

COMPARATIVE ANALYSIS BETWEEN NON-LINEAR WAVELET BASED IMAGE DENOISING TECHNIQUES

Milan Chikanbanjar

Department of Computer Engineering, Khwopa Engineering College, Libali-8, Bhaktapur, Nepal

Abstract

Digital images have been a major form of transmission of visual information, but due to the presence of noise, the image gets corrupted. Thus, processing of the received image needs to be done before being used in an application. Denoising of image involves data manipulation to remove noise in order to produce a good quality image retaining different details. Quantitative measures have been used to show the improvement in the quality of the restored image by the use of various thresholding techniques by the use of parameters mainly, MSE (Mean Square Error), PSNR (Peak-Signal-to-Noise-Ratio) and SSIM (Structural Similarity index). Here, non-linear wavelet transform denoising techniques of natural images are studied, analyzed and compared using thresholding techniques such as soft, hard, semi-soft, LevelShrink, SUREShrink, VisuShrink and BayesShrink. On most of the tests, PSNR and SSIM values for LevelShrink Hard thresholding method is higher as compared to other thresholding methods. For instance, from tests PSNR and SSIM values of lena image for VISUShrink Hard, VISUShrink Soft, VISUShrink Semi Soft, LevelShrink Hard, LevelShrink Soft, LevelShrink Semi Soft, SUREShrink, BayesShrink thresholding methods at the variance of 10 are 23.82, 16.51, 23.25, 24.48, 23.25, 20.67, 23.42, 23.14 and 0.28, 0.28, 0.28, 0.29, 0.22, 0.25, 0.16 respectively which shows that the PSNR and SSIM values for LevelShrink Hard thresholding method is higher as compared to other thresholding methods, and so on. Thus, it can be stated that the performance of LevelShrink Hard thresholding method is better on most of tests.

Keywords: Image denoising, gaussian noise, salt and pepper noise, discrete wavelet transform, thresholding, non-linear wavelet techniques, Peak-Signal-to-Noise-Ratio (PSNR), SSIM (Structural Similarity Index).

1. Introduction

Noise removal is the most common and important preprocessing step in image processing applications. Image processing requires the denoising of image as one of its most vital parts. As the data consist of noise, many great studies have been done in this field. The newest one being the wavelet transforms which can be applied for digital signal and image processing for denoising and compression of data. The main objective of denoising is to recover the best estimate of the original image from its noisy version.

The denoising of a natural image corrupted by Gaussian noise is an abiding problem in signal processing. Images are also frequently corrupted by impulse noise, that is, salt and pepper noise, due to the faulty memory locations or timing errors in analog to digital conversion. Generally, non-linear filtering techniques are used in the removal of salt and pepper noise. The aim of an image-denoising algorithm is to recover the clean image from its noisy version by removing the noise and retaining the maximum possible image information. When the signal corruption is modeled as a Gaussian process, then the accuracy criterion is the mean square error “MSE” (Asem et al., 2013; V. Strela, 2000; Mohen et al., 2006). However, designing a filter based on this assumption frequently results in a filtered image that is more visually displeasing than the original noisy signal, even though the filtering operation successfully reduces the MSE.

*Corresponding author: Milan Chikanbanjar
Department of Computer Engineering, Khwopa Engineering
College, Libali-8, Bhaktapur, Nepal
Email: milancb13.cool@gmail.com
(Received: Nov 15, 2017 Accepted: Jan 02, 2018)

1.1. Non-Linear Threshold Filtering

The most investigated domain in denoising using wavelet transform is the non-linear coefficient thresholding based methods. In the wavelet hard thresholding technique (Mohen et al., 2006), each coefficient after applying wavelet transform is compared with a threshold value. If the coefficient is smaller than threshold, set to zero, else it is preserved.

In order to overcome the demerits of hard thresholding, wavelet transform using soft thresholding was introduced in (Asem et al., 2013; Fodor and Kamath, 2001). An improvement in wavelet thresholding is soft thresholding. In this method, firstly the k level decomposition is performed then, thresholding is applied to noisy coefficient.

One of the most known algorithms in non-adaptive threshold is VisuShrink (Asem et al., 2013; Fodor and Kamath, 2001) which depends only on number of data points. VISUShrink is known to yield overly smoothed images because its threshold choice can be unwarrantedly large due to its dependence on the number of pixels in the image.

SUREShrink uses a hybrid of the universal threshold and the “SURE” (Stein’s Unbiased Risk Estimator) threshold and performs better than VISUShrink. BayesShrink (Asem et al., 2013; Simoncelli and Adelson, 1996; Chipman et al., 1997) minimizes the Bayes’ Risk Estimator function assuming Generalized Gaussian prior and thus yielding data adaptive threshold. BayesShrink outperforms SUREShrink most of the times.

1.2. Image Denoising Techniques

1.2.1 Noise Model

1.2.1.1 Gaussian Noise

Noise having Gaussian-like distribution in the noisy image is the sum of the true pixel value and a random Gaussian distributed noise value (Jain, 1989). This type of noise with a Gaussian distribution has a bell-shaped probability distribution function given by,

$$F(g) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(g-m)^2}{2\sigma^2}} \quad (1)$$

Where g represents the gray level, m is the mean or average of the function and σ is the standard deviation of the noise.

