



數位信號處理實習

LAB5

2014/5/5

電子工程系 蔡偉和 教授

100360318 四子三甲 陳奕璋 學生

[練習 5-1] 將聲音訊號分別進行 $\downarrow 2$ 與 $\uparrow 2$ (不使用 lowpass filters)，繪出其 spectrograms。

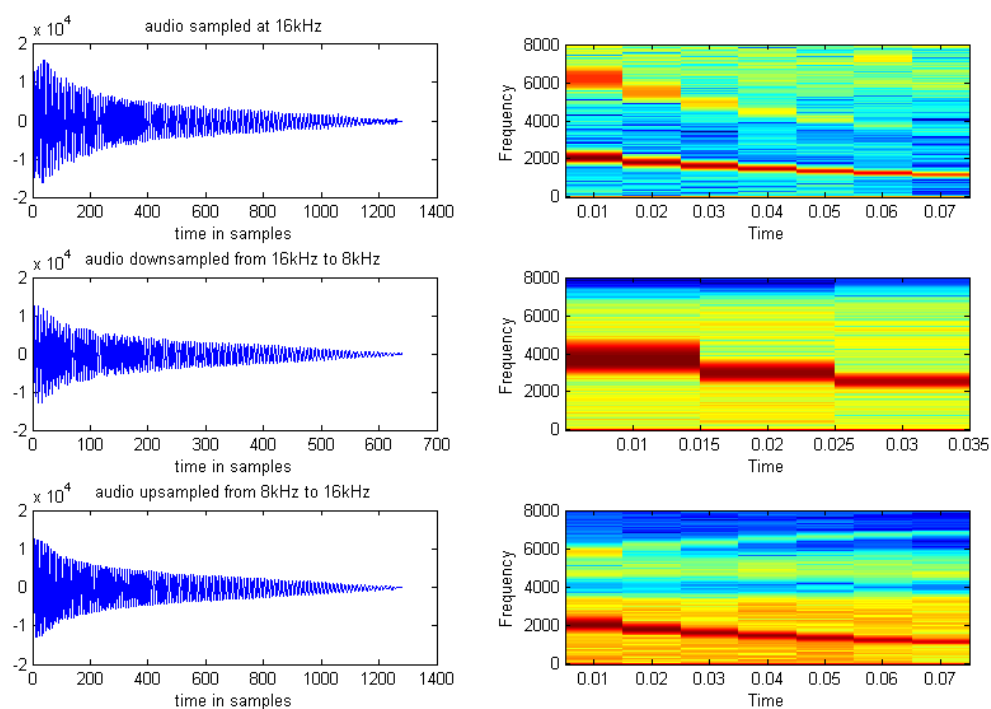
1. 程式碼：

```
fp=fopen('bird.wav','r');
fseek(fp,44,-1);
x=fread(fp,'short');
Fs=16000;
subplot(3,2,1); plot(x);
xlabel('time in samples');
title('audio sampled at 16kHz');
sound(x./32766,Fs,16)
subplot(3,2,2); specgram(x,512,Fs,320);

% downsampling
y=decimate(x,2);
subplot(3,2,3); plot(y);
xlabel('time in samples');
title('audio downsampled from 16kHz to 8kHz');
sound(y./32766,Fs/2,16)
subplot(3,2,4); specgram(y,512,Fs,320);

% upsampling
z=interp(y,2);
subplot(3,2,5); plot(z);
xlabel('time in samples');
title('audio upsampled from 8kHz to 16kHz');
sound(z./32766,Fs,16);
subplot(3,2,6); specgram(z,512,Fs,320);
```

2. Matlab 波形圖：



圖一 升取樣、降取樣之頻域對時間比較圖

3. 心得

由圖一我們可以發現在降取樣後又升取樣，在高頻的部分有明顯的失真，如原本時間點 0 時在 6000Hz 左右之高頻在轉換後幾乎消失了，而本來時間點 0 於 2000Hz 左右之中頻訊號在轉換後也顯的變粗(高頻消失)。

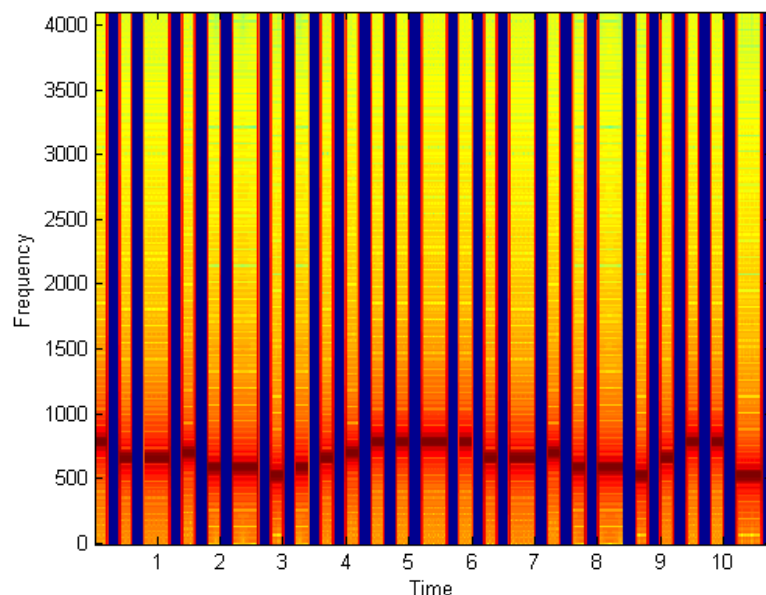
[練習 5-2] 產生一段歌曲訊號 So Mi Mi Fa Re Re Do Re Mi Fa So So So ; So Mi Mi Fa Re Re Do Mi So So Do，並使用 Matlab 函式 sound.m 以取樣頻率 8000Hz 播放出來。

1. 程式碼：

```
notecreate = @(frq,dur) sin(2*pi* [1:dur]/8000* (440*2.^((frq-1)/12)));
notename = {'A' 'A#' 'B' 'C' 'C#' 'D' 'D#' 'E' 'F' 'F#' 'G' 'G#'};
song = { 'G' 'E' 'E' 'F' 'D' 'D' 'C' 'D' 'E' 'F' 'G' 'G' 'G'
         'G' 'E' 'E' 'F' 'D' 'D' 'C' 'E' 'G' 'G' 'C'
         };
dur = [ 1 1 2 1 1 2 1 1 1 1 1 1 2
        1 1 2 1 1 2 1 1 1 1 2
        ];
for k1 = 1:length(song)
    idx = strcmp(song(k1), notename);
    songidx(k1) = find(idx);
end

durA = 0.2*8000;
songnote = [];
for k1 = 1:length(songidx)
    songnote = [songnote; [notecreate(songidx(k1), durA*dur(k1)) zeros(1,durA)]];
end
specgram(songnote,512,8192,320);
soundsc(songnote, 8000);
```

2. Matlab 波形圖：



圖二 小蜜蜂歌曲頻域對時間之波形圖

3. 心得

由頻域對時間之波形圖，我們可以發現其所對的音高與音譜對應之頻率一致，證明了播放出來的結果的確為正確音高，而藍色的空格為停頓時，故無頻率響應。

這題使用了 `function_handle (@)` 的寫法，並搭配 `strcmp` 比較音符，直接利用自母符號作為歌曲的音高，讓程式可讀性更高。