

電腦視覺 Take Home exam

系級:資科所碩一 學號:108753208 姓名:葉冠宏

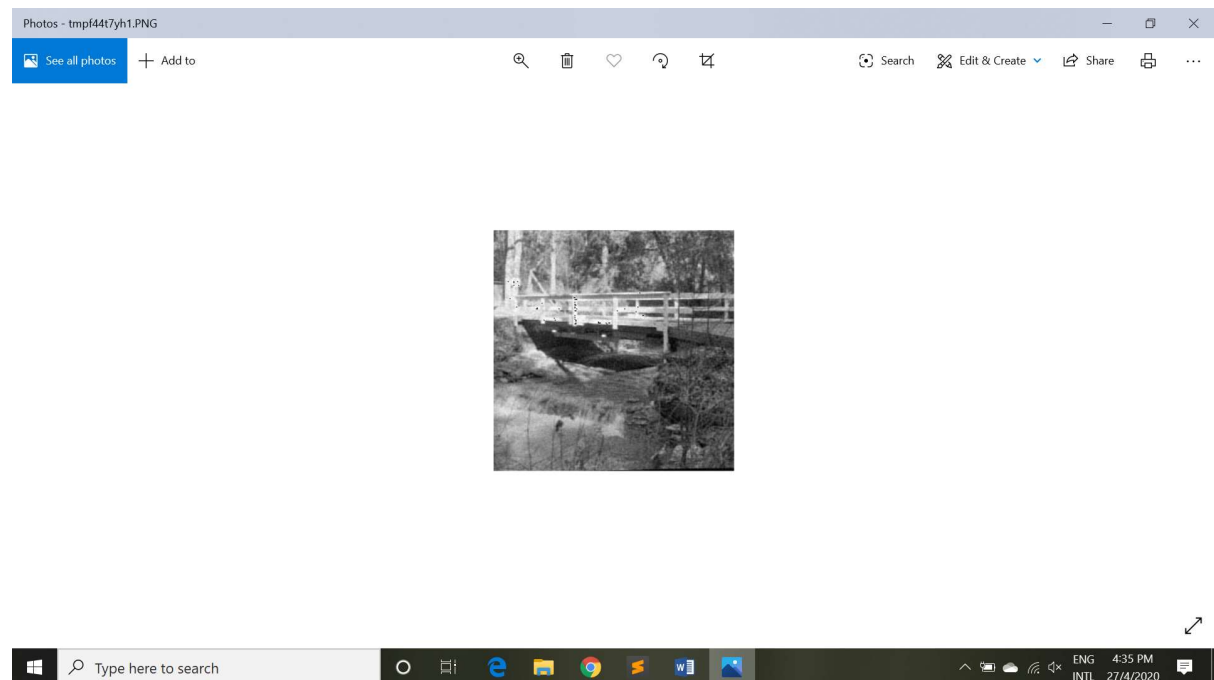
1.

Original image and frequency domain after Fourier transform



(a)