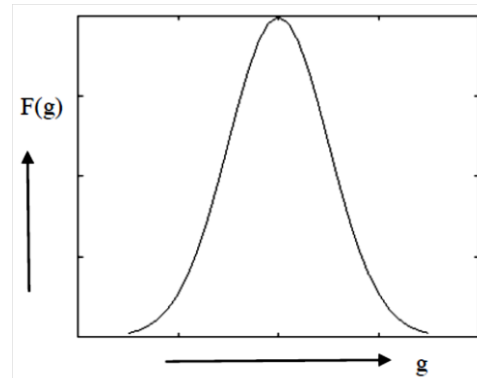


Fig 1 Gaussian distribution

1.2.1.2 Salt and Pepper Noise

Salt and Pepper noise is a kind of impulse noise having intensity spikes. The salt and pepper noise has scattered bright & dark disruptions with different intensities of color as compared with their neighboring pixels. The noisy pixel in case of salt & pepper noise has no relation with the color of the neighboring pixel affecting a small number of pixels. The contaminated image looks like it contains light and dark dots, and hence the name salt & pepper noise. For an 8-bit image, the minimum value i.e. pepper noise is set as 0, and the salt noise which has maximum value is set as 255. Salt and pepper noise occurs due to the defected pixels in the camera sensors and timing errors in the digitization error or faulty memory locations (Harleen et al., 2013; Saxena and Kourav, 2014).

1.2.2 Discrete Wavelet Transform

A Discrete Wavelet Transform (DWT) is any wavelet transform for which the wavelets are discretely sampled. DWT is a fast, linear operation on a data vector having the length with an integer power of 2. An image can be decomposed into a sequence of different spatial resolution image using DWT. In case of an image, an N level decomposition can be performed resulting in $3N+1$ different frequency bands (sub bands) such as LL, LH, HL and HH as shown in Fig 2.

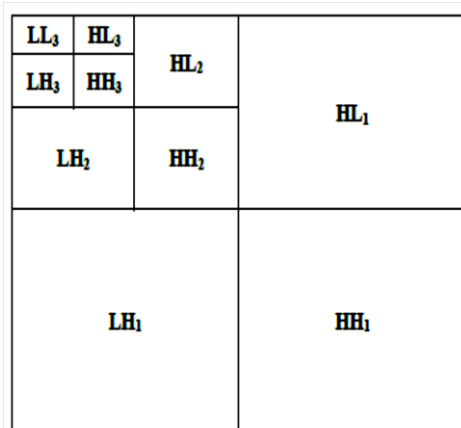


Fig 2 DWT with 3-Level decomposition (Perumal et al., 2008)

The sub-bands HH_k , HL_k , LH_k are called the details coefficients, where $k = 1, 2, \dots, j$; k is the decomposition level and j denotes the largest or coarsest scale in decomposition and LL_k is the approximation coefficient which is low resolution component. The next level of wavelet transform is applied to the low frequency sub band image LL only. The Gaussian noise will nearly be averaged out in low frequency wavelet coefficients. Therefore, only the wavelet coefficients in the high frequency levels need to be threshold. As a final step in the denoising algorithm, the inverse discrete wavelet transform is applied to build back the modified image from its coefficients.

1.2.3 Thresholding Methods

Thresholding methods use a threshold and determine the clean wavelet coefficients based on this threshold.

1.2.3.1 Hard Thresholding Method

The hard-thresholding function chooses all wavelet coefficients whose magnitudes are greater than the selected threshold value λ to remain as they are and the others with magnitudes smaller than λ are set to zero (Parmar, 2015). Mathematically, it is

$$f_h(x) = \begin{cases} x, & |x| \geq \lambda \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

The threshold λ is chosen according to the signal energy and the noise variance (σ^2).

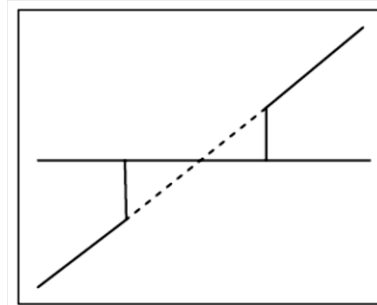


Fig 3 Hard-thresholding (Asem et al., 2013; Rangarajan et al., 2002)

1.2.3.2 Soft Thresholding Method

The soft-thresholding function shrinks the wavelet coefficients by λ towards zero, and hence are also called the wavelet shrinkage function. Soft thresholding is where the coefficients with greater than the threshold are shrunk towards zero after comparing them to a threshold value (Parmar, 2015).

$$f_s(x) = \begin{cases} \text{sign}(x)(|x| - \lambda), & \text{for } |x| \geq \lambda \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

The soft-thresholding rule is chosen over hard-thresholding, for the soft-thresholding method yields more visually pleasant images over hard thresholding.

