Adaptive Wiener Filtering of Noisy Images

Darshan Tank

150020012

Electrical Engineering, IIT Bombay

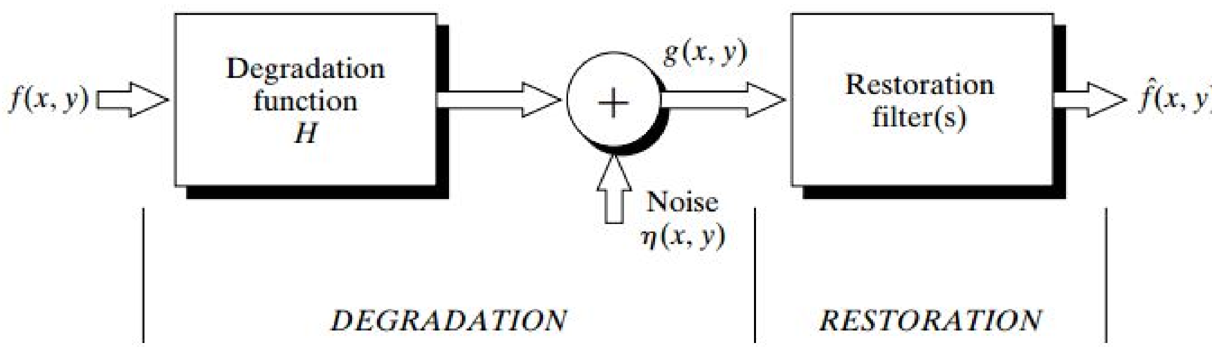
Abstract

Wiener filter is generally used to denoise signals. However, it requires us to know local statistics of actual signal which we generally don't know. Hence, we generally estimate them to which leads to errors. However, by proper assigning proper weights to each pixel in local patches, we can significantly improve the performance. We also utilize wavelet domain to provide a better solution to the problem of denoising.

Keywords: Filtering, Wiener Filter, Image Processing

Introduction

Images and in general, any form of signal, get corrupted during transmission. This noise has to be removed for efficient retrieval of images and signals. This process is known as image restoration. Image restoration is the process to restore the original image from a degraded image. This is generally done with an assumption on the type of degradation. We also define some parameter on which the restored image will be judged to find optimum restored image. Image restoration is quite an important topic of research as images acquired through any means go under some degradation and for proper analysis, we need image captured to be as close to original as possible. The standard model used for a degradation process is as follows:



However, for the purpose of this project , we will limit our self to only noise removal. Hence, we have the following equation:

$$g(x,y) = f(x,y) + n(x,y)$$

Approach

Wiener Filtering

It is also called Minimum Mean Square Error filtering. This method considers images and noise as random variables, and the objective is to find an estimate of the non degraded image such that the mean square error between them is minimized. This error measure is given by:

$$e^2 = E\{ (f - \dot{f})^2 \}$$

where E{ } is the expected value of the argument.

For discrete signals, it is given as:

$$MSE(\hat{\mathbf{x}}) = \frac{1}{N} \sum_{i,j=1}^{N} (\hat{x}(i,j) - x(i,j))^2$$

Wiener filter is optimal filter for stationary Gaussian Processes. When f(x,y) is also white gaussian, wiener filter has this simple form:

$$\hat{x}(i,j) = \frac{\sigma_x^2(i,j)}{\sigma_x^2(i,j) + \sigma_n^2(i,j)} [y(i,j) - \mu_x(i,j)] + \mu_x(i,j)$$

Where σ^2 and μ are variances and mean. Noise is assumed to have zero mean.

