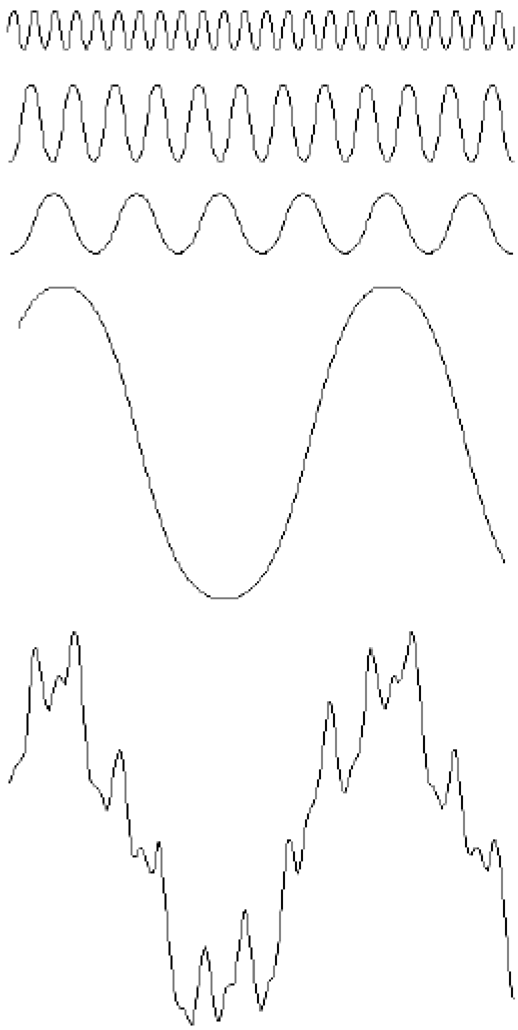
Image Enhencement in the Frequency D

Homework Report

Technical description

Fourier 轉換是說任何週期性重複的函數都可以表示成不同頻率的 sin 或 cos 的和，且每個會乘上不同的係數，稱之為 Fourier series。



(最底下的是上面四個 function 的總合)

就算不是週期性的函數(曲線下的面積是有限的)也可以藉由一個加權的 function 來表達成 sin 或 cos 相乘的積分合，在此情況下形成的結果就是 Fourier transform。

Fourier transform function 可以經由相反的處理(inverse)來完全重建回原本的樣子，而不會有資訊的喪失，這是 Fourier transform 的特色之一，如此便可以在 Fourier 值域上進行操作，並且毫無損失的回到原始的值域。

本次作業要用是二維的 DFT(discrete fourier transform)，去轉換原圖到 frequency domain 上做處理。

一個 MXN 大小的影像 f(x,y)的 DFT 由下列式子表示：

$$F(u,v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi(ux/M+vy/N)},$$

其中 u = 0,1,2,..M-1　　　v = 0,1,2,...,N-1

同理，F(u,v)也可以用 inverse 轉換回 f(x,y)：

$$f(x,y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{j2\pi(ux/M+vy/N)},$$

for $x = 0, 1, 2, \ldots, M-1$ and $y = 0, 1, 2, \ldots, N-1$.

上述為 Fourier transform pair，變數 u 和 v 是 transform 或 frenquency variable，而 x 和 y 是 spatial 或 image variable。

一個常用的做法是在計算 Fourier transform 前把輸入影像乘上

$(-1)^{x+y}$：

$$\Im[f(x,y)(-1)^{x+y}] = F(u-M/2, v-N/2)$$

此式描述 $f(x,y)(-1)^{x+y}$ 之 Fourier transform 的原點，即是 F(0,0) 位於 u=M/2 和 v=N/2，由 2-D 的 DFT 所佔據 MXN 區域的中心，我們稱此頻率區域為一個頻率矩形，為了確保這些位移過的座標為整數，我們需要 M,N 為偶數。