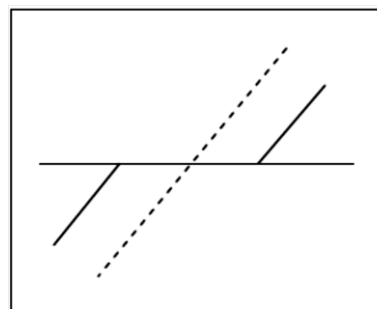


Fig 4 Soft-thresholding (Asem et al., 2013; Rangarajan et al., 2002)

1.2.3.3 Semi-Soft Thresholding Method

By choosing appropriate thresholds, semi-soft shrinkage offers advantages over both hard shrinkage and soft shrinkage (Parmar, 2015).

$$f_{ss}(x) = \begin{cases} 0, & |x| \leq \lambda_1 \\ \text{sign}(x) \left(\frac{\lambda_2 |x| - \lambda_1}{\lambda_2 - \lambda_1} \right), & \lambda_2 < |x| \leq \lambda_1 \\ x, & |x| > \lambda_2 \end{cases} \quad (4)$$

1.2.3.4 Universal Thresholding Method

The most original threshold, Universal Threshold,

was proposed by Donoho and Johnstone and is formulated as (Doye and Ruikar, 2011):

$$\lambda_{univ} = \sqrt{\sigma^2 * 2Ln(M)} \quad (5)$$

Where M is the signal size and σ^2 is noise variance estimated from the sub-band HH₁.

1.2.3.5 LevelShrink Filter

LevelShrink filter is a level-dependent thresholding algorithm which adopts different thresholds for different levels of the wavelet tree to improve the performance of the original wavelet thresholding method. It is calculated as:

$$\lambda_j = \lambda_{univ} \times 2^{\frac{-(J-j)}{2}} \quad (6)$$

Where J is the total number of decomposition levels and j is the scale level where the wavelet coefficient to be thresholded is located.

1.2.3.6 VisuShrink Filter

VisuShrink is thresholding by applying the Universal threshold proposed by Donoho and Johnstone. For denoising images, VisuShrink is found to yield an overly smoothed estimate. It is formulated as:

$$T = \sqrt{\sigma^2 * 2Log(M)} \quad (7)$$

Where σ^2 is the noise variance present in the signal and M represents the signal size or number of samples (Parmar, 2015).

1.2.3.7 SureShrink Filter

An adaptive threshold, called SureShrink, was developed by Donoho and Johnstone, which is named from Stein's unbiased risk estimation (SURE) (Cho, 2004). SureShrink is the combination of the universal threshold and the SURE threshold (Luisier et al, 2007). Sure Shrink (Luisier et al, 2007) is an adaptive thresholding method where the wavelet coefficients are treated in level-by-level fashion. SureShrink is used for suppression of additive noise in wavelet-domain where a threshold 'T' SURE is employed for denoising (Singh and Wadhvani, 2015).

The threshold parameter 'T' SURE is expressed as:

$$T_j = argmin (SURE_j(t, y)) \quad (8)$$

Where the Stein's Unbiased Risk (SURE) is minimized as follows:

$$SURE_j(t, y) = \sigma^2 - \frac{1}{N_j} \left(2\sigma^2 \cdot \#\{i: |y_i| \leq t\} + \sum_{i=1}^{N_j} \min(y_i, t) \right) \quad (9)$$

where N is the number of samples; J is the number of channels; N_j is the number of samples in the channel j; σ^2 is noise variance and y is the coefficient of the sub band.

1.2.3.8 BayesShrink Filter

BayesShrink is an adaptive data-driven threshold driven in a Bayesian framework, and we assume generalized Gaussian distribution (GGD) for the wavelet coefficients in each detail sub band and try to find the threshold T, which minimizes the Bayesian Risk. The goal of this method is to minimize the Bayesian risk, and hence its name, BayesShrink (Doye and Ruikar, 2011). The Bayes threshold, T_B, is defined as:

$$T_B = \frac{\sigma^2}{\sqrt{\sigma_s^2}} \quad (10)$$

Where σ^2 is the noise variance and σ_s^2 is the signal variance without noise. The noise variance σ^2 is estimated from the sub band HH by the median estimator. From the definition of additive-noise we have,

$$w(x, y) = s(x, y) + n(x, y) \quad (11)$$

Since the noise and the signal are independent of each other, it can be stated that

$$\sigma_w^2 = \sigma_s^2 + \sigma^2 \quad (12)$$

Where σ_w^2 can be computed as shown below:

$$\sigma_w^2 = \frac{1}{n^2} \sum_{x,y-1}^n w^2(x, y) \quad (13)$$

The variance of the signal, σ_s^2 is computed as

$$\sigma_s = \sqrt{\max(\sigma_w^2 - \sigma^2, 0)} \quad (14)$$

With σ^2 and σ_s^2 , the Bayes threshold is computed.

Using this threshold, the wavelet coefficients are threshold at each band (Perumal et al, 2008; Poleselet al, 2000).

2. Evaluation

The performance of each algorithm is compared by computing MSE, PSNR and SSIM, besides the visual interpretation. MSE, PSNR and SSIM are the three parameters used in this paper for comparison of denoising techniques. Mean square error (MSE) is calculated as:

$$MSE = \frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N [x(i, j) - x'(i, j)]^2 \quad (15)$$

Where x is the original image and x' is the denoised image.

