HW#3

◘ Question1

◘ Question2

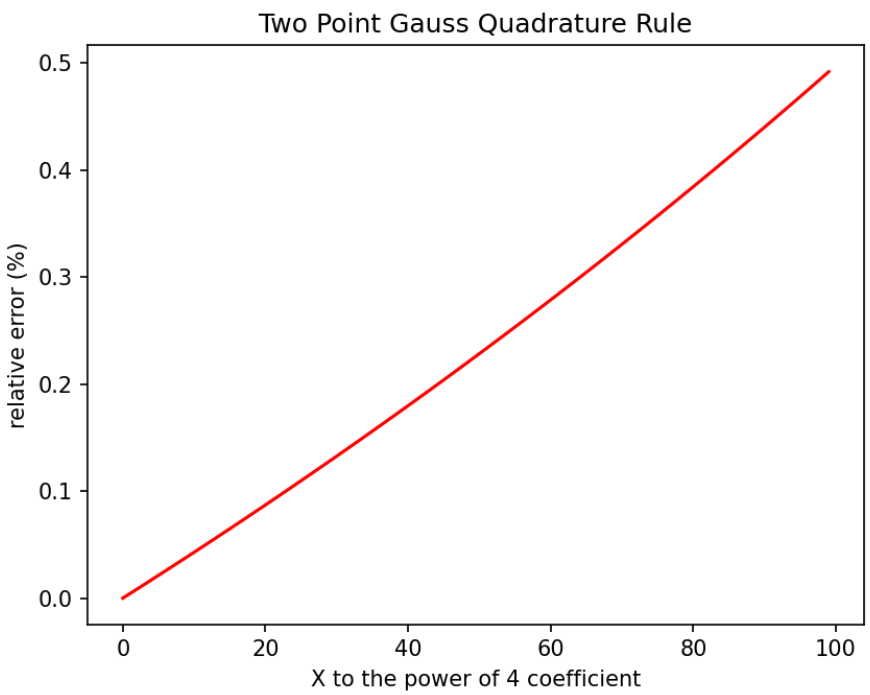
Name: 張嵩禾
StudentID: B083040041

◘ Question1

Compute the definite integral value of the function over [0, 0.8] and the relative error using the two-point Gauss-Legendre formula, which is **3.293615895056088e-14.** It is known that $f1^{(4)}(\xi) = 0$ and indeed, the **relative error is a very small number close to zero**.

```
3.293615895056088e-14
```

Furthermore, since the fourth derivative of f2 is a linear function, I conjecture that the relative error is proportional to the fourth-order coefficient. This result can be seen from the plot.

