Ukrainian Catholic University

Faculty of Applied Sciences

Business Analytics & Computer Science Programmes

Frequency domain of an image and its applications

Linear Algebra second interim report

Authors:
Yuliia Moliashcha
Anastasiia Martsinkovska
Bohdan Hashchuk

10 April 2024

Abstract

Image denoising is a component of digital image processing, aiming to enhance the visual quality of images by reducing noise originating from various sources such as environmental conditions or camera sensor issues. The primary aim of our project is to research different denoising methods primary Fourier-based to denose images while preserving and restoring important information from noisy input.

1 Introduction

The research area of our project lies within the domain of image processing. During image acquisition and transmission susceptibility to noise corruption in digital images arises[3, 5]. Noise appears as undesirable grainy areas, leading to a degradation of image quality and loss of important information. Hence, the task of image denoising emerges, with the primary objective being the elimination of noise from an image, thus introducing a new challenge: preserving critical information such as edges and textures while preventing artifacts and blurring[2]. The aim of our project is to compare the efficacy of Fourier-based denoising methods with other techniques.

2 Spatial and frequency domain of an image

Images can be represented in two basis domains: spatial and frequency.

The spatial domain represents an image in terms of its pixel values, where each pixel corresponds to a specific location and intensity value in the image.

In contrast, the frequency domain represents an image in terms of its frequency components, a transformation accomplished through the Fourier Transform.

Figure 1: Image in Spatial Domain

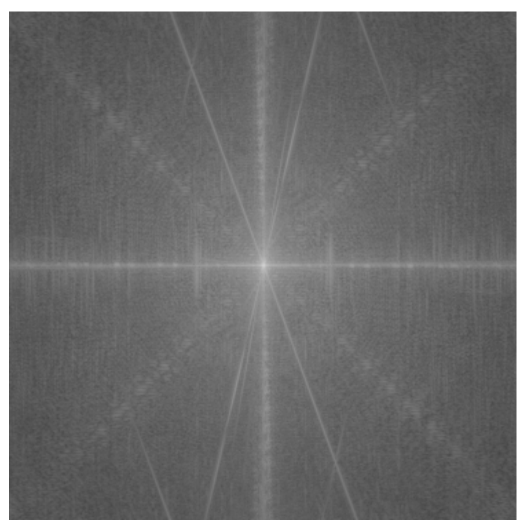


Figure 2: Image in Frequency Domain

In spatial domain an image is commonly represented as a two-dimensional function, denoted as $f(x, y)$, where x and y represent spatial coordinates within the image. The value of f at any given pair of coordinates (x, y) corresponds to the intensity of the image at that point [3].

An image $f(x, y)$ can be represented as a two-dimensional matrix, denoted as $\mathbf{F}$, where each element of the matrix corresponds to the intensity value of the image at a specific spatial coordinate (x, y). Let M and N represent the dimensions of the image in the x and y directions, respectively. Then, the matrix $\mathbf{F}$ can be expressed as:

$$\mathbf{F} = \begin{bmatrix} f(1,1) & f(1,2) & \cdots & f(1,N) \\ f(2,1) & f(2,2) & \cdots & f(2,N) \\ \vdots & \vdots & \ddots & \vdots \\ f(M,1) & f(M,2) & \cdots & f(M,N) \end{bmatrix} \tag{1}$$

3 What is noise?

In practical scenarios images are often corrupted by various sources of noise, degrading the quality of the captured data. One common type of noise, known as additive noise, introduces random fluctuations in pixel values, leading to undesired artifacts in the image. Gaussian noise, commonly known as normal noise, is the most prevalent type of additive noise in the context of natural images.[1]

The resulting degraded image $\hat{I}$ can be described as the sum of the true pixel values I and an additive noise component N:

$$\hat{I} = I + N \tag{2}$$

Each element of the matrices I, $\hat{I}$, and N corresponds to the intensity value of a pixel in the respective image. The equation (2) captures the element-wise addition operation between the true pixel values I and the noise N, resulting in the formation of the degraded image $\hat{I}$.

The matrix form of Gaussian noise can be defined as:

$$N = \begin{bmatrix} n_{11} & n_{12} & \cdots & n_{1N} \\ n_{21} & n_{22} & \cdots & n_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ n_{M1} & n_{M2} & \cdots & n_{MN} \end{bmatrix} \tag{3}$$

The probability distribution function of Gaussian noise is given by:

$$\eta(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{4}$$

where μ denotes the mean and σ^2 denotes the variance.

