

# Hw2\_Image Sharpening

## 1 Technical description

宣告和讀取

```
clc;
clear all;
img_array=cell(1,2);
img_array{1} = imread('skeleton_orig.bmp');
img_array{2} = imread('blurry_moon.tif');
img_array{1}=rgb2gray(img_array{1});
```

spatial domain

用自定義convolution\_spatial函式對圖片做捲積

並顯示結果

```
for i=1:2

    mask=[0,-1,0;-1,4,-1;0,-1,0];
    I_Lap = img_array{i}+convolution_spatial(img_array{i},mask);

    mask=[0,-1,0;-1,5,-1;0,-1,0];
    I_Unsharp = convolution_spatial(img_array{i},mask);

    mask=[0,-1,0;-1,5.5,-1;0,-1,0];
    I_High_Boost = convolution_spatial(img_array{i},mask);

    figure;
    sgtitle('Spatial domain');
    subplot(2,2,1),imshow(img_array{i});title('Original image');
    subplot(2,2,2),imshow(I_Lap);title('Laplacian operator');
    subplot(2,2,3),imshow(I_Unsharp);title('unsharp masking');
    subplot(2,2,4),imshow(I_High_Boost);title('high-boost filtering ');
end
```

frequency domain

用自定義frequency函式對圖片做處理

並顯示結果

```
for i=1:2

    mask=[0,-1,0;-1,4,-1;0,-1,0];
    I_Lap = img_array{i}+frequency(img_array{i},mask);

    mask=[0,-1,0;-1,5,-1;0,-1,0];
    I_Unsharp = frequency(img_array{i},mask);

    mask=[0,-1,0;-1,5.5,-1;0,-1,0];
    I_High_Boost = frequency(img_array{i},mask);
```

```

figure;
sgtitle('frequency domain');
subplot(2,2,1),imshow(img_array{i});title('Original image');
subplot(2,2,2),imshow(I_Lap);title('Laplacian operator');
subplot(2,2,3),imshow(I_Unsharp);title('unsharp masking');
subplot(2,2,4),imshow(I_High_Boost);title('high-boost filtering ');
end

```

### convolution\_spatial 函式

產生一個和image同樣大小的output，將原圖轉成double並拉伸到[0 1]之間，用無視邊界值的方式做convolution，迴圈的起始和結束分別為2到m-1和2到n-1，並將最後的結果轉到uint8

```

function output = convolution_spatial(I,mask)
[m,n]=size(I);
I=im2double(I);
output=zeros(size(I));
for i = 2 : m - 1
    for j = 2 : n - 1
        tmp=mask.*I(i-1:i+1,j-1:j+1);
        output(i-1:i+1,j-1:j+1)=sum(tmp(:));
    end
end
output=im2uint8(output);
end

```

### frequency 函式

frequency domain的影像處理步驟如下

1. Multiply the input image by  $(-1)^{x+y}$  to center the transform
2. Compute  $F(u, v)$ , the DFT of the image from (1).
3. Multiply  $F(u, v)$  by a filter function  $H(u, v)$ .
4. Compute the inverse DFT of the result in (3).
5. Obtain the real part of the result in (4).
6. Multiply the result in (5) by  $(-1)^{x+y}$ .

```

function output = frequency(I,mask)
I = im2double(I);
[m n] = size(I);

% 1.
%影像乘上(-1)^(x+y),為了置中
for i=1:m
    for j=1:n
        I(i,j) = I(i,j)*(-1)^(i+j);
    end
end

% 2.
F = fft2(I);
% Frequency response
H = freqz2(mask,[m n]);

% 3.和4.
%G(u,v) = H(u,v)F(u,v)

```

```

G = H.*F;

% 5.
%取Filtered image的實部
G = real(iff2(G));

% 6.
%乘上(-1)x+y抵銷一開始對影像乘(-1)x+y
for i=1:m
    for j=1:n
        G(i,j) = G(i,j)*(-1)^(i+j);
    end
end

G = im2uint8(G);
output=G;
end

```

## 2 Experimental results



## 3 Discussions

### (1)spatial domain

跟作業一差不多，差別在mask，過程都是對原圖做捲積。

### (2)frequency domain

雖然程式不長，但是因為對 frequency domain 的不熟悉，所以花了不少時間在了解 frequency domain的概念如何在matlab實現。

影像在 spatial domain 的時候需要做 convolution，要對每個點做處理，而frequency domain則是經過轉換後，一次對原圖做處理，最後再還原到spatial domain

在spatial domain和frequency domain處理的圖片看起來差不多

## 4 References and Appendix

<https://www.cs.uregina.ca/Links/class-info/425/Lab5/index.html>

[https://blog.csdn.net/qg\\_30815237/article/details/98655630](https://blog.csdn.net/qg_30815237/article/details/98655630)

<https://www.youtube.com/watch?v=spUNpyF58BY>

<https://theailearner.com/2019/05/14/unsharp-masking-and-highboost-filtering/>