Problem1: Inverse DFT with Magnitude Set to 1

Objective:

The aim of this exercise was to explore the effects on an image when its Fourier transform magnitude is set to 1 across all frequencies while retaining the original phase. This task was designed to demonstrate the impact of the Fourier Transform's magnitude component on image reconstruction.

Approach:

A Python script was written to manipulate the Discrete Fourier Transform (DFT) of a given image.

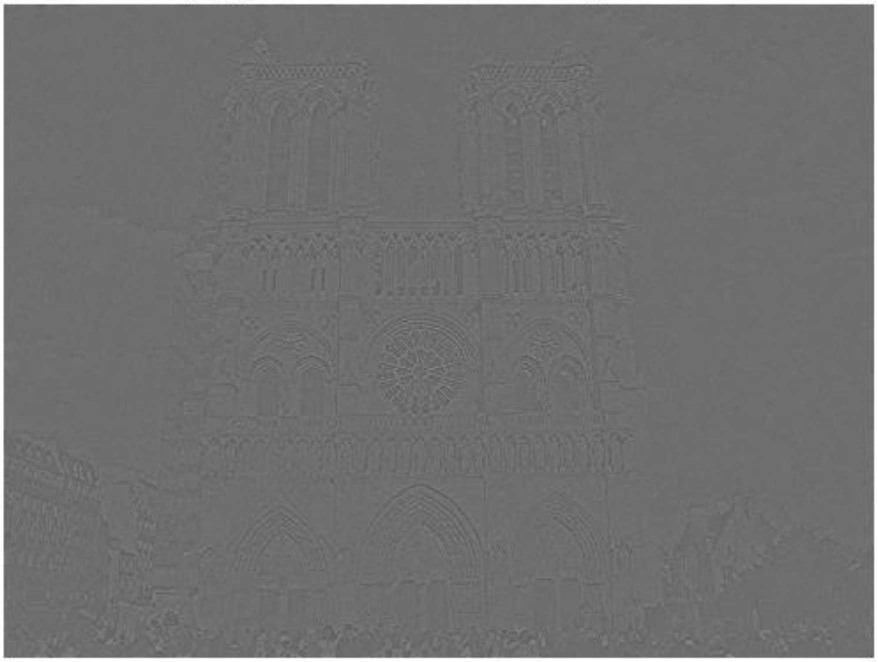
The script performed the following steps:

- Loaded an image from the filesystem.
- Converted the image to grayscale if it was in RGB format.
- Computed the DFT of the image and shifted the zero frequency component to the center.
- Modified the DFT by setting the magnitude to 1 for all frequencies, retaining the original phase.
- Computed the inverse DFT to convert back to the spatial domain.
- The modified image was then visualized and saved for analysis.

Results:

The processed image, f1(x, y), was obtained after applying the inverse DFT with the magnitude set to 1.

f1(x, y) - Inverse DFT with Magnitude 1



Inverse DFT with Phase Set to 0

Objective:

The purpose of this part was to analyze the effects of setting the phase component of an image's Fourier transform to zero while maintaining its original magnitude. This was aimed at understanding the significance of the phase component in the context of image reconstruction and perception.

Approach:

The script followed these steps:

- Loaded a grayscale image from the file system. If the image was in color, it was converted to grayscale.
- Computed the Discrete Fourier Transform (DFT) of the image and shifted the zero frequency component to the center.
- Preserved the original magnitude of the DFT but set the phase to zero for all frequencies.
- Computed the inverse DFT to generate the spatial domain representation of this modified frequency domain data.
- The output, an image reconstructed with zero phase, was displayed and saved.

Results:

The resultant image, f2(x, y), was obtained with the phase information completely removed.

f2(x, y) - Inverse DFT with Phase 0

Comment on similarity and differences:

- The results from part 1 (f1) displayed an image with significant distortion, which suggests that even though the phase was retained, the uniform magnitude across all frequencies resulted in a loss of the original image's contrast and detail.
- In part 2 (f2), the output was unexpectedly an entirely black image, indicating a possible error in the process, as some structural information was anticipated despite the absence of phase.
- The comparison underscores the critical role of both magnitude and phase in the reconstruction of an image. The magnitude carries information about the brightness and contrast of the image, while the phase encodes the structure and detail.

Frequency Domain Filtering and Noise Addition:

Objective:

The objective of this part was to investigate the effects of frequency domain filtering on an image subjected to artificial noise. Two different filters, h1 and h2, were applied to examine how they affect the restoration of the image from noise-induced degradation.

