

## 多媒體技術概論 Assignment2 Report

### Method Description

```
function [y, filter] = createFilter(in, fs, N, window, pass, fc)
```

in 為用 wavread 讀取的振幅原值，fs 則為取樣頻率。

N 是 filter 的 order，取 1001。

window 是選用的 window function，pass 則是 filter 的種類。

fc 則為 cutoff frequency。

```
% Normalization
fc=fc/fs;
middle=round(N/2);
fltr=zeros(1, N);
```

將 fc 標準化使得  $\pi$  等於 Nyquist 角頻率

```
if strcmp(pass, 'low-pass')==1
    for n=-round(N/2)+1:round(N/2)
        if n==0
            fltr(middle)=1;
        else
            fltr(n+middle)=sin(2*pi*fc*n)/(pi*n);
        end
    end
    fltr(middle)=2*fc;

elseif strcmp(pass, 'high-pass')==1
    for n=-round(N/2)+1:round(N/2)
        if n==0
            fltr(middle)=1;
        else
            fltr(n+middle)=-sin(2*pi*fc*n)/(pi*n);
        end
    end
    fltr(middle)=1-2*fc;

elseif strcmp(pass, 'band-pass')==1
    for n=-round(N/2)+1:round(N/2)
        if n==0
            fltr(middle)=1;
        else
            fltr(n+middle)=(sin(2*pi*fc(2)*n)-sin(2*pi*fc(1)*n))/(pi*n);
        end
    end
    fltr(middle)=2*(fc(2)-fc(1));
end
```

依照給入的 pass 參數做不同的 filtering。

```
% window
if strcmp(window, 'Hanning')==1
    for n = -round(N/2)+1:round(N/2)
        fltr(middle+n) = fltr(middle+n)*(0.5+0.5*cos((2*pi*n)/N));
    end
elseif strcmp(window, 'Hamming')==1
    for n = -round(N/2)+1:round(N/2)
        fltr(middle+n) = fltr(middle+n)*(0.54+0.46*cos((2*pi*n)/N));
    end
elseif strcmp(window, 'Blackman')==1
    for n = -round(N/2)+1:round(N/2)
        fltr(middle+n) = fltr(middle+n)*(0.42+0.5*cos((2*pi*n)/(N-1))+0.08*cos((4*pi*n)/(N-1)));
    end
end
```

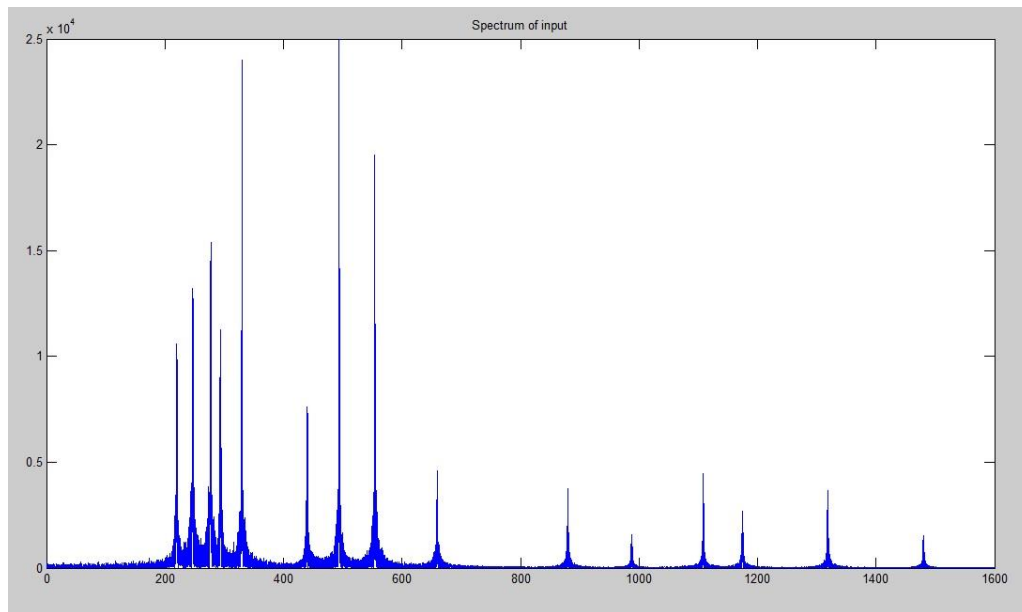
依照給入的 window 參數使用不同的 window function 完成 filter 並回傳。

```
for i=1:leni
    for j=1:lenj
        if(i-j>0)
            y(i) = y(i) + in(i-j)*fltr(j);
        end
    end
end
```

