Digital Image Processing: Lab Assignment 1

Vaibhav Sharma: T23156

September 9, 2024

1 To acquire an image, store it in different formats and display the properties of the images.

1.1 `subplot`

The `subplot` function divides the current figure window into a grid of smaller panes. Each pane can contain a separate plot. This is useful for comparing multiple plots in a single figure window.

```
x = 0:0.01:2*pi ;
y1 = sin ( x ) ;
y2 = cos ( x ) ;
subplot ( 2, 1, 1 ) ;
plot ( x , y1 ) ;
subplot ( 2, 1, 2 ) ;
plot ( x , y2 ) ;
```

Code 1: MATLAB code using subplot

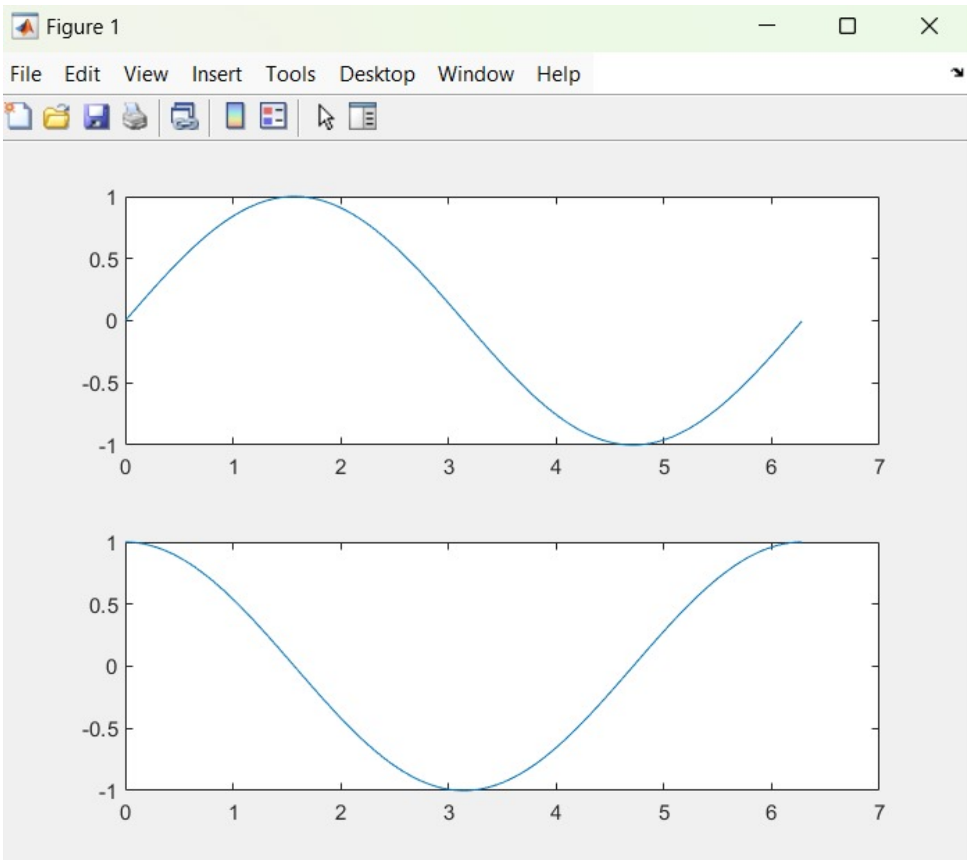


Figure 1: Results of the `subplot` function.

1.2 imshow

The `imshow` function displays an image stored in a file or in the MATLAB workspace. This function is essential for visualizing images in various formats.

```
imshow('image.png');
```

Code 2: MATLAB code using `imshow`

Figure 2: Displaying an image using `imshow`.

1.3 `impixelinfo`

The `impixelinfo` function creates a tool in the current figure that displays pixel information. As you move the cursor over an image, it shows the pixel's coordinates and colour values.

```
imshow('image.png');
impixelinfo;
```

Code 3: MATLAB code using `impixelinfo`

Figure 3: Using `impixelinfo` to display pixel information.

