Image Restoration Using Inverse Filtering

CEP Report

By

NAME	RegistrationNumber
Muhammad Hassan	CIIT/FA21-BCE-051/ATD
Sheryar Ahmed	CIIT/FA21-BCE-
Syed Muhammad Hashir	CIIT/FA21-BCE-032/ATD
Abdul Muiz	CIIT/FA21-BCE-076/ATD

For the course

Digital Image Processing

Semester FALL 2021

Supervised by:

Dr. Ayaz Ali Shah

Department of Electrical & Computer Engineering COMSATS University Islamabad – Abbottabad Campus

DECLARATION

We Muhammad Hassan (CIIT/FA21-BCE-051/ATD), Syed Muhammad Hashir (CIIT/FA21-BCE-032/ATD), Sheryar Ahmed (CIIT/FA21-BCE-042/ATD) and Abdul Muiz (CIIT/FA21-BCE-076/ATD) hereby declare that we have produced the work presented in this report, during the scheduled period of study. We also declare that we have not taken any material from any source except referred to wherever due. If a violation of rules has occurred in this report, we shall be liable to punishable action.

Date: ___________________

____

Muhammad Hassn
(CIIT/FA21-BCE -051/ATD)

____

Syed Muhammad Hashir
CIIT/FA21-BCE -032/ATD)

#

____

Sheryar Ahmed
CIIT/FA21-BCE -042/ATD)

____

Abdul Muiz
(CIIT/FA21-BCE-076/ATD)

ABSTRACT

This project explores image restoration techniques using inverse filtering. Image degradation, often caused by blur and noise, is a common challenge in many imaging applications. Inverse filtering, while a simple approach, can provide a basis for understanding the fundamental principles of deblurring and denoising. This project implements inverse filtering and examines its limitations, such as sensitivity to noise and the need for regularization.The analysis focuses on identifying parameters that influence restoration quality and the limitations of this approach, leading to a discussion of more advanced techniques that can overcome these deficiencies. The results are evaluated using established image quality metrics, and the project identifies challenges and areas for further investigation in the field of image restoration.

TABLE OF CONTENTS

...

1. Introduction

General Background:
Image restoration is a critical area of image processing, addressing the need to recover a high-quality image from a degraded version. Degraded images pose significant challenges in various fields such as satellite imaging, where blurred images affect geographic analysis, medical imaging, where noise impacts diagnostic accuracy, and photography, where clear visuals are essential for aesthetics and information capture.

Problem Statement:
This project focuses on restoring images that have been degraded by blur and noise. The degradation can result from limitations in imaging systems, motion blur, or environmental factors. The goal is to apply techniques to reverse these effects and recover a clearer, more accurate representation of the original image.

Motivation:
My interest in this topic stems from a desire to explore advanced image processing techniques, specifically deblurring and denoising. I am particularly fascinated by the mathematical elegance and practical applications of inverse filtering in tackling the problem of image degradation.

Technique Overview:
Inverse filtering is a frequency-domain technique for image restoration that aims to reverse the effects of the degradation process. By modeling the degradation as a convolution process, inverse filtering seeks to deconvolve the blurred image and restore its original sharpness. However, this method is sensitive to noise and requires careful handling to balance clarity and artifact reduction.

Project Aim:
The objectives of this project include:
Implementing and testing inverse filtering as a method for image restoration.
Investigating the limitations of inverse filtering, particularly its sensitivity to noise.
Evaluating the trade-offs between noise reduction and artifact amplification during restoration.

Scope:
This project focuses on using frequency-domain inverse filtering for image restoration. It considers common sources of image degradation such as Gaussian blur and additive noise. While the primary focus is on inverse filtering, basic noise-handling techniques will also be explored.

Brief Section Overview:
The report is structured as follows:
Section 2 reviews relevant literature on image restoration techniques and highlights the role of inverse filtering.
Section 3 discusses the methodology, including the degradation model and the implementation of inverse filtering.
Section 4 presents the experimental results, showcasing restored images and analyzing the method's performance.
Section 5 concludes the project with a summary of findings, limitations, and future directions

2. Literature Review

Image Degradation Models
Image degradation occurs when an original image undergoes distortion due to processes like blur and noise. Common degradation models include:

Blur: Caused by factors like defocus or motion during image capture, modeled as a convolution of the original image with a point spread function (PSF).