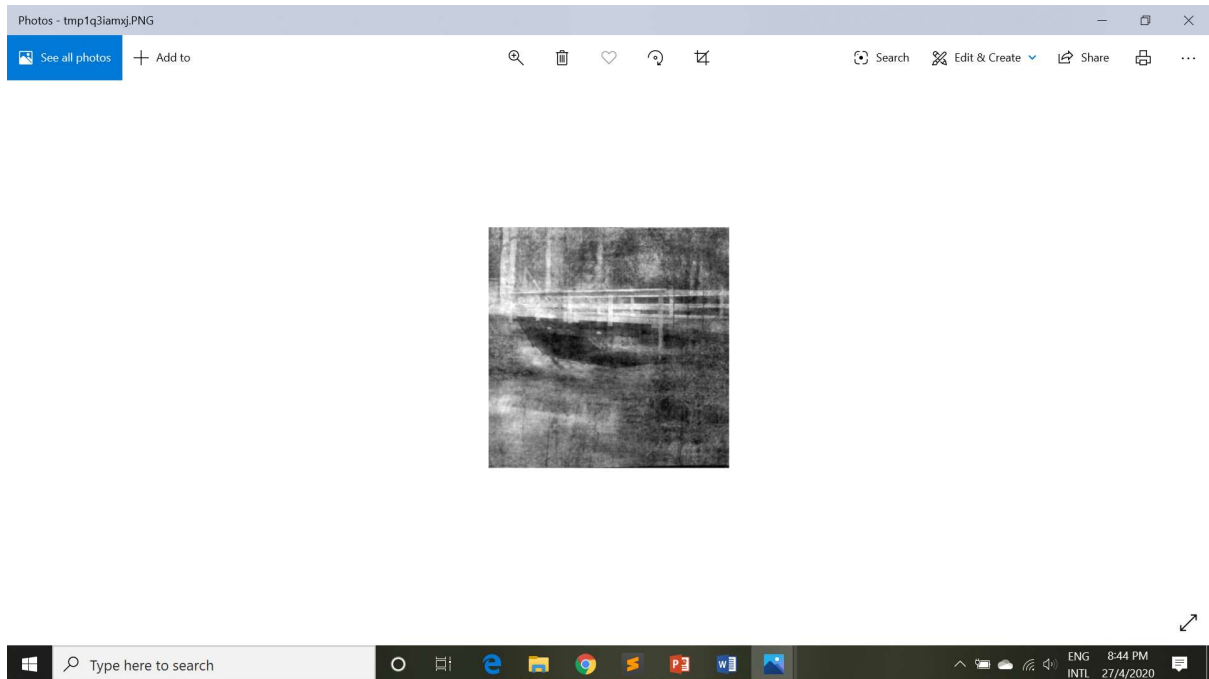
Image A: (Keeping only $\frac{1}{4}$ largest coefficients, and perform inverse transform)



可以看到影像變化不大，但有些躁點

(b)

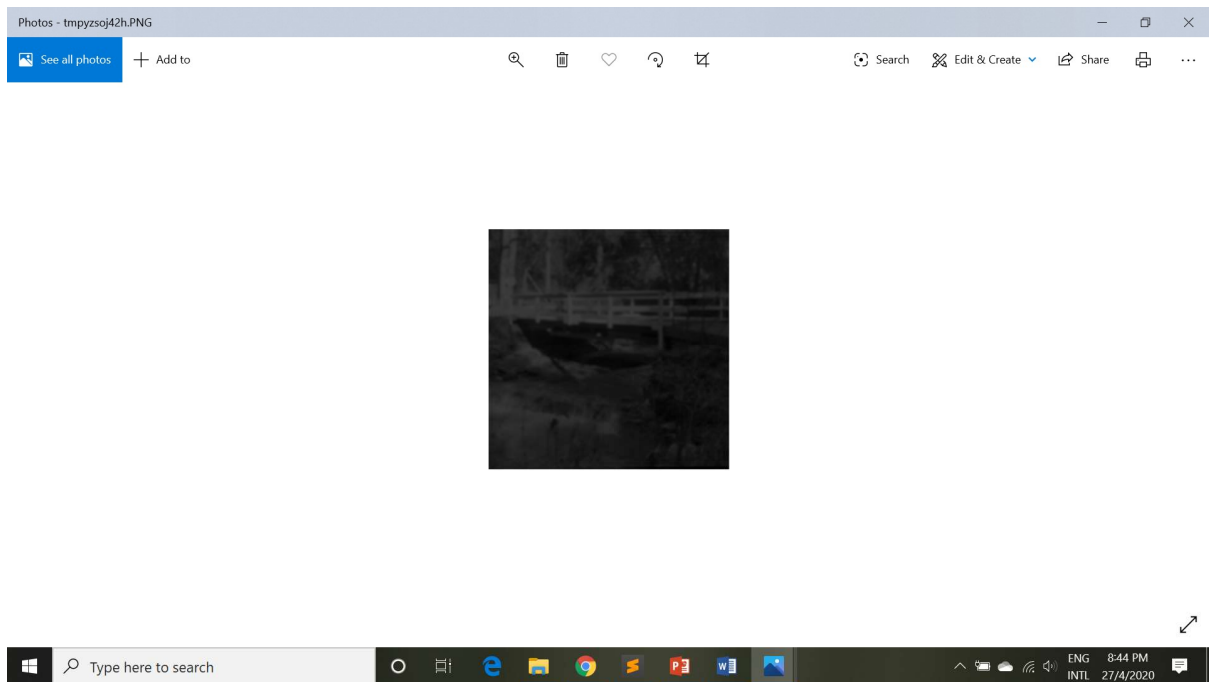
Image B: (Divide the image into 256 16x16 blocks. Perform Fourier transform on each block, keeping only $\frac{1}{4}$ largest coefficients in each block, and take inverse transform)