PSNR is the peak signal to noise ratio. PSNR is the most commonly used parameter to measure the quality of reconstruction image with respect to the original image. A higher PSNR would normally indicate that the reconstruction is of higher quality. PSNR is usually expressed in terms of the logarithmic decibel scale (dB). PSNR of reconstructed image is formulated as:

$$PSNR = 10 \log_{10} \left(\frac{255^2}{MSE} \right) \quad (16)$$

The Structural SIMilarity (SSIM) index is a method for measuring the similarity between two images. The SSIM index can be viewed as a quality measure of one of the images being compared provided the other image is regarded as of perfect quality. SSIM is designed to improve on traditional methods such as peak signal-to-noise ratio (PSNR) and mean squared error (MSE).

The SSIM index is calculated on various windows of an image. The measure between two windows x and y of common size $N \times N$ is:

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

with:

μ_x the average of x ;

μ_y the average of y ;

σ_x^2 the variance of x ;

σ_y^2 the variance of y ;

σ_{xy} the covariance of x and y ;

$c_1 = (k_1L)^2$, $c_2 = (k_2L)^2$ two variables to stabilize the

division with weak denominator;

L the dynamic range of the pixel-values (typically this is $2^{\text{\#bits per pixel}} - 1$);

$k_1 = 0.01$ and $k_2 = 0.03$ by default.

The SSIM index satisfies the condition of symmetry: $SSIM(x, y) = SSIM(y, x)$.

3. Illustration of Proposed Algorithm

1. Input image $x_{i,j}$ corrupted with visual noise.
2. Compute the Discrete Wavelet Transform

$$F(t) = \sum_k \sum_j \alpha_{j,k} \beta_{j,k} \quad (18)$$

Where $\alpha_{j,k}$ and $\beta_{j,k}$ are the transform coefficients and basic functions respectively.

3. Apply different wavelet schemes.
4. Estimate the wavelet coefficients of an image produced by different wavelet schemes.
5. Compute the inverse discrete wavelet transform.

4. Simulated Results And Discussion

Different test images i.e. lena.jpeg, cameraman.jpeg, peppers.jpeg and mandrill.jpeg, of size 256 x 256 are taken for Gaussian noise application and airplane.png, barbara.png, and boat.png (512 x 512) images for Salt & Pepper Noise application as input images for simulation of results.

Test images for Gaussian Noise Application:

Original Images

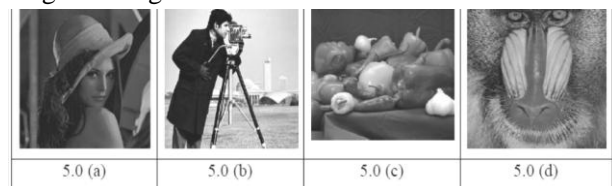


Fig 5 (a)Lena.jpeg (b) Cameraman.jpeg (c)Peppers.jpeg (d) Mandrill.jpeg

Test images for Salt & Pepper Noise Application:

Original Images

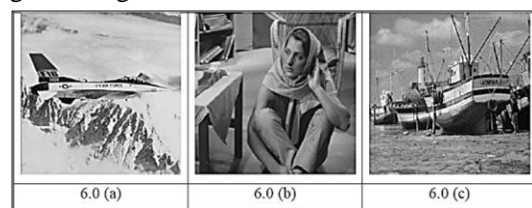


Fig 6 (a)airplane.png (b) barbara.png (c) boat.png

Table 1 PSNR Values for Lena image (Level 3)

Methods	Variance			
	10	20	30	40
VISU hard	26.45	29.51	27.51	34.54
VISU soft	25.18	25.95	25.29	25.92
VISU semi soft	26.56	29.9	36.86	31.06
Level Shrink hard	26.24	28.05	28.83	33.65
Level Shrink soft	25.52	27.87	30.36	32.17
Level Shrink semi soft	26.31	29.84	30.36	51.85
SURE shrink	26.23	28.39	29.03	31.84
Bayes Shrink	29.9	30.1	30.49	32.15

Table 3 PSNR Values for Cameraman image (Level 3)

Methods	Variance			
	10	15	20	25
VISU hard	49.61	53.77	49.11	37.37
VISU soft	26.77	25.31	24.16	22.34
VISU semi soft	47.92	47.98	51.04	36.2
Level Shrink hard	52.73	51.96	52.25	49.27
Level Shrink soft	46.93	43.25	37.07	31.73
Level Shrink semi soft	50.42	51.35	45.56	44.34
SURE shrink	26.77	25.31	24.16	22.34
Bayes Shrink	19.44	19.34	19	18.61

Table 2 SSIM Values for Lena image (Level 3)

Methods	Variance			
	10	20	30	40
VISU hard	0.28	0.23	0.23	0.14
VISU soft	0.28	0.25	0.25	0.22
VISU semi soft	0.28	0.22	0.09	0.16
Level Shrink hard	0.29	0.2	0.15	0.11
Level Shrink soft	0.28	0.22	0.17	0.14
Level Shrink semi soft	0.22	0.13	0.17	0.12
SURE shrink	0.24	0.2	0.18	0.14
Bayes Shrink	0.16	0.16	0.15	0.14