在(u,v) = (0,0)，套用上式可得到：

(4.2-16),

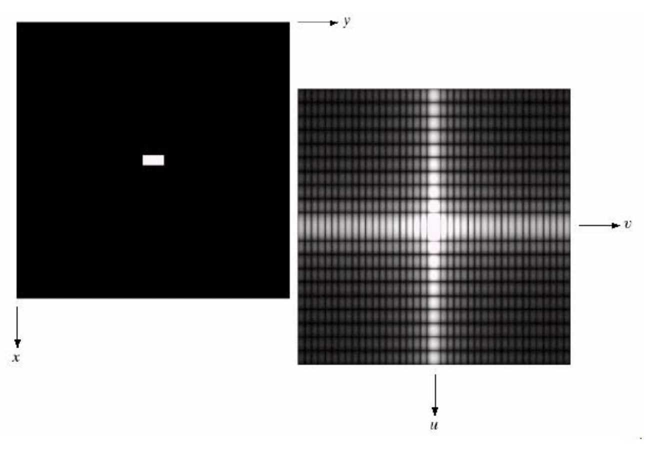
$$F(0,0) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y),$$

可以看出這個結果就是 f(x,y)的平均，也就是說 f(x,y)這個影像的平均灰階等於該影像做完 Fourier tramsform 到 frequency domain 中，原點 F(0,0)的值。

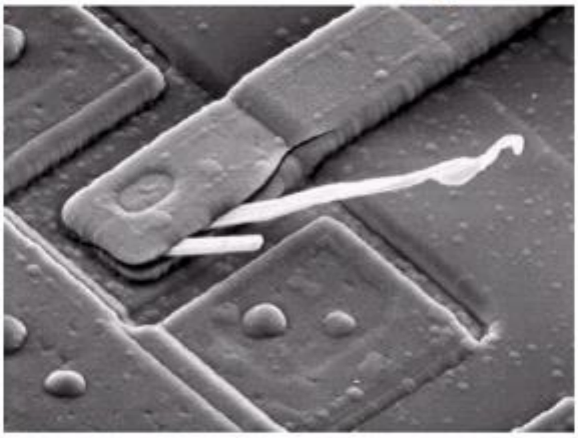
假如 f(x,y)為實數，則 Fourier transform 是共軛對秤的

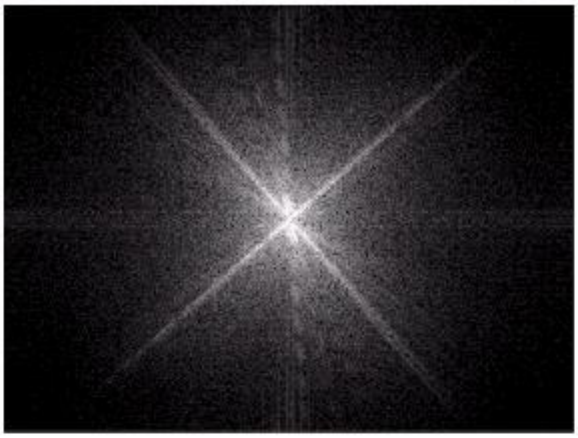
F(u,v) = F*(-u,-v)　=>　| F(u,v) | = | F(-u,-v) |