可以看到影像變模糊，且多了需多灰色朦朧

(c)

Image C: (Reduce the size of the original image to 128x128 by subsampling. Perform Fourier Transform on the 128x128 image. Zero-pad the coefficient matrix to a 256x256 array. Then perform inverse transform)



可以看到圖案變得很暗

(d)

Mean square error:

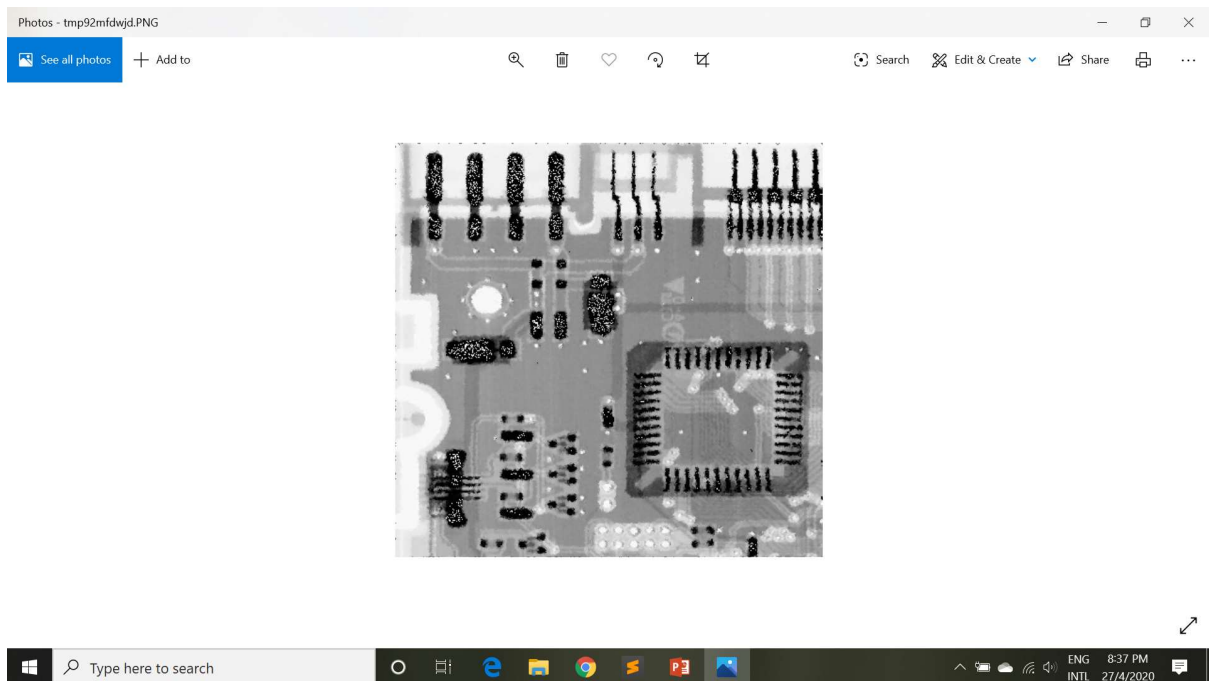
```
C:\Users\chrystal212\Desktop\pp\problem1.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

problem1.py
48 plt.subplot(121),plt.imshow(img, cmap = 'gray')
49 plt.title('Input Image'), plt.xticks([], plt.yticks([]))
50 plt.subplot(122),plt.imshow(magnitude_spectrum, cmap = 'gray')
51 plt.title('Magnitude Spectrum'), plt.xticks([], plt.yticks([]))
52 plt.show()
53
54
55 MASK=filt(magnitude_spectrum)
56 res=np.multiply(fshift,MASK)
57
58 f_ishift = np.fft.ifftshift(res)
59 img_back = np.fft.ifft2(f_ishift)
60 img_back = np.abs(img_back)
61
62 new_im = Image.fromarray(img_back)
63 new_im.show(title="Image A")
64 cv2.waitKey(0)
65
66 #####
67

Mean square error between image and image A: 80.72322571998994
Mean square error between image and image B: 314.9300294007736
Mean square error between image and image C: 9042.411695864123
[Finished in 8.0s]
```