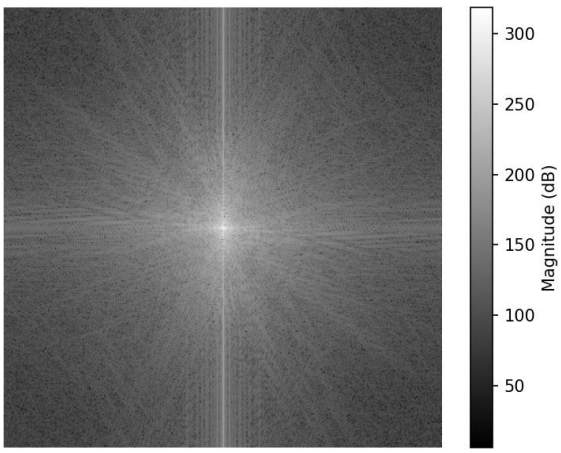


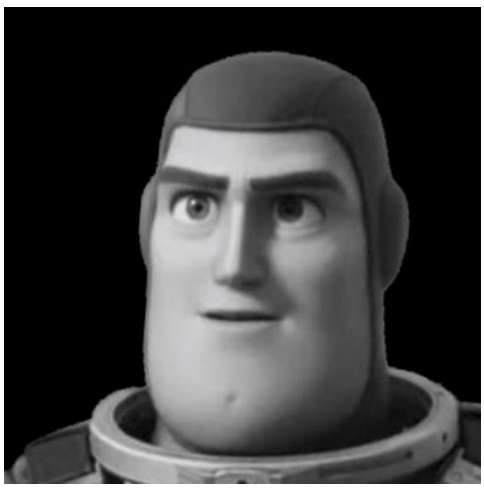
◘ Question2

2a

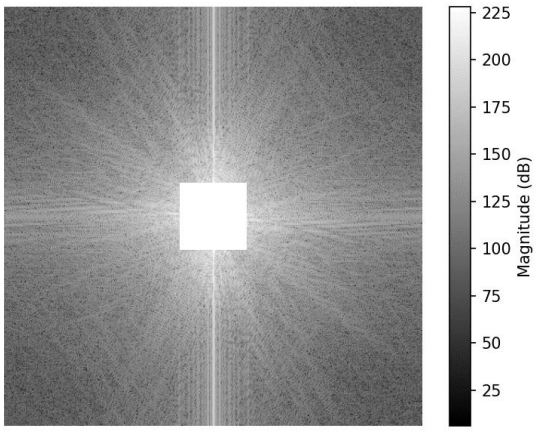
2a and 2b are written in 2a.py. For the high-frequency wave part, I set the range to 30 and delete its signal, resulting in the following effect in the image. **If the High-frequency filtering range smaller, which results in a clearer image.**

Origin image & Spectrum





After HPF

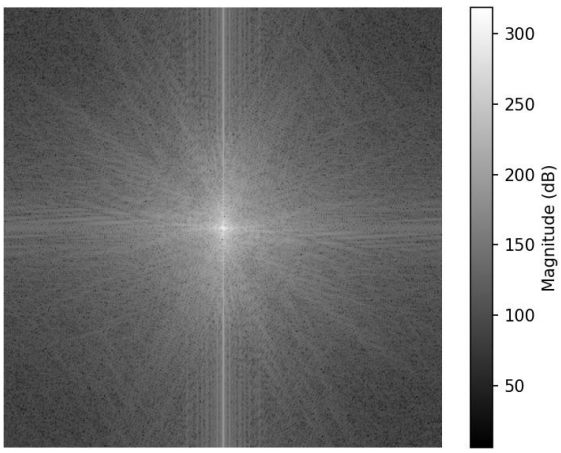


◘ Question2

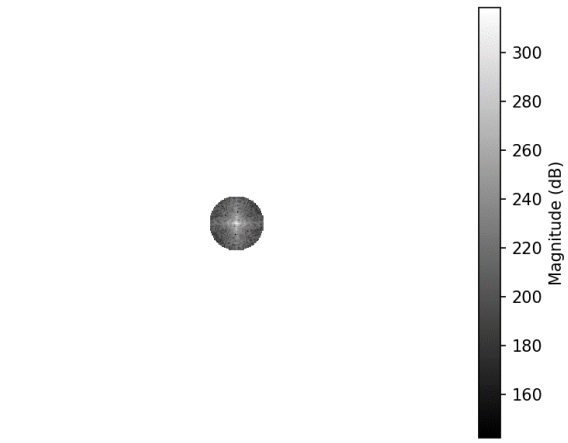
2a

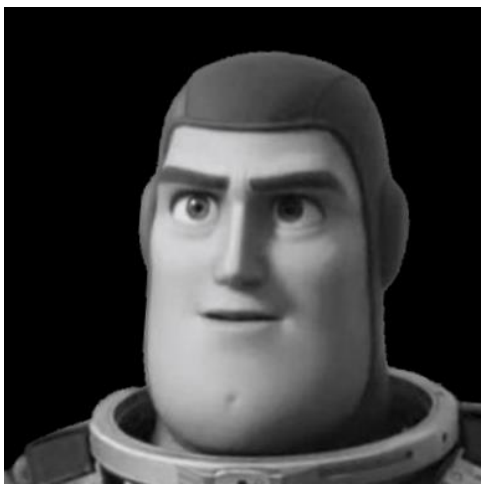
2a written in 2a.py. For the high-frequency wave part, I set the range to 30 and delete its signal, resulting in the following effect in the image. **If we increase the range of low-frequency filter, the image will become clearer.**