Table 4 SSIM Values for Cameraman image (Level 3)

Methods	Variance			
	10	15	20	25
VISU hard	0.61	0.37	0.25	0.23
VISU soft	0.5	0.35	0.22	0.17
VISU semi soft	0.6	0.41	0.22	0.22
Level Shrink hard	0.69	0.5	0.32	0.27
Level Shrink soft	0.55	0.38	0.24	0.2
Level Shrink semi soft	0.49	0.29	0.17	0.1
SURE shrink	0.5	0.35	0.22	0.17
Bayes Shrink	0.5	0.33	0.18	0.13

Visual Results:



Fig 7 (a) Original image (b) Noisy Image, Denoised by: (c) VISUshrink Hard (d) VISUshrink Soft (e) VISUshrink semi-soft (f) Level Shrink Hard (g) Level Shrink Soft (h) Level Shrink semi-soft (i) SureShrink (j) Bayes Shrink

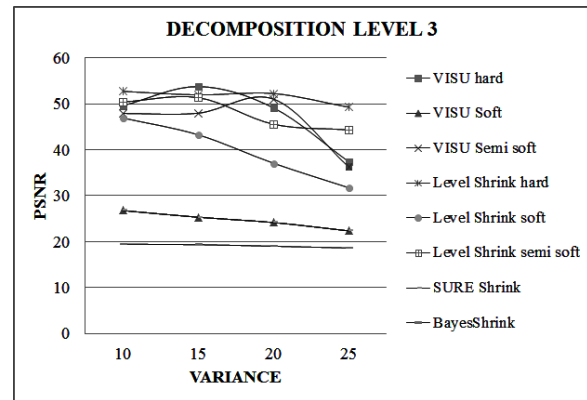


Fig 9 PSNR vs. VARIANCE for Cameraman image

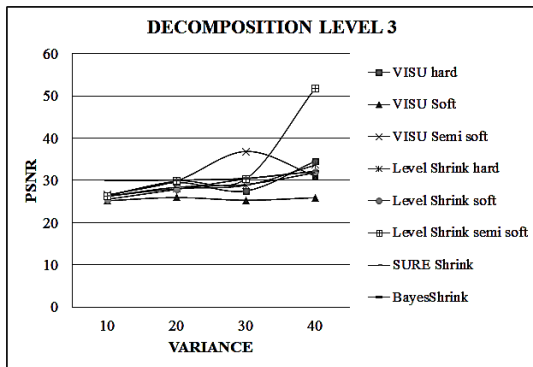


Fig 8 PSNR vs VARIANCE for Lena image

Table 5 PSNR Values for Peppers image (Level 3)

Methods	Variance			
	10	15	20	25
VISU hard	38.36	36.99	39.77	35.27
VISU soft	35.48	33.66	34.07	33.57
VISU semi soft	38.36	37.37	38.83	34.87
Level Shrink hard	40.53	36.08	38.07	36.53
Level Shrink soft	36.47	35.96	36.28	34.84
Level Shrink semi soft	40.16	41.49	42.93	42.67
SURE shrink	37.01	35.98	36.36	34.75
Bayes Shrink	36.55	35.92	36.29	34.71

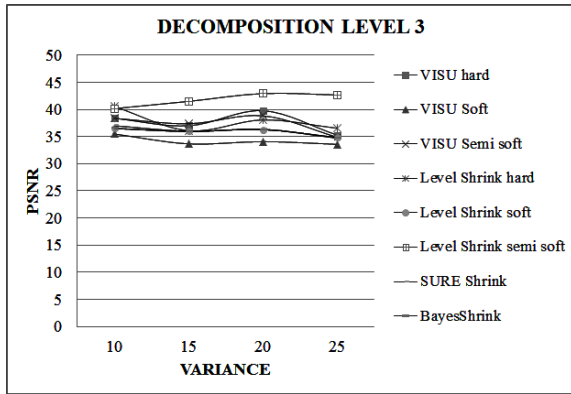


Fig 10 PSNR vs. VARIANCE for Peppers image

Table 6 SSIM Values for Peppers image (Level 3)

Methods	Variance				SSIM Values of images
	10	15	20	25	
VISU hard	0.27	0.19	0.11	0.11	
VISU soft	0.27	0.19	0.12	0.1	
VISU semi soft	0.27	0.18	0.11	0.1	
Level Shrink hard	0.25	0.21	0.14	0.12	
Level Shrink soft	0.28	0.18	0.11	0.09	
Level Shrink semi soft	0.21	0.12	0.06	0.04	
SURE shrink	0.24	0.16	0.1	0.08	
Bayes Shrink	0.23	0.15	0.1	0.08	

Table 7 PSNR Values for mandrill image (Level 3)