如圖為 image 和 image A,B,C 間의 MSE 差。C 最大，A 最小

2. Implement the adaptive medium filter

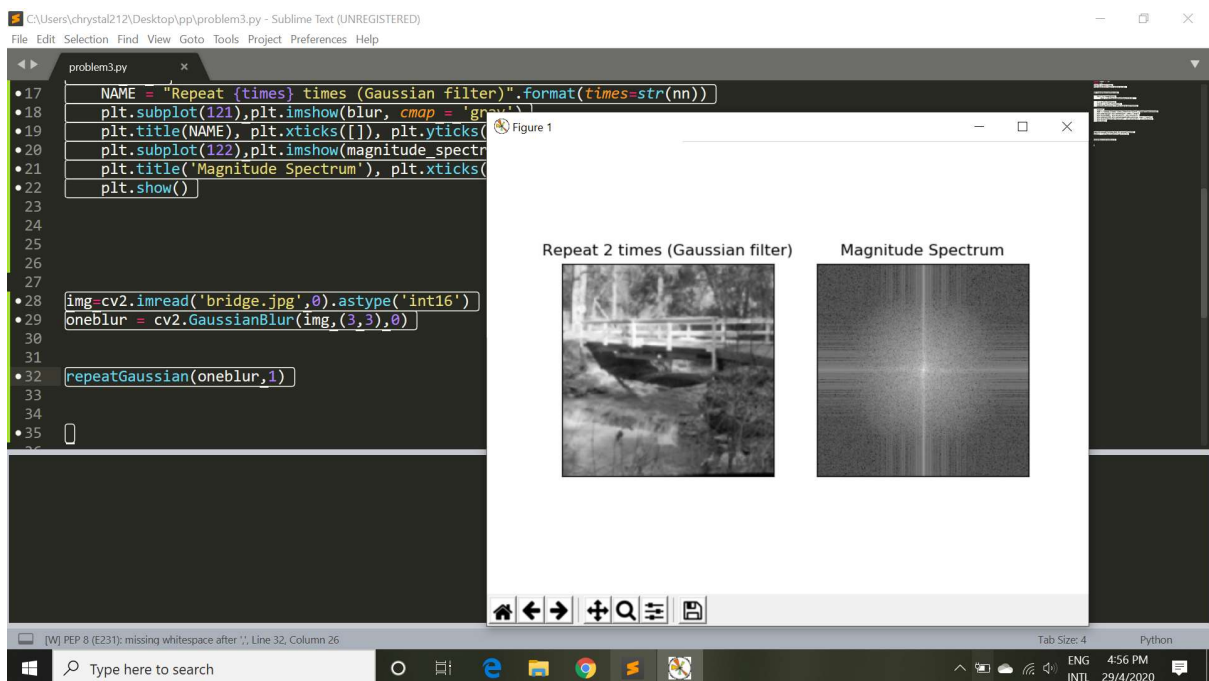


呈現結果如上，可以看到的確噪點消除了許多，變平滑了。但還是有些許噪點在細處。

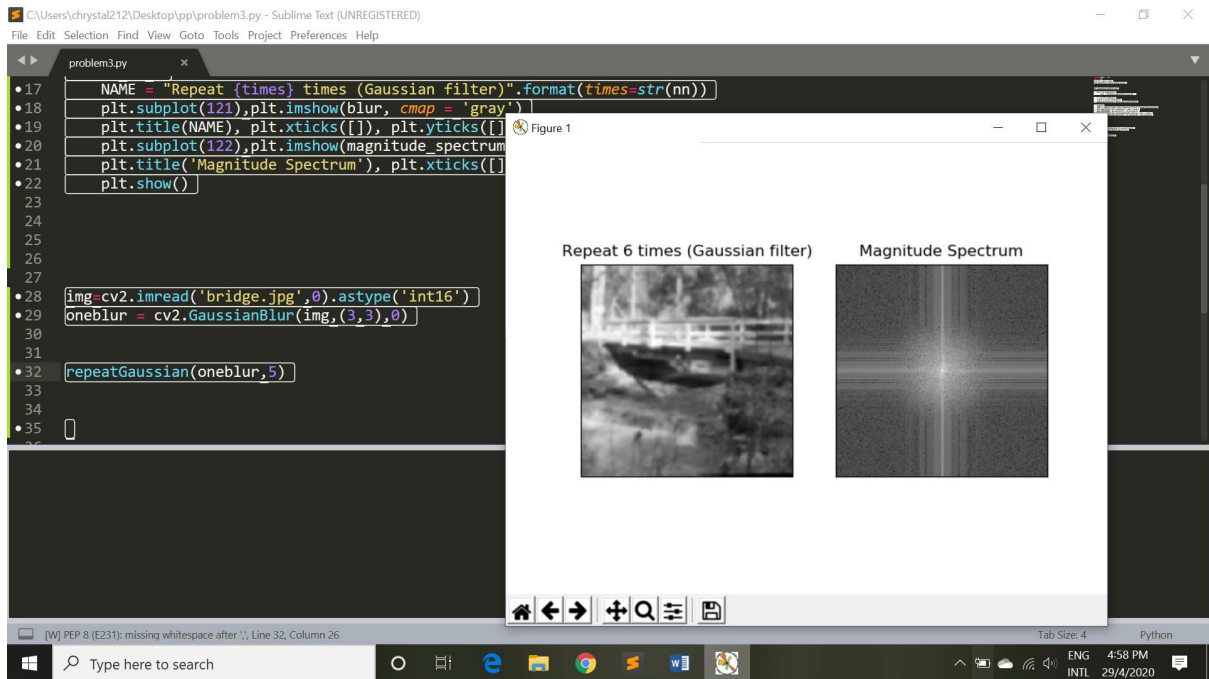
3.

(a)