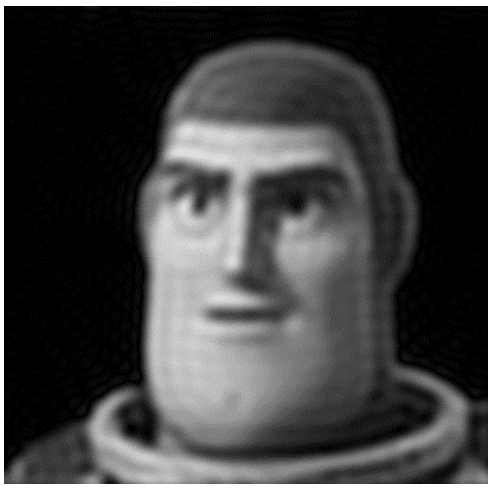
Origin image & Spectrum



After LPF







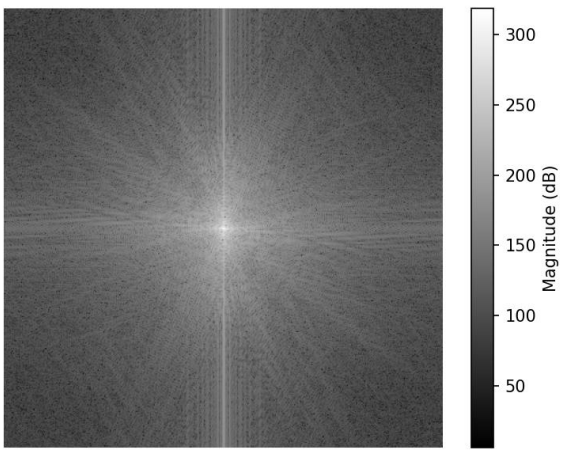
◘ Question2

2b

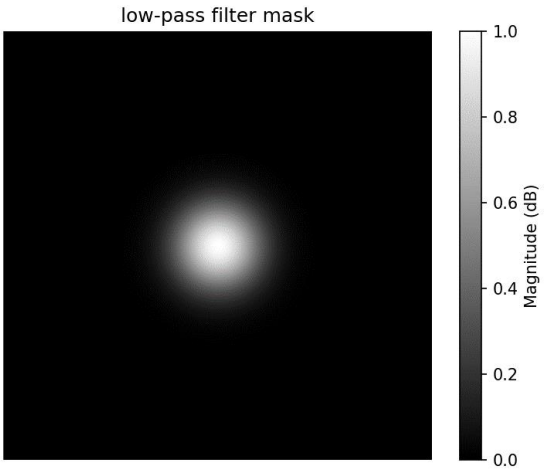
In 2b, we use a Gaussian low-pass filter to process the image, and since FWHM = 64, we can calculate the sigma and perform subsequent processing.

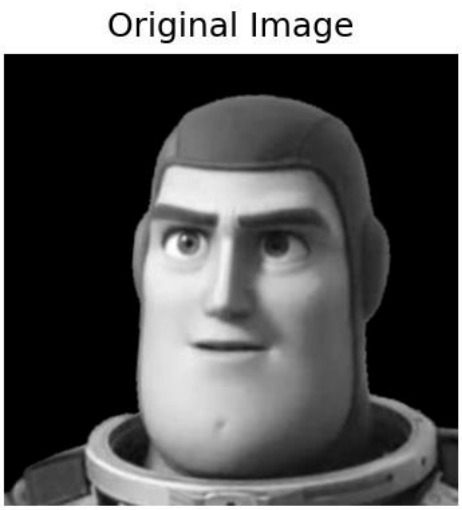
```
sigma = 64 / (2 * np.sqrt(2 * np.log(2)))
```

