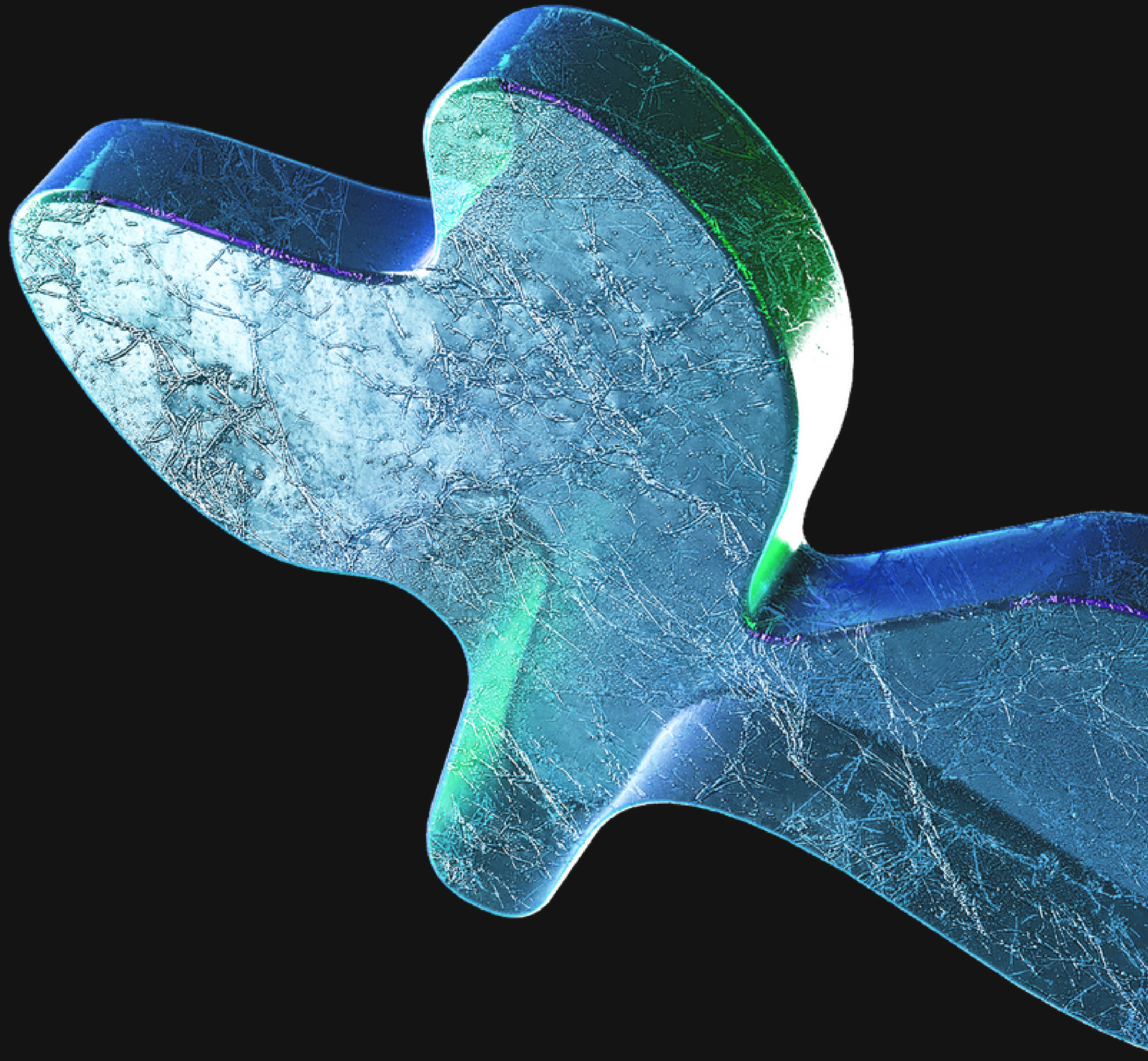


Applications of Fourier Transform in Digital Image Processing

What is Fourier Transform?

- Virtually everything in the world can be described via a waveform - a function of time, space or some other variable. For instance, sound waves, electromagnetic fields, the elevation of a hill versus location, a plot of VSWR versus frequency, the price of your favorite stock versus time, etc. The Fourier Transform gives us a unique and powerful way of viewing these waveforms.

- Fourier Transform is a mathematical model which helps to transform the signals between two different domains, such as transforming signal from frequency domain to time domain or vice versa. Fourier transform has many applications in Engineering and Physics, such as signal processing, RADAR, and so on

- The generalisation of the complex Fourier series is known as the Fourier transform. The term “Fourier transform” can be used in the mathematical function, and it is also used in the representation of the frequency domain. The Fourier transform helps to extend the Fourier series to the non-periodic functions, which helps us to view any functions in terms of the sum of simple sinusoids.

Mathematical Representation

The Fourier transform is considered to be a generalisation of the complex Fourier series in the limit L→∞. Also, convert discrete An to the continuous F(k)dk and let n/L→k. Finally, convert the sum to an integral.Thus, the Fourier transform of a function f(x) is given by:

$$f(x) = \int_{-\infty}^{\infty} F(k)e^{2\pi ikx}dk$$
$$F(k) = \int_{-\infty}^{\infty} f(x)e^{-2\pi ikx}dx$$

Fourier sine transform

The Fourier sine transform is defined as the imaginary part of full complex Fourier transform, and it is given by:

$$F_x^{(s)}[f(x)](k) = I[F_x[f(x)](k)]$$
$$F_x^{(s)}[f(x)](k) = \int_{-\infty}^{\infty} sin(2\pi kx)f(x)dx$$

Fourier cosine transform

The Fourier transform for cosines of a real function is defined as the real part of a full complex Fourier transform.

$$F_x^{(c)}[f(x)](k) = R[F_x[f(x)](k)]$$
$$F_x^{(c)}[f(x)](k) = \int_{-\infty}^{\infty} cos(2\pi kx)f(x)dx$$

Fourier transform in Image processing

The Fourier Transform is an important image processing tool which is used to decompose an image into its sine and cosine components. The output of the transformation represents the image in the *Fourier* or frequency domain, while the input image is the spatial domain equivalent. In the Fourier domain image, each point represents a particular frequency contained in the spatial domain image.

