear transform. The blocks that are identical to Z_{xR} is

$$S_{xR}^{ht}=\left\{x\in X: d(Z_{xR},Z_x)\le\tau_{match}^{ht}\right\} \qquad (5)$$

Here, τ_{match}^{ht} is pre-defined hard threshold.

The inverse 3D transform is provided by

$$\widehat{Y}_{S_{xR}^{ht}}^{ht}=T_{3D}^{ht^{-1}}\left[\Upsilon(T_{3D}^{ht}(Z_{S_{xR}^{ht}}))\right] \qquad (6)$$

Given the primary assessment $\widehat{Y}^{basic}$ of the true image, the denoising can be enhanced by carrying out grouping inside this fundamental estimate and collaborative empirical Wiener filtering. Due to the fact that the noise in $\widehat{Y}^{basic}$ is considered to be considerably lessened, we supplant the thresholding-based d-distance (4) with the standardized squared L^2-distance estimated inside the primary estimate. Consequently, the matched blocks' coordinates are the components of the set

$$S_{xR}^{wie}=\left\{x\in X:\frac{\left\|\widehat{Y}_{xR}^{basic}-\widehat{Y}_x^{basic}\right\|_2^2}{(N_1^{wie})^2}<\tau_{match}^{wie}\right\} \qquad (7)$$

We make use of the set S_{xR}^{wie} for the end goal of forming two groups. Out of these two, one will be from the basic estimate while the other one from the noisy observation:

The empirical Wiener shrinkage coefficients could be defined in terms of the energy of the 3D transform coefficients of the group of basic estimate as given here underneath

$$W_{S_{xR}^{wie}}=\frac{\left|T_{3D}^{wie}(\widehat{Y}_{S_{xR}^{wie}}^{basic})\right|^2}{\left|T_{3D}^{wie}(\widehat{Y}_{S_{xR}^{wie}}^{basic})\right|^2+\sigma^2} \qquad (8)$$

Finally, the inverse transform $T_{3D}^{wie^{-1}}$ develops the group of estimates

$$\widehat{Y}_{S_{xR}^{wie}}^{wie}=T_{3D}^{wie^{-1}}\left[\Upsilon(T_{3D}^{wie}(Z_{S_{xR}^{wie}}))\right] \qquad (9)$$

With the help of above for hard thresholding (for all $x_R\in X$), we could assign the weight

$$\omega_{xR}^{ht}=\begin{cases}\dfrac{1}{\sigma^2 N_{har}^{xR}}, & \text{if } N_{har}^{xR}\ge 1\\ 1, & \text{otherwise}\end{cases} \qquad (10)$$

for the group of estimates $\widehat{Y}_{x\in S_x^{ht}}^{ht,xR}$. Similarly, for wiener filtering for each $x_R\in X$, we assign the weight

$$\omega_{xR}^{wie}=\frac{1}{\sigma^2\left\|W_{S_{xR}^{wie}}\right\|_2^2} \qquad (11)$$

for the group of estimates $\widehat{Y}_{x\in S_x^{wie}}^{wie,xR}$

Aggregation by weighted average: We can compute the global basic estimate $\widehat{Y}^{basic}$ with a weighted average of the blockwise estimates $\widehat{Y}_{x\in S_x^{ht}}^{ht,xR}$ obtained, by making use of the weights ω_{xR}^{ht} illustrated in (10), i.e.

$$\widehat{y}^{basic}(x)=\frac{\sum_{xR\in X}\sum_{xm\in S_{xR}^{ht}}\omega_{xR}^{ht}\widehat{Y}_{xm}^{ht,xR}(x)}{\sum_{xR\in X}\sum_{xm\in S_{xR}^{ht}}\omega_{xR}^{ht}\chi_{x_m}(x)},\forall x\in X \qquad (12)$$

Here in the equation written above χ_{x_m} is 0 or 1. The global final estimate $\widehat{y}^{final}$ is estimated by making use of the above equation in which we replaced each variable *ht* by *wie*. This method is termed as BM3D.