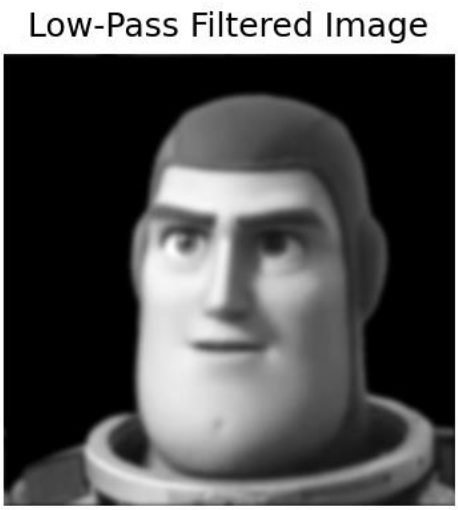
Origin image & Spectrum



After Gaussian LPF







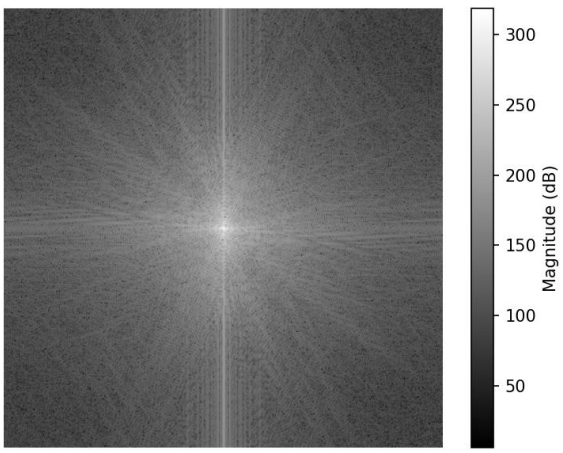
◘ Question2

2c

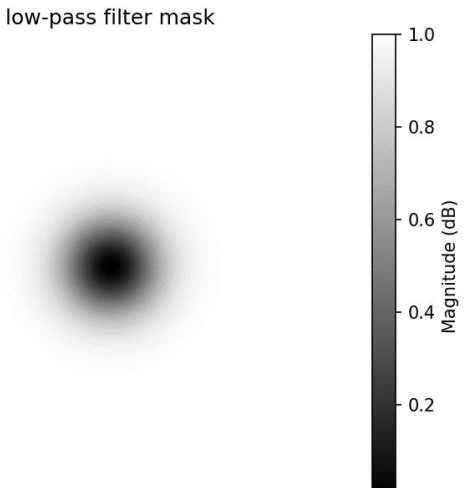
In 2c, we use a Gaussian high-pass filter to process the image, and since FWHM = 64, we can calculate the sigma and perform subsequent processing.

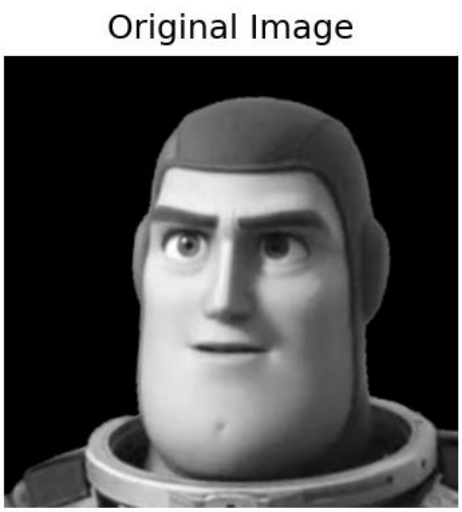
```
sigma = 64 / (2 * np.sqrt(2 * np.log(2)))
```

