Wavelet Based Image Denoising

Tejash More, Dwaipayan Haldar
M Tech, Signal Processing
Indian Institute of Science

November 16, 2025

Introduction

- Spatial denoising smooths pixels directly, so it often blurs edges and fine textures, especially at higher noise levels. It cannot distinguish noise from real high-frequency details.
- While wavelet based denoising, noise is spread mostly as low coefficients values in the wavelet domain. Applying the threshold removes the low coefficient noise while preserving the high coefficient edges.
- The basic workflow of our denoising algorithm works as follows:

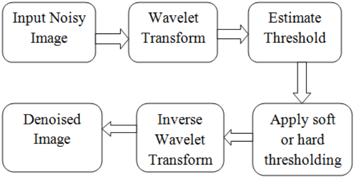


Figure 1: Flowchart of the Overall Algorithm

The Wavelet Transform

- The Discrete Wavelet Transform (DWT) decomposes an image into multiple frequency subbands (LL, HL, LH, HH), capturing coarse structure and fine details at different scales.
- It provides a sparse representation where important features concentrate in a few large coefficients.
- Noise mostly appears in the high-frequency bands, making it easy to remove using thresholding.

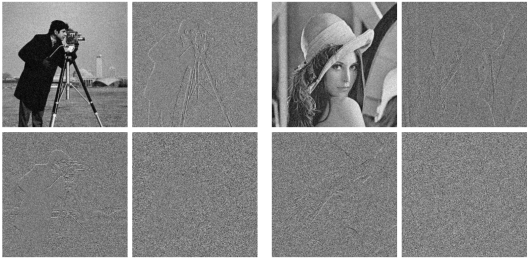


Figure 2: LL,HL,LH,HH representation for cameraman and lena with Haar Wavelet (1-Level)

Thresholding / Shrinking Algorithms

VisuShrink

- Simplest of all. Oversmooths. Uses universal threshold: $T = \sigma\sqrt{2\log n}$, where σ is estimated by the robust median estimator $\hat{\sigma} = \frac{Median(|Y_{ij}|)}{0.6745}$, $Y_{ij} \in$ subband HH1

SUREShrink

- Uses Stein's Unbiased Risk Estimate(SURE) for estimating bayesian risk. Picks the threshold that minimizes $SURE(t; \mathbf{x})$. It is given by:

$$T_S = argmin_{t\geq 0} SURE(t; \mathbf{x}) = d - 2 \cdot \sum_{i=1}^{d} \mathbb{1}_{|x_i| \leq t} + \sum_{i=1}^{d} min(x_i^2, y^2)$$

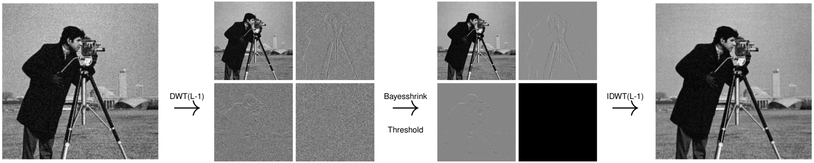
BayesShrink

- Minimizes the Bayesian risk using a GGD prior.

$$T_B = \frac{\sigma^2}{\sigma_X}; \hat{\sigma}_X = \sqrt{max(\hat{\sigma}_Y^2 - \hat{\sigma}^2, 0)}; \hat{\sigma}_Y^2 = \frac{1}{n^2} \sum_{i,j=1}^{n} Y_{ij}^2$$

Complete Workflow: Cameraman

The complete workflow is given cameraman image. Level-1 wavelet decompostion is shown for easy visualization. Experiments are carried on Level-4. Here Haar filter is used. Denoising is done using BayesShrink.



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

Figure 3: Denoising Using whileavelet Transform Flowchart: (a) Denoised Image, (b) DWT with Haar Wavelet Level-1, (c) Thresholded Wavelet coefficients using BayesShrink threshold, (d) Denoised Image

Evaluation Metrics

Mean Squared Error (MSE)

$$\text{MSE} = \frac{1}{MN} \sum (I - \hat{I})^2$$

Peak Signal-to-Noise Ratio (PSNR)

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

SSIM (Structural Similarity Index)

$$\text{SSIM}(I, \hat{I}) = \frac{(2\mu_I \mu_{\hat{I}} + C_1)(2\sigma_{I\hat{I}} + C_2)}{(\mu_I^2 + \mu_{\hat{I}}^2 + C_1)(\sigma_I^2 + \sigma_{\hat{I}}^2 + C_2)}$$

Visual Results (Example: Cameraman)

- Show Noisy Image
- VisuShrink result (PSNR: X dB, SSIM: Y)
- SUREShrink result (PSNR: X dB, SSIM: Y)
- BayesShrink result (PSNR: X dB, SSIM: Y)

[Insert 4-image comparison grid]

Quantitative Comparison (Averaged Over 12 Images)

- **Average PSNR Table**

Noise	VisuShrink	SUREShrink	BayesShrink
Gaussian	XX.X	XX.X	XX.X
Salt-Pepper	XX.X	XX.X	XX.X
Random	XX.X	XX.X	XX.X

- **Average SSIM Table**

Noise	VisuShrink	SUREShrink	BayesShrink
Gaussian	X.XXX	X.XXX	X.XXX
Salt-Pepper	X.XXX	X.XXX	X.XXX
Random	X.XXX	X.XXX	X.XXX

Graphical Comparison

- Bar plot: PSNR vs Algorithm for Gaussian Noise
- Bar plot: PSNR vs Algorithm for Salt-and-Pepper Noise
- Bar plot: SSIM comparison

[Insert graphs/placeholders here]

Failure Cases and Limitations

- Textured images (e.g., Barbara) show artifacts
- VisuShrink oversmooths edges
- Salt-and-pepper noise not well handled by wavelets
- High-frequency details often lost
- Threshold selection sensitive to noise variance estimation

[Insert failure case images]

Conclusion

- BayesShrink gives highest PSNR for Gaussian noise
- SUREShrink gives best detail preservation
- VisuShrink is simplest but oversmooths
- Wavelet denoising effective but limited for non-Gaussian noise
- Future work:
  - Non-local Means + Wavelets
  - BLS-GSM
  - Deep Learning-based Hybrid Models