BM3D-Principal Component Analysis:

In the BM3D-PCA algorithm, input image is being processed in raster scan. In this scan, the below mentioned operations are carried out at each processed pixel:

1) Get adaptive-shape neighborhood focused at the present pixel by making use of the 8-directional LPA-ICI just like as in [7], [4]. The neighborhood is confined inside a constant-size and non-adaptive square block, which we known as *reference*

block. We denote The pixels' number in the neighborhood could be defined as N_{el}.

2) Look for the blocks that are identical to the reference one with the help of block-matching and acquire an adaptive-shape neighborhood from all of these matched blocks by making use of the shape got in Step 1. We define the number of matched blocks by N_{gr}.

3) Find out the transform used for the application on the adaptive-shape neighborhoods. We got two conditions, which depend on the fact whether $\frac{N_{gr}}{N_{el}}$ is smaller or larger than a pre defined threshold τ.

a) In the case of $\frac{N_{gr}}{N_{el}} \geq \tau$, we assume that we got an enough quantity of mutually similar neighbourhoods in order to estimate a second-moment matrix in a reliable manner. The eigenvectors of this matrix frame the shape-adaptive PCA basis. Thereafter, only those eigenvectors are retained whose corresponding eigenvalues are more than a predefined threshold. In this way, we obtain a *trimmed* shape-adaptive PCA trans-form.

b) In the event of $\frac{N_{gr}}{N_{el}} < \tau$, we consider that there are not sufficient similar neighborhoods to be used as training data and in this way, we resort to the fixed (i.e. non data-adaptive) SA-DCT, just like as in [4].

4) Develop a 3-D array (termed as *group*) by stacking together the $\min(N_{gr}, N_2)$ adaptive-shape neighborhoods with greatest similarity to the reference one, where N_2 is a pre defined parameter that prevents the number of filtered neighborhoods.

5) Put the transform we obtained in Step 3 on all of the grouped adaptive-shape neighborhoods. Afterward, a 1-D orthogonal transform in other words Haar wavelet decomposition is applied along the third dimension of the 3-D group.

6) Carry out the shrinkage (hard-thresholding or empirical Wiener filtering) on the 3-D spectrum.

7) Now, invert the 3-D transform from Step 5 in order to get estimates for all of the grouped adaptive-shape neighborhoods.

8) Make the acquired estimates to return to their original positions with the help of weighted averaging in the event of overlapping.

Trimmed Principal Component Analysis:

As the primary commitment of the proposed strategy is the utilization of the shape- and data-adaptive PCA transform on adaptive-shape neighbourhoods' groups, we clarify in a detailed way what is performed in Step 3a, while we allude the reader to our past assignments [5], [9], [10],[13] for information on the other algorithm steps. The input given in the Step 3a is a group of N_{gr} adaptive-shape neighborhoods that are discovered to be mutually identical. Each of these 2-D neighborhoods is represented as a 1-D column vector $\vec{u_i}$ of length N_{el}, $i = 1,2,...N_{gr}$. After this, an $N_{el} \times N_{el}$ sample second-moment matrix is estimated by matrix multiplication,

$$A = \left[\vec{u_i}\ \vec{u_2}\ \dots\ \overrightarrow{u_{N_{gr}}}\right]\left[\vec{u_i}\ \vec{u_2}\ \dots\ \overrightarrow{u_{N_{gr}}}\right]^T$$

$$D^T AD = B = diag(b_1, b_2, ...b_{N_{el}})$$

and thereafter its eigenvalue decomposition produces where D defines the orthonormal matrix and *B* represents the diagonal matrix that contains eigenvalues ordered by magnitude $b_1 > b_2 > ... > b_{N_{el}}$. At last, the PCs which are used for the purpose of decomposition of the adaptive.shape neighborhoods are the initial N_t columns of D, in which N_t represents the number of eigenvalues more than $\lambda\sigma^2$, *λ* being a pre defined threshold.