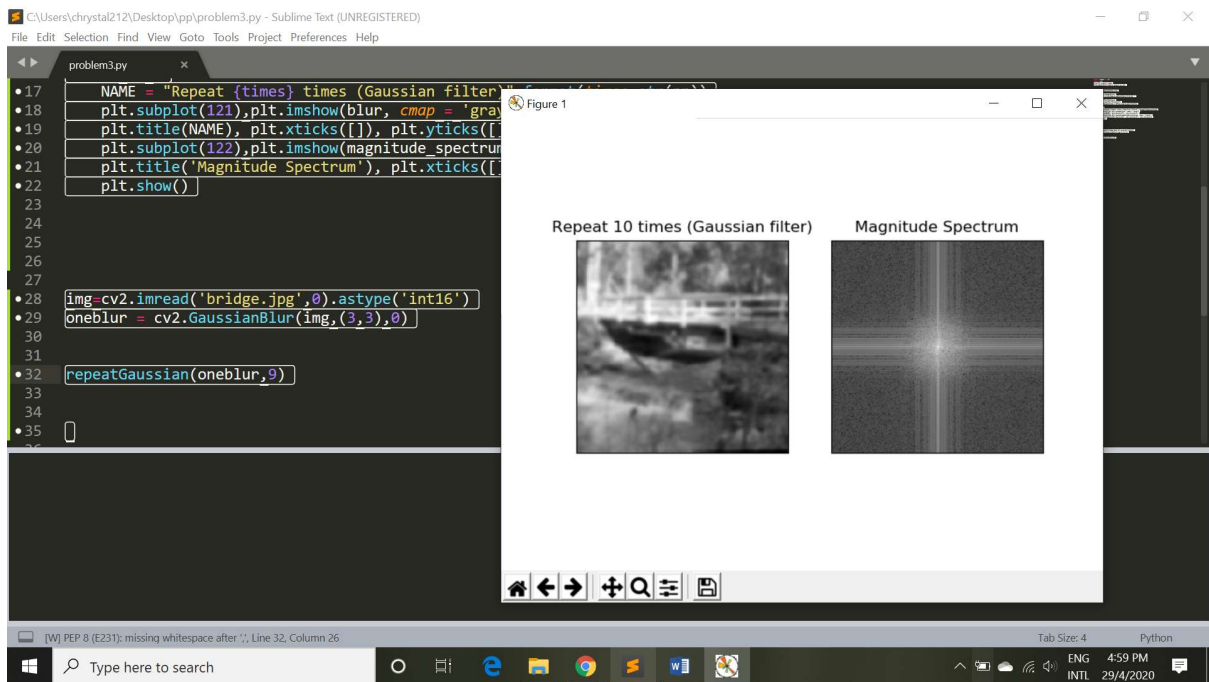
Repeat Gaussian low-pass filter:



可以看到，因為 Gaussian 是 low-pass filter，所以中間亮很多。



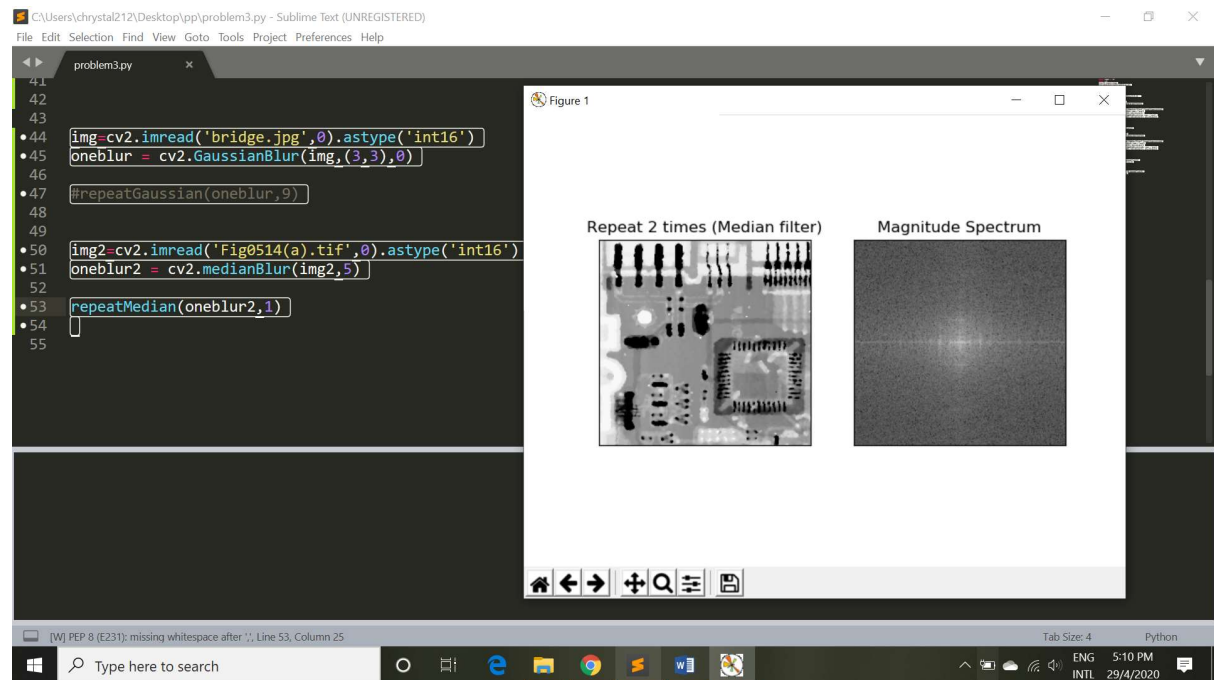
在重複 6 次 Gaussian filter 時，可以看到影像模糊很多，而高頻的部分感覺有變少。