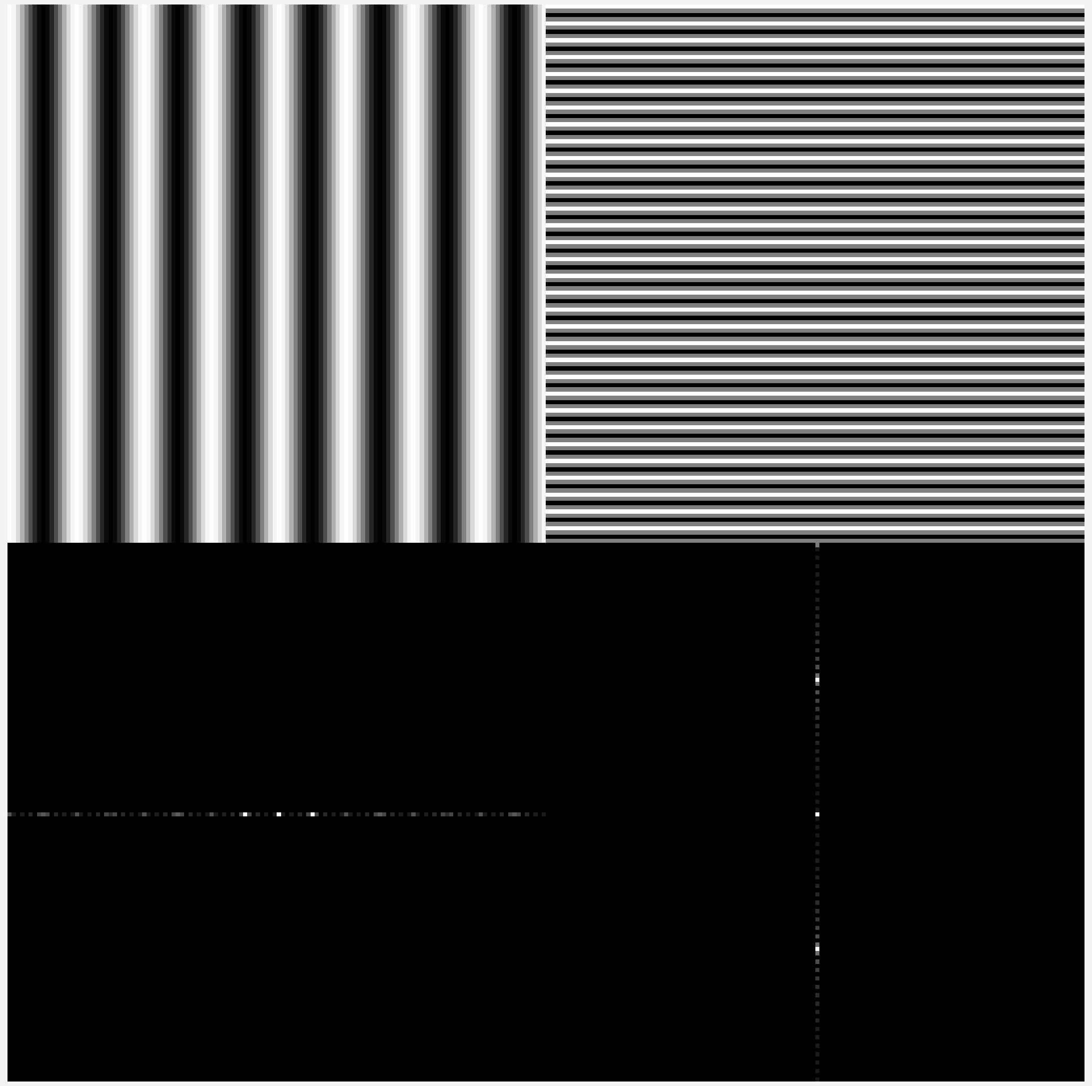
The Fourier Transform is used in a wide range of applications, such as

- Image analysis,
- Image filtering,
- Image reconstruction
- Image compression

Effects of fourier Transform

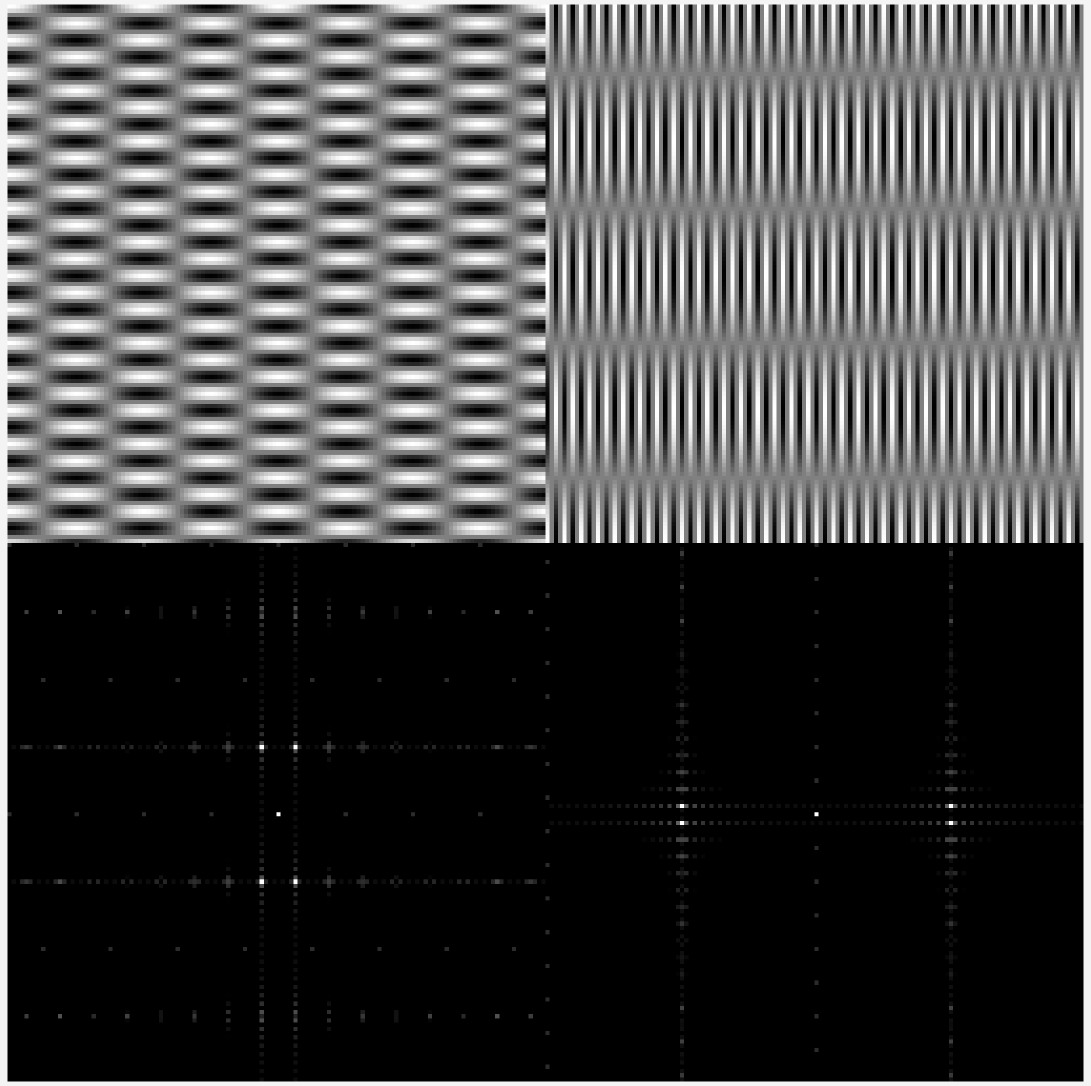
EXAMPLE 1

First we will investigate the "basis" functions for the Fourier Transform (FT). The FT tries to represent all images as a summation of cosine-like images. Therefore images that are pure cosines have particularly simple FTs.



