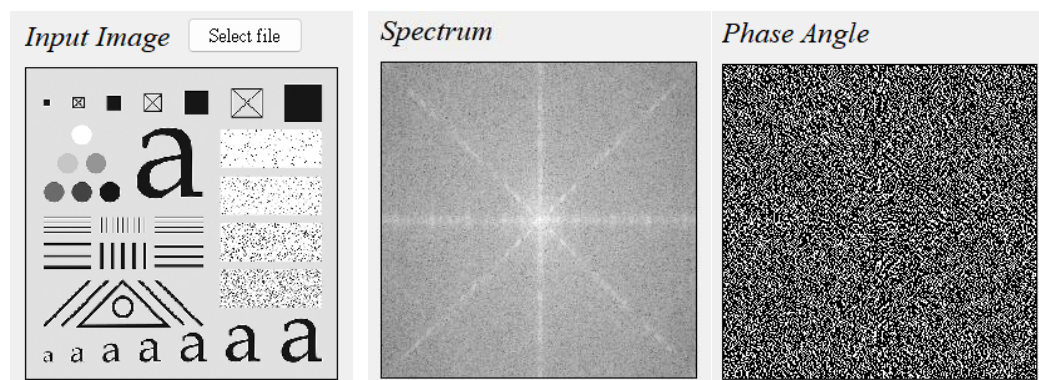
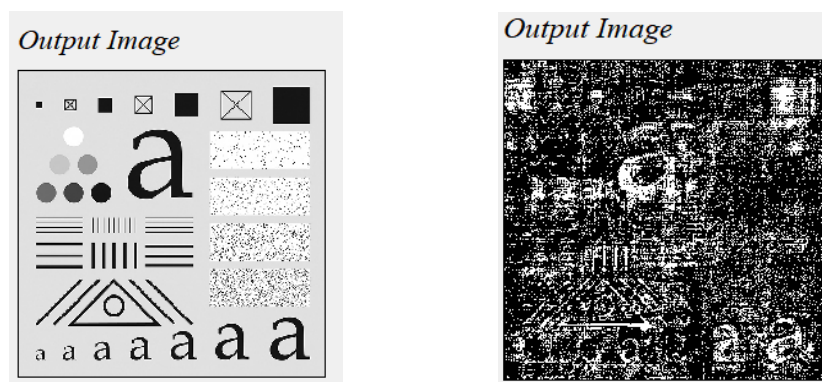


此次作業目標是設計一介面，供使用者對輸入的 jpg 影像進行傅立葉轉換，並在頻率域進行高、低通濾波等處理。

● Part 1



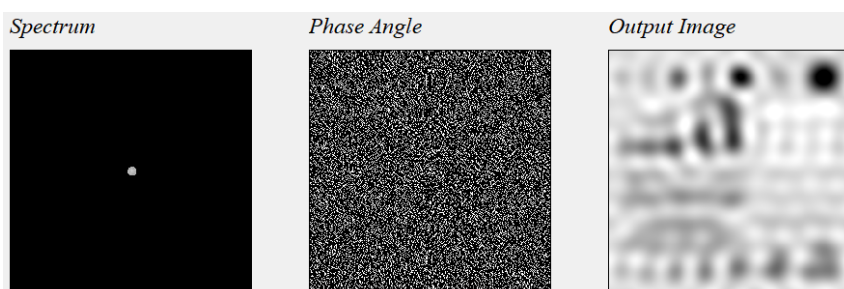
將一張 size 為 (688X688) 的影像經過程式轉換，我們可以得到 spectrum 和 phase angle (上圖)，以及反轉換回來的影像 (下左圖)。可以看出反轉換後的影像，內容與大小和原圖大致相同。但透過將轉換前後的影像相減 (下右圖)，可以看出轉換後的影像與轉換前相比仍略有差異，並非完全相同。



