Digital Image Processing Project Using MATLAB

Faik Doruk Akgüney
Student, *Department of Electrical Electronics Engineering*
Cankaya University
Ankara, Turkey
dorukakguney@hotmail.com

***Abstract*—This project demonstrates digital image processing using MATLAB. The process involves reading an image, converting it to grayscale, creating a negative of the grayscale image, blurring the negative image, and finally generating a processed image. The steps and results are visualized using MATLAB.**

***Index Terms*—MATLAB, Digital Image Processing, Grayscale Image, Image Blurring**

I. Introduction

Digital image processing is a method to perform operations on an image to enhance it or extract useful information. This project, completed as part of the EE-462 Digital Image Processing course, showcases basic image processing techniques using MATLAB.

II. Methodology

The methodology involves several key steps:

- Reading an image.
- Converting the image to grayscale.
- Creating a negative of the grayscale image.
- Blurring the negative grayscale image.
- Generating the final processed image.

III. Implementation

The MATLAB code for this project is provided below:

```matlab
close all;
clear all;
clc;

% EE-462 Digital Image Processing
% Faik Doruk Akguney
% 201726003

% Read an image
original_image = imread("Image-3.jpg");

% Invert original image to grayscale image
grayscale_image = rgb2gray(original_image);

[r,c] = size(grayscale_image);

% Convert grayscale image to negative grayscale image
negative_grayscale_image = zeros(r,c);
for i=1:r
    for j=1:c
        negative_grayscale_image(i,j)= 255 - double(grayscale_image(i,j));
    end
end

% Convert Negative Grayscale Image to Blurred Image
sigma = 20;
filter = 7 * sigma;
H = fspecial(’gaussian’,[filter,filter],sigma);
blurred_image = imfilter(negative_grayscale_image,H);

% Invert blurred image to negative blurred image
negative_blurred_image=zeros(r,c);
for i=1:r
    for j=1:c
        negative_blurred_image(i,j)=255-blurred_image(i,j);
    end
end

% Final Image
F = imdivide(double(grayscale_image), negative_blurred_image);
Finale_image = round(255*F);

figure(’Name’,’Step by Step’,NumberTitle=’off’);
subplot(3,2,1);
imshow(original_image);
title("Original Image");
subplot(3,2,2);
imshow(grayscale_image);
title("Grayscale Image");
subplot(3,2,3);
imshow(uint8(negative_grayscale_image));
title("Negative Grayscale Image")
subplot(3,2,4);
imshow(uint8(blurred_image));
title("Blurred Image");
subplot(3,2,5);
imshow(uint8(negative_blurred_image));
title("Negative Blurred Image");
subplot(3,2,6);
imshow(uint8(Finale_image));
title("Final Image");

figure("Name",’Final Image’,NumberTitle=’off’,Units=’normalized’);
imshow(uint8(Finale_image));
```

Listing 1. Digital Image Processing in MATLAB

IV. Results and Discussion

The following steps were performed and visualized using MATLAB:

1) **Original Image:** The original image read using ‘imread‘.
2) **Grayscale Image:** Conversion of the original image to grayscale using ‘rgb2gray‘.
3) **Negative Grayscale Image:** Inversion of the grayscale image.

4) **Blurred Image:** Blurring the negative grayscale image using a Gaussian filter.
5) **Negative Blurred Image:** Inversion of the blurred image.
6) **Final Image:** Division of the grayscale image by the negative blurred image to produce the final processed image.

Each step is visualized using MATLAB's 'imshow' function. The final image demonstrates the effectiveness of the image processing techniques applied.

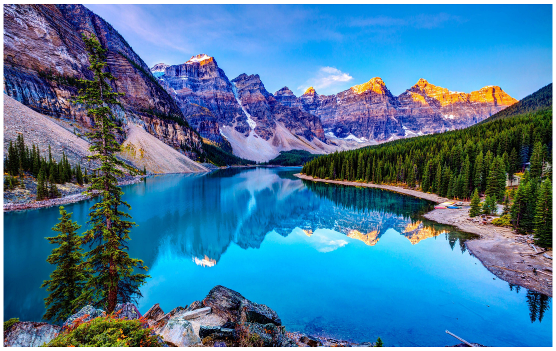

Fig. 1. Original Image

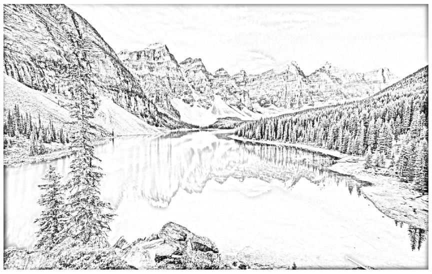

Fig. 2. Final Processed Image

V. Conclusion

This project successfully demonstrates basic digital image processing techniques using MATLAB. The steps taken provide a clear understanding of how image processing can be applied to manipulate and enhance images.

VI. Future Work

Future work could include implementing more advanced image processing techniques such as edge detection, image segmentation, and image enhancement to further explore the capabilities of MATLAB in digital image processing.

References

[1] MathWorks, "MATLAB Documentation," 2024. [Online]. Available: https://www.mathworks.com/help/matlab/