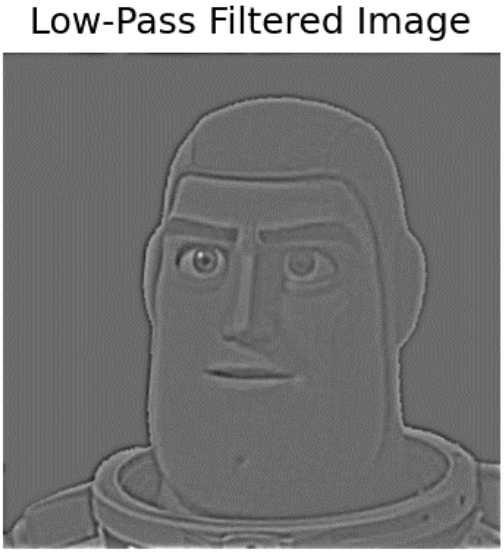
Origin image & Spectrum



After Gaussian LPF





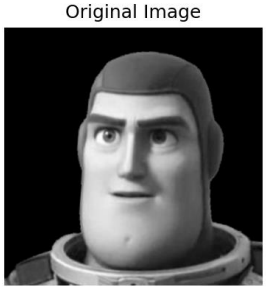


◘ Question2

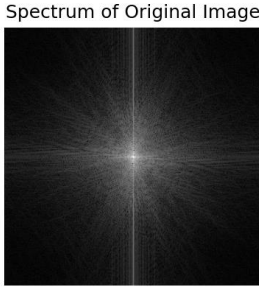
2d

For 2d, we need to combine two images using Gaussian high-pass and low-pass filters to generate the X-Y illusion image**. I set the FWHM of the low-pass filter to 32 and the FWHM of the high-pass filter to 128, with LPF threshold = 50dB and HPF threshold = 25dB**, which can produce a better effect as seen below:

Original Image

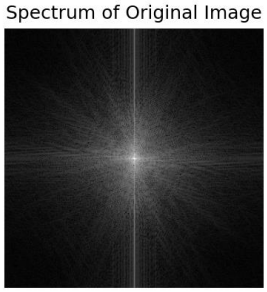


Spectrum of Original Image



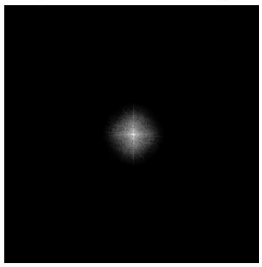
Original Image

Spectrum of Original Image



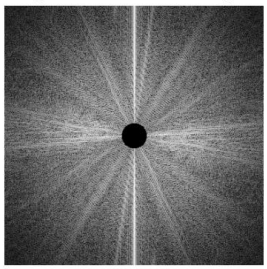
Low-Pass Filtered Image
Threshold: 50 dB

Spectrum of LPF Image



High-Pass Filtered Image
Threshold: 25 dB

Spectrum of HPF Image



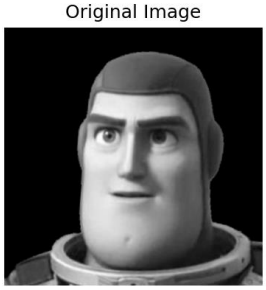
X-Y Illusion

◘ Question2

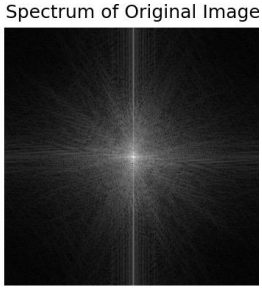
2e

After going through these steps, I discovered that designing an X-Y illusion image requires finding similar images and adjusting the filter threshold and FWHM to achieve the best effect. After several tests, I found that the effect shown in 2d was the best, so I presented this result.

Original Image

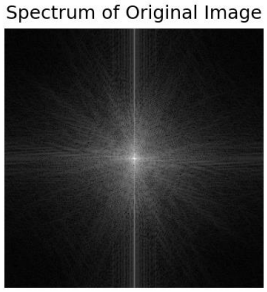


Spectrum of Original Image



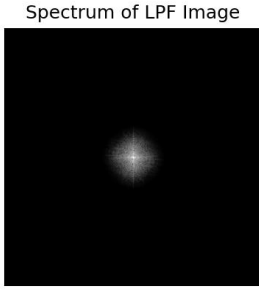
Original Image

Spectrum of Original Image



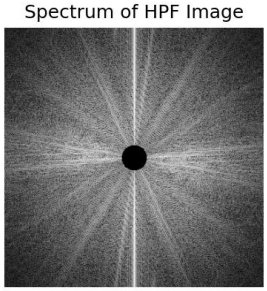
Low-Pass Filtered Image
Threshold: 50 dB

Spectrum of LPF Image



High-Pass Filtered Image
Threshold: 25 dB

Spectrum of HPF Image



X-Y Illusion