1.4 imageinfo

The `imageinfo` function opens a separate window displaying detailed information about the image in the current figure, including image size, type, and colour depth.

```
imshow('image.png');
imageinfo;
```

Code 4: MATLAB code using `imageinfo`

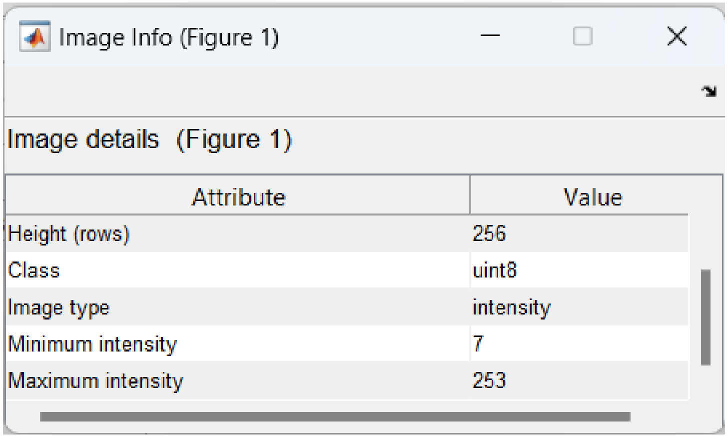


Figure 4: Using `imageinfo` to display image details.

1.5 title

The `title` function adds a title to the top of the current axes. This is useful for labelling plots within subplots.

```
plot(x, y1);
title('Camaraman');
```