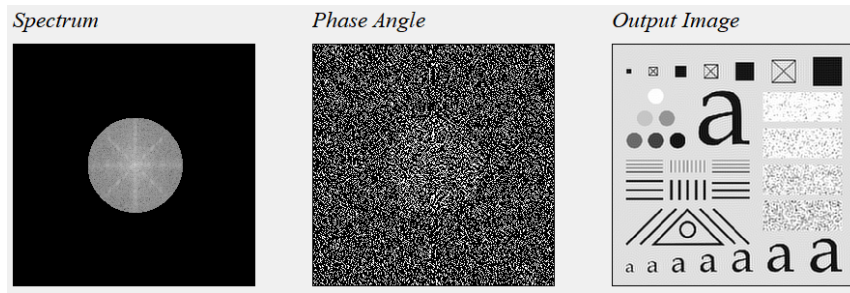
● Part 2

此小題目標為在頻率域對輸入影像分別進行三種遮罩的高、低通處理 (共六種結果)，以下將針對其中三種結果進行討論。

1. Ideal Low Pass Filter



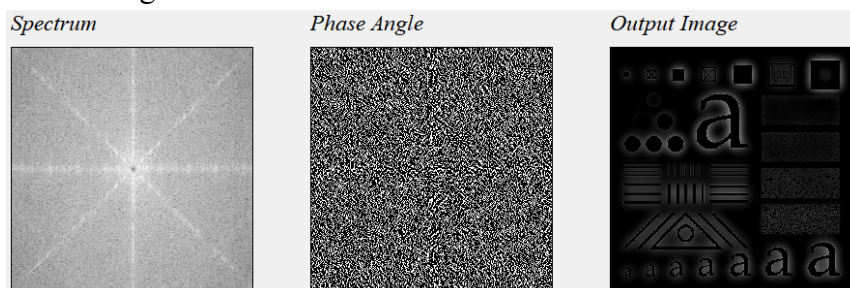
圖一、cut-off frequency = 10



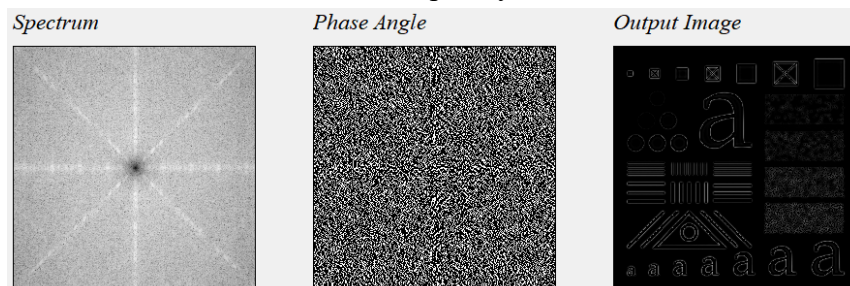
圖二、cut-off frequency = 100

由上圖可以看出對於 ILPF，cut-off frequency 越小，只留下越低頻的範圍，因此越模糊。

2. Butterworth High Pass Filter



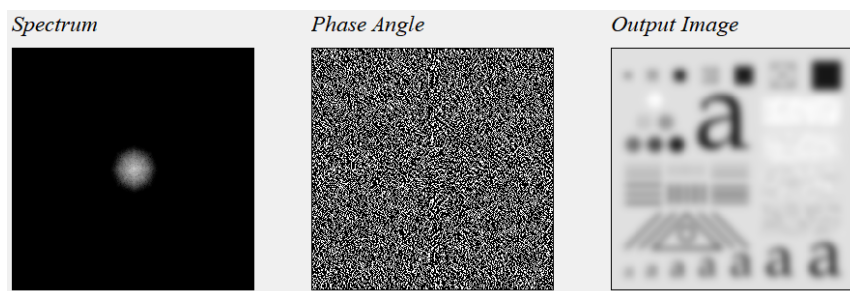
圖三、cut-off frequency = 10，n = 2



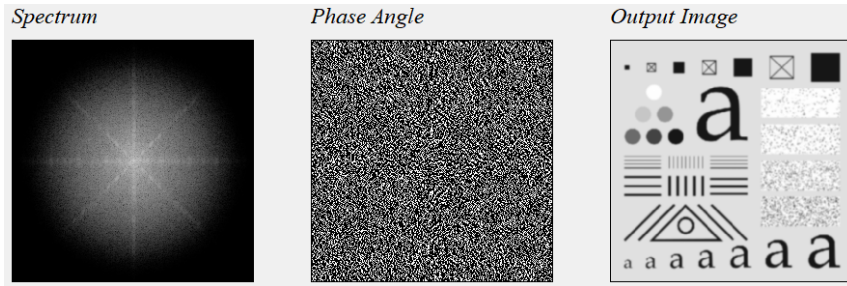
圖四、cut-off frequency = 50，n = 2

對於 BHPF，在 cut-off frequency 內會漸進的被消除，而 cut-off frequency 外附近會漸進地被保留，cut-off frequency 越大影像輪廓越能被突顯。

3. Gaussian Low Pass Filter



圖五、cut-off frequency = 10



圖六、cut-off frequency=50

對於 GLPF，不像 ILPF 只是簡單的將 cut-off frequency 內的訊號保留，而是以曲線的方式慢慢將範圍內的訊號減弱，直到 cut-off frequency 後為 0，而 cut-off frequency 越小，影像也就越模糊。

● Part3

```
for(int j = 0; j < 512; j++)
{
    for(int i = 0; i < 512; i++)
    {
        QRgb inputimagergb = propimage.pixel(i,j);
        z_array[i][j] = (double) qGray(inputimagergb);
        z_array[i][j] = log(z_array[i][j] + epsilon);
    }
}

Complex ** output_seq = new Complex * [512];
for (int j = 0; j < 512; j++) {