Methods	Variance				PSNR Values of images
	10	15	20	25	
VISU hard	48.03	43.48	45.99	32.26	
VISU soft	34.58	27.83	26.29	22.36	
VISU semi soft	47.45	41.58	40.44	29.19	
Level Shrink hard	53.64	44.65	54.39	41.5	
Level Shrink soft	41.28	37.21	40.15	28.9	
Level Shrink semi soft	56.04	52.85	45.47	48.82	
SURE shrink	37.13	32.08	31.71	28.88	
Bayes Shrink	32.09	31.47	31.09	28.8	

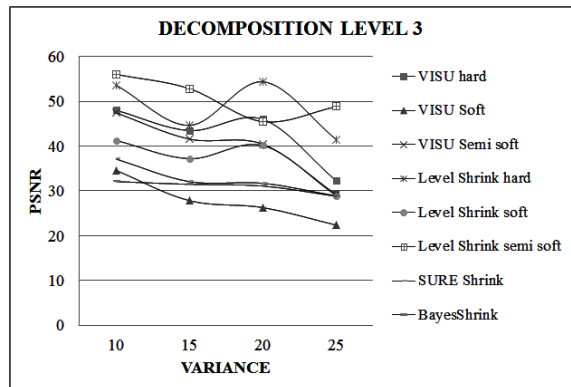


Fig 11 PSNR vs. VARIANCE for Mandrill image

Table 8 SSIM Values for mandrill image (Level 3)

Methods	Variance				SSIM Values of images
	10	15	20	25	
VISU hard	0.29	0.19	0.11	0.11	
VISU soft	0.27	0.16	0.11	0.09	
VISU semi soft	0.28	0.18	0.12	0.1	
Level Shrink hard	0.27	0.23	0.09	0.14	
Level Shrink soft	0.28	0.16	0.09	0.08	
Level Shrink semi soft	0.17	0.08	0.04	0.03	
SURE shrink	0.25	0.15	0.1	0.08	
Bayes Shrink	0.24	0.14	0.09	0.08	

Table 9 PSNR Values for Airplane image (Level 3)

Methods	Variance				PSNR Values of images
	12	17	22	27	
VISU hard	32.21	31.65	36.36	29.66	
VISU soft	23.46	21.3	19.96	18.58	
VISU semi soft	36.92	31.22	35.34	23.2	
Level Shrink hard	38.37	37.94	41.51	38.83	
Level Shrink soft	27.67	25.68	28.41	22.57	
Level Shrink semi soft	50.95	41.33	39.2	42.97	
SURE shrink	25.72	23.84	23.35	22.43	
Bayes Shrink	24.44	23.24	22.9	22.18	

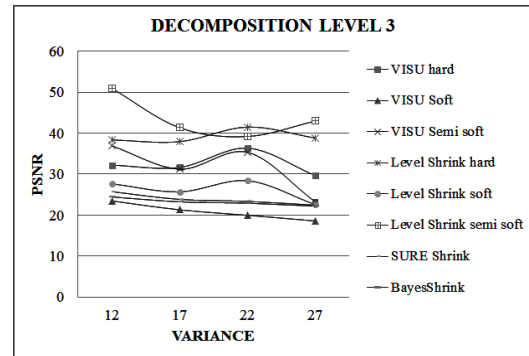


Fig 12 PSNR vs. VARIANCE for Airplane image

Table 10 SSIM Values for Airplane image (Level3)

Methods	Variance				SSIM Values of images
	12	17	22	27	
VISU hard	0.66	0.42	0.26	0.19	
VISU soft	0.46	0.34	0.2	0.14	
VISU semi soft	0.57	0.4	0.24	0.18	
Level Shrink hard	0.75	0.59	0.54	0.39	
Level Shrink soft	0.54	0.37	0.21	0.15	
Level Shrink semi soft	0.49	0.28	0.13	0.07	
SURE shrink	0.46	0.34	0.2	0.14	
Bayes Shrink	0.41	0.31	0.18	0.13	

Table 11 PSNR Values for Barbara image (Level 3)

Methods	Variance				PSNR Values of images
	10	15	20	25	
VISU hard	21.72	40.21	30.19	33.76	
VISU soft	14.72	26.95	20.08	19.35	
VISU semi soft	19.22	36.74	29.18	33.51	
Level Shrink hard	18.51	40.62	38.25	36.02	
Level Shrink soft	18	32.41	24.08	27.95	
Level Shrink semi soft	32.43	55.12	38.28	39.39	
SURE shrink	15.58	26.95	22.78	20.72	
Bayes Shrink	15.61	23.37	22.55	20.64	

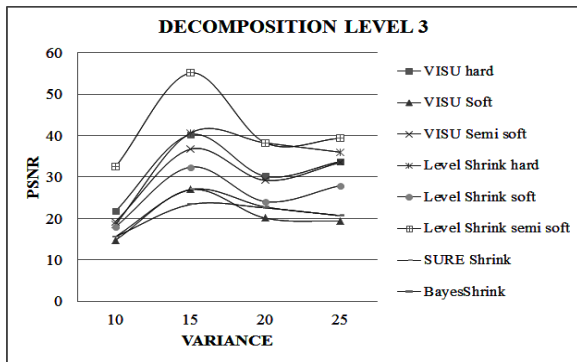


Fig 13 PSNR vs. VARIANCE for Barbara image

Table 12 SSIM Values for Barbara image (Level 3)

