Sukkur Institute of Business Administration University

Computer Systems Engineering Department

Signals & Systems

s

IMAGE DEBLURRING

PROJECT REPORT

Instructor:

Dr. Junaid Ahmed Bhatti

SUBMITTED BY:

- **Ibad Ur Rahman (133-22-0004)**
- **Khalid Hussain (133-22-0011)**

INDEX

	PAGE NO.
CERTIFICATION	
ACKNOWLEDGEMENT	
ABSTRACT	
CHAPTER 1: INTRODUCTION TO IMAGE DEBLURRING	1
1.1 INTRODUCTION	1
1.2 MOTIVATION	1
1.3 OVERVIEW OF IMAGE DEBLURRING	2
CHAPTER 2: OBJECTIVES & TOOLS	3
2.1 INTRODUCTION	3
2.2 OBJECTIVES	3
2.3 TOOLS	4
CHAPTER 3: LITERATURE REVIEW & ALGORITHM	5
3.1 MATLAB OVERVIEW	5
3.2 LITERATURE REVIEW ON IMAGE DEBLURRING	7
3.3 DECONVOLUTION ALGORITHMS	8
3.4 WIENER FILTER	9
3.5 NOISE ESTIMATION	11
3.6 IMAGE BLURRING MODELS	12
3.7 MOTION BLUR	13
3.8 GAUSSIAN NOISE	15
3.9 ALGORITHM IMPLEMENTATION	17
FLOW CHART DISCRIPTION	18
CHAPTER 4: RESULTS & DISCUSSION	19
APPENDIX I: COMMANDS	21
REFERENCES	28

CERTIFICATE

It is certified that **Ibad Ur Rahman & Khalid Hussian** students of **BE-CSE IV** has carried out the necessary work of **Signals & Systems** as per course of studies prevailed at the Computer Systems Engineering Department, Sukkur Institute of Business Administration for **Spring-2024.**

Date: ________________ Instructor's Signature

ACKNOWLEDGMENTS

We extend our gratitude to Dr. Junaid Ahmed Bhatti for his valuable suggestion of this project during our Signals & Systems course. Additionally, we would like to express our special thanks to the individuals who generously assisted us through social media platforms and the Internet. Their support and guidance have been instrumental in the completion of this project.

ABSTRACT

This report presents a comprehensive study on image deblurring, focusing on the restoration of images degraded by motion blur and Gaussian noise. The project leverages MATLAB and its Image Processing Toolbox to implement and analyze various deblurring techniques. We employ a motion blur Point Spread Function (PSF) to simulate blurred images and use the Wiener filter for restoration. The study explores different scenarios, including images with and without added Gaussian noise, and evaluates the impact of noise estimation on deblurring effectiveness. Additionally, we examine the effects of image quantization on the deblurring process. Results demonstrate that accurate noise estimation significantly improves image restoration quality, highlighting the importance of parameter tuning in deblurring algorithms. This work underscores the practical applications of image deblurring in enhancing image quality for various fields such as photography, medical imaging, and forensic analysis.

Chapter 1: Introduction to Image Deblurring

1.1 Introduction

Image deblurring is a critical task in image processing aimed at reconstructing a sharp image from a blurred one. Blurring can occur due to various reasons such as camera shake, motion of the subject, or defocus. The goal of deblurring is to reverse these effects to obtain an image that is as close as possible to the original scene.

1.2 Motivation

The motivation behind image deblurring lies in its wide range of applications in fields such as photography, medical imaging, forensic analysis, and scientific research. Enhancing image quality by reducing blur can lead to more accurate interpretations and analyses.

1.3 Overview of Image Deblurring

Image deblurring involves the use of mathematical models to approximate and invert the blurring process. This typically includes defining a Point Spread Function (PSF) which characterizes the blur, and then using various algorithms to restore the image. Common methods include Wiener filtering, Richardson-Lucy deconvolution, and blind deconvolution.

Chapter 2: Objectives & Tools

2.1 Introduction

This chapter outlines the specific objectives of the image deblurring project and the tools used to achieve these goals.

2.2 Objectives

1. To understand the process of image blurring and deblurring.
2. To implement deblurring algorithms using MATLAB.
3. To analyse the effectiveness of different deblurring techniques.
4. To handle various types of noise and evaluate their impact on image restoration.

2.3 Tools

- **MATLAB**: A high-level programming language and environment for numerical computation, visualization, and programming. It provides built-in functions for image processing and deblurring.
- **Image Processing Toolbox**: A MATLAB toolbox that provides a comprehensive set of reference-standard algorithms and functions for image processing, analysis, visualization, and algorithm development.