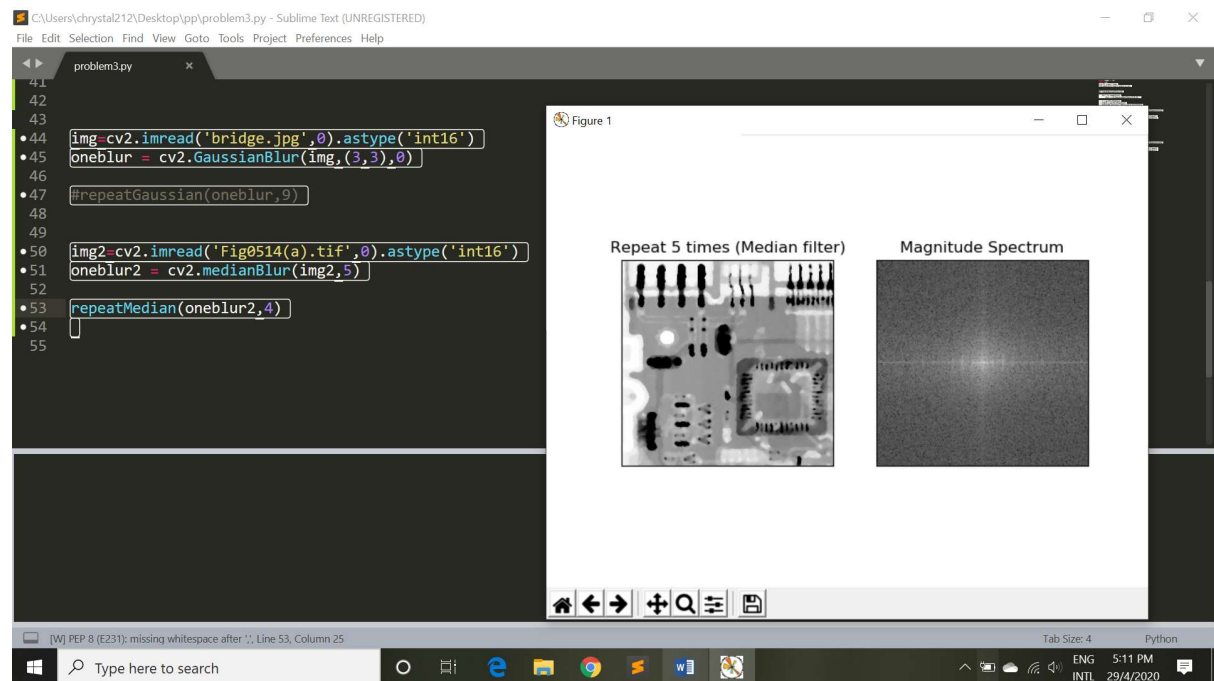
在做 10 次 Gaussian filter 時，可以看到和做 6 次時不太有明顯差距。可能是因為平滑到一定程度後函式有收斂。

(b)