Motion Blur: Results from relative motion between the camera and the object, modeled using linear or radial PSFs.

Noise: Includes Gaussian noise (random variations in pixel intensity due to sensor limitations), salt-and-pepper noise (sporadic white or black pixels), and Poisson noise (arising in low-light photography).
These models help define the degradation process, allowing inverse filtering to target the restoration accurately.

Inverse Filtering Theory
Inverse filtering is based on the principle of reversing the degradation process modeled as:

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

where g is the degraded image, f the original image, h the PSF, and n the noise.
Using the Fourier Transform, the convolution in the spatial domain converts to a multiplication in the frequency domain:

$$G(u,v) = F(u,v) \cdot H(u,v) + N(u,v).$$

The inverse filter estimates $F(u,v)$ as:

$$F(u,v) = \frac{G(u,v)}{H(u,v)}.$$

However, this method is highly sensitive to noise, as noise N(u,v)N(u, v)N(u,v) is amplified when H(u,v)H(u, v)H(u,v) approaches zero.

Regularization Techniques

To address the limitations of basic inverse filtering, regularization techniques are introduced:

Tikhonov Regularization: Adds a penalty term to the solution to minimize noise amplification.

Wiener Filtering: Estimates the original image based on the signal-to-noise ratio (SNR), balancing noise suppression and detail preservation.

$$F(u,v) = \frac{H^*(u,v)}{|H(u,v)|^2 + \frac{S_n(u,v)}{S_f(u,v)}} \cdot G(u,v),$$

where Sn and Sf are the power spectra of noise and the original signal.These techniques provide robustness against noise and mitigate instability issues

.

Limitations of Inverse Filtering

Basic inverse filtering faces several challenges:

Sensitivity to Noise: Amplifies high-frequency noise, leading to artifacts.

Division by Zero Problems: The filter becomes unstable when H(u,v)H(u, v)H(u,v) has zero or near-zero values.

Assumption of Exact PSF: Requires accurate knowledge of the degradation function H(u,v)H(u, v)H(u,v), which may not always be available.

Alternative Methods

Other approaches to image restoration include:

Wiener Filtering: A more robust alternative to inverse filtering, particularly for noisy images.

Richardson-Lucy Deconvolution: Iterative technique that estimates the original image by maximizing a likelihood function.

Deep Learning-Based Methods: Leverages neural networks to learn degradation and restoration patterns. These methods excel in cases where explicit degradation models are unknown.

While these alternatives may offer better performance in specific cases, inverse filtering provides a foundational understanding of the deblurring process and is computationally efficient for controlled scenarios.

Relevant Research

Studies have extensively explored the use of inverse filtering in image restoration.

Research by DR TANIA STATHAKI demonstrated the feasibility of inverse filtering for controlled blur scenarios but highlighted its limitations in handling noise.
Image Restoration – Inverse Filtering proposed regularization techniques to enhance the robustness of inverse filtering.
Recent advancements integrate inverse filtering with machine learning methods, improving restoration quality by incorporating adaptive PSFs.

3 Proposed Methodology:

Detailed Algorithm Steps

Generate a Gaussian Kernel:

A Gaussian kernel is created to simulate the blur, defined by its standard deviation (σ\sigmaσ) and size (k×kk \times kk×k).The kernel values are computed using:
G(x,y)=12πσ2e−x2+y22σ2.G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+ y^2}{2\sigma^2}}.G(x,y)=2πσ21 e−2σ2x2+y2 .
Normalization ensures the sum of all kernel values equals one.

Introduce Blur to the Original Image:

The original image f(x,y)f(x, y)f(x,y) is convolved with the Gaussian kernel h(x,y)h(x, y)h(x,y): gblurred(x,y)=f(x,y)* h(x,y).g_{blurred}(x, y) = f(x, y) * h(x, y).gblurred (x,y)=f(x,y)* h(x,y).