Code 5: MATLAB code using `title`

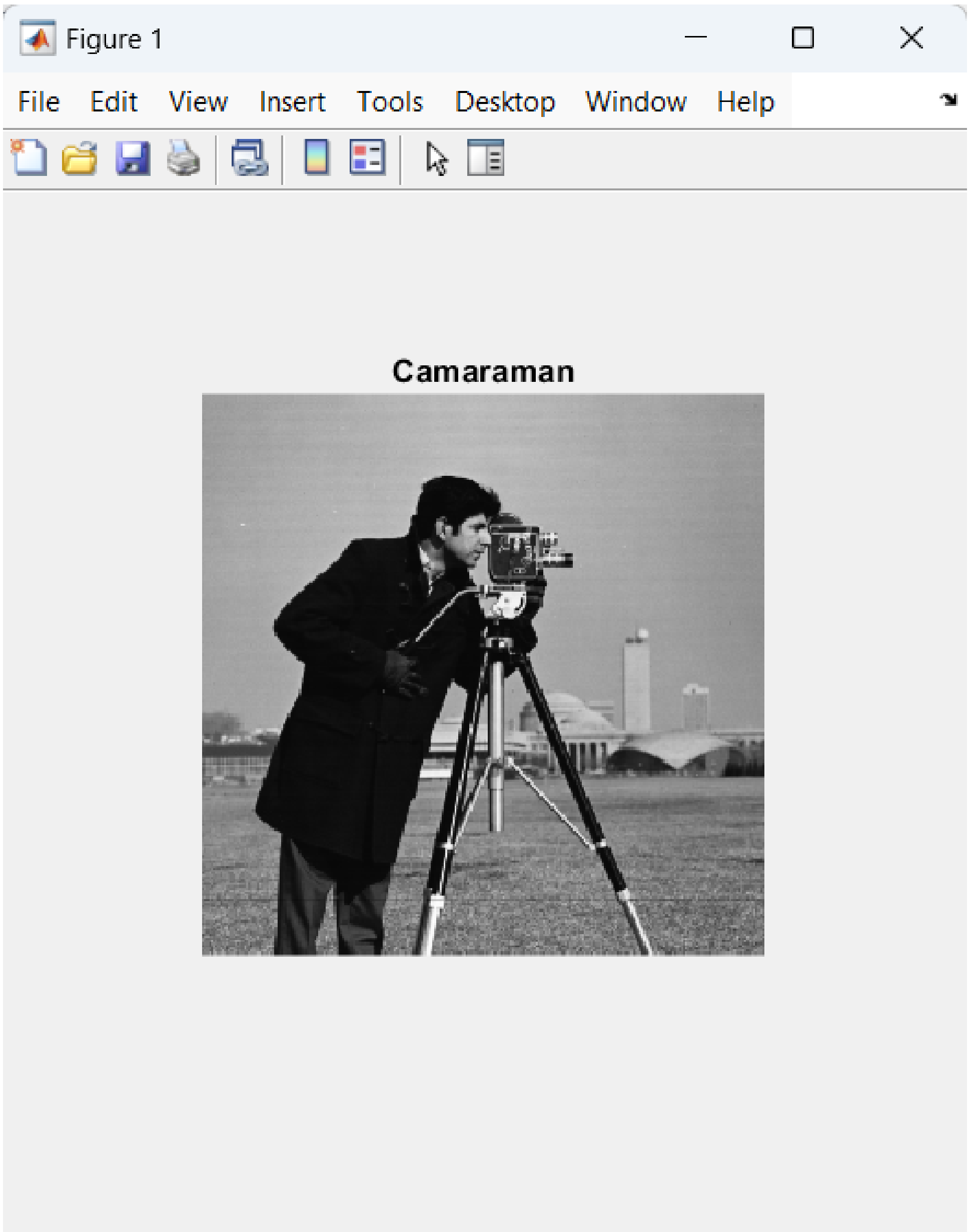


Figure 5: Adding a title to a plot using `title`.

1.6 Conclusion

This report covered the usage of several MATLAB functions that are crucial for image processing and visualization. By using these functions, we can effectively display, analyze, and annotate images and plots within MATLAB.

2 To find the discrete Fourier transform of a grey scale image and perform inverse transform to get back the image.

2.1 Introduction

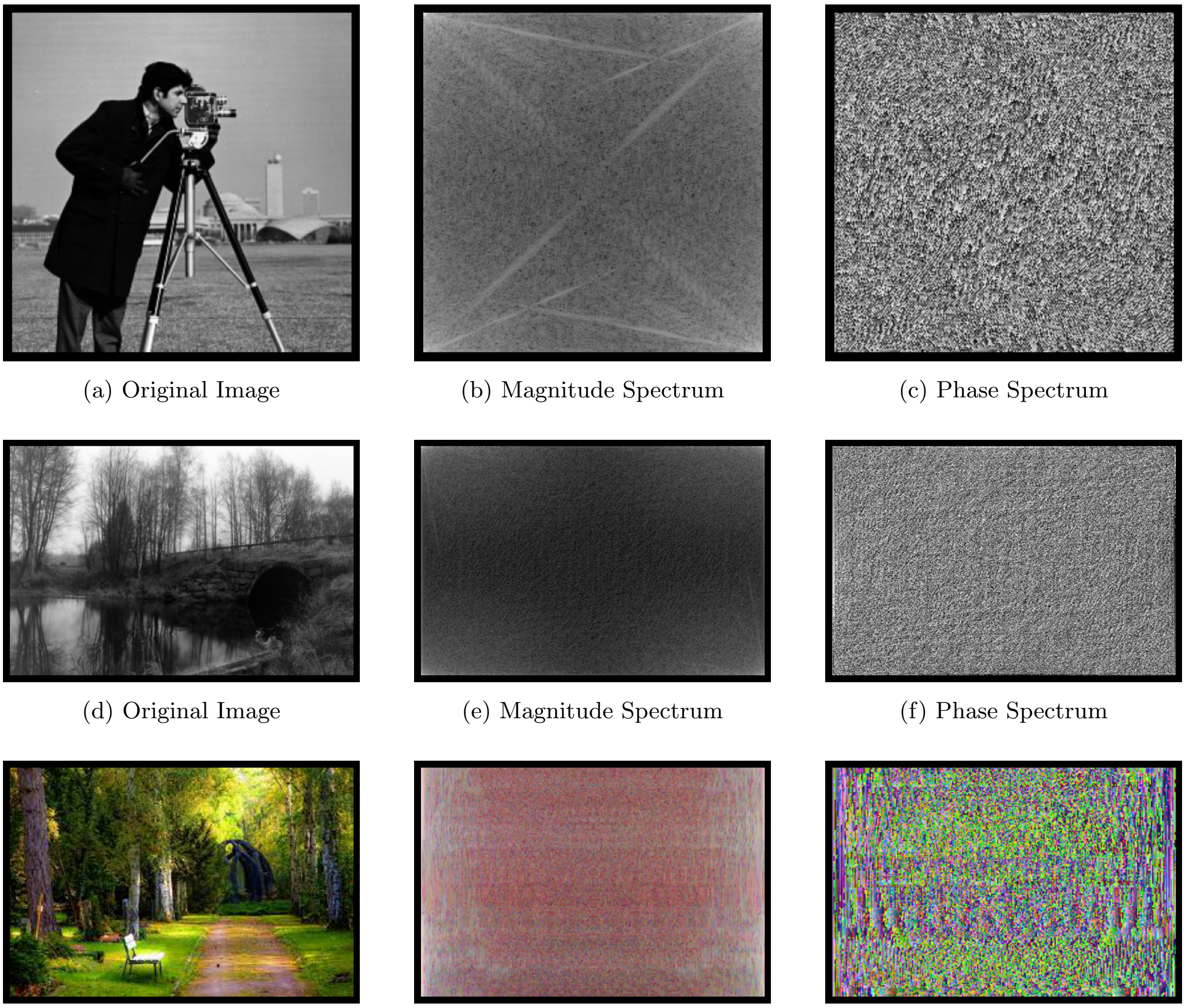
The Fourier Transform, developed by Jean-Baptiste Joseph Fourier, is a crucial tool in image processing. It allows us to transform an image from the spatial domain into the frequency domain, facilitating the analysis of spatial variations. This report illustrates various aspects of image processing using the Fourier Transform, including rotation and convolution properties.