- This shows 2 images with their Fourier Transforms directly underneath. The images are a pure horizontal cosine of 8 cycles and a pure vertical cosine of 32 cycles.
- Notice that the FT for each just has a single component, represented by 2 bright spots symmetrically placed about the center of the FT image. The center of the image is the origin of the frequency coordinate system.
- The u-axis runs left to right through the center and represents the horizontal component of frequency.
- The v-axis runs bottom to top through the center and represents the vertical component of frequency.
- In both cases there is a dot at the center that represents the (0,0) frequency term or average value of the image. Images usually have a large average value (like 128) and lots of low frequency information so FT images usually have a bright blob of components near the center.
- Notice that high frequencies in the vertical direction will cause bright dots away from the center in the vertical direction. And that high frequencies in the horizontal direction will cause bright dots away from the center in the horizontal direction

EXAMPLE 2

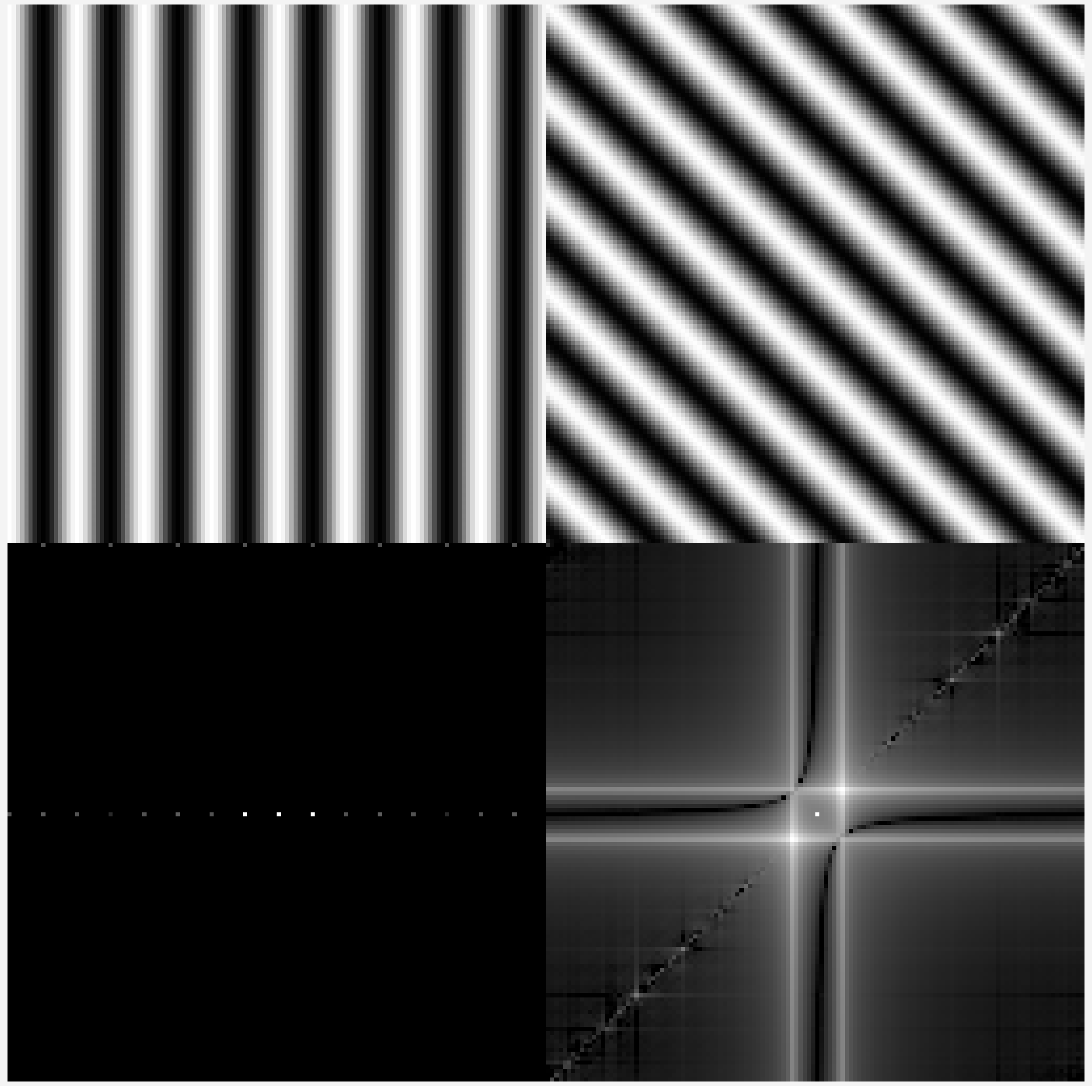


- Here are 2 images of more general Fourier components.
- They are images of 2D cosines with both horizontal and vertical components.
- The one on the left has 4 cycles horizontally and 16 cycles vertically. The one on the right has 32 cycles horizontally and 2 cycles vertically. (Note: You see a gray band when the function goes through gray = 128 which happens twice/cycle.) You may begin to notice there is a lot of symmetry.
- For all REAL (as opposed to IMAGINARY or COMPLEX) images, the FT is symmetrical about the origin so the 1st and 3rd quadrants are the same and the 2nd and 4th quadrants are the same.
- If the image is symmetrical about the x-axis (as the cosine images are) 4-fold symmetry results.

EXAMPLE 3

Rotation and Edge effects:

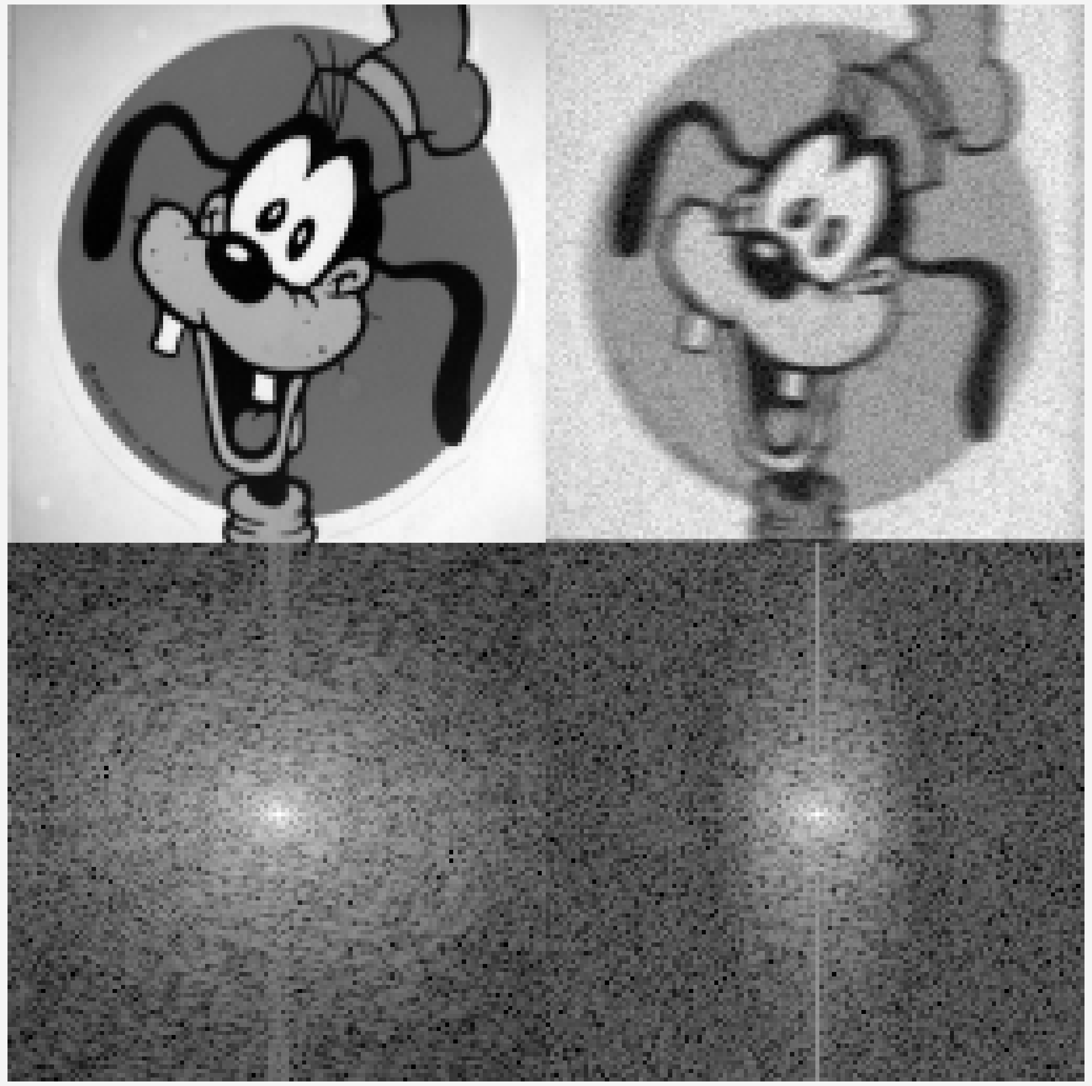
In general, rotation of the image results in equivalent rotation of its FT. To see that this is true, we will take the FT of a simple cosine and also the FT of a rotated version of the same function. The results can be seen by:



- At first, the results seem rather surprising. The horizontal cosine has its normal, very simple FT.
- But the rotated cosine seems to have an FT that is much more complicated, with strong diagonal components, and also strong "plus sign" shaped horizontal and vertical components.
- The question is, where did these horizontal and vertical components come from? The answer is that the FT always treats an image as if it were part of a periodically replicated array of identical images extending horizontally and vertically to infinity.

EXAMPLE 4

Some image transforms:



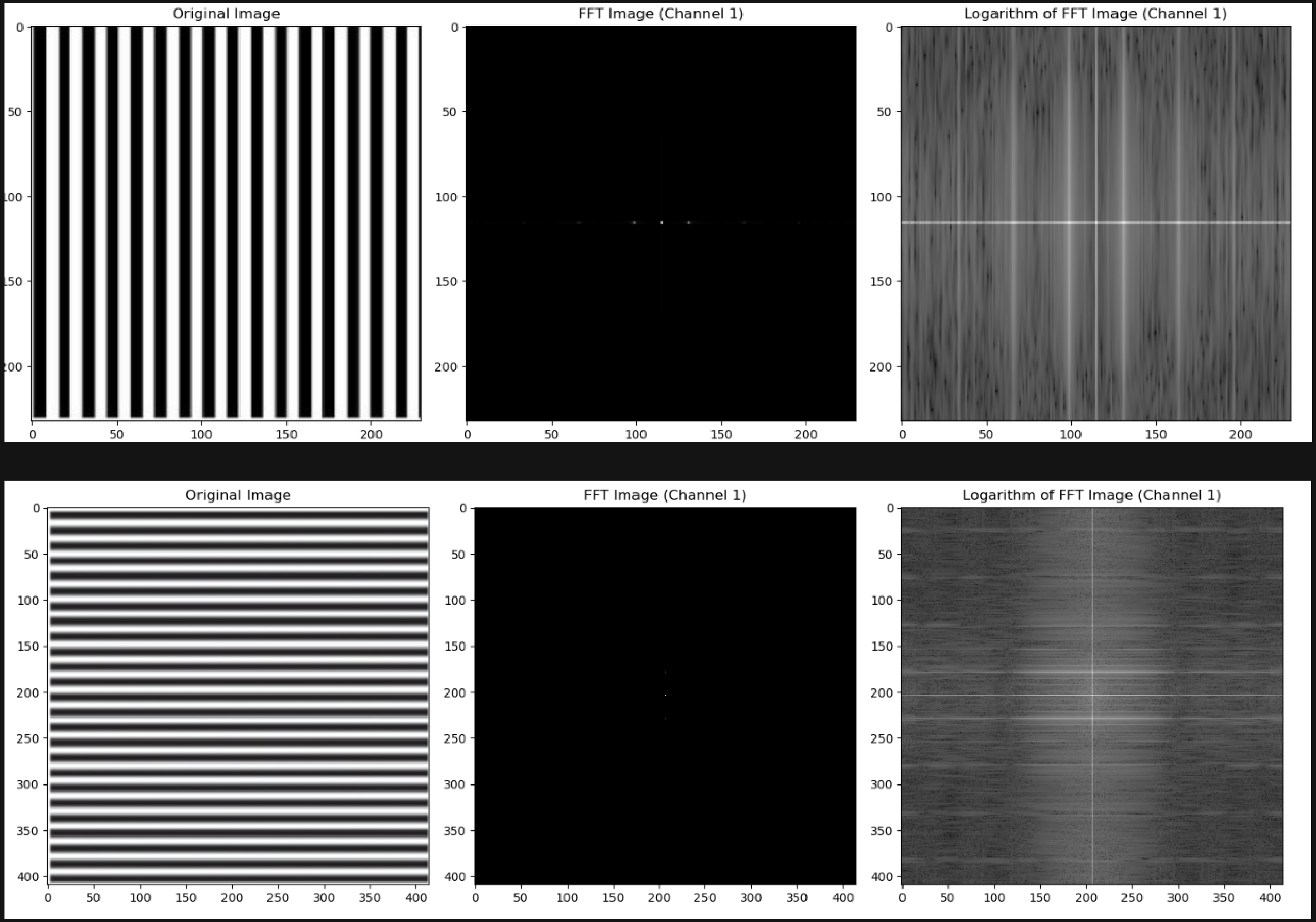
- There are 2 images, goofy and the degraded goofy, with FTs below each. Notice that both suffer from edge effects as evidenced by the strong vertical line through the center.
- The major effect to notice is that in the transform of the degraded goofy the high frequencies in the horizontal direction have been significantly attenuated. This is due to the fact that the degraded image was formed by smoothing only in the horizontal direction.
- Also, if you look carefully you can see that the degraded goofy has a slightly larger background noise level at high frequencies.
- This is difficult to see and perhaps not even meaningful because the images are scaled differently, but if really there, it is due to the random noise added to the degraded goofy.
- Notice also that it is difficult to make much sense out of the low frequency information. This is typical of real life images.