說明 Fourier 的頻譜是對稱的。



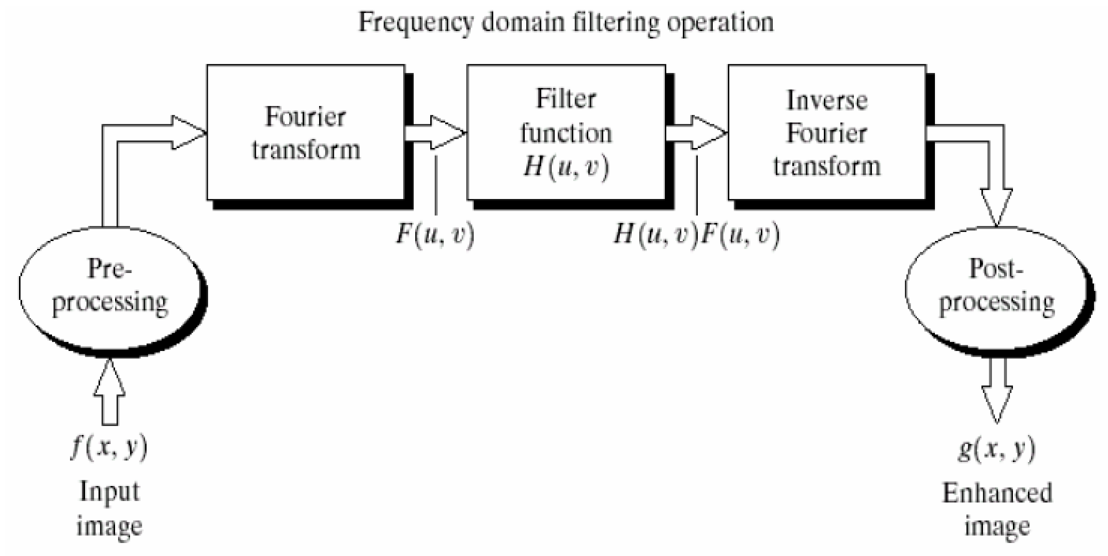
F(u,v)的每一項都包含 f(x,y)的所有值，並依**指數項**的值來修改，最慢變化的頻率 u=v=0 相當於影像的平均灰階，而頻率與速率改變有關，故 Fourier transform 的頻率高低和影像中灰階強度變化有所關聯。低頻對應到影像變化較慢的成分，高頻對應到影像越來越快的灰階變化(像是影像中的邊界、雜訊等等)。





上圖中透過 Fourier transform 可以清楚觀察到較強的灰階變化出現在影像中何處。

在 frequency domain 上的基本濾波步驟圖：



H(u,v)為 filter function，它抑制轉換中的某些頻率，同時讓其他的不變。輸出影像的 Fourier transform 為：

G(u,v) = F(u,v) H(u,v)

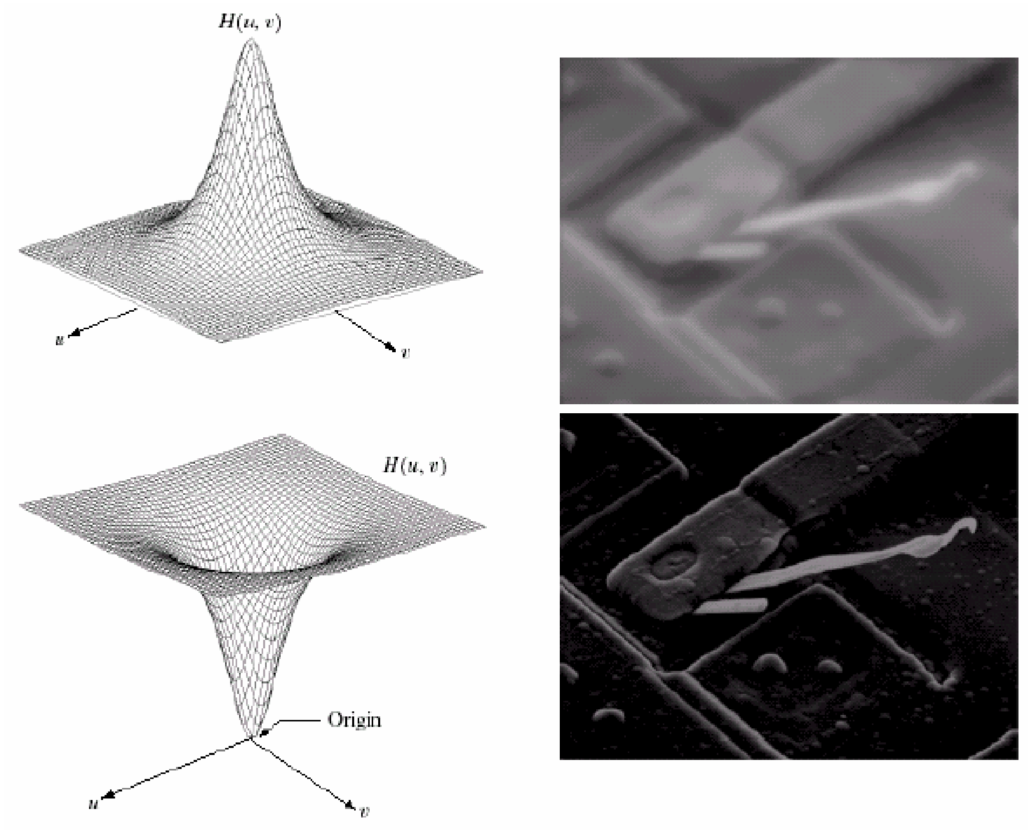
F 的組成是複數，但通常 filter 通常是常數，在此情形下，H 的每一個成分乘上所對應的 F 成分的實部和虛部，這樣的 filter 稱為 zero-phase-shift filter，經過 filtered 的影像可以由取得 G(u,v) 的 inverse Fourier transform 獲得：