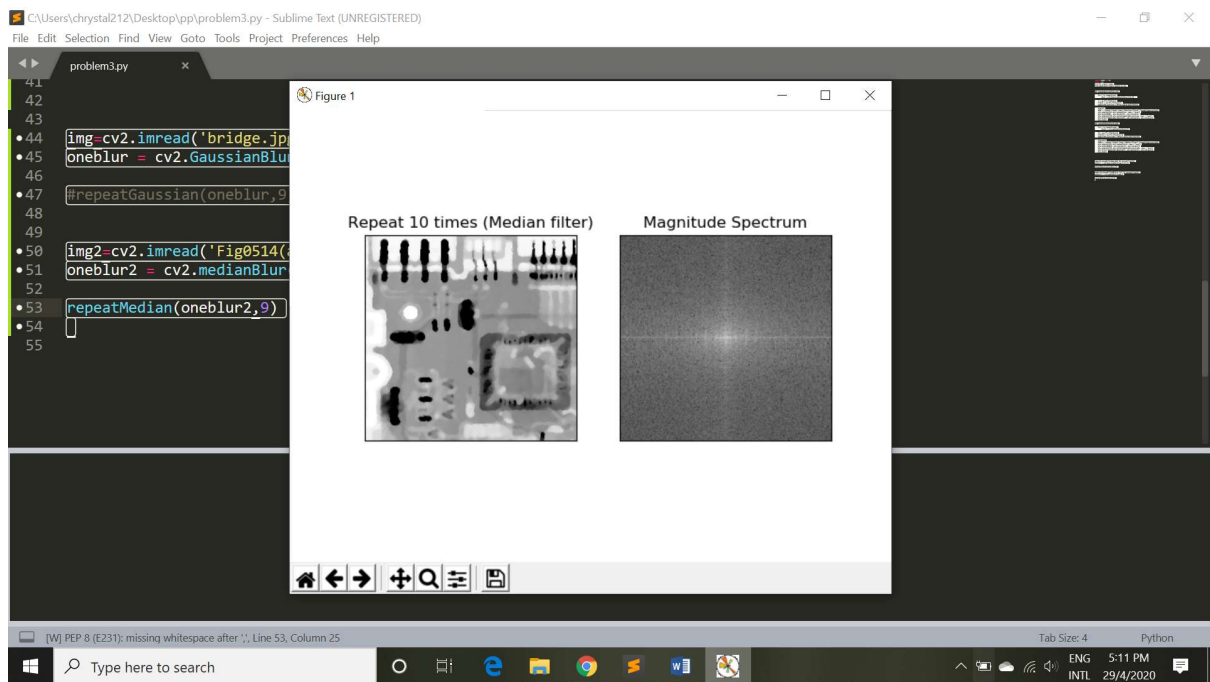
Repeat median filter



可以看到在做兩次時就可以有消除所有雜點，而中間低頻的部分感覺比高頻多一點點。



在做 5 次時從 magnitude spectrum 來看感覺和做兩次時差不多，而圖案則感覺有飽和一點。



做 10 次時感覺和 5 次時相差不大，而似乎有些細節被削掉了。

Frequency domain analysis 可以用 median filter，但可能要先把 Fourier transform 後的複數座標取其絕對值做值的轉換再處理。

4.

$$f(x', y') = \sum_{i=0}^M \sum_{j=0}^N \delta(x' - x_i, y' - y_j) f(x_i, y_j)$$

$$\Rightarrow f(T_1 k, T_2 k) = \sum_{i=0}^M \sum_{j=0}^N \delta(T_1 k - T_1 i, T_2 k - T_2 j) f(T_1 i, T_2 j)$$

$$\Rightarrow G(u, v) = F(u, v) H(u, v) + N(u, v)$$

$$\frac{G(u, v) - N(u, v)}{F(u, v)} = H(u, v)$$

$$H(u, v) = \frac{\left(\frac{T_1 + T_2}{2}\right)}{\pi(u_k + v_k)} \sin[\pi(u_k + v_k)] e^{-j\pi(u_k + v_k)}$$