

程式網址：

<https://colab.research.google.com/drive/1QKoiwRpuluWYVsBDisdDasB411lh17Ba?usp=sharing>

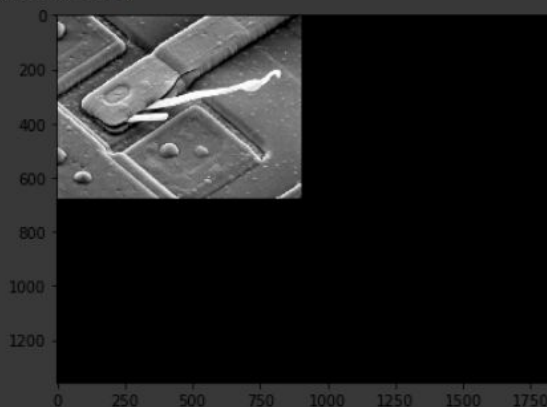
```
[ ] 1 import skimage.io as io
    2 import numpy as np
    3 import matplotlib.pyplot as plt
```

程式所用到的函式庫。

```
1 # (b) Padded image, fp of size PxQ
2 P = 2 * M
3 Q = 2 * N
4 img_b = np.zeros((P, Q), dtype = img.dtype)
5 for i in range(M):
6     for j in range(N):
7         img_b[i][j] = img[i][j]
8 io.imshow(img_b)
9 print(img_b.shape)
10 print("{} {}".format(P, Q))
11 #print(img_b.dtype)
```

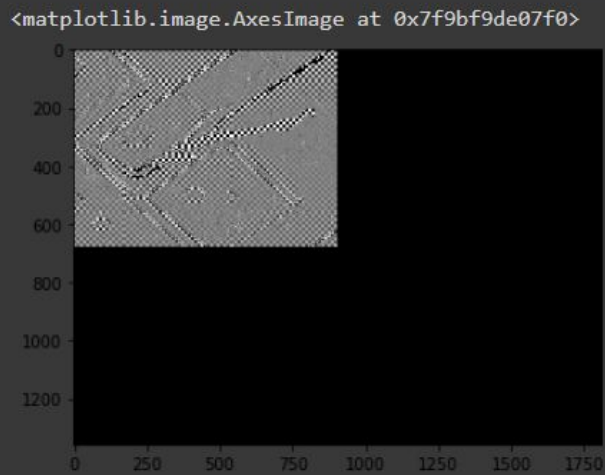
(1356, 1812)

(1356, 1812)



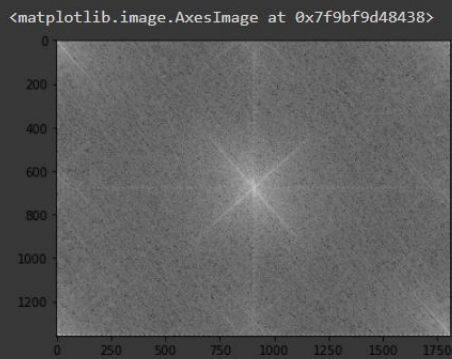
此為圖片(b)及程式碼。

```
[ ] 1 # (c) Result of multiplying fp by  $(-1)^{(x+y)}$ 
2 img_c = np.copy(img_b)
3 for x in range(P):
4     for y in range(Q):
5         k = pow(-1, (x + y))
6         img_c[x][y] = img_c[x][y] * k
7 io.imshow(img_c)
```



這是圖片(c)及其程式碼，將圖片(b)的每一個內容乘以-1的(x+y)次方。

```
[ ] 1 # (d) Spectrum of F
2 # Compute the DFT,  $F(u,v)$ , of the image from (c)
3 img_fft = np.fft.fft2(img_c)
4 #print(img_fft.dtype)
5 # fft結果為複數，其絕對值結果為振幅
6 img_d = np.log(1 + np.abs(img_fft))
7 img_d = 255 * (img_d - img_d.min()) / (img_d.max() - img_d.min())
8 img_d = img_d.astype(img.dtype)
9 io.imshow(img_d, cmap=plt.cm.gray)
```