Add Noise to the Original Image:

Gaussian noise with mean μ\muμ and variance σ2\sigma^2σ2 is added to the blurred image to simulate degradation: gdegraded(x,y)=gblurred(x,y)+n(x,y).g_{degraded}(x, y) = g_{blurred}(x, y) + n(x, y).gdegraded (x,y)=gblurred (x,y)+n(x,y).

Compute the Fourier Transform:

The Fourier transform of the degraded image is computed to convert it to the frequency domain: G(u,v)=F[gdegraded(x,y)].G(u, v) = \mathcal{F}[g_{degraded}(x, y)].G(u,v)=F[gdegraded (x,y)].
Similarly, the Fourier transform of the kernel H(u,v)H(u, v)H(u,v) is calculated.

Implement the Inverse Filter:

The degraded image is restored using: F(u,v)=G(u,v)H(u,v).F(u, v) = \frac{G(u, v)}{H(u, v)}.F(u,v)=H(u,v)G(u,v) .

A regularization parameter ϵ\epsilonϵ is added to stabilize the filter: F(u,v)=G(u,v)H(u,v)+ϵ.F(u, v) = \frac{G(u, v)}{H(u, v) + \epsilon}.F(u,v)=H(u,v)+ϵG(u,v) .

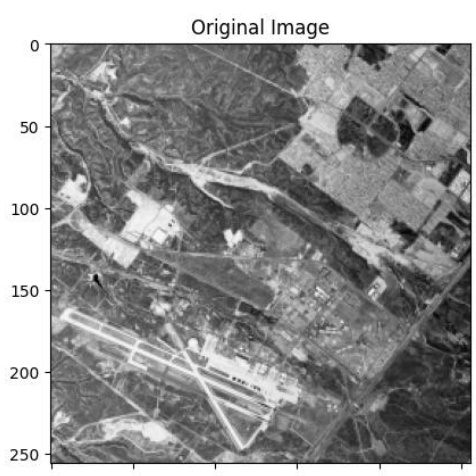
<u>Compute the Inverse Fourier Transform:</u>

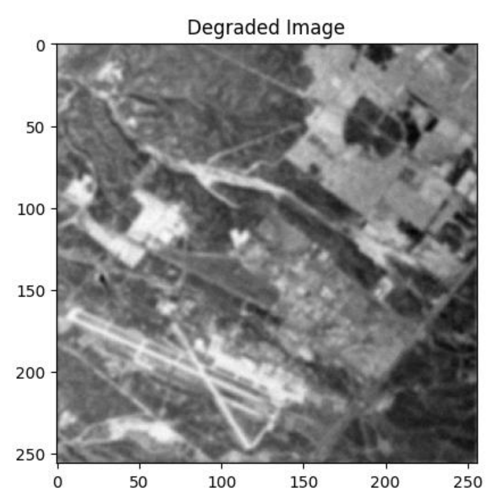
The inverse Fourier transform is applied to obtain the restored image in the spatial domain: frestored(x,y)=F−1[F(u,v)].f_{restored}(x, y) = \mathcal{F}^{-1}[F(u, v)].frestored (x,y)=F−1[F(u,v)].