The Fourier Transform $F(u,v)$ of an image $f(x,y)$ is given by:

$$F(u,v) = \int\int f(x,y)\, e^{-j2\pi(ux+vy)}\, dx\, dy$$

The Inverse Fourier Transform is:

$$f(x,y) = \int\int F(u,v)\, e^{j2\pi(ux+vy)}\, du\, dv$$



(a) Original Image (b) Magnitude Spectrum (c) Phase Spectrum

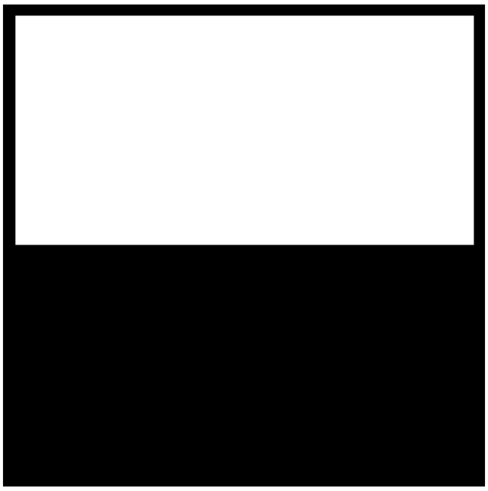
(d) Original Image (e) Magnitude Spectrum (f) Phase Spectrum

(g) Original Image (h) Magnitude Spectrum (i) Phase Spectrum

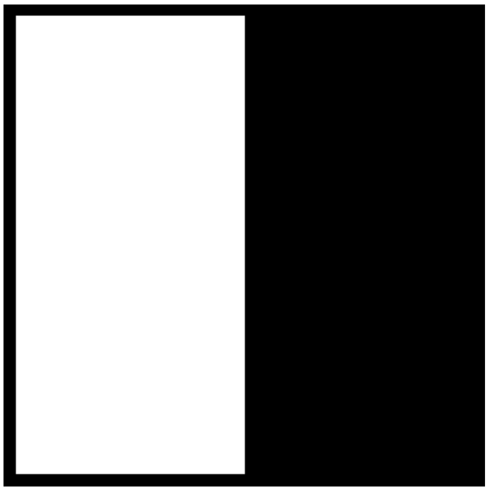
Figure 6: Few Images and Visualization of their Discrete Fourier Transform.