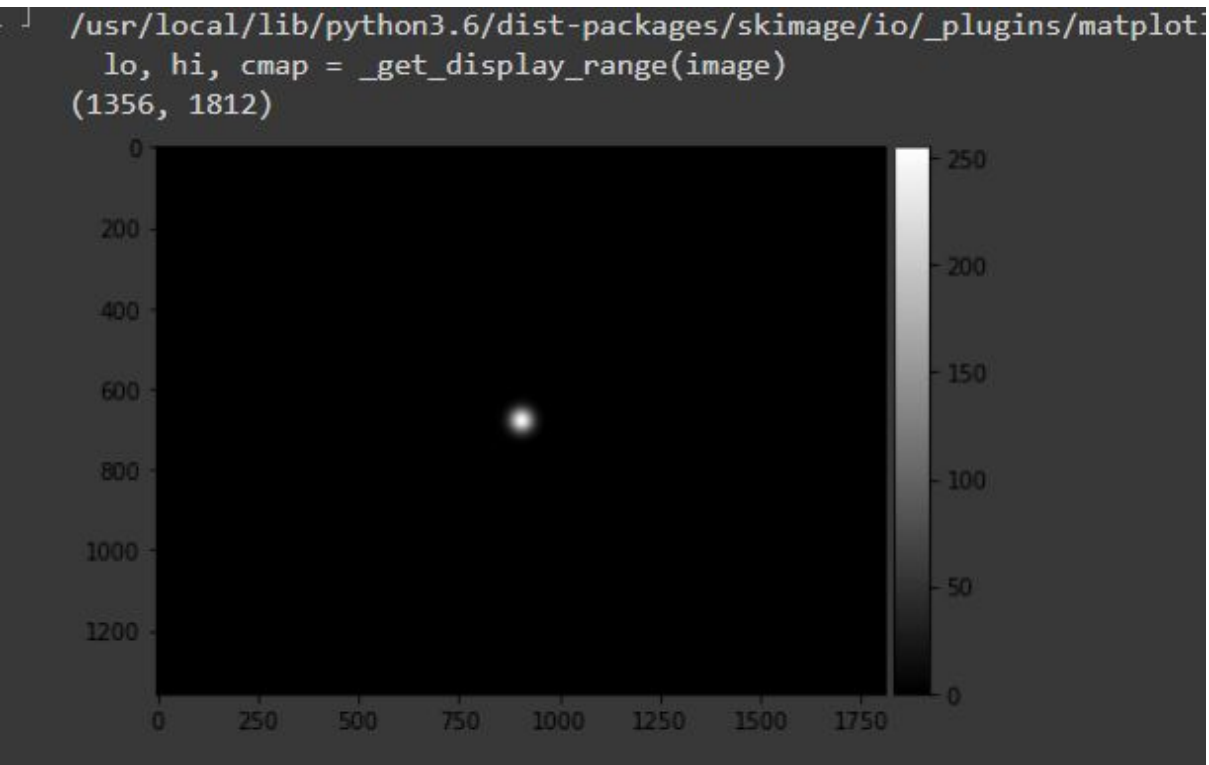


這是圖片(d)，先給圖片(c)做傅立葉轉換，然後秀出它的spectrum。

```

1 # (e) Centered Gaussian lowpass filter transfer function, H, of size PxQ
2 sigma = 20
3 x, y = np.mgrid[-(M):M, -(N):N]
4 # Construct Gaussian low pass filter
5 gaussian_filter = np.exp(-(x**2 + y**2) / (2 * (sigma**2)))
6 gaussian_filter = gaussian_filter / gaussian_filter.sum()
7 gaussian_filter = 255 * (gaussian_filter - gaussian_filter.min()) / (gaussian_filter.max() - gaussian_filter.min())
8 io.imshow(gaussian_filter, cmap=plt.cm.gray)
9 print(gaussian_filter.shape)

```



這是圖片(e)及其程式碼，參考網路上找來的方法做出Gaussian lowpass filter。

```

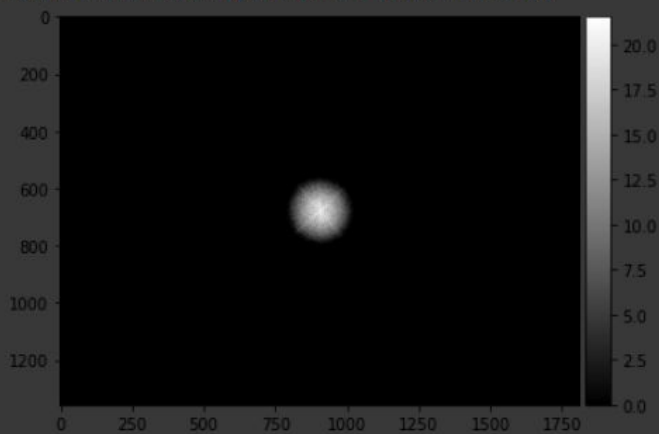
1 # (f) Spectrum of the product HF
2 img_HF = np.empty(img_fft.shape, dtype = img_fft.dtype)
3 gaussian_filter = gaussian_filter.astype(img_fft.dtype)
4 for i in range(P):
5     for j in range(Q):
6         img_HF[i][j] = img_fft[i][j] * gaussian_filter[i][j]
7 img_f = np.log(1 + np.abs(img_HF))
8 io.imshow(img_f, cmap=plt.cm.gray)