We can make the evaluation of the performance of the algorithm on the basis of the peak-signal to noise ratio and is as follows

$$PSNR(\hat{y}) = 10\log_{10}\left[\frac{(255)^2}{|X|^{-1}\sum_{x\in X}\left(y(x) - \hat{y}(x)\right)^2}\right] \tag{13}$$

4. RESULTS

In the simulation three values 0.7, 1 and 1.3 for τ are considered. The value of *λ* is taken to be 13. The standard variance of noise (σ) is taken to be 25.

From figure 2 to figure 7, in part (a) noisy images are shown and in part (b) denoised image is shown.

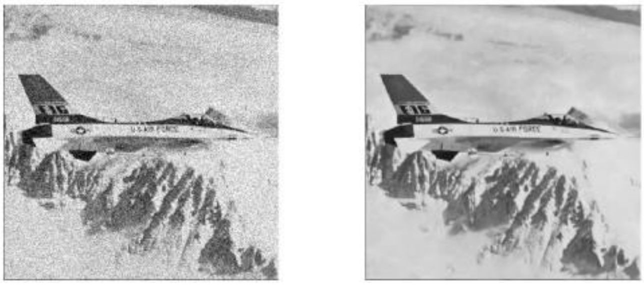

(a) (b)

Figure 2: (a) noised image (b) de-noised image (Airplane)

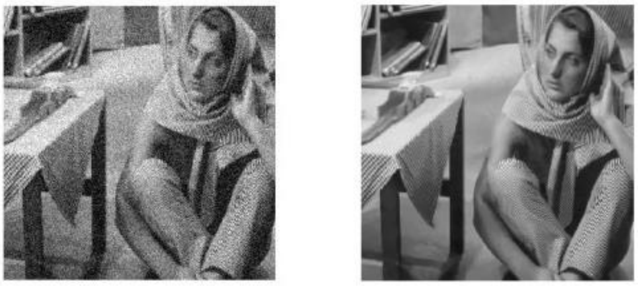

(a) (b)

Figure 3: (a) noised image (b) de-noised image (Barbara)

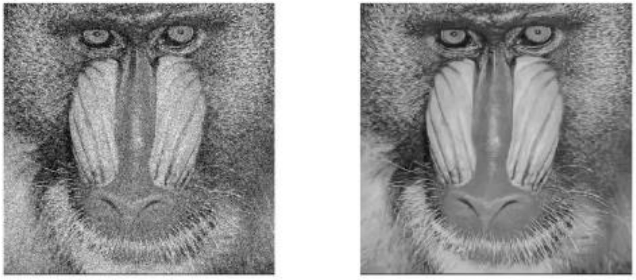
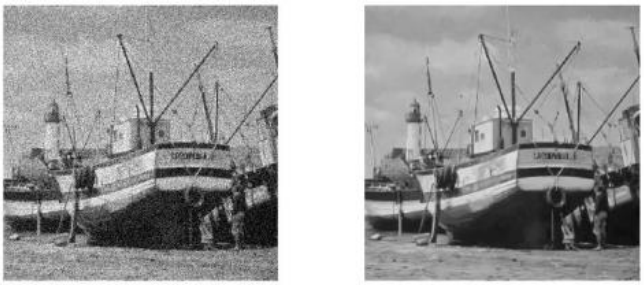
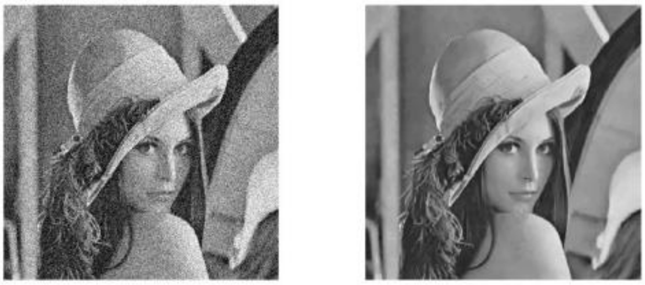

(a) (b)

Figure 4: (a) noised image (b) de-noised image (Baboon)