Local Wiener Filtering

Images have edges, constant regions and gradients and hence they can't be considered stationary. However given a small patch, they can considered stationary in that patch. Under this assumption, we can apply wiener filter to local patches taking local means and variances. Estimates of local means and variances are given by:

$$\hat{\mu}_x(i,j) = \frac{1}{(2r+1)^2}\sum_{p=i-r}^{i+r}\sum_{q=j-r}^{j+r} y(p,q)$$

$$\hat{\sigma}_x^2(i,j) = \frac{1}{(2r+1)^2}\sum_{p=i-r}^{i+r}\sum_{q=j-r}^{j+r} (y(p,q) - \hat{\mu}_x(i,j))^2 - \sigma_n^2$$

Where it is assumed that we know variance of noise which can be estimated through standard techniques. We will concentrate only on removal here.

The main problem in this assumption is that image might still be non stationary in the given local patch. This is generally seen in patches having edges. Image will have different variances on different sides of edges and hence we won't get an effective removal in such cases. Hence, local wiener filter fails at the edges.

Adaptive Weighted Averaging

Instead of just taking values of each pixel directly in calculating means and variances, we assign a weights to them such that pixels have values similar to center value of patch are given higher weightage.

$$\hat{\mu}_x(i,j) = \sum_{p=i-r}^{i+r}\sum_{q=j-r}^{j+r} w(i,j,p,q)y(p,q)$$

$$\hat{\sigma}_x^2(i,j) = \sum_{p=i-r}^{i+r}\sum_{q=j-r}^{j+r} w(i,j,p,q)(y(p,q) - \hat{\mu}_x(i,j))^2$$

Where w(i,j,p,q) is weight given by:

$$w(i,j,p,q) = \frac{K(i,j)}{1 + a(max[\epsilon^2, (y(i,j) - y(p,q))^2])}$$

Where K(i,j) is normalization constant over a patch.

The parameter $\epsilon = 2.5\ \sigma^2$ and a>0, $a\epsilon^2 >> 1$ to remove outliers.

Local Wavelet Base Wiener Filter

Since we have considered image to be non-stationary mean and non-stationary variances, its wavelet coefficient can be considered the same. Hence all of above methods can be applied to wavelet domain. However, taking mean in wavelet domains leads to artifacts. Hence, mean is taken zero for wavelet domain.

Combined Approach

Here final estimate is taken as a linear combination of estimate from spatial domain and wavelet domain. Linear coefficients can be estimated through correlation between spatial and wavelet domain, that is left as future work and we just take coefficients as 0.4 and 0.6 for estimates from spatial and wavelet domains respectively.

Results

Code link in the form of jupyter notebook is give at the end of the report. I took the standard lena image for comparing results.

Original Image and Noisy Image with PSNR = 22dB

Noisy image was formed by adding a white gaussian noise with standard deviation 0.45 times the standard deviation of original image.


Local Wiener Filtering (PSNR = 25.54dB)


Adaptive Local Wiener Filtering (PSNR = 25.68dB)


Local Wavelet Wiener Filtering (PSNR = 25.9525dB)


Adaptive Local Wavelet Wiener Filtering (PSNR = 25.9537 dB)


Combined Result(PSNR = 26.52dB)

(Please visit the code link to view images in high resolution if differences are not properly visible in printed version)

It can be seen that adaptive filter works well in both spatial and wavelet domains. The artifacts appearing due to wavelet denoising can be solved through combined approach.

References

1. Adaptive Wiener filtering of noisy images and image sequences, F. Jin ; P. Fieguth ; L.Winger ; E. Jernigan https://ieeexplore.ieee.org/document/1247253
2. Digital Image Processing, R.Gonzalez and R.Woods https://www.amazon.com/Digital-Image-Processing-Rafael-Gonzalez/dp/013168728X
3. Numpy reference http://www.numpy.org/
4. Digital image enhancement and noise filtering by use of local statistics https://ieeexplore.ieee.org/document/4766994
5. Spatially adaptive wavelet thresholding with context modeling for image denoising https://ieeexplore.ieee.org/document/862630

Code Link

All the results and code implementation can be found at:

https://github.com/darshan3/adaptive-wiener-filter/blob/master/Estimation.ipynb