Implementation

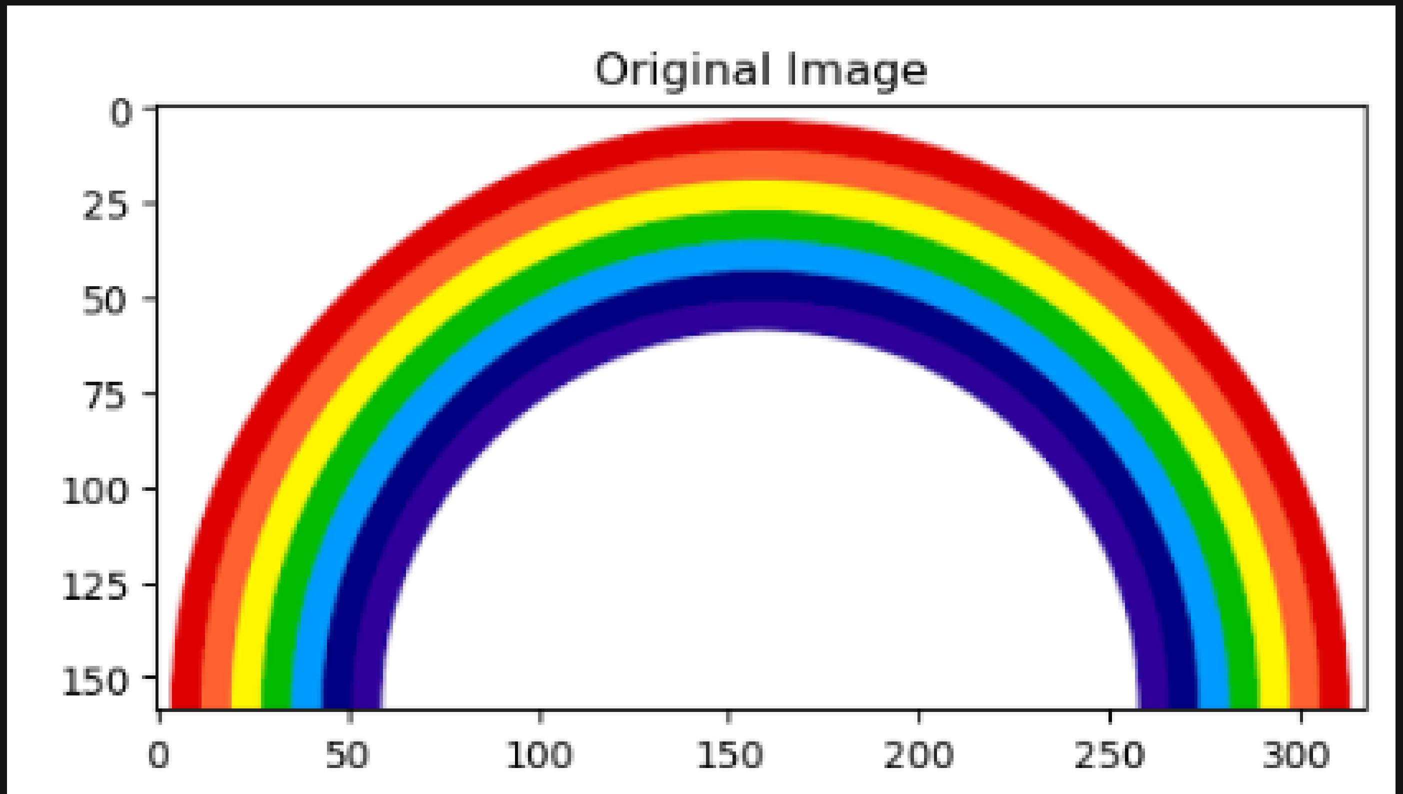
CODE

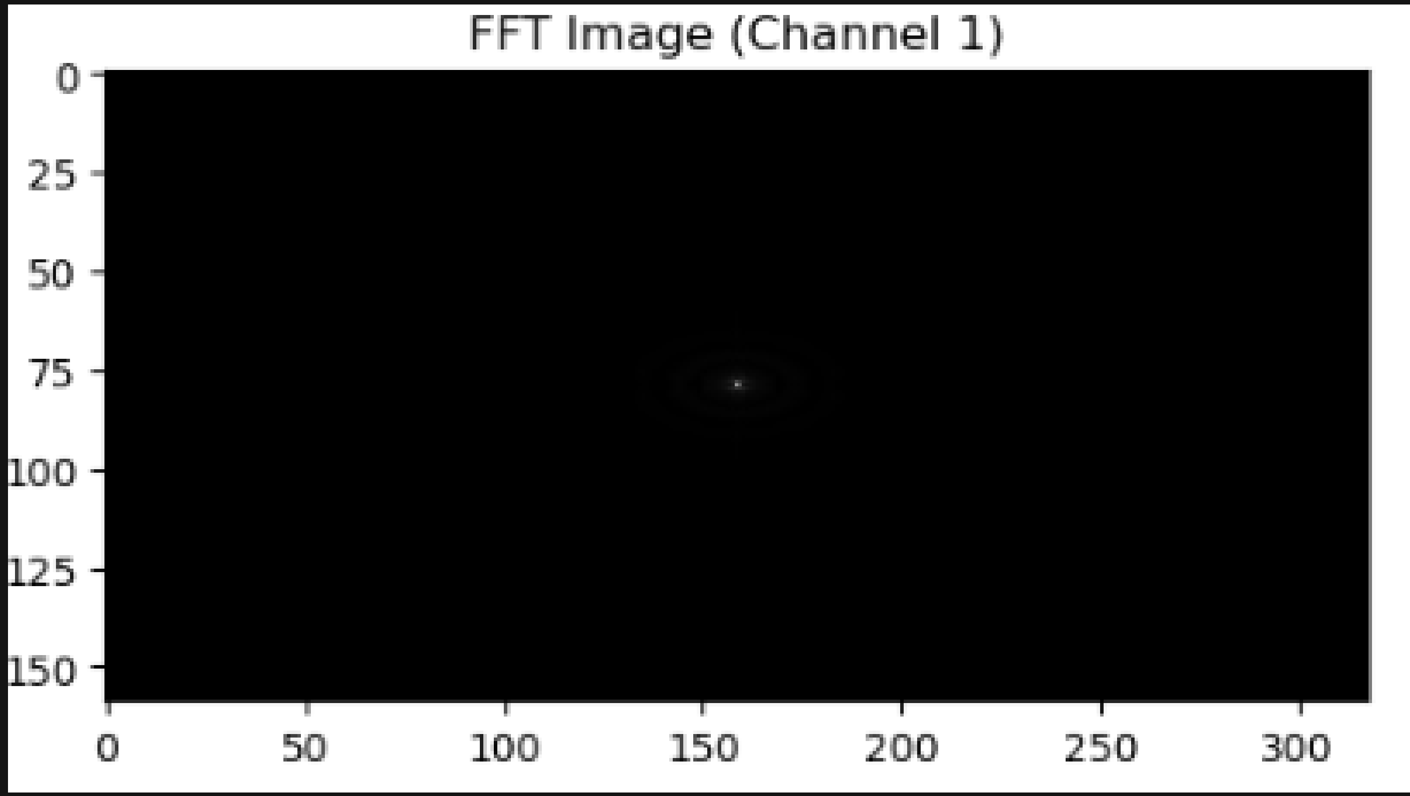
```python
import cv2
import numpy as np
import matplotlib.pyplot as plt
def rgb_fft(image):
    f_size = 30
    fft_images = []
    fft_images_log = []
    for i in range(3):
        rgb_fft = np.fft.fftshift(np.fft.fft2(image[:, :, i]))
        fft_images.append(rgb_fft)
        fft_images_log.append(np.log(abs(rgb_fft)))
    return fft_images, fft_images_log
# Load an image using OpenCV
image_path = 'C:/Users/Hp-D/OneDrive/Desktop/mask2.png'
image = cv2.imread(image_path)
plt.figure(figsize=(15, 5))
# Display the original image
plt.subplot(131)
plt.imshow(cv2.cvtColor(image, cv2.COLOR_BGR2RGB))
plt.title('Original Image')
# Call the rgb_fft function
fft_images, fft_images_log = rgb_fft(image)
# Display the first FFT image
plt.subplot(132)
plt.imshow(np.abs(fft_images[0]), cmap='gray')
plt.title('FFT Image (Channel 1)')
# Display the first Logarithm of FFT Image
plt.subplot(133)
plt.imshow(fft_images_log[0], cmap='gray')
plt.title('Logarithm of FFT Image (Channel 1)')
plt.tight_layout()
plt.show()
```