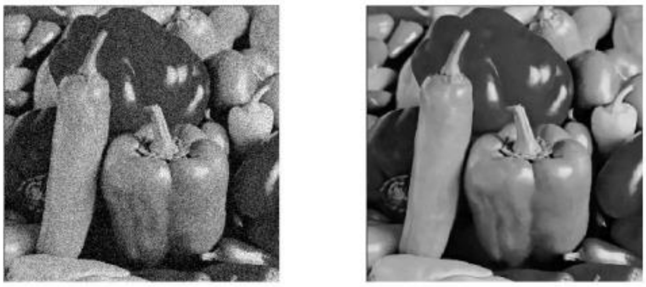
(a) (b)

Figure 5: (a) noised image (b) de-noised image (Boat)



(a) (b)

Figure 6: (a) noised image (b) de-noised image (Lenna)



(a) (b)

Figure 7: (a) noised image (b) de-noised image (Peppers)

In these figures, image is corrupted by Gaussian noise with sigma considered to be 25.

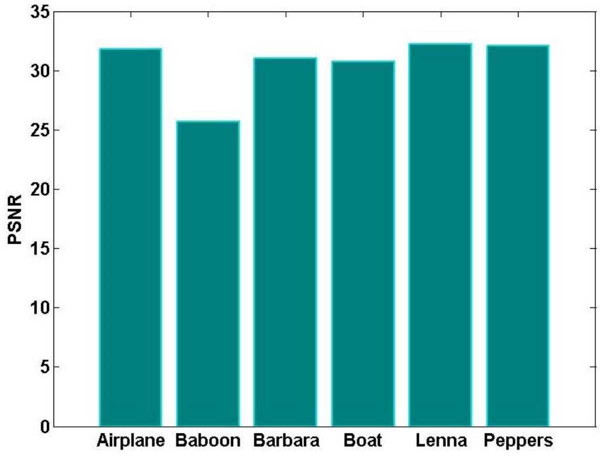


Figure 8: Comparison of PSNR (dB) at σ equals 25.

The obtained PSNR of the denoised images is shown in figure 8. Over here, the PSNR of the most of the images is around 32 dB. However, the PSNR of Baboon image is nearly 25.

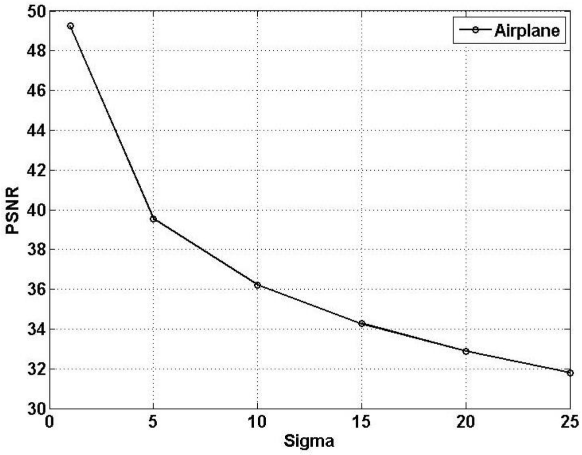


Figure 9: PSNR vs. Sigma for Airplane image

In figure 9, PSNR vs. sigma is plotted for Airplane image. For sigma value of 1, the PSNR is nearly 49 dB, and at the sigma of 25, the PSNR is nearly 32 dB. Initially fall in PSNR is huge with sigma, but when sigma reaches a value of 10, the PSNR value is 36 dB, thereafter fall in PSNR with respect to sigma is very slow.

In figure 10, PSNR vs. sigma is plotted for Baboon image. For sigma value of 1, the PSNR is nearly 48 dB, and at the sigma of 25, the PSNR is nearly 25 dB. Thus, noise reduction is Baboon image is lesser in comparison to Baboon image.