Methods	Variance				SSIM Values of images
	10	15	20	25	
VISU hard	0.22	0.41	0.26	0.16	
VISU soft	0.24	0.34	0.21	0.15	
VISU semi soft	0.24	0.39	0.25	0.14	
Level Shrink hard	0.19	0.61	0.46	0.36	
Level Shrink soft	0.24	0.37	0.22	0.14	
Level Shrink semi soft	0.16	0.18	0.09	0.04	
SURE shrink	0.25	0.34	0.21	0.15	
Bayes Shrink	0.25	0.32	0.2	0.14	

Table 13 PSNR Values for Boat image (Level 3)

Methods	Variance				PSNR Values of images
	11	16	21	26	
VISU hard	56.31	39.19	38.89	34.49	
VISU soft	27.01	25.64	23.17	18.56	
VISU semi soft	45.91	38.65	34.83	34.21	
Level Shrink hard	45.02	50.09	39.27	40.06	
Level Shrink soft	39.3	33.69	29.04	27.88	
Level Shrink semi soft	47.58	52.86	47.43	38.61	
SURE shrink	27.01	25.64	23.17	21.11	
Bayes Shrink	24.36	23.31	22.35	20.98	

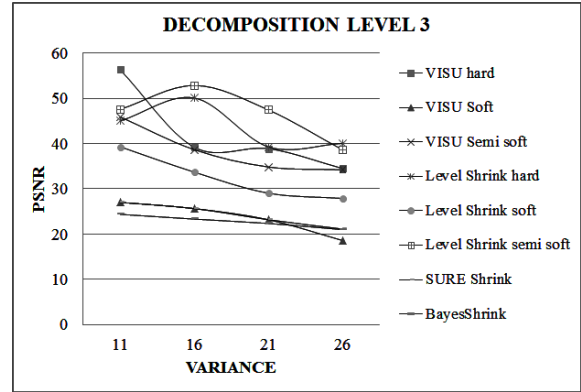


Fig 14 PSNR vs. VARIANCE for Boat image

Table 14 SSIM Values for Boat image (Level 3)

Methods	Variance				SSIM Values of images
	11	16	21	26	
VISU hard	0.52	0.35	0.23	0.16	
VISU soft	0.42	0.28	0.19	0.13	
VISU semi soft	0.51	0.33	0.23	0.15	
Level Shrink hard	0.69	0.52	0.42	0.33	
Level Shrink soft	0.47	0.3	0.2	0.13	
Level Shrink semi soft	0.4	0.19	0.1	0.05	
SURE shrink	0.42	0.28	0.19	0.13	
Bayes Shrink	0.37	0.25	0.18	0.13	

From above test results, Thresholding Techniques that obtained higher values of PSNR and SSIM with Gaussian Noise and Salt and Pepper Noise are summarized below:

Table 15 Summarized test results of Lena image, Cameraman image, Pepper image and Mandrill image with Gaussian Noise (Level 3)

Image	Variance				High SSIM value
	10	20	30	40	
Lena	Level Shrink Hard	VISU Soft	VISU Soft	VISU Soft	High PSNR value
	BayesShrink	BayesShrink	VISU Semi Soft	Level Shrink Semi Soft	
Cameraman	Variance				High SSIM value
	10	15	20	25	
	Level Shrink Hard	Level Shrink Hard	Level Shrink Hard	Level Shrink Hard	
Pepper	Level Shrink Hard	VISU Hard	Level Shrink Hard	Level Shrink Hard	High PSNR
	Variance				
	10	15	20	25	
	Level Shrink Soft	Level Shrink Hard	Level Shrink Hard	Level Shrink Hard	

	Hard	semi soft	semi soft	semi soft	value
Mandrill	Variance				
	10	15	20	25	
	VISU Hard	Level Shrink Hard	VISU Soft	Level Shrink Hard	High SSIM value
	Level Shrink semi soft	Level Shrink semi soft	Level Shrink Hard	Level Shrink semi soft	High PSNR value

Table 16 Summarized test results of Airplane image, Barbara image and Boat image with Salt and Pepper Noise (Level 3)

Image	Variance				
	12	17	22	27	
Airplane	Level Shrink Hard	Level Shrink Hard	Level Shrink Hard	Level Shrink Hard	High SSIM value
	Level Shrink semi soft	Level Shrink semi soft	Level Shrink Hard	Level Shrink semi soft	High PSNR value
Barbara	Variance				
	10	15	20	25	
	Level Shrink Hard	Level Shrink Hard	Level Shrink Hard	Level Shrink Hard	High SSIM value
	Level Shrink Soft	Level Shrink Soft	Level Shrink Soft	Level Shrink Soft	High PSNR value
Boat	Variance				
	11	16	21	26	
	Level Shrink Hard	Level Shrink Hard	Level Shrink Hard	Level Shrink Hard	High SSIM value
	VISU Hard	Level Shrink Soft	Level Shrink Soft	Level Shrink Hard	High PSNR value

5. Conclusion

From the results obtained above, it can be concluded that LevelShrink Semi-Soft thresholding method has best PSNR values whereas LevelShrink Hard thresholding method has better PSNR and SSIM values on most of tests. VISU Shrink Soft method has poor PSNR and SSIM for the output image as compared to the other methods considered.