Chapter 3: Literature Review & Algorithm

3.1 MATLAB Overview
MATLAB is widely used in academia and industry for data analysis, algorithm development, and numerical computation. Its Image Processing Toolbox includes tools for image segmentation, enhancement, noise reduction, geometric transformations, and deblurring.

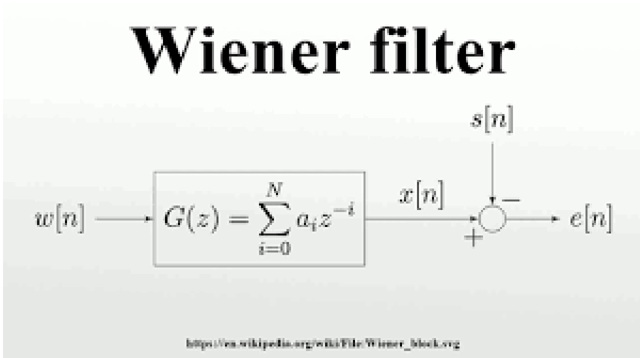
3.2 Literature Review on Image Deblurring
Image deblurring has been extensively studied, with various approaches developed over the years. Traditional methods include linear filtering and inverse filtering, while modern approaches often involve regularization techniques and optimization algorithms.

3.3 Deconvolution Algorithms
Deconvolution is a key technique in image deblurring, aiming to reverse the convolution process that caused the blur. Common deconvolution methods include:

- **Inverse Filtering**: Directly inverts the convolution operation but is sensitive to noise.
- **Wiener Filtering**: Incorporates noise reduction by balancing inverse filtering and noise smoothing.

3.4 Wiener Filter

The Wiener filter is designed to minimize the mean square error between the estimated and true images. It requires knowledge of the PSF and the noise characteristics. The filter is defined as:



$Hwiener(u,v) = H*(u,v)|H(u,v)|2+Sn(u,v)Sx(u,v)H_{wiener}(u,v) = |H(u,v)|2+S_x(u,v)S_n(u,v)H_*(u,v)$ where $H(u,v)H(u,v)$ is the PSF, $Sn(u,v)S_n(u,v)$ is the power spectrum of the noise, and $Sx(u,v)S_x(u,v)$ is the power spectrum of the original image.

3.5 Noise Estimation
Estimating noise characteristics is crucial for effective image deblurring. Noise in digital images can often be modelled as Gaussian noise with known mean and variance. Accurate estimation of these parameters is essential for the Wiener filter and other noise-sensitive deblurring methods.

3.6 Image Blurring Models

Blurring can be modelled using different PSFs depending on the cause of the blur. For motion blur, the PSF can be a linear function representing the path of movement. For out-of-focus blur, the PSF is often a circular function representing the defocused lens aperture.

3.7 Motion Blur

Motion blur occurs when there is relative motion between the camera and the object during exposure. The PSF for motion blur can be modeled as a linear motion function, characterized by the length and angle of the motion.

3.8 Gaussian Noise

Gaussian noise is a statistical noise having a probability density function equal to that of the normal distribution. In image processing, Gaussian noise is commonly added to simulate realistic noisy conditions.

3.9 Algorithm Implementation

The algorithm for image deblurring in this project involves several steps:

1. **Reading the Original Image**: Load the image into MATLAB.
2. **Defining the PSF**: Create a motion blur PSF using MATLAB's **fspecial** function.
3. **Blurring the Image**: Apply the PSF to the image using convolution.
4. **Restoring the Image**: Use the Wiener filter for deblurring, first without noise, then with added Gaussian noise.
5. **Handling Quantized Images**: Analyze the effect of image quantization and apply deblurring techniques accordingly.