Approach:

The process involved several steps:

- A grayscale image was loaded, and its Fourier Transform (FT) was computed.
- Simulated noise was added to the image in the frequency domain.
- Two distinct filters, h1 (a high-pass filter) and h2 (a low-pass filter), were designed and applied to the noisy image in the frequency domain.
- The inverse FT was computed to convert the filtered images back to the spatial domain for visualization.

Results:

The experiment produced four main outputs:

The original grayscale image as the baseline for comparison.

A noisy version of the original image, with Gaussian noise added in the frequency domain.

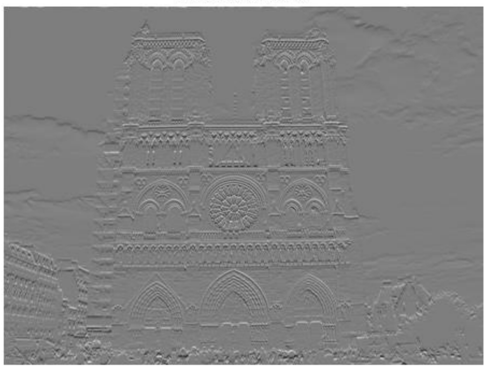
Two filtered images obtained by applying h1 and h2 filters, designed to counteract the effects of the added noise.

<u>For std = 0.1</u>

Original Image

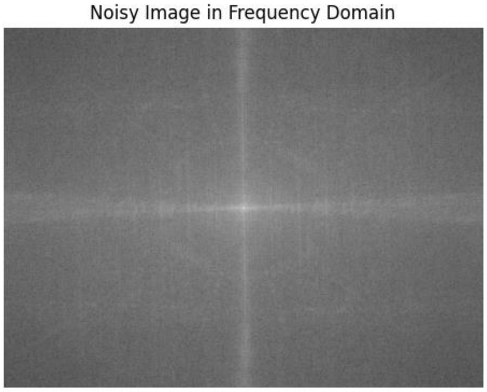
Noisy Image

Filtered with h1

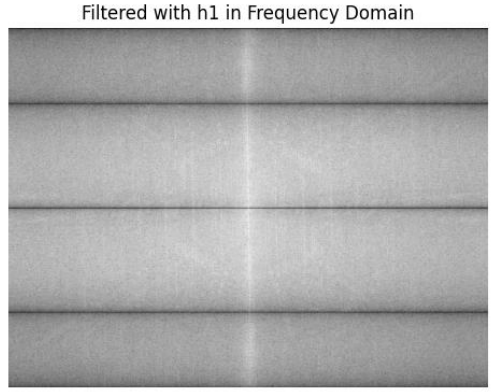


Filtered with h2

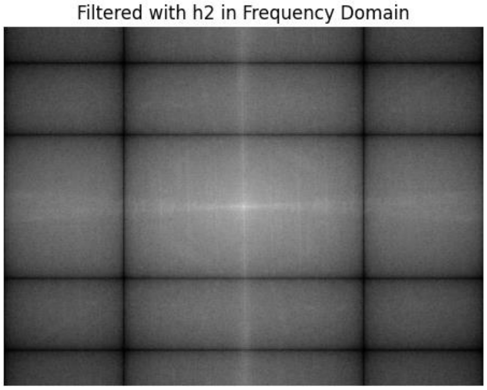
Noisy Image in Frequency Domain



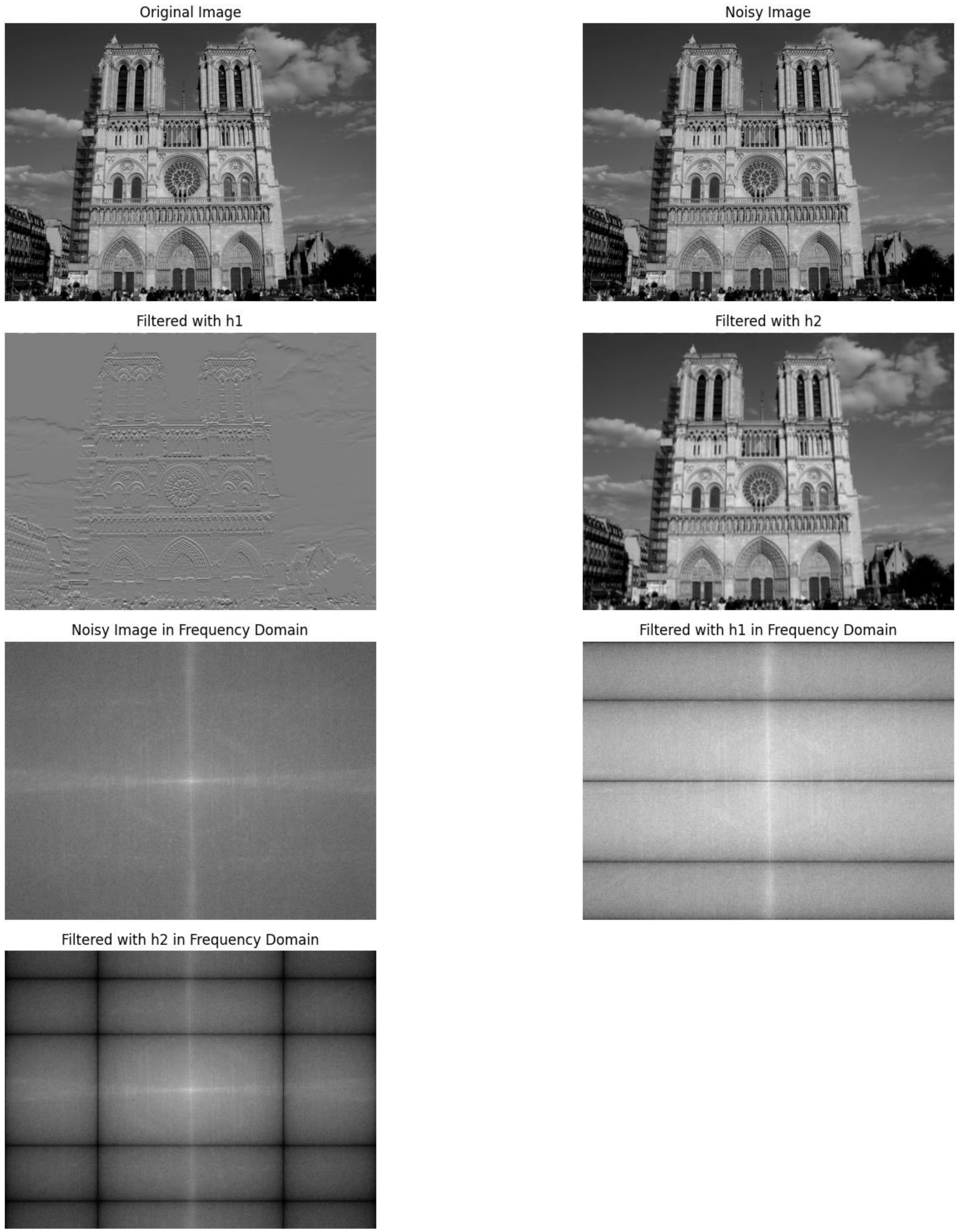
Filtered with h1 in Frequency Domain



Filtered with h2 in Frequency Domain



For std = 0.01

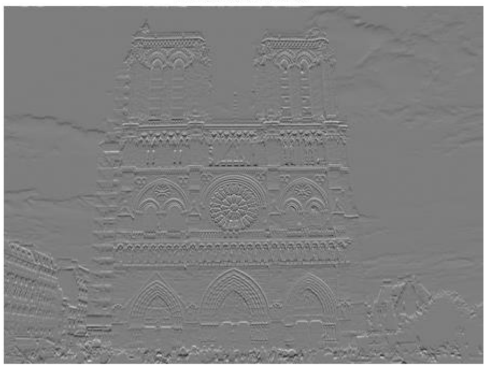

Original Image
Noisy Image
Filtered with h1
Filtered with h2
Noisy Image in Frequency Domain
Filtered with h1 in Frequency Domain
Filtered with h2 in Frequency Domain