2.2 Rotation Property of Fourier Transform

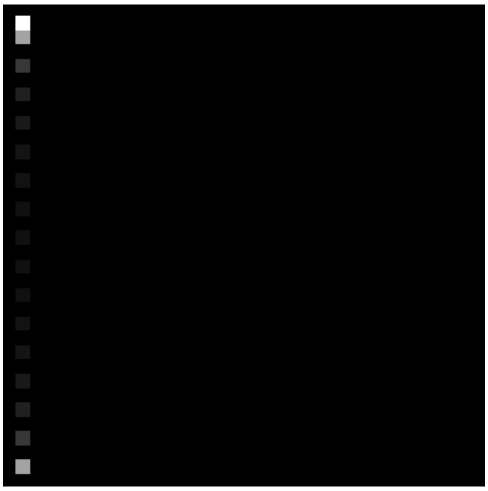
The rotation property of the Fourier Transform implies that rotating an image by a certain angle results in a corresponding rotation of its Fourier Transform.



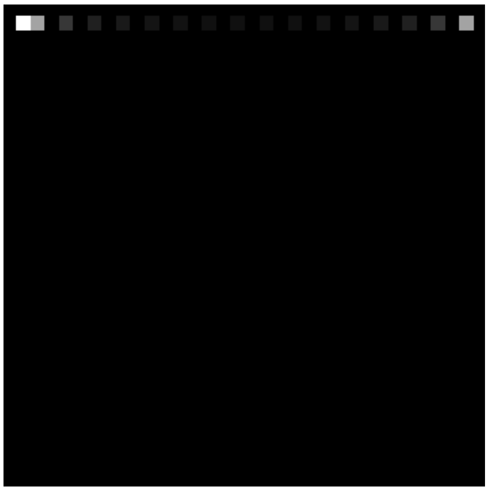
(a) Image



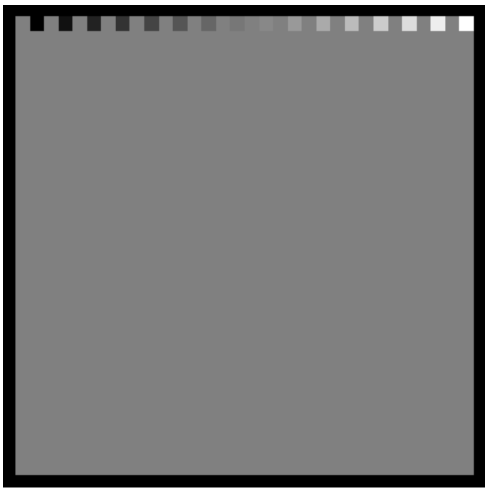
(b) Rotated Image



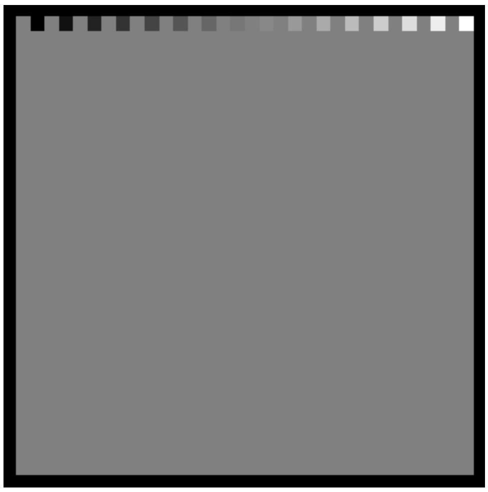
(c) DFT(Magnitude) of Image



(d) DFT(Magnitude) of Rotated Image



(e) DFT(Phase) of Image



(f) DFt(Phase) of Rotated Image

Figure 7: Visualization of Rotation property of Discrete Fourier Transform

2.3 Convolution Property of Fourier Transform

The convolution property states that the convolution of two images in the spatial domain corresponds to the multiplication of their Fourier Transforms in the frequency domain.

