AIP Assignment 3

G Ramesh Babu
SR. No. 20950

Abstract

This report consists the results and interpretations of AIP Assignment 3. Python code for this assignment is in ***Ramesh_AIP3.ipynb***

1. Linear combinations of order statistics for uniformly distributed noise

1.1. Objective

Consider the noise model $Y_i = x + Z_i, i = 1, 2, 3, ..., N$ where Y_i are in the neighbourhood of given x. The objective of this problem is to find the coefficients α_i of $\hat{X} = \sum_{i=1}^{N} \alpha_i Y_{(i)}$ which would minimise the Mean Square Error between $\hat{X}$ and x where $Y_{(i)}$ are the order statistics of Y_i. In this problem, we are assuming that the noise Z_i involved are iid samples drawn from $\sim$ U$[-1, 1]$

1.2. Mathematical Background

$$Y_i = x + Z_i, i = 1, 2, 3, ..., N \tag{1}$$

For an Unbiased Estimate,

$$\sum_{i=1}^{N} \alpha_i = 1, \quad \alpha_i = \alpha_{N-i+1}$$

Mean Square Error is given by

$$MSE = \mathbb{E}[(\hat{X} - x)^2)] \tag{2}$$

where $\hat{X} = \sum_{i=1}^{N} \alpha_i Y_{(i)}$ and $Y_i = x + Z_i$

$$\alpha^* = \arg \min_{\sum_{i=1}^{N} \alpha_i = 1} \mathbb{E}[(\hat{X} - x)^2)] \tag{3}$$

This equation reduces to

$$\alpha^* = \arg \min_{e^T\alpha=1} \alpha^T H \alpha \tag{4}$$

where e is the vector $[1, ..., 1]^T$ of dimension N and H is the covariance matrix of order statistics of Z_i

By solving this equation using Lagrangian multipliers, Solution to this Optimization problem is

$$\alpha = \frac{H^{-1}e}{e^T H^{-1} e} \tag{5}$$

We use this equation to find the coefficients of the linear combination of order statistics of Y_i

H matrix is computed as follows:

$$H \in \mathbb{R}^{N\times N}, H_{ij} = \mathbb{E}[Z_{(i)}Z_{(j)}]$$

Density Function of $Z_{(i)}$ (ith Order Statistic of Z)

$$g_{z_{(i)}}(z) = K_{ij} \cdot F_Z^{(i-1)}(z)(1 - F_Z(z))^{N-i} \cdot f_Z(z) \tag{6}$$
$$\text{where, } K_{ij} = \frac{N!}{(i-1)!(N-i)!}$$

Joint Density Function of $Z_{(i)}, Z_{(j)}$ (ith, jth Order Statistic of Z, $i < j$)

$$g_{z_{(i)}z_{(j)}}(z_1, z_2) = K_{ij} \cdot F_Z(z_1)^{i-1} \cdot (F_Z(z_2) - F_Z(z_1))^{j-i-1} \cdot (1 - F_Z(z_2))^{N-j} \cdot f_Z(z_1) \cdot f_Z(z_2) \tag{7}$$
$$\text{where, } K_{ij} = \frac{N!}{(i-1)!(j-i-1)!(N-j)!}$$

Distribution function of Z_i

$$F_z(z) = \begin{cases} 0, & z < -1 \\ \frac{z+1}{2}, & -1 \le z < 1 \\ 1, & z \ge 1 \end{cases} \tag{8}$$

Density Function of Z_i

$$f_z(z) = \begin{cases} 0, & z < -1 \\ 1/2, & -1 \le z < 1 \\ 0, & z \ge 1 \end{cases} \tag{9}$$

Computation of H using above equations:

$$H_{ii} = \int_{-\infty}^{\infty} z^2 g_{z_{(i)}}(z)\, dz \tag{10}$$

$$H_{ij} = \iint_{z_1 < z_2} z_1 z_2\, g_{z_{(i)}z_{(j)}}(z_1, z_2)\, dz_1\, dz_2 \tag{11}$$

