SOFTWARE PROJECT LAB - 01

NAME: RASHIK RAIHAN

BSSE ROLL: 1619

Project Title –
ImgProc++

Supervised By
Mohd. Zulfiquar Hafiz
Professor
IIT, University of Dhaka

Project Overview

Objective : Building a Command-Line image processing system capable of:

- Scaling (Upscaling\Downscaling)
- Denoising
- Sharpening
- Edge Detection

Key Features :

- CLI menu for selecting operations
- Basic vs Advanced algorithm comparisons
- Multithreading toggle
- Performance log to CSV file for comparison

Algorithms/Methods

Features	Basic	Advanced
Scaling	Bilinear Interpolation	Lanczos
Denoise	Median Filter	Non-Local Means
Sharpening	Laplacian	Unsharp Masking

- For **Edge Detection**, only sobel operator will be used.

Progress So Far

Scaling:

- So far bilinear intepolation is in working condition

Before
Resolution: 512 x 512

After down sampling
Resolution : 128 x 128

Progress So Far

Denoising:

- Median Filtering and Gaussian blur have been implemented

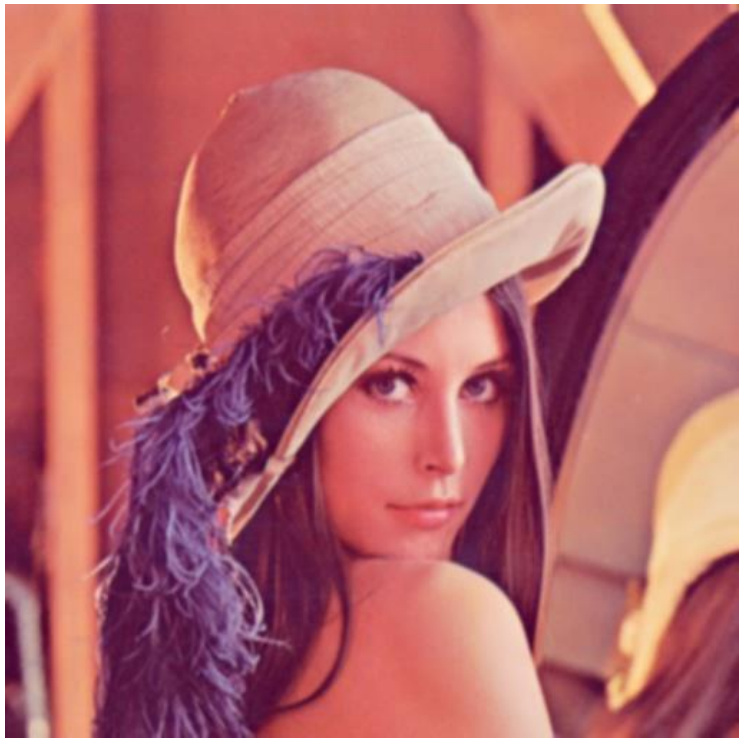
Before
Filled with sand-pepper noise

After applying median filter
A bit blurry but with less noise

Progress So Far

Sharpening:

- Unsharp Masking is in working order for now



Before
Blurry Image

After sharpening
Less blurry but a bit choppy

Progress So Far

Edge Detection:

- Sobel Operator is fully functional

Before

After outling the edges

File Structure and Code Organization

High Level Structure:

src/

- ├ main.c
- ├ scaling.c
- ├ denoise.c
- ├ sharpen.c
- ├ edge.c
- ├ utils.c
- ├ bmp.c

include/

- ├ scaling.h
- ├ denoise.h
- ├ sharpen.h
- ├ edge.h
- ├ utils.h
- ├ bmp.h

File Structure and Code Organization

Module Breakdown

main.c (Controller Layer)

- Entry point of the program
- Displays the menu-driven interface
- Handles user input
- Calls appropriate image processing functions
- Does not contain image-processing logic

BMP Handling Module (bmp.c / bmp.h)

- Handles:
  - Reading BMP headers
  - Writing output images
- Keeps file-format logic separate from image processing

File Structure and Code Organization

Image Processing Modules

Each major feature has its own source and header file.

Scaling Module

- Handles image resizing
- Implements bilinear interpolation
- Responsible only for pixel mapping and size conversion

Denoising Module

- Applies smoothing filters
- Uses convolution with a kernel
- Reduces noise while preserving image structure

File Structure and Code Organization

Sharpening Module

- Enhances edges and fine details
- Uses high-pass filtering techniques

Edge Detection Module

- Detects object outlines
- Uses Sobel Operator (gradient based)

Utility Module (utils.c / utils.h)

- Common helper functions:
  - ASCII representation
  - Menu Overhaul
  - Copying Image
- Avoids code duplication across modules

Next Steps

Implementing remaining advanced algorithms

Multithreading support for functions

Exporting the data to CSV file for comparison

THANK YOU