    output_seq[j] = new Complex[512];
}
//輸入待轉換的實部、虛部
for (int i = 0; i < 512; i++) {
    for (int j = 0; j < 512; ++j) {
        output_seq[i][j] = Complex(z_array[i][j] * pow(-1, i + j), 0);
    }
}

//進行2D FFT，正轉換為false，反轉換為true
fft2(output_seq, 512, 512, false);

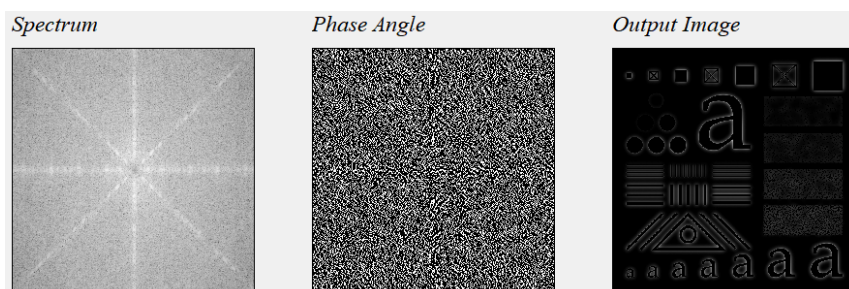
for(int j = 0; j < 512; j++)
{
    for(int i = 0; i < 512; i++)
    {
        // output_seq[i][j] = Complex(input_seq[i][j].real, input_seq[i][j].imag);
        mask[i][j] = (gamma_H - gamma_L) * (1 - exp(-5 * (pow(i-256,2) + pow(j-256,2)) / pow(cutoff_freq,2))) + gamma_L;
        output_seq[i][j] *= mask[i][j];
    }
}

MainWindow::MainWindow::show_processed_spec_pa(output_seq);

fft2(output_seq, 512, 512, true);

for(int j = 0; j < 512; j++)
{
    for(int i = 0; i < 512; i++)
    {
        // output_seq[i][j].real = exp(output_seq[i][j].real) * -1 * / ;
        output_seq[i][j].real = output_seq[i][j].real * -1 * / ;
        output_seq[i][j].real *= pow(-1, i + j);
    }
}
```

此遮罩共有四個參數能夠進行調整，分別為 γ_H 、 γ_L 、 c 與 d_0 。其中， γ_H 和 γ_L 分別代表遮障的上下限， c 代表上下限間的調整斜率，而 d_0 則為 cut-off frequency。處理結果如高通濾波，能夠進行邊緣檢測（圖七）。



圖七、 $\gamma_H=1.4$ 、 $\gamma_L=0.3$ 、 $c=1$ 、 $d_0=60$