

Image Processing Basics with MATLAB

Introduction

This assignment explores the basic techniques of image processing using MATLAB. The primary tasks include loading and displaying images, enhancing image quality through brightness adjustment, contrast enhancement, and noise reduction, and finally, implementing edge detection algorithms such as Sobel and Prewitt.

Image Loading and Display

The image was loaded using the 'imread' function in MATLAB, and the original image was displayed using the 'imshow' function. Below is the MATLAB code snippet used for this task:

```
img = imread('image.jpeg');
imshow(img);
title('Original Image');
```

Original Image:



Image Enhancement:

Three techniques were applied to enhance the image:

1. Brightness Adjustment: The brightness was adjusted using 'imadjust'.
2. Contrast Enhancement: The contrast was improved using 'histeq'.
3. Noise Reduction: Noise was reduced using Gaussian filtering with 'imgaussfilt'.

Below is the code for brightness adjustment:

```
bright_img = imadjust(img, [], [], 1.2); %Increase Brightness  
imshow(bright_img);  
title('Brightness Adjusted Image');
```

Brightness Adjusted Image:



Below is the code for contrast enhanced:

```
contrast_img = histeq(img);  
imshow(contrast_img);  
title('Contrast Enhanced Image');
```

Contrast Enhanced Image:



Below is the code for noise reduction:

```
noise_reduction_img = imgaussfilt(img, 2); %sigma value 2  
imshow(noise_reduction_img);  
title('Noise Reduced Image');
```

Noise Reduced Image:

Noise Reduced Image



Edge Detection

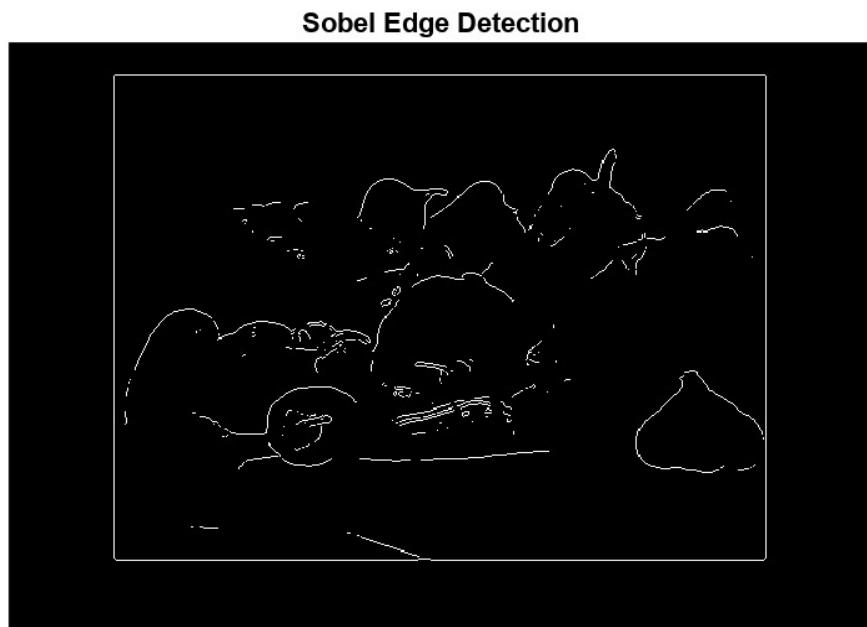
Edge detection was performed using two methods:

1. Sobel Edge Detection: Implemented using the 'edge' function with the 'Sobel' method.
2. Prewitt Edge Detection: Implemented using the 'edge' function with the 'Prewitt' method.

The following MATLAB code was used for Sobel edge detection:

```
edges_sobel = edge(rgb2gray(img), 'sobel');  
imshow(edges_sobel);  
title('Sobel Edge Detection');
```

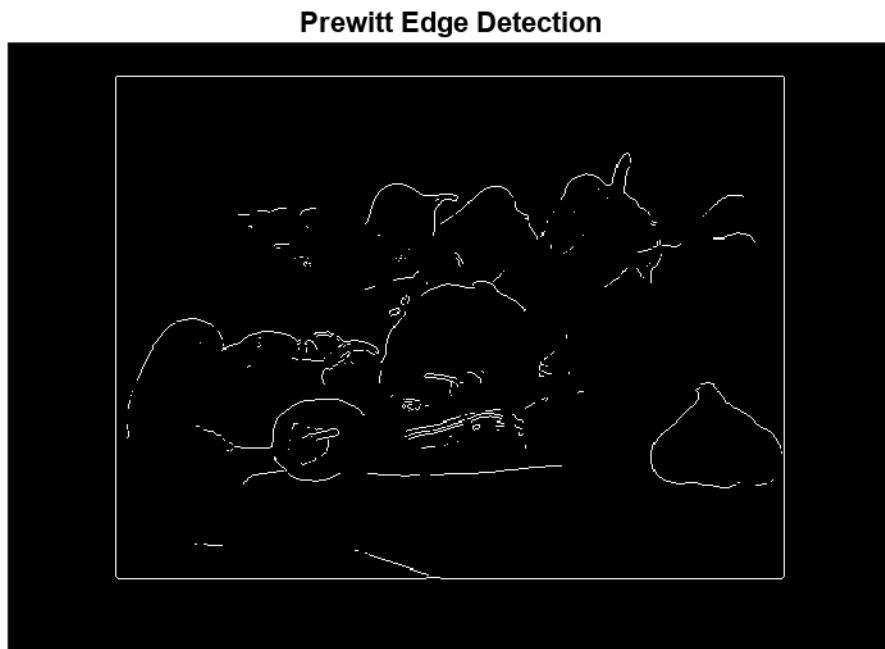
Sobel Edge Detected Image:



The following MATLAB code was used for Prewitt edge detection:

```
edges_prewitt = edge(rgb2gray(img), 'prewitt');
imshow(edges_prewitt);
title('Prewitt Edge Detection');
```

Prewitt Edge Detection:



Documentation

Code Comments

```
% Load the image
```

```
img = imread('image.jpeg');
```

```
% Display the original image
```

```
imshow(img);
```

```
title('Original Image');
```

```
% Adjust brightness
```

```
bright_img = imadjust(img, [], [], 1.2); %Increase Brightness
```

```
imshow(bright_img);
```

```
title('Brightness Adjusted Image');
```

```
% Enhance contrast
```

```
contrast_img = histeq(img);
```

```
imshow(contrast_img);
```

```
title('Contrast Enhanced Image');
```

```
% Reduce noise
```

```
noise_reduction_img = imgaussfilt(img, 2 ); %sigma value 2
```

```
imshow(noise_reduction_img);
```

```
title('Noise Reduced Image');
```

```
% Sobel edge detection
```

```
edges_sobel = edge(rgb2gray(img), 'sobel');
```

```
imshow(edges_sobel);
```

```
title('Sobel Edge Detection');
```

```
% Prewitt edge detection
```

```
edges_prewitt = edge(rgb2gray(img), 'prewitt');
imshow(edges_prewitt);
title('Prewitt Edge Detection');
```

Conclusion

In this assignment, we successfully implemented basic image processing techniques in MATLAB. We loaded and displayed images, enhanced image quality through brightness and contrast adjustments, reduced noise, and performed edge detection using Sobel and Prewitt methods. The results demonstrated the effectiveness of these techniques in improving image quality and extracting important features