$$\text{Filtered Image} = \Im^{-1}[G(u,v)]$$

最後的影像是取 Filtered image 的實部並乘上$(-1)^{x+y}$來抵銷一開始對輸入影像乘$(-1)^{x+y}$的結果。

總而來講就是由經過某個 filter function H(u,v)，以某種形式修改影像的 Fourier transform，然後再取其 inverse Fourier transform 得到 spatial domain 中的 filter h(x,y)，原始影像 f(x,y) 再和 h(x,y)做相應的處理得到 enhanced image g(x,y)。

- ✧ low-pass filter：衰減高頻，通過低頻，使影像模糊平滑
- ✧ high-pass filter：衰減低頻，通過高頻，使影像銳利清晰



- ◆ Laplacian in frequency domain

$$\Im\left[\frac{d^n f(x)}{dx^n}\right] = (j2\pi u)^n F(u),$$

$$\Im\left[\frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2}\right] = (j2\pi u)^2 F(u,v) + (j2\pi v)^2 F(u,v)$$

$$= -4\pi^2(u^2 + v^2)F(u,v).$$

Laplacain of f(x,y) has the result in frequency domain：

$$\Im\left[\nabla^2 f(x,y)\right] = -4\pi^2(u^2+v^2)F(u,v),$$

故推導出 Laplacian 可以使用以下 filter function 來實現：

$$H(u,v) = -4\pi^2(u^2+v^2).$$

filter function 也需要被 shift 到中心點

$$H(u,v) = -4\pi^2\left[(u-M/2)^2+(v-N/2)^2\right].$$

藉由 inverse Fourier transform 可以得到在 spatial domain 的 Laplacian filter：

$$\nabla^2 f(x,y) = IDFT\{H(u,v)F(u,v)\}$$

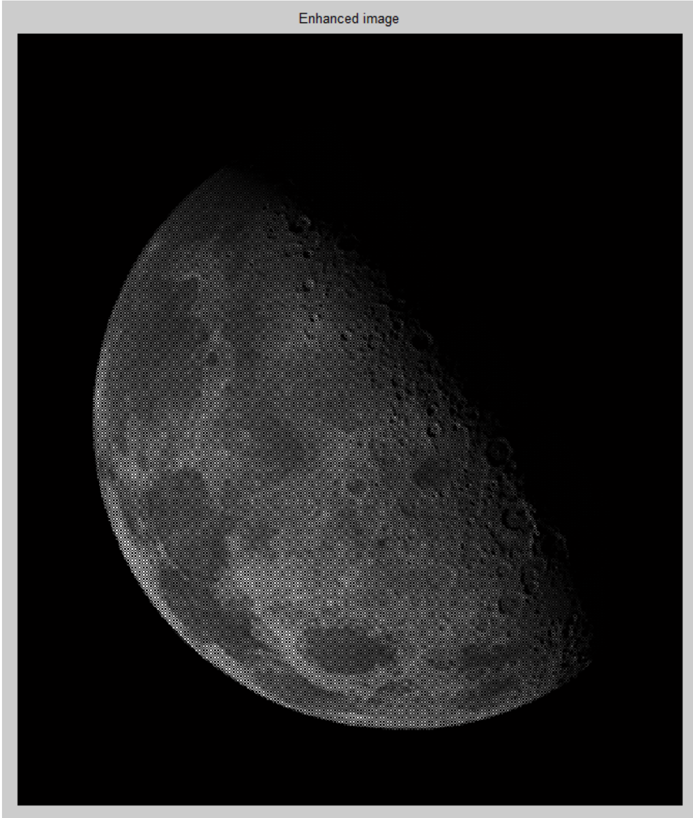
將原始影像減去 Laplacian filter 來獲得 Sharpened image