1.3. Results

The given problem is tried out for various values of N and the following results are for N = 5

$$H = \begin{bmatrix} 0.52380952 & 0.28571429 & 0.04761905 & -0.19047619 & -0.42857143 \\ 0.28571429 & 0.23809524 & 0.0952381 & -0.04761905 & -0.19047619 \\ 0.04761905 & 0.0952381 & 0.14285714 & 0.0952381 & 0.04761905 \\ -0.19047619 & -0.04761905 & 0.0952381 & 0.23809524 & 0.28571429 \\ -0.42857143 & -0.19047619 & 0.04761905 & 0.28571429 & 0.52380952 \end{bmatrix} \tag{12}$$

$$\alpha = \begin{bmatrix} 0.5 & 0 & 0 & 0 & 0.5 \end{bmatrix} \tag{13}$$

While Computation of H matrix, **scipy.integrate** is used to compute the definite integrals

The optimal OSF estimator for x is given by: $\hat{X} = 0.5Y_{(1)} + 0.5Y_{(5)}$ which is equivalent to averaging the lowest and highest observations. This holds for any value of N

2. Block Matching and 3D Filtering

2.1. Objective

The Objective of this problem is to perform various denoising experiments on the BM3D Algorithm

2.2. Experiments

The given lighthouse image is grayscaled and a white Gaussian noise with variance $\sigma_z^2 = 100$ is added to it. The pixels values are ensured to be within the range of $0 - 255$

- BM3D Algorithm is applied on the noisy lighthouse image and the output images are compared before and after the application of Wiener filter. Mean Square Error is calculated with the original grayscaled image for these two cases
- Performance variation (in terms of MSE) of entire Algorithm with respect to the choice of the input noise variance $\sigma_z^2 = 100$ in the Algorithm is observed by changing σ_{psd}^2 between $1 - 400$
- Wiener filter in the second stage is replaced with a hard thresholding estimate and the performance is compared with the former in terms of MSE

2.3. Results

Experiment	Mean Square Error with grayscaled Image
Denoised Image before Wiener filter	32.8946
Denoised Image after Wiener filter	27.9683
Denoised Image after hard thresholding again	63.3816
Min. Mean Square Error	27.9683

Table 1. Mean Square Errors obtained in the below mentioned experiments

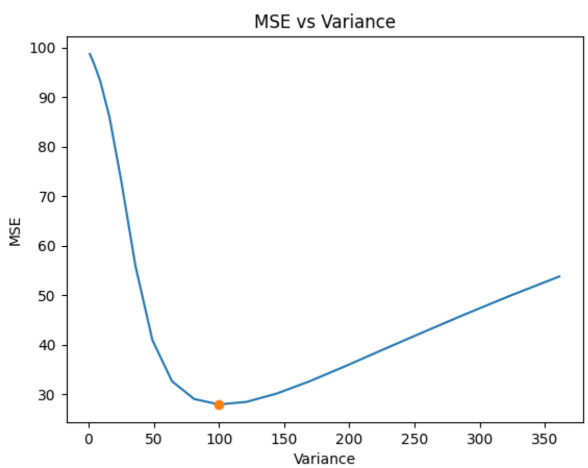

Figure 1. Mean Square Error vs Variance, minimum at $\sigma_{psd}^2 = 100$

- Table 1 shows the Mean Square Errors with the denoised images befor Wiener filter, after Wiener filter and also after hard thresholding the ouput image from 1st stage again.
- Figure 1 shows the variation of Mean Square Error for a choice of the input noise variance $\sigma_z^2 = 100$, with respect to the change in Variance σ_{psd}^2