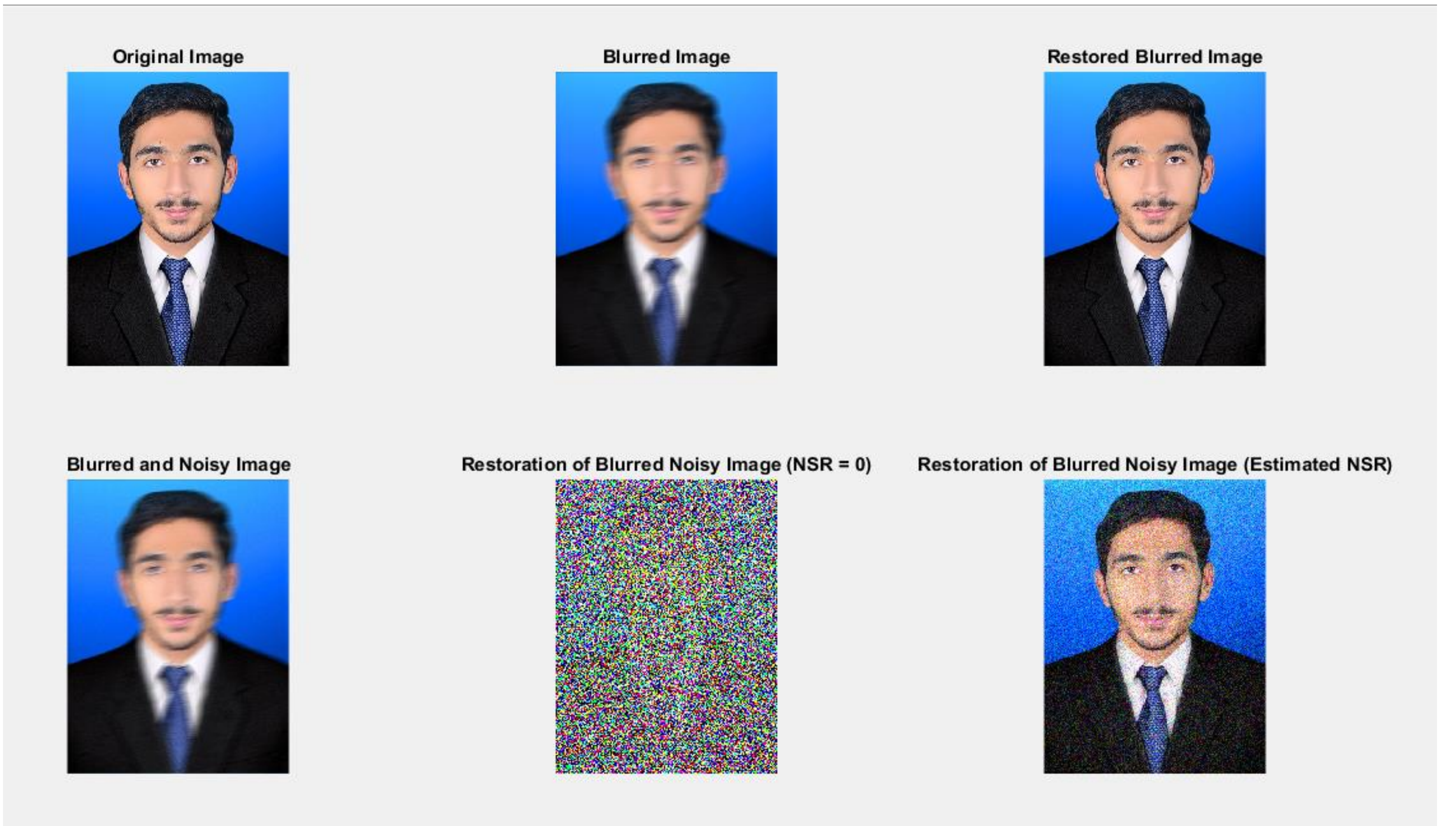
Flow Chart Description:

1. Start
2. Read Original Image
    - Input: Path to the image file
3. Convert Image to Double Precision
    - Function: im2double
4. Define PSF (Point Spread Function)
    - Function: fspecial('motion', length, angle)
5. Blur the Image
    - Function: imfilter
6. Display Blurred Image
7. Apply Wiener Filter for Restoration (without noise)
    - Function: deconvwnr
8. Display Restored Blurred Image
9. Add Gaussian Noise to Blurred Image
    - Function: imnoise
10. Display Blurred and Noisy Image
11. Apply Wiener Filter for Restoration (with noise, NSR=0)
    - Function: deconvwnr
12. Display Restoration of Blurred Noisy Image (NSR=0)
13. Estimate Noise-to-Signal Ratio (NSR)
    - Calculate: NSR = noise_var / signal_var
14. Apply Wiener Filter for Restoration (with estimated NSR)
    - Function: deconvwnr
15. Display Restoration of Blurred Noisy Image (Estimated NSR)
16. Blur and Quantize Image
    - Function: imfilter
17. Display Blurred Quantized Image

18. Apply Wiener Filter for Restoration (Quantized Image, NSR=0)
    - Function: deconvwnr
19. Display Restoration of Blurred Quantized Image (NSR=0)
20. Estimate NSR for Quantized Image
    - Calculate: uniform_quantization_var
21. Apply Wiener Filter for Restoration (Quantized Image, Estimated NSR)
    - Function: deconvwnr
22. Display Restoration of Blurred Quantized Image (Estimated NSR)
23. End

Chapter 4: Results & Discussion

In this chapter, the results of the image deblurring experiments are presented and discussed. Different scenarios, including blur with and without noise, and quantized images, are evaluated to demonstrate the effectiveness of the Wiener filter and the impact of noise estimation on image restoration quality.



APPENDIX I: Commands

This appendix provides the MATLAB commands used in the project, including image reading, blurring, adding noise, and applying the Wiener filter for deblurring.