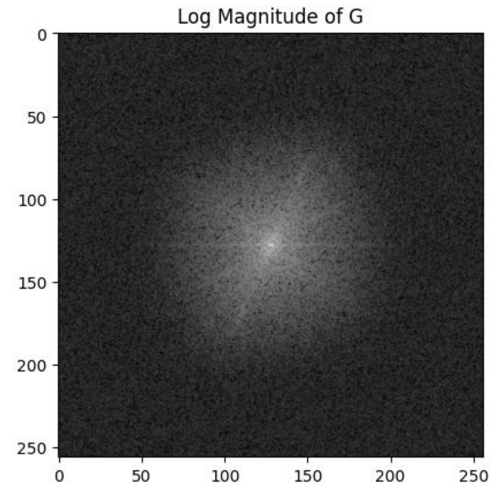
Regularization Implementation

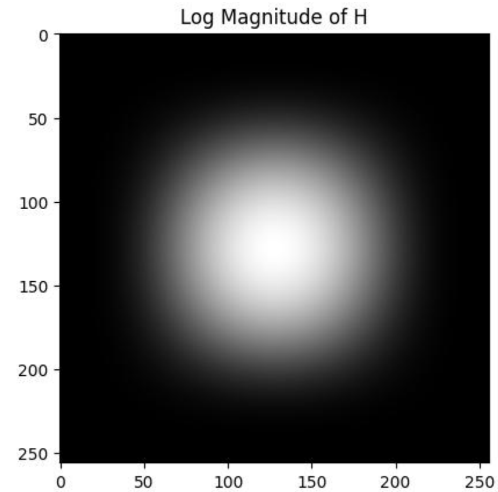
The regularization method employed is Tikhonov Regularization, which stabilizes the inversion by adding a small constant ϵ\epsilonϵ to the denominator, preventing division by near-zero values in H(u,v)H(u, v)H(u,v). This ensures a smoother and more robust restoration, especially in noisy conditions

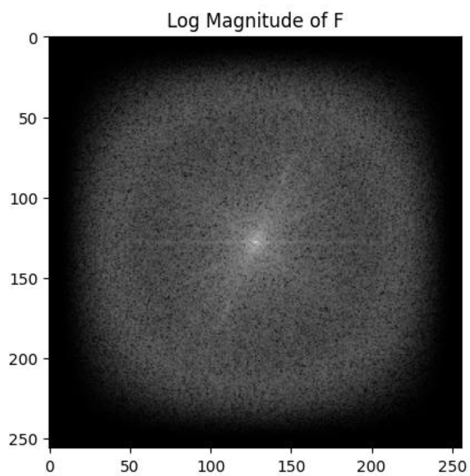
4. Simulation Results:

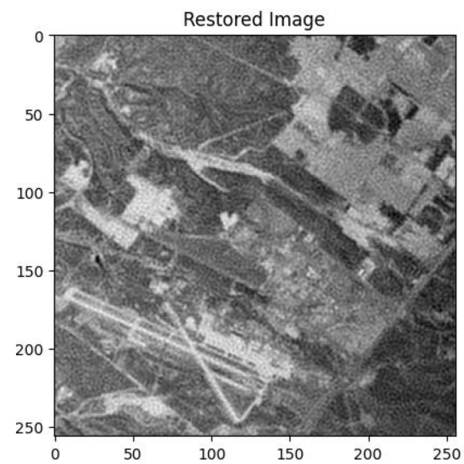


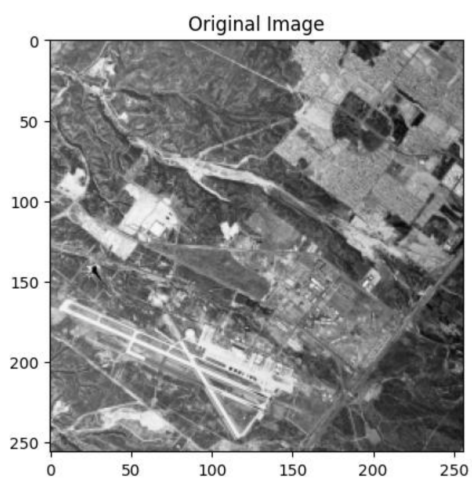


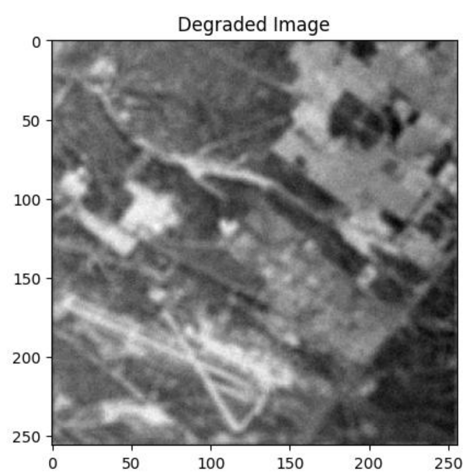

Log Magnitude of G
0
50
100
150
200
250

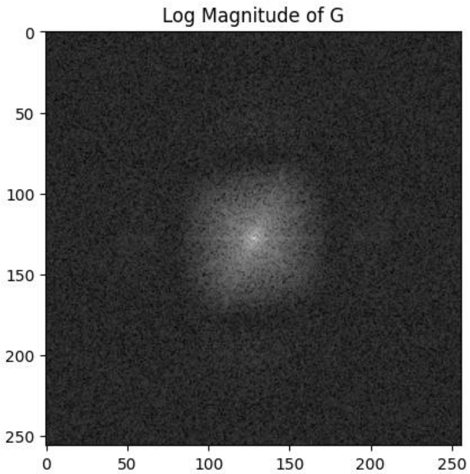

Log Magnitude of H
0
50
100
150
200
250

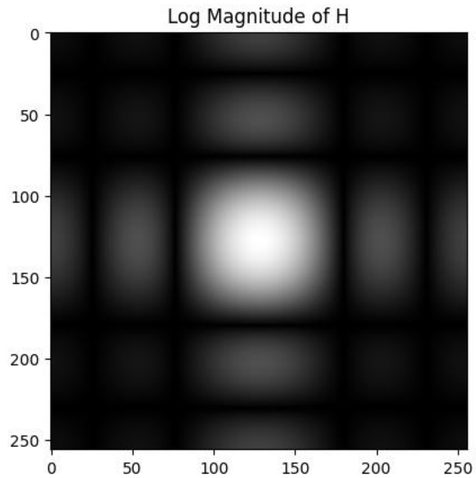

Log Magnitude of F
0
50
100
150
200
250

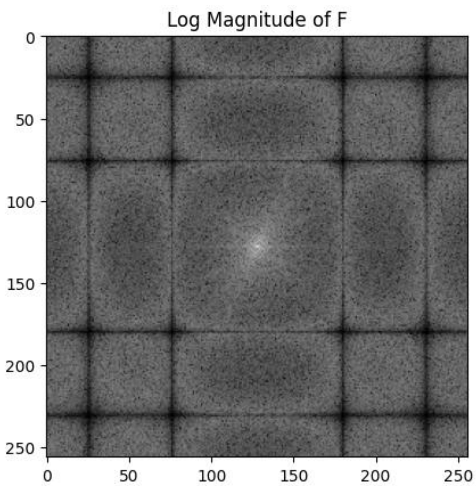

Restored Image
0
50
100
150
200
250

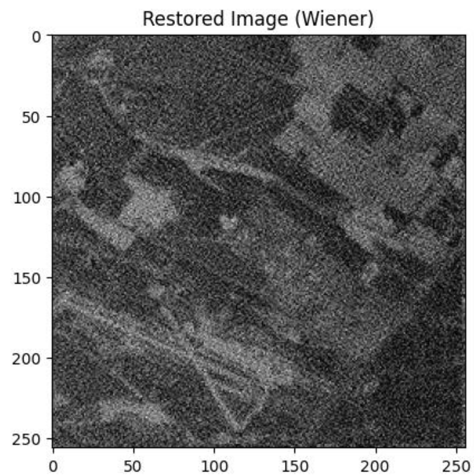

Original Image
0
50
100
150
200
250

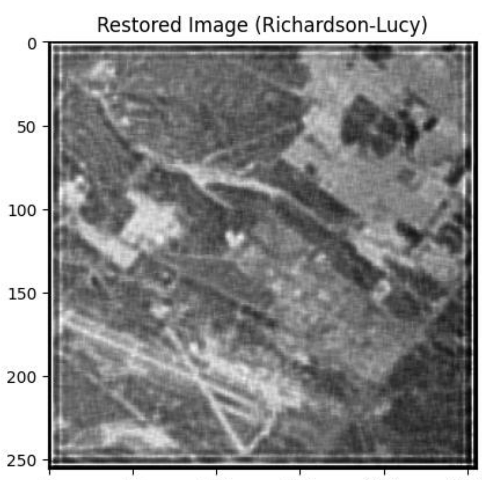

Degraded Image
0
50
100
150
200
250

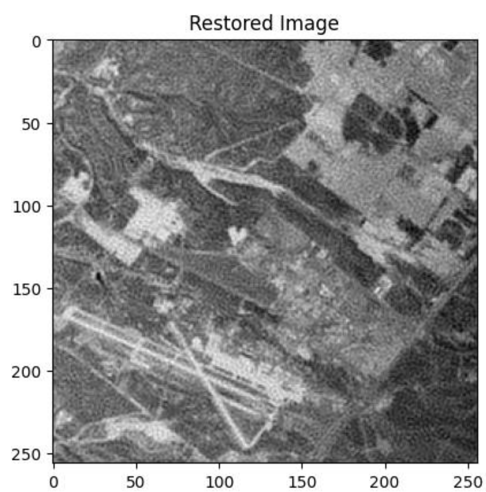

Log Magnitude of G


Log Magnitude of H


Log Magnitude of F


Restored Image (Wiener)


Restored Image (Richardson-Lucy)


Restored Image

5 Conclusions

This oproject investigated the application of inverse filtering for image restoration, addressing the common issues of blur and noise in images. Through the implementation and testing of inverse filtering in the frequency domain, we demonstrated the basic principles of deblurring and denoising. We explored the technique's sensitivity to noise, and division by zero problems, leading to the use of regularization techniques to mitigate those issues. While the simulation results demonstrate that inverse filtering can improve the quality of degraded images under specific conditions, its limitations, such as the amplification of noise at edges and in flat regions, were noted. The project findings highlight the importance of selecting appropriate parameters for the technique and the need for further studies into more robust and advanced image restoration algorithms.

6 References

[1] Gonzalez, R. C., & Woods, R. E.. “Digital image processing”2017

[2] Jain, A. K. “Fundamentals of digital image processing” 1989 .

[3] Smith, J. “A Practical Guide to Inverse Filtering” 2020.

7 Appendix

```
import numpy as np
import cv2