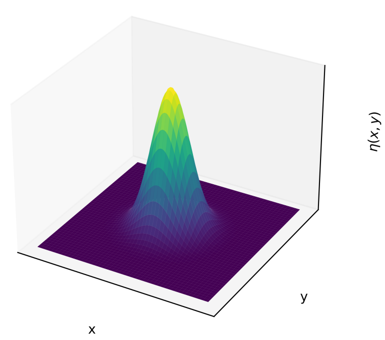


Figure 3: $\mu = 0$, $\sigma^2 = 1$

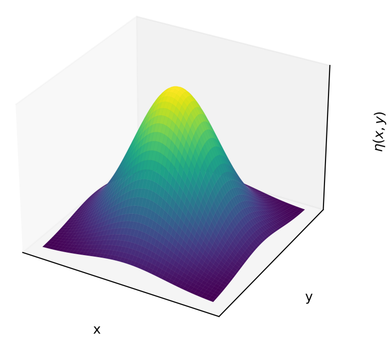


Figure 4: $\mu = 0$, $\sigma^2 = 5$

4 Fourier series and transform

Joseph Fourier claimed that any function of a variable can be expressed as a sum of sines and cosines of various amplitudes and frequencies.

These sine and cosine waves form a complete basis set that spans all possible functions.

4.1 Discrete Fourier Transform

Discrete Fourier Transform is employed for discrete and finite signals, as commonly encountered in image processing.

Let's define an input vector of N dimensions:

$$\vec{x} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_{N-1} \end{bmatrix} \qquad x[n] = x_n \tag{5}$$

The Discrete Fourier Transform converts $\vec{x}$ from the spatial domain to its corresponding frequency domain representation $\vec{X}$.

The output vector is denoted as $\vec{X}$, and representing frequency components:

$$\vec{X} = \begin{bmatrix} X_0 \\ X_1 \\ X_2 \\ \vdots \\ X_{K-1} \end{bmatrix} \qquad X[k] = X_k \tag{6}$$

The Discrete Fourier Transform:

$$X_k = \sum_{n=0}^{N-1} x_n \, e^{-i2\pi \frac{k}{N} n} \tag{7}$$

And the Inverse Discrete Fourier Transform:

$$x_n = \frac{1}{N}\sum_{k=0}^{N-1} X_k \; e^{i2\pi \frac{k}{N}n} \tag{8}$$

Lets define coefficients of Fourier Transform as

$$\omega_N = e^{\frac{i2\pi}{N}} \tag{9}$$

Employing Euler's formula, we can express $e^{-i2\pi ux}$ as a decomposition into functions which form an orthogonal basis:

$$e^{\frac{i2\pi}{N}} = \cos\left(\frac{2\pi}{N}\right) + i\sin\left(\frac{2\pi}{N}\right) \tag{10}$$

Now we can form a Fourier basis:

$$f_k = \begin{bmatrix} 1 \\ \omega_N^k \\ \omega_N^{2k} \\ \vdots \\ \omega_N^{(N-1)k} \end{bmatrix} \tag{11}$$

The following basis is orthogonal as

$$\langle f_k, f_\ell \rangle = \begin{cases} N & \text{if } k = \ell \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

To define the Fourier matrix, F, using the given coefficients, we can represent it as:

$$F = \frac{1}{\sqrt{N}} \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \omega_N & \omega_N^2 & \dots & \omega_N^{N-1} \\ 1 & \omega_N^2 & \omega_N^4 & \dots & \omega_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega_N^{N-1} & \omega_N^{2(N-1)} & \dots & \omega_N^{(N-1)^2} \end{pmatrix} \tag{13}$$

The Discrete Fourier Transform can be expressed as:

$$\mathbf{X} = \frac{1}{\sqrt{N}} F x \tag{14}$$

And the Inverse Discrete Fourier Transform:

$$x = \frac{1}{\sqrt{N}} F^T \mathbf{X} \tag{15}$$

where x is the input data (5), X is vector with corresponding frequency components (6), F is the Fourier matrix (13)

F matrix is unitary matrix $\Rightarrow F^H F = I$, therefore the Inverse Discrete Fourier Transform is the Hermitian transpose of the Fourier matrix.

$$x = F * \mathbf{X}_u \tag{16}$$

Further, the Discrete Fourier Transform can be extended to any number of dimensions.