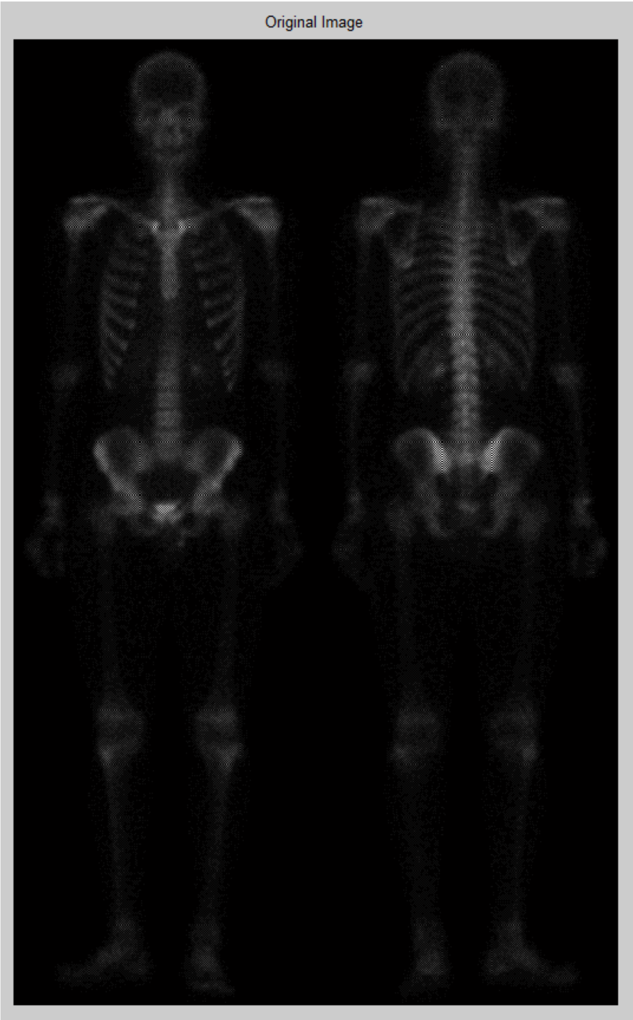
$$g(x,y) = f(x,y) - \nabla^2 f(x,y),$$

可以推導出：

$$g(x,y) = IDFT\{F(u,v) - H(u,v)F(u,v)\} =$$
$$IDFT\{[1-H(u,v)]F(u,v)\} = IDFT\{[1+4\pi^2 D^2(u,v)]F(u,v)\}$$

- Experimental results





- ## Disscussions

實作過程中因為對 frequency domain 不熟悉，故只要結果有錯也很難看出是錯在哪，需要對 Fourier transform 的機制有一定程度的瞭解才行，像是如果沒有做位移到 center 的動作，跑出來的結果就會非常詭異，而只在 Fourier transform 之前做一次位移，最後 inverse 之後沒有相對位移回來結果也一樣詭異。而 laplacian filter 取得的方式跟在 spatial domain 不太一樣，在 frequency domain 上是取得 laplacian filter 的頻域。

有了上次在 spatial domain 作 Laplacian 的 image sharpening 的經驗，對於如何應用 laplacian 處理銳化，已有基本的概念，影像在 spatial domain 作 filtering 的時候需要做 convolution 的計算，實作上比較麻煩一點，而轉換到 frequency domain 只要直接相乘即可，省下不少力氣。所以事實上只是換個方式去做 laplacian filter，結果和上次沒什麼差，故可以發現若沒有先做高斯濾波先去除雜訊，銳化後的 image 還是有雜訊變強的問題。

- ## References and Appendix

http://ppt.cc/vfkf

http://ppt.cc/7gHi

http://ppt.cc/A2xL

http://ppt.cc/d6ZA