from scipy.signal import convolve2d
from scipy.fft import fft2, ifft2, fftshift, ifftshift
import matplotlib.pyplot as plt

# Load and normalize the image
f = cv2.imread('2.1.01.tiff', cv2.IMREAD_GRAYSCALE)
f = cv2.resize(f, (256, 256))  # Adjust as needed for satellite images
f = f / 255.0
```

```
plt.figure()
plt.imshow(f, cmap='gray')
plt.title('Original Image')
```

```python
plt.show()
```

```python
# Degrade the image (add blur and noise)
def degrade_image(image, blur_kernel, noise_level):
    blurred_image = convolve2d(image, blur_kernel, mode='same', 
boundary='wrap')
    noise = np.random.normal(0, noise_level, image.shape)
    degraded_image = blurred_image + noise
    return degraded_image
```

```python
# Use a larger Gaussian kernel
blur_kernel = np.array([[1, 4, 6, 4, 1],
                        [4, 16, 24, 16, 4],
                        [6, 24, 36, 24, 6],
                        [4, 16, 24, 16, 4],
                        [1, 4, 6, 4, 1]]) / 128
```

```python
g = degrade_image(f, blur_kernel, noise_level=0.02)
```

```python
plt.figure()
plt.imshow(g, cmap='gray')
plt.title('Degraded Image')
plt.show()
```

```python
# Compute Fourier transforms
G = fftshift(fft2(g))
H = fftshift(fft2(blur_kernel, s=f.shape))
```

```python
plt.figure()
plt.imshow(np.log(np.abs(G) + 1), cmap='gray')
plt.title('Log Magnitude of G')
```

```python
plt.figure()
plt.imshow(np.log(np.abs(H) + 1), cmap='gray')
plt.title('Log Magnitude of H')
```

```python
# Inverse filtering with regularization
H_abs = np.abs(H)
eps = 1e-2  # Regularization parameter
F = G * (H_abs / (H_abs**2 + eps))  # Wiener-like filtering
```

```python
plt.figure()
plt.imshow(np.log(np.abs(F) + 1), cmap='gray')
plt.title('Log Magnitude of F')
```

```python
# Restore the image
f_restored = np.abs(ifft2(ifftshift(F)))
```

```
plt.figure()
plt.imshow(f_restored, cmap='gray')
plt.title('Restored Image')
plt.show()
```