To compute the two-dimensional Discrete Fourier Transform of an $m \times n$ matrix, the process involves two sequential steps. Initially, the DFT of each of the n columns of the matrix is computed. Subsequently, the DFT is applied to each of the m resulting rows, forming the final transformed matrix. The computation of the two-dimensional Inverse Discrete Fourier Transform follows a similar sequential procedure, albeit in reverse order.

5 Overview of approaches

Two primary approaches for image denoising are spatial and transform domain filtering methods [5].

5.1 Spatial domain filtering methods

Spatial filtering involves the direct manipulation of pixel values within the image and encompasses both linear and non-linear techniques.

Linear filters, exemplified by Mean filtering [3], are implemented through convolution operations. Each pixel in an image undergoes processing via a filter window, wherein the central value is replaced by the average mean of all pixels within the window. Gaussian filter

$$\mathrm{GC}[I_p] = \sum_{q \in S} G_\sigma(||p - q||) I_q \tag{17}$$

In contrast, non-linear filters, such as Median filtering, replace each pixel value with a non-linear function of its surrounding pixels. The Median filter involves the substitution of the central pixel of the window with the middle value of the window.

$$f(x, y) = \mathrm{median}\{g(s, t)\} \mid (s, t) \in S_{xy} \tag{18}$$

where $f(x, y)$ denotes the pixel value at location (x, y) in the denoised image, $g(s, t)$ the pixel value at location (s, t) in the noisy image, S_{xy} denotes a square-shaped window centered at pixel location (x, y)

Spatial filters, despite their efficacy in reducing noise by employing low pass filtering techniques, often introduce a trade-off: while they effectively reduce noise, they tend to induce edge corruption and image blurring [2, 4].

5.2 Transform domain filtering methods

In transform domain filtering, we consider frequency domain. Image denoising in the frequency domain is the process of transforming an image from the spatial domain to the frequency domain, which involves linear transformation from the Euclidean basis to the Fourier basis, and designing a frequency domain filter, by which we can target and filter out high-frequency values, commonly associated with image noise [5].

Alternatively, in the frequency domain, Equation (2) can be represented as:

$$\mathbf{G} = \mathbf{F} + \mathbf{N} \tag{19}$$

where $\mathbf{G}$ denotes the Fourier transform of the observed image, $\mathbf{F}$ the Fourier transform of the original image, and $\mathbf{N}$ the Fourier transform of independent Gaussian noise with zero mean and variance σ^2.

In the frequency domain, a filter is applied by performing multiplication of a filter and a noisy image. This multiplication is based on the Convolution Theorem.

5.2.1 Convolution Theorem

The Convolution Theorem provides a link between operations in the spatial domain and their counterparts in the frequency domain.

Let $\vec{f}$ and $\vec{g}$ be two vectors of length N, representing discrete signals. The convolution operation, denoted $f * g$, yields a new vector $\vec{h}$ of length N, where each element of $\vec{h}$ is the result of convolving $\vec{f}$ with a shifted version of $\vec{g}$.

$$(f * g)_k = \sum_{j=0}^{N-1} f_{k-j} \cdot g_j \tag{20}$$

The Convolution Theorem states that convolution in the spatial domain is equivalent to element-wise multiplication in the frequency domain, and vice versa.

$$F_n(f * g) = n(F_n f) \cdot (F_n g) \tag{21}$$

The Convolution Theorem in its inverse form, where F_n denotes the Fourier transform operator and $*$ represents the convolution operation, can be represented as:

$$f * g = nF_n^{-1}\left((F_n f) \cdot (F_n g)\right) \tag{22}$$

5.2.2 Overview of low-pass filters in frequency domain

The reduction of noise is achieved by applying a low-pass filter, H(u, v). This filter attenuates all frequencies above a specific cut-off frequency.

$$H(u, v) = \begin{cases} 1 & \text{if } D(u, v) \leq D_0 \\ 0 & \text{if } D(u, v) > D_0 \end{cases} \tag{23}$$

A Butterworth low-pass filter [**?**]

$$H(u, v) = \frac{1}{1 + \left(\frac{D(u,v)}{D_0}\right)^{2n}} \tag{24}$$

where n is an order, D0 as the cut-off distance from the origin
Gaussian low-pass filter

$$H(u, v) = e^{-\frac{D^2(u,v)}{2D_0^2}} \tag{25}$$

6 Implementation Pipeline

The denoising process proceeds as follows:

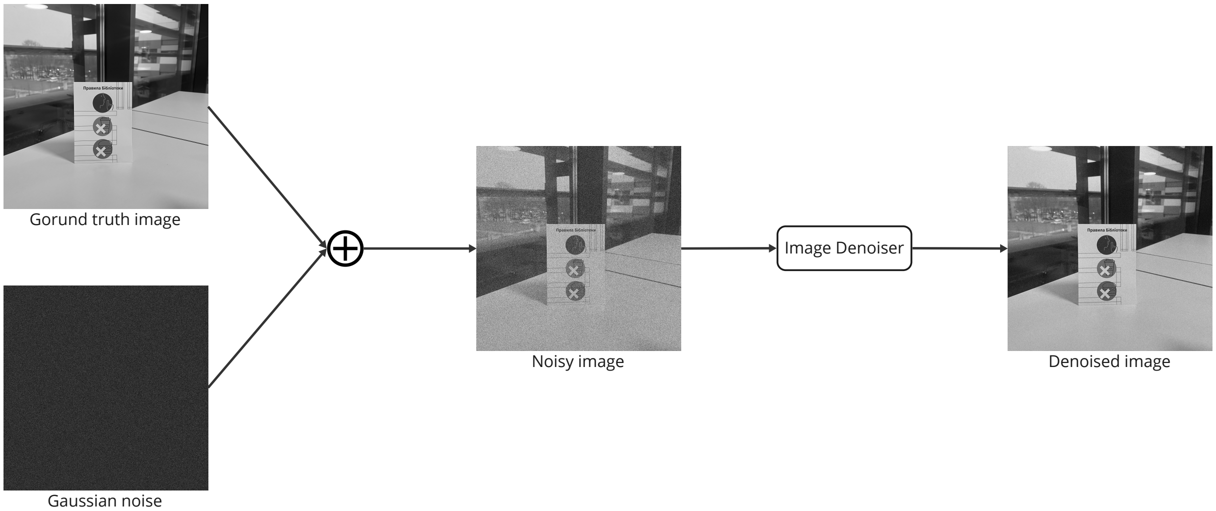


Figure 5: Denoising concept

Initially, the image is transformed from its native Euclidean basis to the Fourier basis via Fourier transformation. This conversion enables the representation of the image in terms of frequency components. Following the transformation, we can observe the image in the form of frequency with high and low frequency components, where high-frequency components are typically indicative of noise present within the image. The next step is to reduce this noise by filtering out the high-frequency components. After filtering out the high-value frequencies, the next step is to perform an inverse Fourier transformation to convert the image back to its spatial domain. This step restores the image to its original form but with reduced noise, resulting in a denoised representation.

However, it's essential to strike a balance between noise reduction and preservation of image details. Over-filtering can lead to loss of important information and result in a blurred image. Therefore careful adjustment of the filter parameters is required.

7 Testing

To test our implementation, we use the CIFAR-10 dataset with Gaussian noise added to these images to simulate real-world conditions. For evaluation purposes, we devise a metric to compare the denoised images with the ground truth images. This metric includes measures such as mean squared error (MSE) and peak signal-to-noise ratio (PSNR).

7.1 Mean Square Error

The Mean Squared Error (MSE), quantifies the cumulative square error between the denoised $\hat{I}$ and ground truth image I, is defined by:

$$MSE(I, \hat{I}) = \frac{1}{N} \|I - \hat{I}\|_F^2 \tag{26}$$

where $\|\cdot\|_F$ denotes the Frobenius norm.

7.2 Peak Signal-to-Noise Ratio

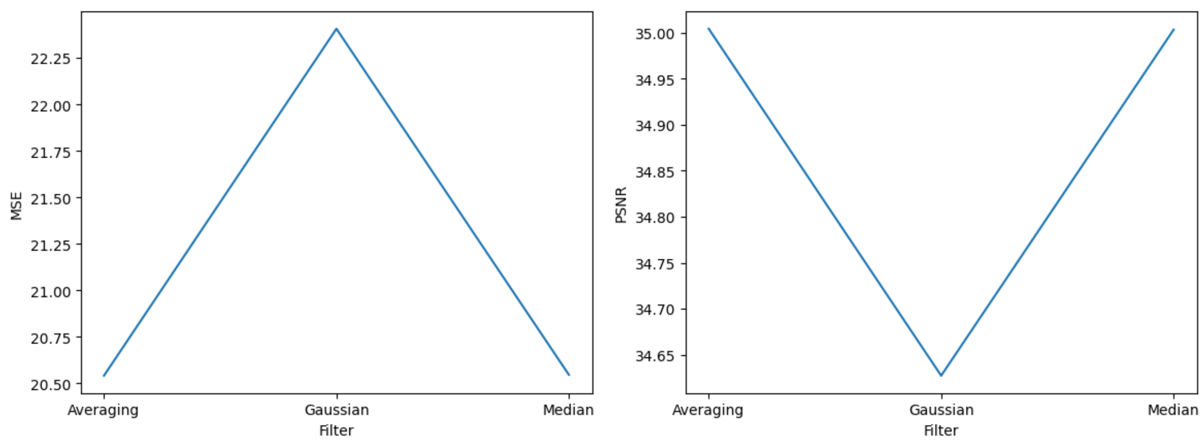
Peak Signal-to-Noise Ratio (PSNR), a ratio of signal power to noise power, defined by:

$$\mathrm{PSNR} = 10 \cdot \log_{10}\left(\frac{255^2}{\mathrm{MSE}}\right) \tag{27}$$

8 Challenges and Potential Limitations

One significant challenge we anticipate is striking a balance between image denoising and preserving valuable image information. Following the Fourier transform, the resulting high frequencies, typically associated with noise, can also encode essential image features such as edges and textures. The challenge lies in accurately distinguishing between high frequencies representing noise and those embodying meaningful image information.

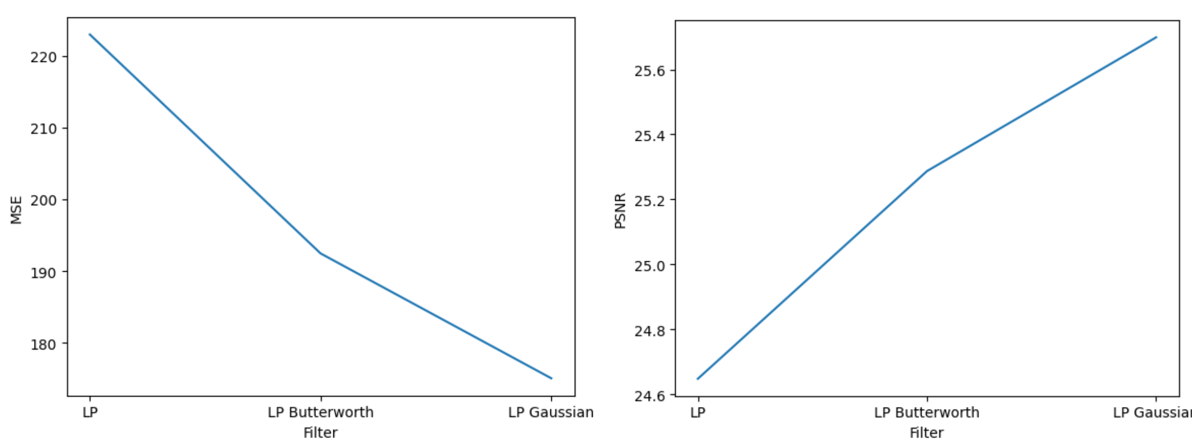
9 Benchmarks



(a) MSE, Spatial Filters

(b) PSNR, Spatial Filters

Figure 6: MSE & PSNR, Spatial Filters



(a) MSE, Frequency Filters

(b) PSNR, Frequency Filters

Figure 7: MSE & PSNR, Frequency Filters

10 Conclusions

In conclusion, we have conducted thorough research into the problem of image noise, explored various denoising algorithms, and their theoretical background with the stress on linear algebra methods. We compared spatial filters (mean , median and Gaussian) with frequency domain filters (low-pass, Butterworth, and low-pass Gaussian). Our analysis revealed that the low-pass Gaussian filter emerges as the most effective choice for a broad range of denoising scenarios.

References

[1] Alan C. Bovik. *The Essential Guide to Image Processing*. Academic Press, Inc., USA, 2009.

[2] Linwei Fan, Fan Zhang, Hui Fan, and Caiming Zhang. Brief review of image denoising techniques, 12 2019.

[3] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Pearson Education, 2018.

[4] Bhawna Goyal, Ayush Dogra, Sunil Agrawal, B.S. Sohi, and Apoorav Sharma. Image denoising review: From classical to state-of-the-art approaches, 2020.

[5] Mukesh Motwani, Mukesh Gadiya, Rakhi Motwani, and Frederick Harris. Survey of image denoising techniques, 01 2004.