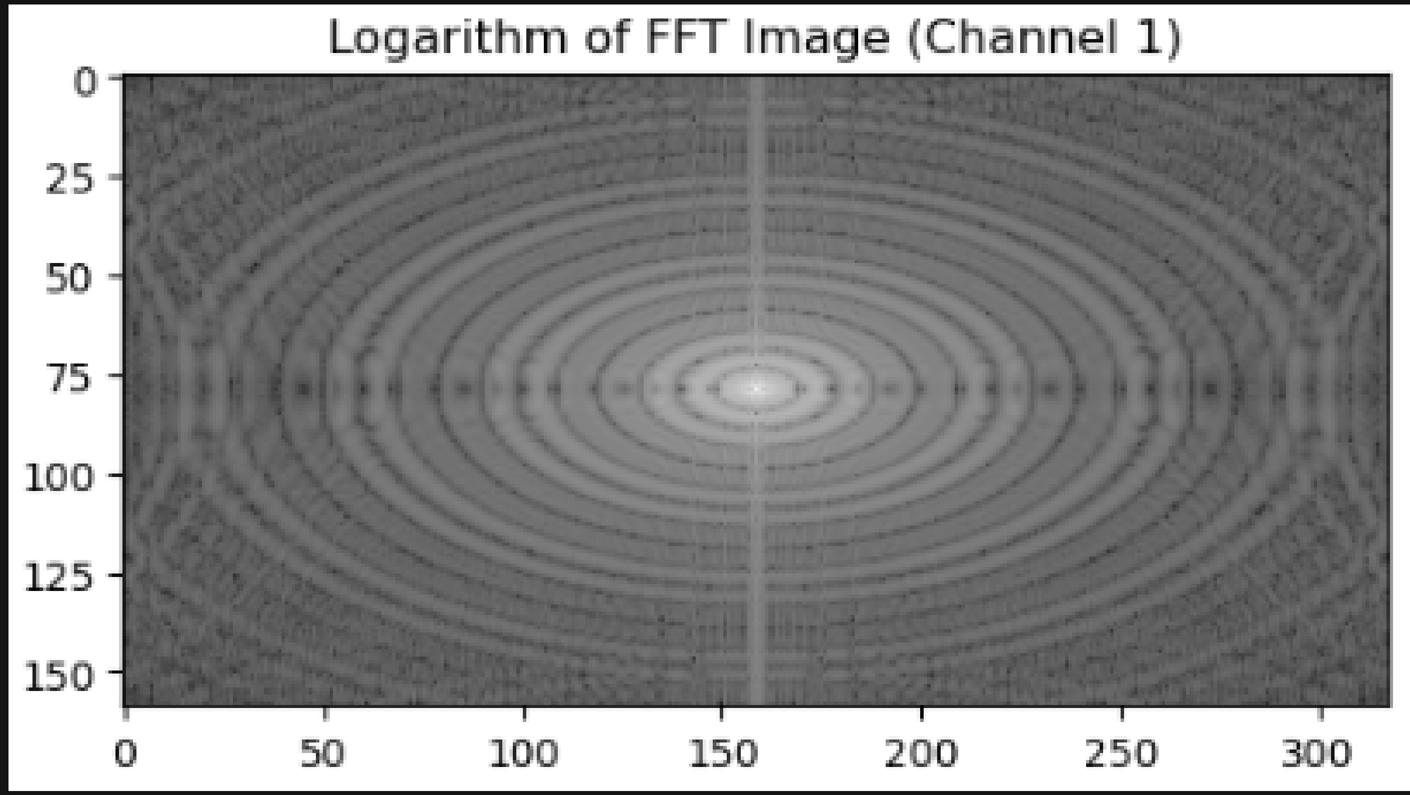
OUTPUT


Original Image
FFT Image (Channel 1)
Logarithm of FFT Image (Channel 1)
Original Image
FFT Image (Channel 1)
Logarithm of FFT Image (Channel 1)

OUTPUT


Original Image
0
25
50
75
100
125
150
0 50 100 150 200 250 300


FFT Image (Channel 1)
0
25
50
75
100
125
150
0 50 100 150 200 250 300


Logarithm of FFT Image (Channel 1)
0
25
50
75
100
125
150
0 50 100 150 200 250 300

Fourier Transform in MRI

Fourier Transform is a fundamental mathematical tool used in Magnetic Resonance Imaging (MRI) to convert the raw data acquired during an MRI scan into a meaningful image. Here's how it's used in the MRI process:

Signal Acquisition: During an MRI scan, the MRI machine emits radiofrequency pulses into the body. These pulses cause the hydrogen nuclei (protons) in the body's tissues to emit signals. These signals are received by the MRI machine's detectors as raw data in the time domain. Each data point represents the amplitude of the signal at a specific time.

Data Sampling: The raw MRI data is sampled at discrete time intervals. This means that the continuous signal received by the detectors is converted into a series of discrete data points.

Fast Fourier Transform (FFT): The next step is to apply the Fast Fourier Transform (FFT) to the sampled data. The FFT is a mathematical algorithm that converts the data from the time domain to the frequency domain. In the frequency domain, the data is represented as a combination of sinusoidal waves at different frequencies. Each frequency component represents different tissue properties in the body.

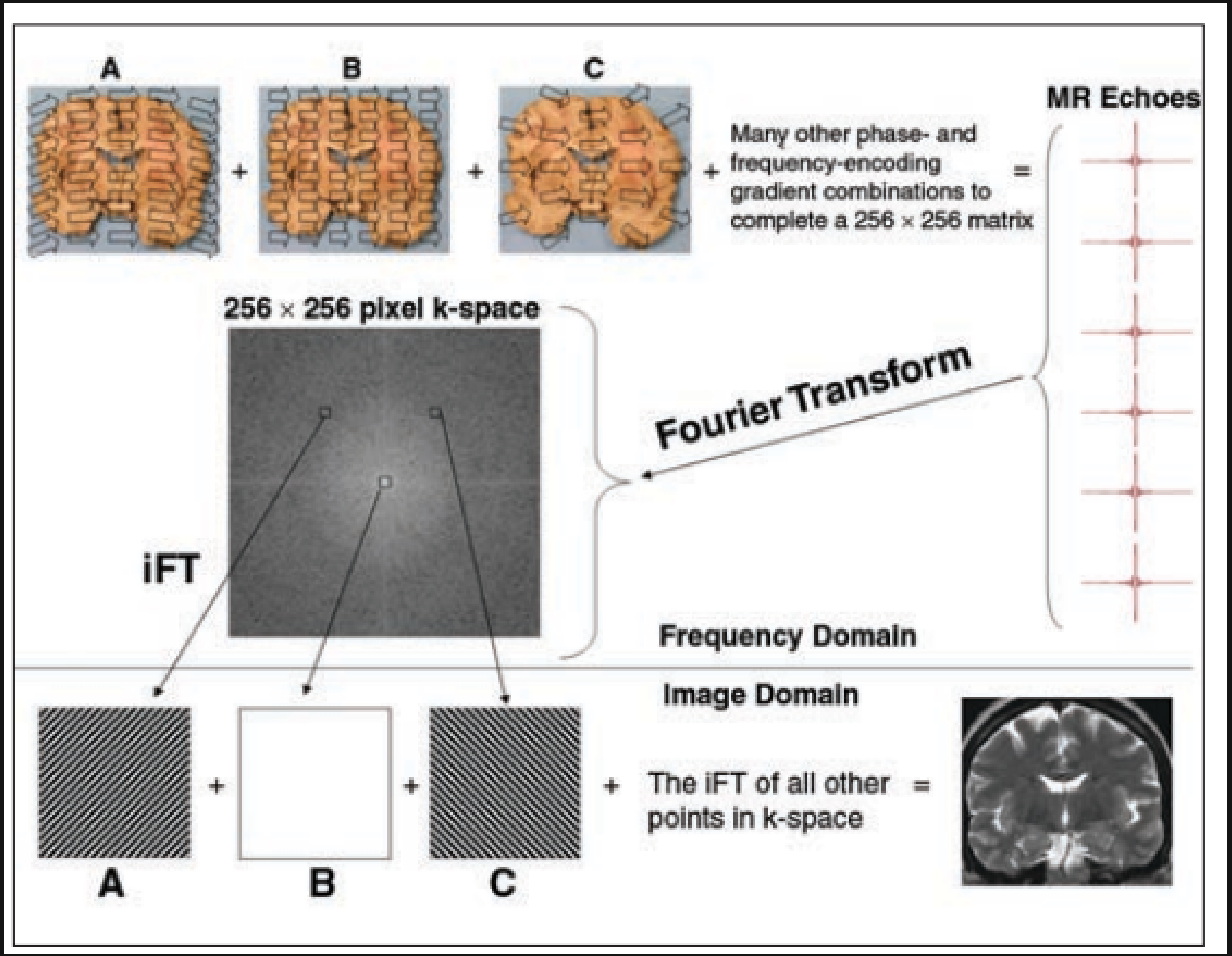
Frequency Spectrum: The result of the FFT is a complex data set that represents the frequency spectrum of the MRI signal. This data contains information about the different frequencies present in the signal and their respective amplitudes and phases.