Though, LevelShrink Semi-Soft and LevelShrink Hard thresholding methods have better PSNR values, but they cannot denoise properly the salt & pepper noise. SURE Shrink, Bayes Shrink and VISU Shrink are poor in terms of PSNR values but work well for denoising Salt & Pepper Noise with visually pleasing results.

References

[1] A. Jain et al (2013). Performance Comparison of Two Image Denoising Algorithm at Different Noises,

International Journal of Emerging Technology & Advanced Engineering, Volume 3, Issue 12.

[2] A. Khmag et al (2013). Review of Image Denoising Algorithms Based on the Wavelet Transformation, International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE), Vol.2, No.5, Pages: 01-08, Special Issue of ICACET.

[3] A. K. Jain (1989). Fundamentals of Digital Image Processing, Prentice Hall, Englewood cliffs.

[4] A. Polesel et al (2000). Image Enhancement Via Adaptive Unsharp Masking, IEEE Transactions On Image Processing, Vol. 9, No. 3, pp. 505-509.

[5] B. Sudharani (2015). A Better Thresholding Technique For Image Denoising Based On Wavelet Transform, International Journal of Innovative Research in Computer and Communication Engineering Vol. 3, Issue 5.

[6] C. Saxena et al (2014). Noises and Image Denoising Techniques: A Brief Survey, International Journal of Emerging Technology and Advanced Engineering, ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 4, Issue 3.

[7] D. Cho (2004). Image Denoising Using Wavelet Transforms, in his dissertation.

[8] D. D. Doye et al (2011). Wavelet Based Image Denoising Technique, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 2, No.3.

[9] D. Sharma (2008). A Comparative analysis of thresholding technique used in image denoising through wavelets, in her dissertation.

[10] E. P. Simoncelli et al (1996). Noise removal via Bayesian wavelet coring, in Third Int'l Conf on Image Proc, Lausanne, IEEE Signal Proc Society, volume I, pages 379-382.

[11] F. Luisier et al (2005). Which wavelet bases are the best for image denoising?, from Conference, Volume 5914, San Diego, California, USA, Proc. SPIE 5914, Wavelets XI, 59140E; doi:10.1117/12.614999.

[12] F. Luisier et al (2007). A new SURE approach to image denoising: Interscale orthonormal wavelet thresholding, IEEE Transactions on Image Processing, vol. 16, no. 3, pp. 593-606.

[13] H. A. Chipman et al (1997). Adaptive Bayesian wavelet shrinkage, vol. 92, No 440, pp.1413-1421.

- [14] H. Choi et al (1998). Analysis of wavelet domain Wienerfilters, in IEEE Int.Symp, Pittsburgh. Mathematics, Progress in Mathematics series, BirkhauserVerlag.
- [15] H. K. Khehra et al (2013). Performance Enhancement of Wavelet Techniques, International Journal of Emerging Technology and Advanced Engineering, Volume 3, Issue 5.
- [16] I. K. Fodor et al (2001). Denoising through wavlet shrinkage: An empirical study, Center for applied science computing Lawrence Livermore National Laboratory.
- [17] M. Ghazel et al (2006). Fractal- Wavelet Image Denoising Revisited, IEEE transaction on image processing, vol. 15, no. 9, pp. 2669-2675.
- [18] M. Salarian et al (2007). Wavelet Based Echocardiography Image Denoising, International conference on Wavelet Analysis and Multirate Systems, Arcachon, France, (pp. 153-157).
- [19] P. Hedaoo et al (2011). Wavelet Thresholding Approach For Image Denoising, International Journal of Network Security & Its Applications (IJNSA), Vol.3, No.4, DOI: 10.5121/ijnsa.2011.3402 16.
- [20] P. L. Parmar (2015). Image Denoising Using VISUShrink Algorithm, ISSN:0975, Volume – 02, Issue – 02.
- [21] R. K. Rai et al (2011). Implementation of Image Denoising using Thresholding Techniques, International Journal of Computer Technology and Electronics Engineering (IJCTEE),ISSN 2249-6343, Volume 1, Issue 2, pp.6-10.
- [22] R. Pavithra et al (2015). Wavelet Based Non Local Means Algorithm for Efficient Denoising of MRI Images, International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 2.
- [23] R. Rangarajan et al (2002). Image Denoising Using Wavelets - Wavelets & Time Frequency.
- [24] S. K. Mohideen et al (2008). Image De-noising using Discret Wavelet transform,IJCSNS (International Journal of Computer Science and Network Security), vol.8, no.1.
- [25] S. Singh et al (2015). Medical Image Denoising Using Sub Band Adaptive Thresholding Techniques Based on Wavelet 2D Transform, International Journal of Bio-Science and Bio-Technology Vol.7, No.5, pp.327-334.
- [26] V. Strela (2000). Denoising via block Wiener filtering in wavelet domain,Proc. Third European Congress of