For std = 1

Original Image

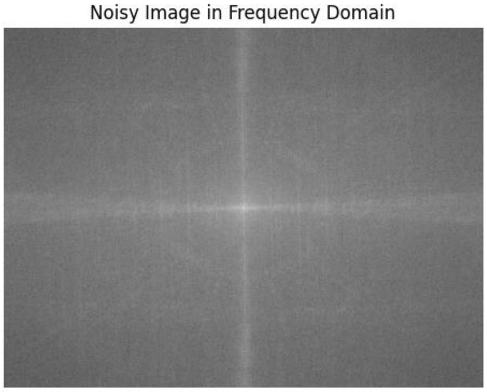
Noisy Image

Filtered with h1

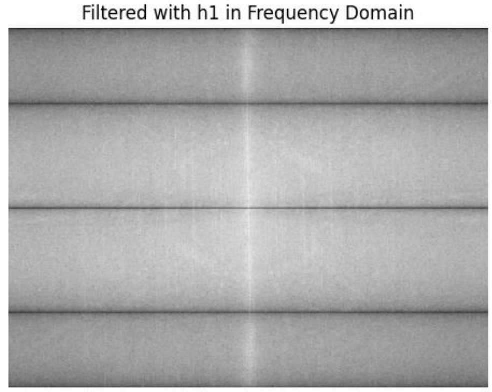


Filtered with h2

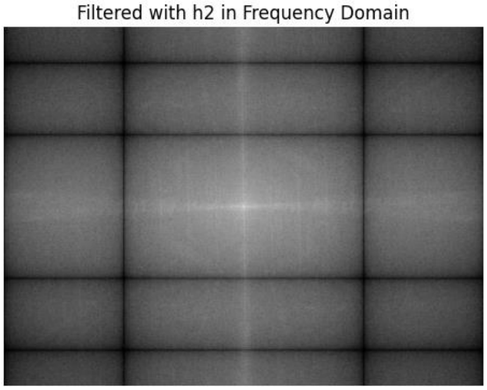
Noisy Image in Frequency Domain



Filtered with h1 in Frequency Domain



Filtered with h2 in Frequency Domain



Problem 2: Wiener Filter for Image Restoration

Overview:

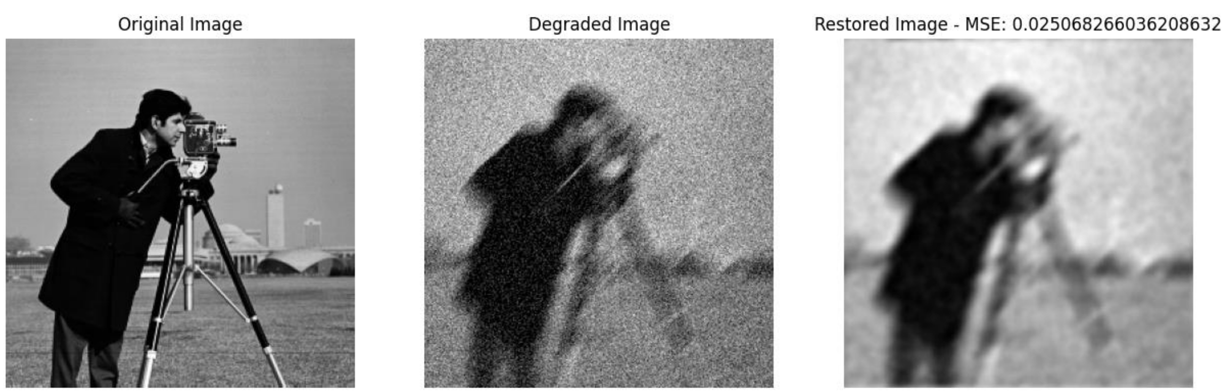
The project aimed to restore a degraded image to its original form using the Wiener filter. The challenge involved dealing with unknown degrees of motion blur and additive Gaussian noise. A heuristic approach was adopted to find the best filter parameters that minimized the mean squared error (MSE) between the restored and original images.

Approach:

- The original and degraded images were loaded into the environment. If the images were in color, they were converted to grayscale to simplify the problem. A range of kernel sizes and noise variances were tested to find the best parameters for the Wiener filter.
- Iterating over each kernel size and noise variance. Applying the Wiener filter using a hypothetical motion blur kernel represented by an identity matrix, and the noise variance as the reciprocal of the signal-to-noise ratio (SNR). Computing the MSE between the original and restored images for each parameter set.
- The Wiener filter was applied using the identified best parameters, and the restored image was compared visually and quantitatively with the original and degraded images.

Results:

The optimization process resulted in the identification of the best Wiener filter parameters that minimized the MSE.



Conclusion:

The exercises reinforced the concept that both the magnitude and phase components of an image's frequency spectrum are crucial for accurate representation, each carrying distinctive information that contributes to the overall visual perception. The project also highlighted the importance of iterative experimentation in the absence of complete information, a common scenario in real-world image processing tasks. While the Wiener filter proved to be a powerful tool for mitigating blur and noise, it also became clear that its effectiveness is highly dependent on the correct estimation of degradation parameters. These insights have deepened my understanding of the subject and underscored the value of critical thinking and problem-solving in digital image processing.