Spatial Encoding: In MRI, spatial information is also encoded during the scan. This is done using gradient magnetic fields. These gradients create variations in the magnetic field across space. By changing the gradients, different regions of the body are excited and emit signals at different frequencies.

Inverse Fourier Transform: To create an image, the frequency domain data obtained from the FFT is subjected to an inverse Fourier Transform. This process converts the data back from the frequency domain to the spatial domain, producing an image in which each pixel represents a specific location in the body.

I**mage Reconstruction:** Once the data is transformed back into the spatial domain, the MRI scanner can combine this spatial information from different frequency components to create an anatomical image of the scanned area. The varying signal strengths at different locations contribute to the contrast in the final image, allowing us to visualize the internal structures of the body.

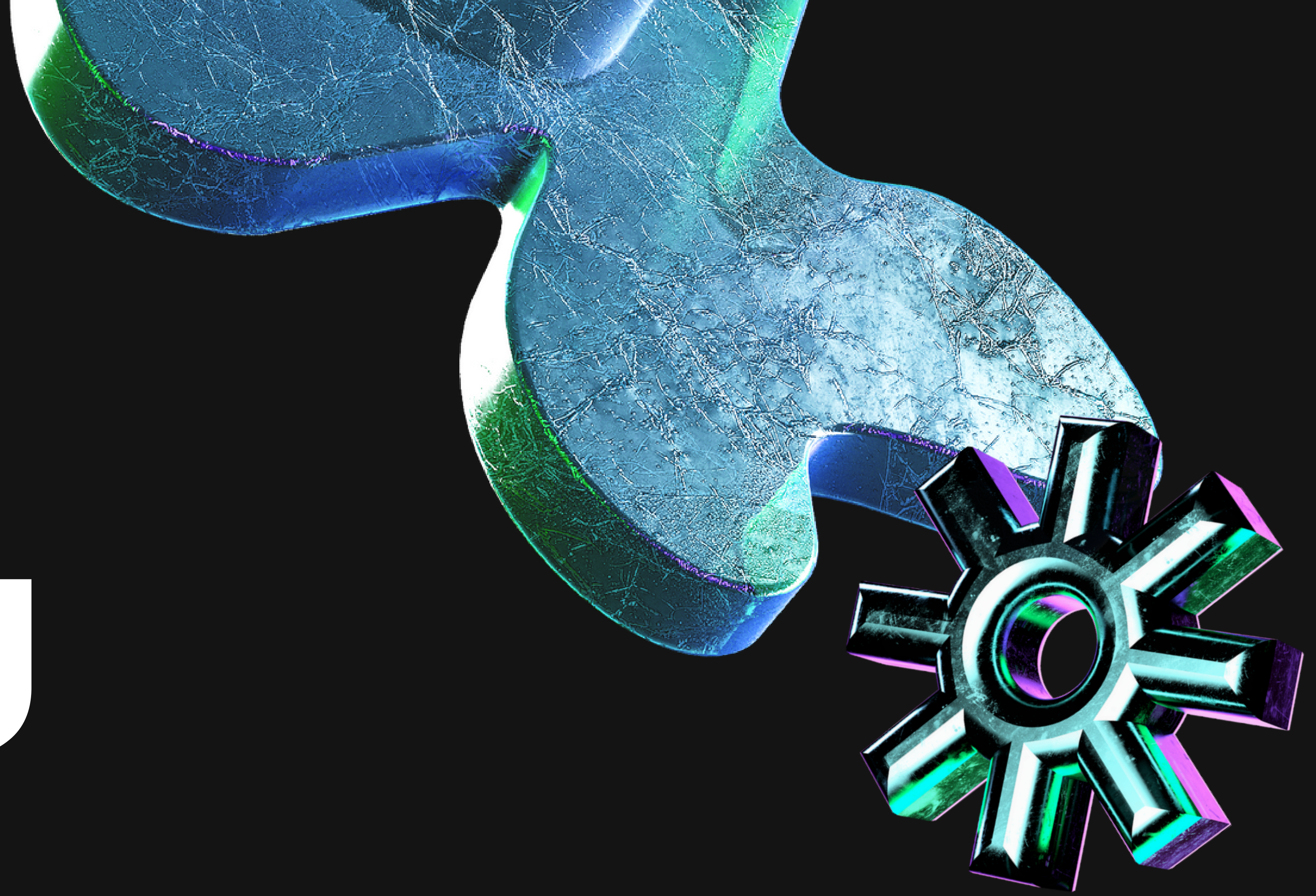
In summary, the Fourier Transform is a crucial part of MRI as it helps convert the raw data acquired in the frequency domain to meaningful images in the spatial domain. This process enables us to create detailed and high-resolution images of the human body's internal structures, making MRI a valuable tool in medical diagnosis and research.



MRI. This coronal slice of a brain is interrogated for all its different spatial frequencies by successively altering magnetic field gradients (open arrows in
top three images) during frequency- and phase-encoding. Although only three examples are shown here, many different gradient combinations are necessary to fill k-space. Inverse Fourier transform (iFT) of k-space essentially adds the relative contributions of all spatial frequencies to give the final image.