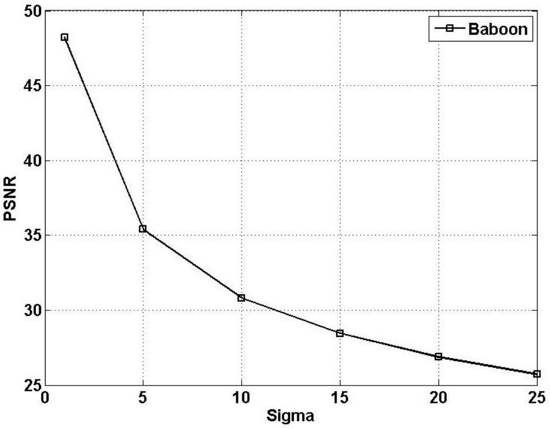


Figure 10: PSNR vs. Sigma for Baboon image

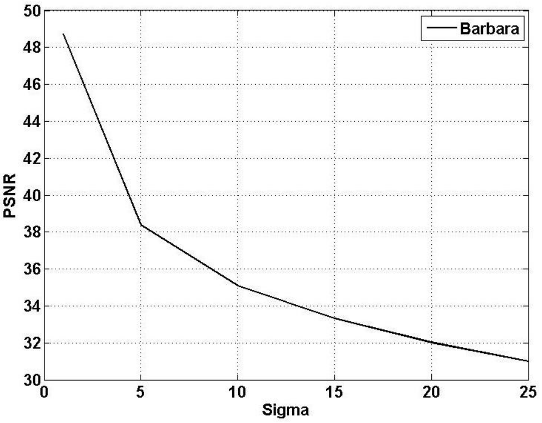


Figure 11: PSNR vs. Sigma for Barbara image

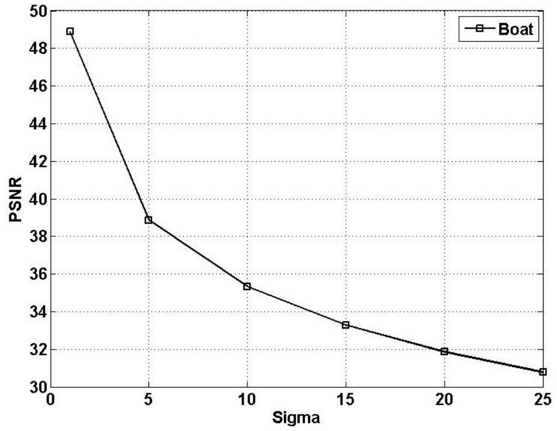


Figure 12: PSNR vs. Sigma for Boat image

From figure 11 to figure 14, other images PSNR performance is shown and these figures also follow the same trends as other images (Figure 9, 10).