References

1. Gonzalez, R. C., & Woods, R. E. (2008). Digital Image Processing. Prentice Hall.
2. MATLAB Documentation. (2024). Image Processing Toolbox. MathWorks.
3. Jain, A. K. (1989). Fundamentals of Digital Image Processing. Prentice Hall.

Appendix: MATLAB Commands:

```
% Reading the original image
Ioriginal = imread('C:\Users\princ\Documents\Ibad Documents\PIcs\formal pic.jpg');
figure; imshow(Ioriginal)
title('Original Image')

% Defining the PSF
PSF = fspecial('motion',21,11);

% Converting image to double
Idouble = im2double(Ioriginal);

% Blurring the image
blurred = imfilter(Idouble, PSF, 'conv', 'circular');
figure; imshow(blurred)
title('Blurred Image')

% Restoring the blurred image using Wiener filter
wnr1 = deconvwnr(blurred, PSF);
figure; imshow(wnr1)
title('Restored Blurred Image')

% Adding Gaussian noise
noise_mean = 0;
noise_var = 0.0001;
blurred_noisy = imnoise(blurred, 'gaussian', noise_mean, noise_var);
figure; imshow(blurred_noisy)
title('Blurred and Noisy Image')

% Restoring blurred noisy image (NSR = 0)
wnr2 = deconvwnr(blurred_noisy, PSF);
figure; imshow(wnr2)
title('Restoration of Blurred Noisy Image (NSR = 0)')

% Estimating noise-to-signal ratio (NSR)
signal_var = var(Idouble(:));
NSR = noise_var / signal_var;
wnr3 = deconvwnr(blurred_noisy, PSF, NSR);
figure; imshow(wnr3)
title('Restoration of Blurred Noisy Image (Estimated NSR)')

% Blurring and quantizing the image
blurred_quantized = imfilter(Ioriginal, PSF, 'conv', 'circular');
figure; imshow(blurred_quantized)
title('Blurred Quantized Image')

% Restoring the blurred quantized image (NSR = 0)
wnr4 = deconvwnr(blurred_quantized, PSF);
figure; imshow(wnr4)
title('Restoration of Blurred Quantized Image (NSR = 0)')
```

```
% Estimating NSR for quantized image
uniform_quantization_var = (1/256)^2 / 12;
signal_var = var(Idouble(:));
NSR = uniform_quantization_var / signal_var;
wnr5 = deconvwnr(blurred_quantized, PSF, NSR);
```

Ibad Ur Rahman (Team Member 1): Algorithm Developer and Data Analyst Responsibilities:

1. **Algorithm Development:**
    - Implement the image deblurring algorithms in MATLAB.
    - Develop and optimize the Wiener filter functions.
    - Define and simulate the Point Spread Function (PSF) for motion blur.
    - Implement the addition of Gaussian noise to the blurred images.
2. **Data Processing:**
    - Convert images to double precision format for accurate processing.
    - Perform noise estimation and calculation of Noise-to-Signal Ratio (NSR).
    - Handle image quantization and its impact on the deblurring process.
3. **Code Documentation:**
    - Comment the code comprehensively to ensure clarity and maintainability.
    - Prepare detailed documentation on the implementation of algorithms and functions.
4. **Testing and Validation:**
    - Test the algorithms with various images to ensure robustness and accuracy.
    - Validate the results of deblurring algorithms and adjust parameters for optimal performance.
5. **Result Analysis:**
    - Analyze the performance of the deblurring algorithms on blurred and noisy images.
    - Compare the effectiveness of different NSR values on image restoration quality.

Khalid Hussain (Team Member 2): Report Writer and Visualization Specialist
Responsibilities:

1. **Project Documentation:**
    - Write the project report including the introduction, objectives, methodology, and results.
    - Create detailed descriptions of each step in the deblurring process.
    - Compile a comprehensive literature review on image deblurring and noise reduction techniques.
2. **Flow Chart and Diagrams:**
    - Design a flow chart to visually represent the project workflow.
    - Create diagrams and illustrations to explain the PSF, noise addition, and deblurring process.
    - Develop a clear and attractive visual layout for the report.
3. **Presentation Preparation:**
    - Prepare a presentation summarizing the project's goals, methods, and findings.
    - Design slides with key points, visual aids, and examples of before-and-after images.
    - Practice and coordinate the presentation delivery with team member
4. **Results Visualization:**
    - Generate visual representations of the blurred, noisy, and restored images.
    - Create plots and charts to illustrate the performance metrics of the algorithms.
    - Highlight significant findings and improvements in image quality.
5. **References and Appendices:**
    - Compile and format references for all sources cited in the report.
    - Create an appendix with MATLAB commands and code snippets used in the project.
    - Ensure proper formatting and consistency throughout the document.