If $f(x, y)$ and $g(x, y)$ are two images, their convolution $h(x, y)$ is defined as:

$$h(x, y) = \int \int f(x', y') \cdot g(x - x', y - y')\, dx'\, dy'$$

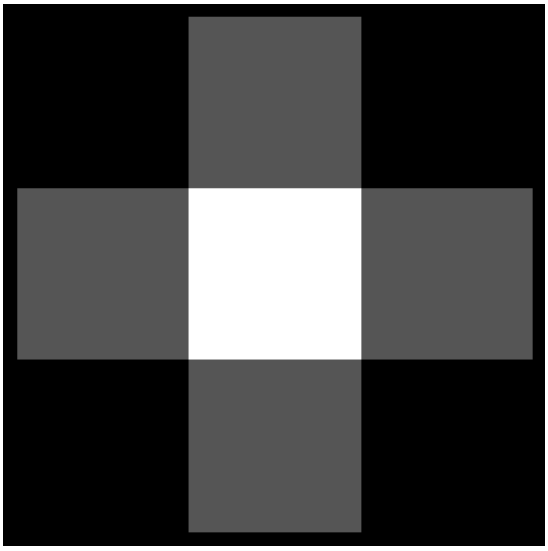
In the Fourier domain:

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

The Inverse Fourier Transform retrieves the convolution result:

$$h(x, y) = \int \int H(u, v)\, e^{j2\pi(ux+vy)}\, du\, dv$$

(a) Image 1

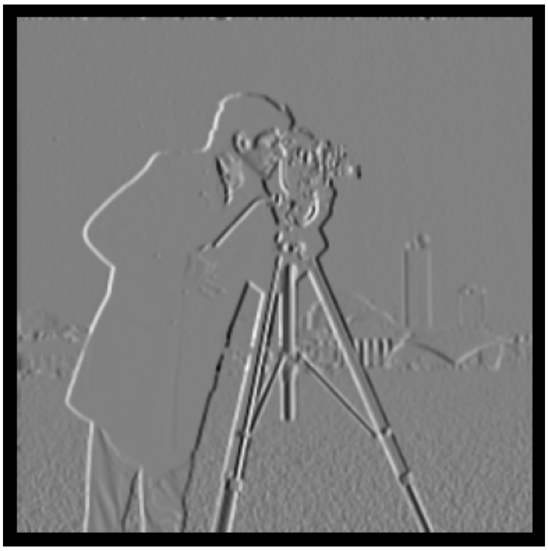


(b) Gaussian Kernel of 3*3 size scaled to fit

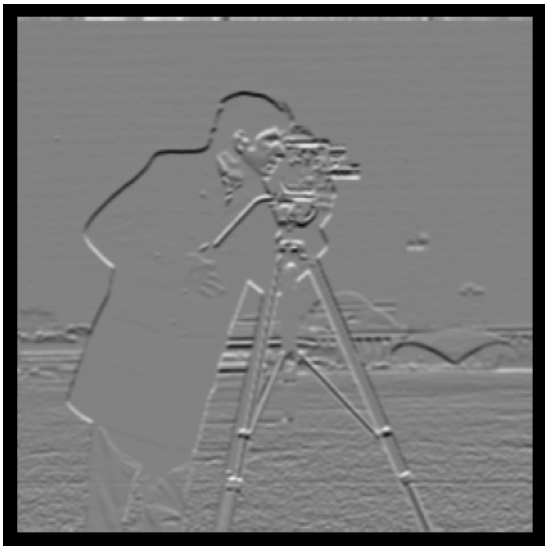
(c) Convolution of Image 1 and Gaussian Kernel

(d) IDFT of the product of DFTs of two images

Figure 8: Visualization of convolution property of Fourier Transform



(a) sobelx



(b) soblely

(c) laplacian

Figure 9: Convolution using Discrete Fourier Transform

2.4 Conclusion

This section demonstrates the application of the Fourier Transform in image processing. Fourier Transform could be used to perform convolution due to the convolution property of DFT, and we also demonstrated the rotation property of Fourier Transform.

Code Availability

All the code used in this project is available in a public GitHub repository. You can access it at: https://github.com/Computer-Science-Practicum/DIP-Lab-Assignment.