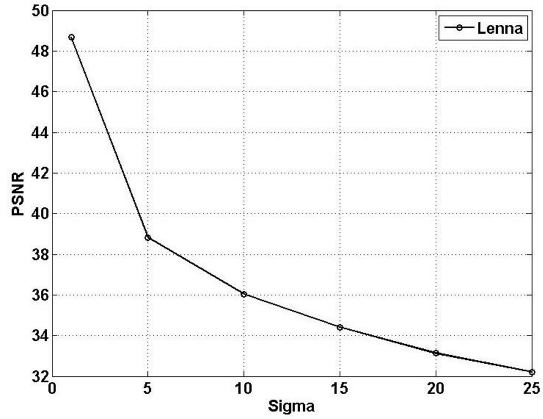


Figure 13: PSNR vs. Sigma for Lenna image

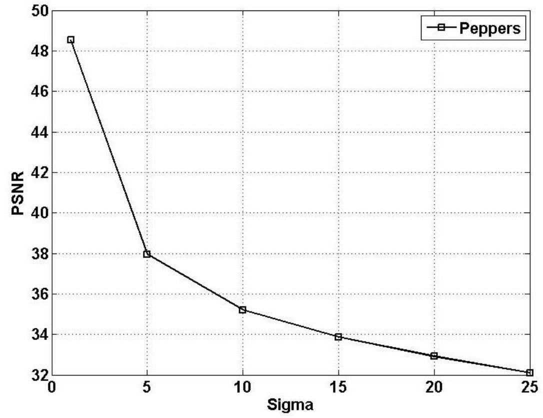


Figure 14: PSNR vs. Sigma for Peppers image

Table 1. Comparative Study of PSNR (dB) for BM3D and BM3D-PCA

	Noisy	BM3D	BM3D-PCA
Airplane	20.18	31.57	**31.80**
Baboon	20.16	25.45	**25.71**
Barbara	20.16	30.72	**31.00**
Boat	20.16	30.62	**30.78**
Lenna	20.16	32.05	**32.21**
Peppers	20.16	31.98	**32.11**

In table 1, results for BM3D and BM3D-PCA algorithms are compared in terms of PSNR. Results are compared for six images while considering σ to be 25. The noisy PSNR of each image is 20.16 dB. It is clear from the table that the performance of BM3D-PCA is better in comparison to BM3D.

5. CONCLUSIONS

This paper discusses techniques for noise removal in digital images. Block matching in addition with collaborative filtering is detailed. The use of PCA in noise reduction is also highlighted and resulted algorithm is defined as BM3D-PCA. The results of both the algorithms are generated using computer simulation. In the analysis, six digital images are considered. This has been found that both the algorithms are comparable to each other; however BM3D-PCA shows slightly better results for most of the images. This happens as PCA only retains the eigen values which are very close to principal components, which show better similarity with original noise free images.

REFRENCES

[1] L. Sendur and I. W. Selesnick, “Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency,” IEEE Trans. Signal Processing, 50(11), pp. 2744.2756, 2002.

[2] J. Guerrero-Colon and J. Portilla, “Two-level adaptive denoising using Gaussian scale mixtures in over-complete oriented pyramids,” in Proc. IEEE Int. Conf. Image Processing, vol. 1, Genova, Italy, September 2005.

[3] D. Muresan and T. Parks, “Adaptive principal components and image denoising,” in Proc. IEEE Int. Conf. Image Process., vol. 1, September 2003.

[4] A. Foi, V. Katkovnik, and K. Egiazarian, “Pointwise Shape-Adaptive DCT for high-quality denoising and deblocking of grayscale and color images,” *IEEE Trans. Image Process.*, 16(5), 1395.1411, 2007.

[5] Dabov, K., Foi, A., Katkovni k, V. and Egiazarian, K., “ Image denoising by sparse 3-D transform- domain collaborative filtering,” IEEE Transactions on Image Processing, 16(8), 2080-2095, 2007.

[6] Dabov, K., Foi, A., Katkovni k, V. and Egiazarian, K., “Image restoration by sparse 3D transform - domain collaborative filtering,” Proceedings of SPIE, 681207-12, 2008.

[7] Deledalle, C., Denis, L. and Tupin, F., “Iterative weighted maximum likelihood denoising with probabilistic patch-based weights,” IEEE Transactions on Image Processing, 18(12), 2661-2672, 2009.

[8] A. Pizurica, W. Philips, "Estimating the probability of the presence of a signal of interest in multiresolution single- and multiband image denoising" ,IEEE Transaction on Image Processing , Vol. 15, No.3, pp. 654–665,2006.

[9] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform domain collaborative filtering", IEEE Trans. Image Process., vol. 16, no. 8, pp. 2080–2095, Aug. 2007.

[10] L. Zhang, W. Dong, D. Zhang, and G. Shi, "Two-stage image denoising by principal component analysis with local pixel grouping", Pattern Recognition, vol. 43, pp. 1531-1549, Apr. 2010.

[11] Yingkun Hou, Chunxia Zhao, Deyun Yang, and Yong Cheng, "Comment on Image denoising by sparse 3D transform-domain collaborative filtering", IEEE Trans. Image Process., 20(1), 268–270, 2011.

[12] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries", IEEE Trans. Image Process., 15(12), 3736–3745, 2006.

[13] P. Scheunders, "Wavelet thresholding of multivalued images", IEEE Trans. Image Proc., 13(4), 475–483, 2004.