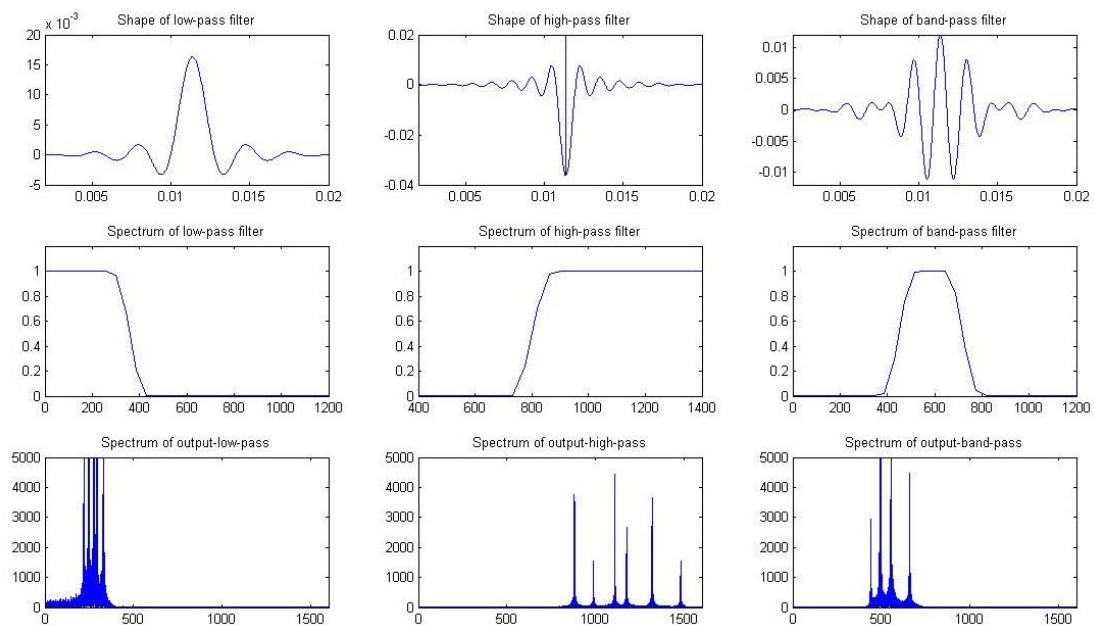
對 input signal 做 convolution，並回傳新的 output signal。

