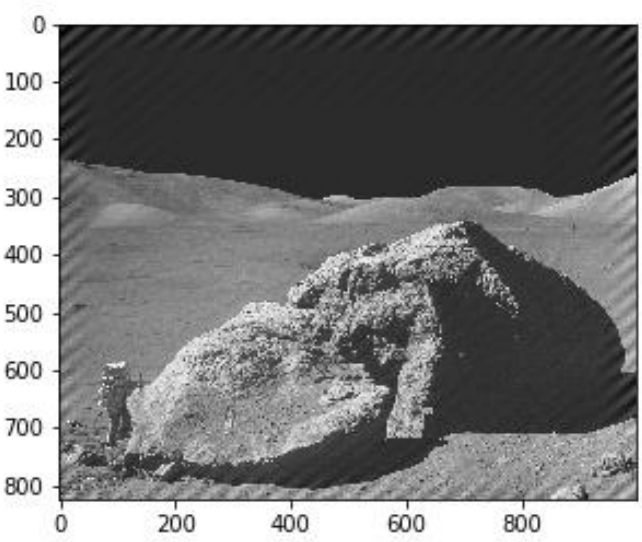
309513061 電信碩 簡詠倫 Ch4_2

1.

Processed image:

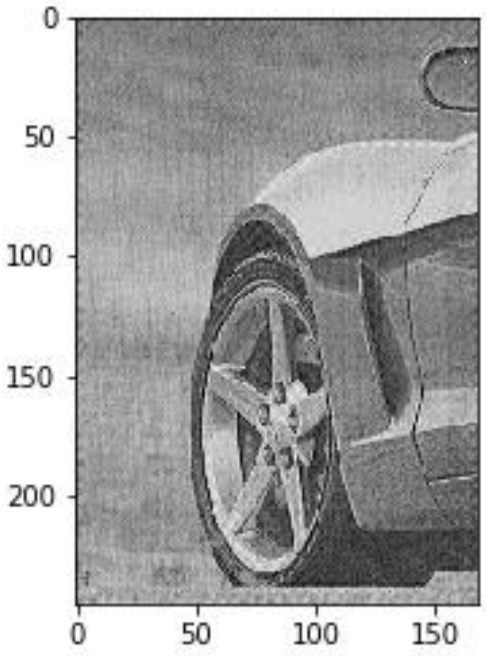


Source code:

```
from PIL import Image
import numpy as np
import matplotlib.pyplot as plt
import cv2
img_as = Image.open('astronaut-interference.tif')
img_as = np.asarray(img_as)
img_as = cv2.copyMakeBorder(img_as,0,img_as.shape[0],0,img_as.shape[1],cv2.BORDER_CONSTANT,value=0)
f = np.fft.fft2(img_as)
fshift_as = np.fft.fftshift(f)
dft_as = np.log(1+np.abs(fshift_as))
points_as = [[774.5,950.5],[874.5,1050.5]]
m = fshift_as.shape[0]
n = fshift_as.shape[1]
for u in range(m):
    for v in range(n):
        for d in range(len(points_as)):
            u0 = points_as[d][0]
            v0 = points_as[d][1]
            d1 = pow(pow(u - u0, 2) + pow(v - v0, 2), 0.5)
            d2 = pow(pow(u + u0, 2) + pow(v + v0, 2), 0.5)
            fshift_as[u][v] *= (1.0/(1+pow((9/d1),4)))*(1.0/(1+pow((9/d2),4)))
f_ishift_as = np.fft.ifftshift(fshift_as)
img_back_as = np.fft.ifft2(f_ishift_as)
img_back_as = np.abs(img_back_as)
img_recon = img_back_as[:int(img_as.shape[0]/2),:int(img_as.shape[1]/2)]
plt.imshow(img_recon,'gray')
plt.savefig('as_recon.jpg')
plt.show()
```

2.

Processed image:

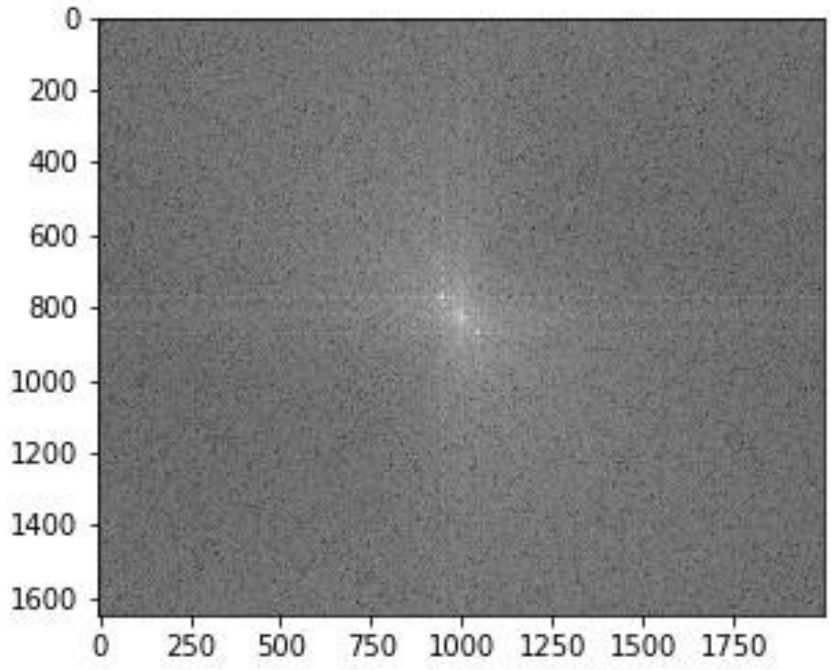


Source code:

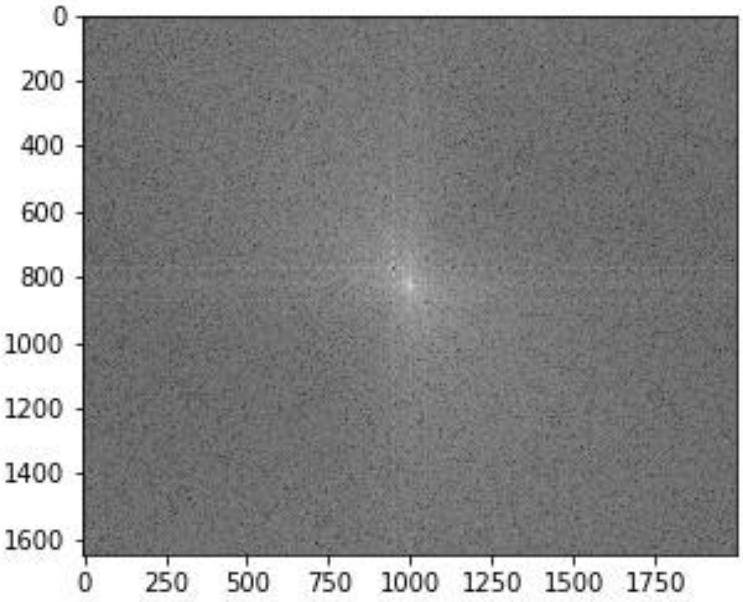
```
img_car = Image.open('car-moire-pattern.tif')
img_car = np.asarray(img_car)
f_car = np.fft.fft2(img_car)
fshift_car = np.fft.fftshift(f_car)
dft_car = np.log(1+np.abs(fshift_car))
points_car = [[81.5,111.5],[40.5,111.5],[44.5,54.5],[85.5,55.5],[165.5,57.5],[206.5,57.5],[161.5,113.5],[202.5,114.5]]
m = fshift_car.shape[0]
n = fshift_car.shape[1]
for u in range(m):
    for v in range(n):
        for d in range(len(points_car)):
            u0 = points_car[d][0]
            v0 = points_car[d][1]
            d1 = pow(pow(u - u0, 2) + pow(v - v0, 2), 0.5)
            d2 = pow(pow(u + u0, 2) + pow(v + v0, 2), 0.5)
            fshift_car[u][v] *= (1.0/(1+pow((9/d1),4)))*(1.0/(1+pow((9/d2),4)))
f_ishift_car = np.fft.ifftshift(fshift_car)
img_back_car = np.fft.ifft2(f_ishift_car)
img_back_car = np.abs(img_back_car)
plt.imshow(img_back_car,'gray')
plt.savefig('car_recon.jpg')
plt.show()
```

3.

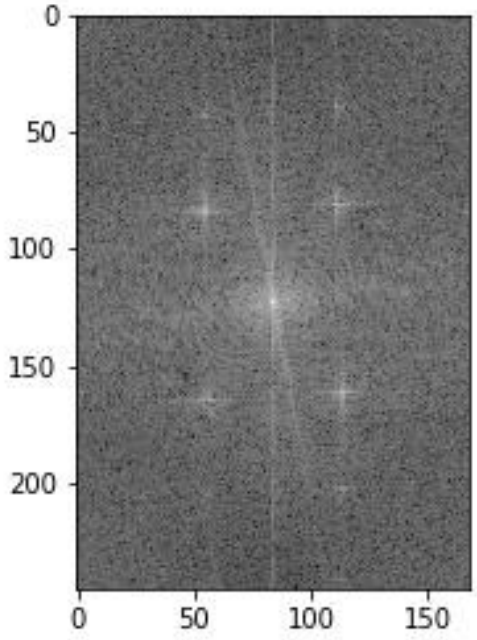
第一題若直接將原圖做 FFT 觀察不太到雜訊的頻譜，因此我將原圖作 zero-padding，長寬變為原本的兩倍再做 FFT，順利觀察到雜訊在頻譜上的兩點(約位於中央兩側)，如下圖所示。



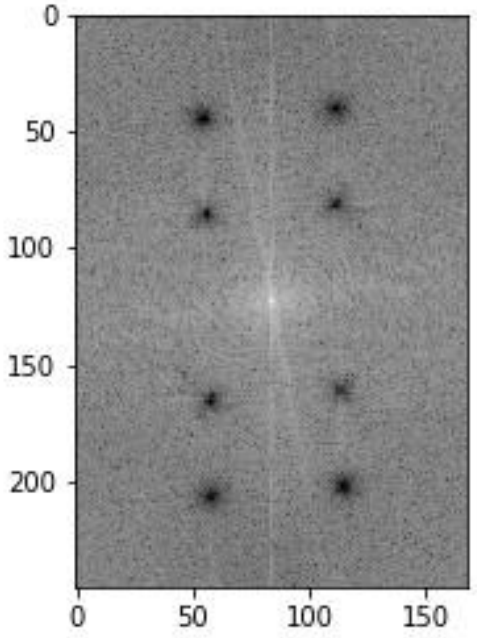
我先用 Matlab 的 imtool 工具確認雜訊點的位置，接下來我使用 4 階巴特沃斯 Notch filter 來過濾雜訊，D0 設為 9。結果如下圖所示，可觀察到原本的雜訊點確實被濾除，第一題結果圖也顯示大部分弦波雜訊被過濾掉。



第二題直接將原圖做 FFT 能發現明顯的脈衝串，如下圖所示。



我一樣選擇 4 階巴特沃斯 Notch filter 來衰減這些脈波串，D0 設為 9，結果圖如下。



可發現脈衝串點被濾波器濾除，由第二題的結果圖也能發現圖片確實有被改善。