References

- https://www.cs.unm.edu/~brayer/vision/fourier.html
- https://hicraigchen.medium.com/digital-image-processing-using-fourier-transform-in-python-bcb49424fd82
- https://byjus.com/maths/fourier-transform/
- https://medium.com/crossml/fourier-transformation-in-image-processing-84142263d734

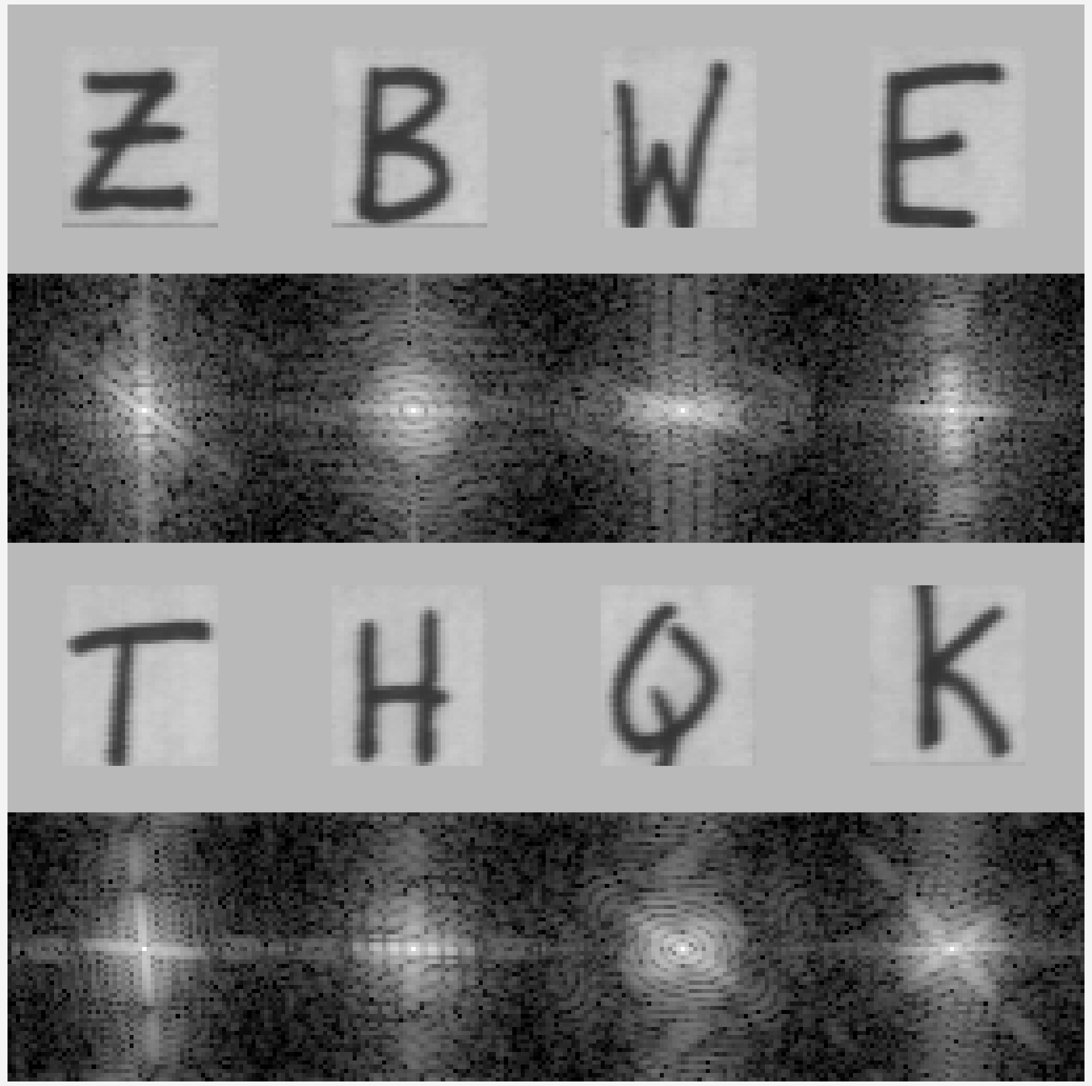


THANK YOU

Presented by:
Anshika Agarwal

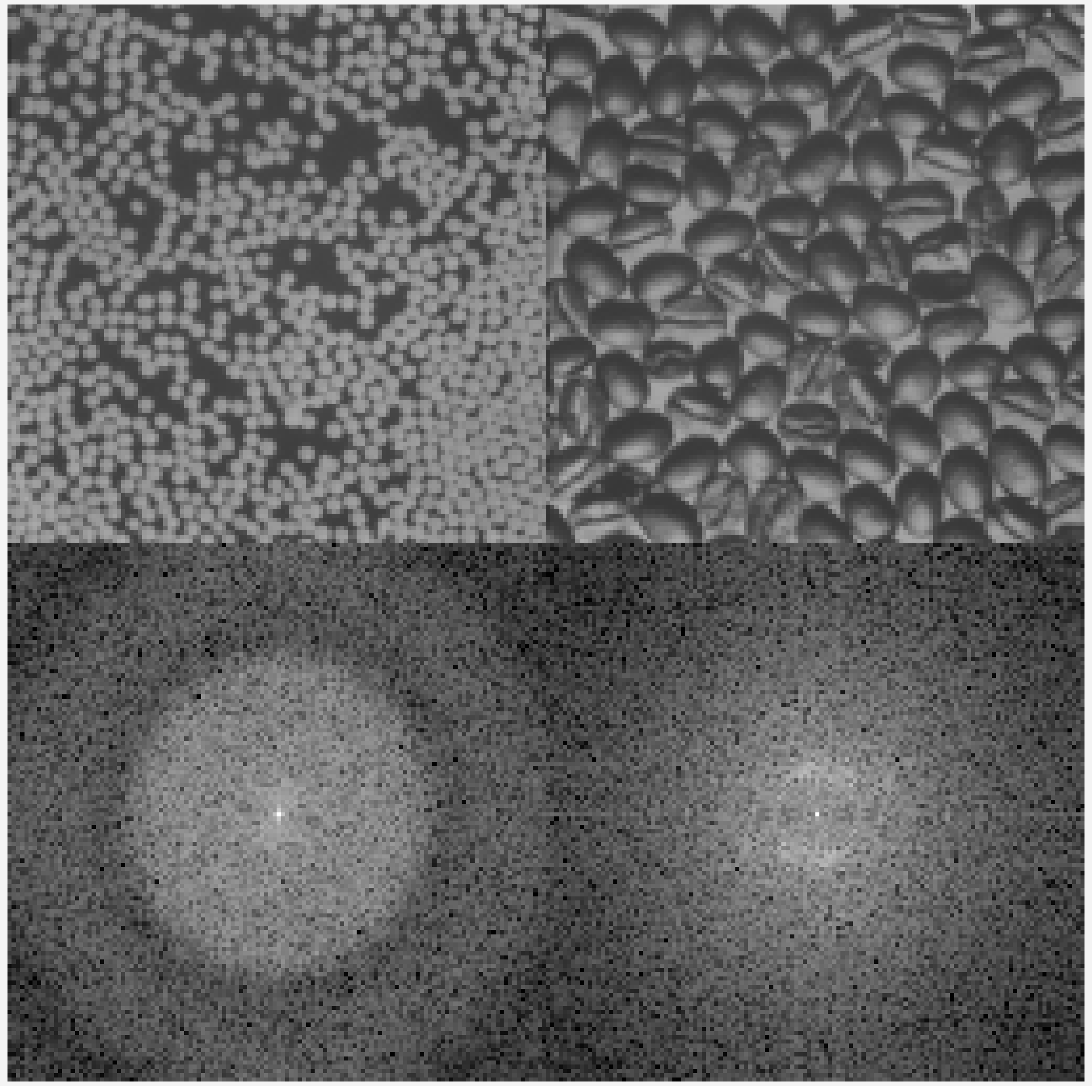
EXAMPLE 5

A bunch of different shapes and their FTs.



Notice that the letters have quite different FTs, especially at the lower frequencies. The FTs also tend to have bright lines that are perpendicular to lines in the original letter. If the letter has circular segments, then so does the FT.

Some collections of similar objects:



The concentric ring structure in the FT of the white pellets image. It is due to each individual pellet. That is, if we took the FT of just one pellet, we would still get this pattern. Remember, we are looking only at the magnitude spectrum. The coffee beans have less symmetry and are more variably colored so they do not show the same ring structure.