## Results

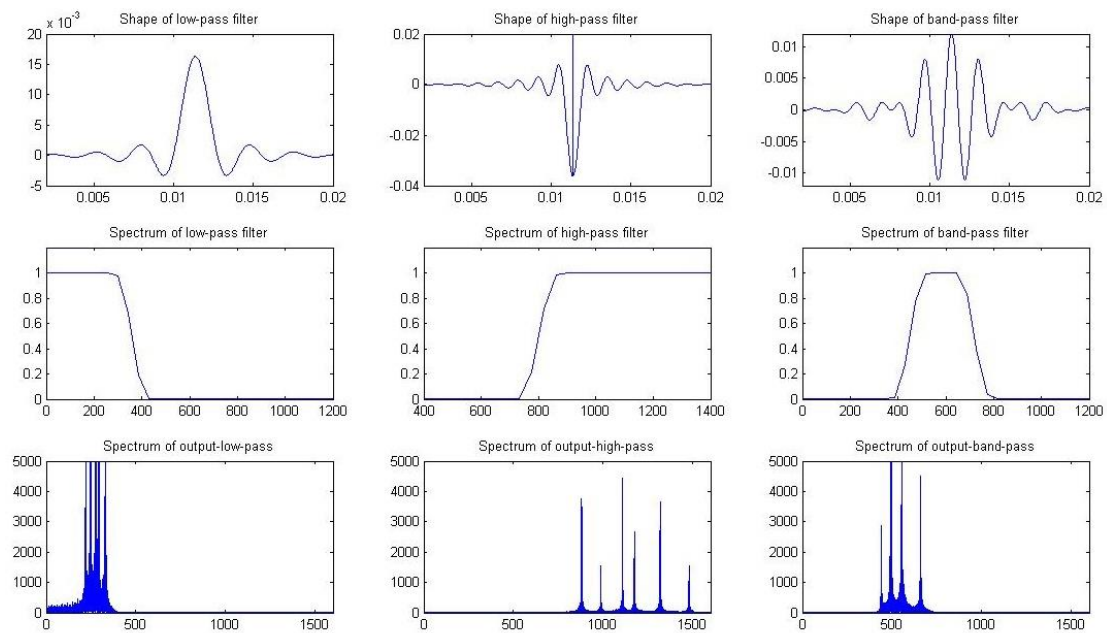
### Spectrum of Input signal



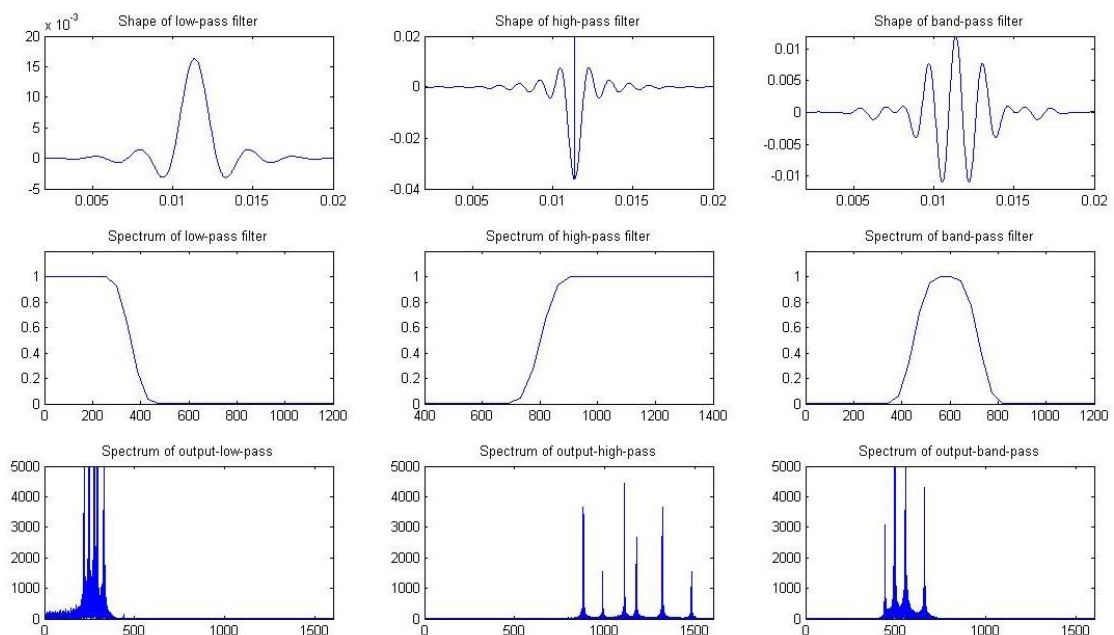
### Results of Hanning window function



## Results of Hamming window function



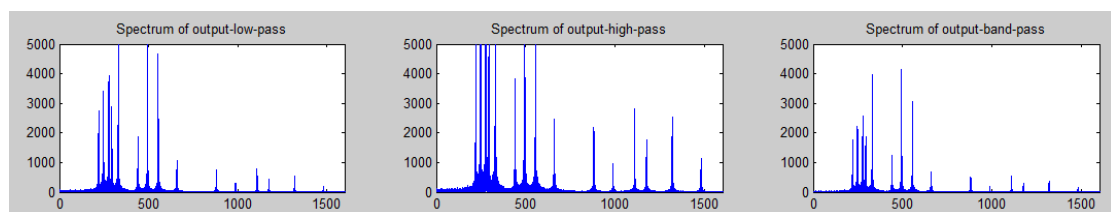
## Results of Blackman window function



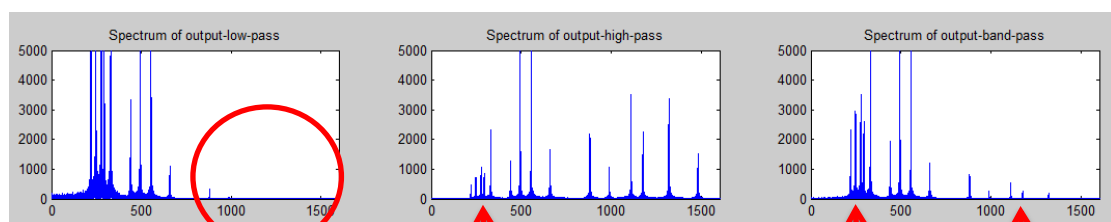
## Discussion

先用 input spectrum 猜測低、中、高頻資料的大概範圍，再以 Hanning window function 嘗試了幾個 N 的值，反覆多次找出效果最佳的頻率範圍以及 N 值，並用 low-pass, high-pass, band-pass 分別過濾出低頻、高頻、中頻資料。

N=31，三首歌完全分離不出來。



N=91，low-pass 成功把高頻資料篩掉；high-pass 則篩掉了低頻資料，但仔細聽還是有些很小聲的餘音；band-pass 完全沒有篩掉任何歌，但已把高頻及低頻資料的音量降低許多。

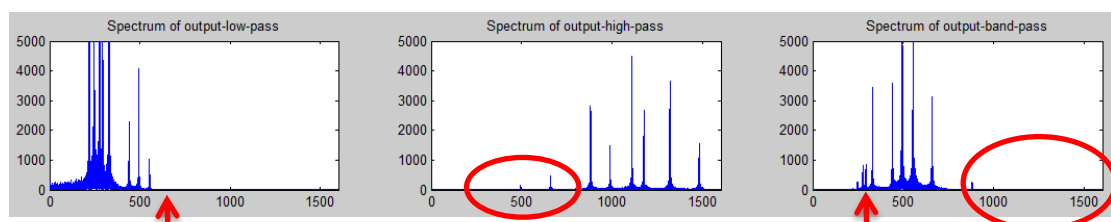


將高頻資料完全篩掉

沒篩完全的低頻資料

低頻及高頻資料音量降低