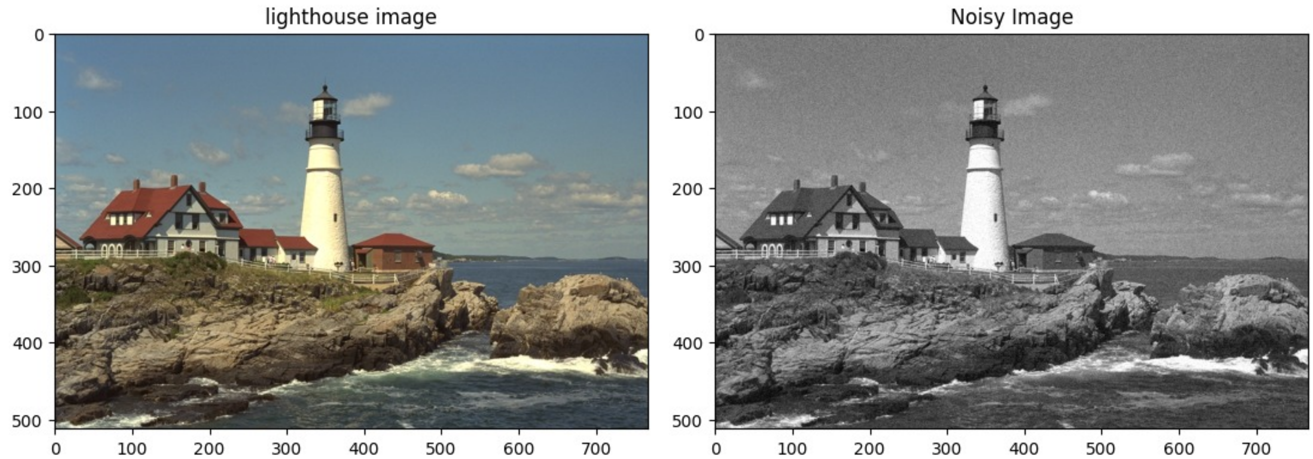

Figure 2. Lighthouse Image and its Grayscaled Noisy version

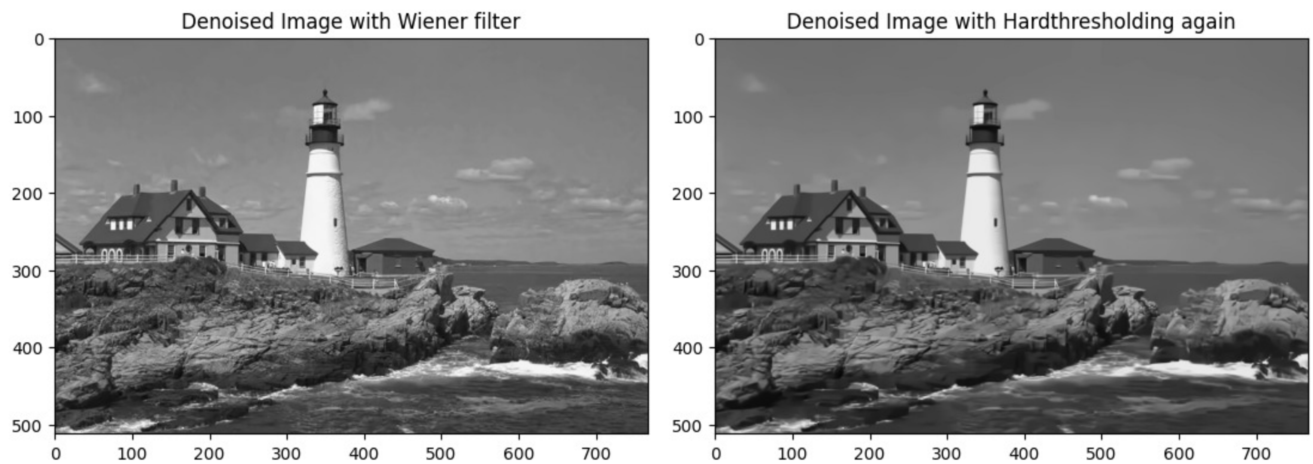

Figure 3. BM3D applied with Wiener filter and with hard thresholding

- **2.1** The MSE obtained after 1st stage should be higher than the MSE obtained after 2nd stage. This is because after the 1st stage, the noise in the image is reduced to some extent but some artifacts are introduced due to non-linear thresholding applied to the transform coefficients which are reduced by the Wiener filter in the 2nd stage. So the MSE drops after the 2nd stage. Table 1 shows that the MSE after 2nd stage is lesser than the MSE after 1st stage

- **2.2** When the variance σ^2_{psd} is lesser than the true variance, BM3D algorithm maynot remove much noise in 2nd stage, as the variance assumed by the Wiener filter is lesser than the true value. When the variance σ^2_{psd} is higher than the true variance, BM3D algorithm may remove true image details which increases the MSE. Thus MSE vs Variance follows a U shaped curve, with minimum value at $\sigma^2_{psd} = \sigma^2_Z$. Hence BM3D algorithm performs best when the variance of the noise is known

- **2.3** The first stage of BM3D applies a non-linear filter that removes coefficients below a certain threshold. The second stage of BM3D applies a Wiener filter (which is linear) which assumes that the noise added is Gaussian and based on the SNR, preserves the image details while suppressing the noise. Thus Hard thresholding might introduce artifacts at the edges which increase MSE. Hence replacing Wiener filter with Hard thresholding increases MSE eventhough it is computationally faster