```

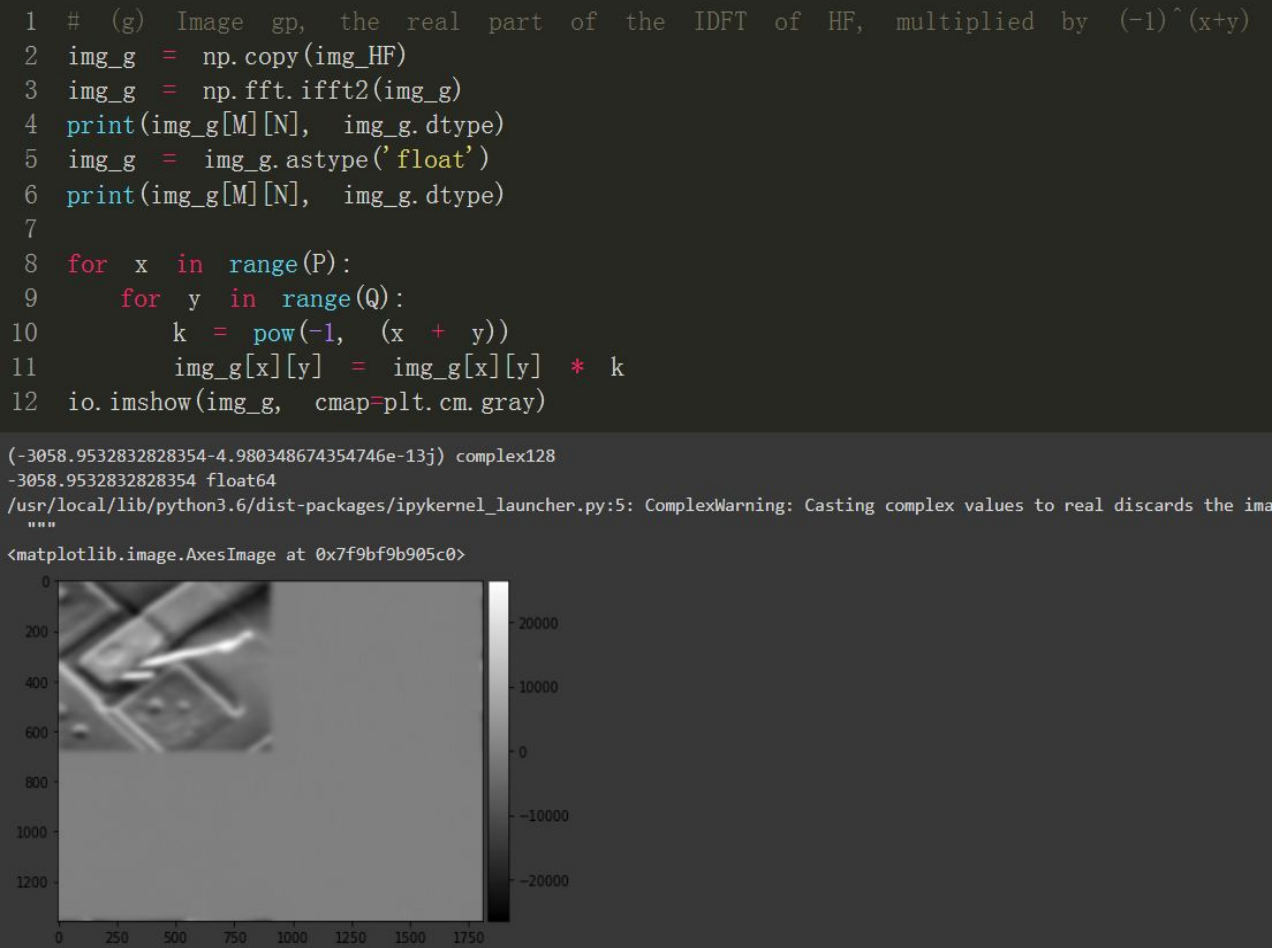
```

/usr/local/lib/python3.6/dist-packages/skimage/io/_plugins/matplotlib_plugin.py:150: UserWarning:
  lo, hi, cmap = _get_display_range(image)
<matplotlib.image.AxesImage at 0x7f9bf9c5c358>

```



圖片(f)及其程式碼，將做過傅立葉轉換的圖乘上剛剛得到的filter，然後秀出圖片的spectrum。



圖片(g)及其程式碼，將剛剛做好運算的圖做逆傅立葉轉換，然後得出的圖片裡每一個內容乘以-1的(x+y)次方。



將圖片(g)做切片，即得到課本上的圖片(h)。

以下是圖片(a)到圖片(h)。