N=301，low-pass 篩掉了中頻資料的最高音；high-pass 完全篩掉了低頻及中頻資料，但中頻資料的些許高音仍未篩掉；band-pass 篩掉了高頻資料，且大幅降低低頻資料的音量，但仍未完全篩除。



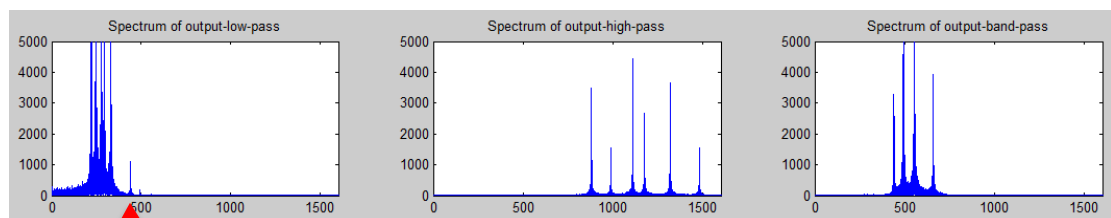
篩掉中頻資料最高音

沒篩完全的中頻高音

低頻資料音量降低

篩除高頻資料

N=601，low-pass 篩掉了中頻資料，但其最低音仍未被篩除；high-pass 至此已成功篩除低頻及中頻資料；band-pass 至此也成功篩除低頻及高頻資料。



中頻資料最低音仍未篩除

N=1001，成功分離出低、中、高頻三首歌，其結果如 Results 所示。

low-pass threshold=360；high-pass threshold=800；band-pass band=450~720

Hamming 及 Hanning 的效果相差極小，而 Blackman 的結果就比較不一樣，尤其是在 low-pass 的部分，仍會有些微中頻資料未被篩除，在一般環境下應該是聽不出來，要用耳機才聽得出來，因此尚不影響要求的效果；但在相同的參數之下，Hamming 及 Hanning 的 low-pass 都可以完全篩除中頻資料。

## Execution

檔案：hw2.m、createFilter.m、hw2\_mix.wav 需在同個目錄下。

執行：在 hw2() 中傳入要運用的 window function，例：hw2('Hamming') 即可，但因要處理十張圖，大約需時三分鐘。