**Homework 2: Gabor transforms**

Yicheng Duan

AMATH 482

02/07/2020

**Part I**

1. **Abstract**

In this part of the homework, we will analyze a portion of Handel's Messiah with time-frequency analysis. We will use the Gabor filtering method at first with simple Gaussian. Then we test for different window width and window type of Gabor transform. Oversampling and Under-sampling will be included in the discussion. Then we will analyze two music and find their music scale.

1. **Introduction and Overview**

First, we import a portion of Handel's Messiah directly form MATLAB. We use Gabor transformation with different time steps on this signal to create spectrograms. By altering the Gabor filter’s window width and using two different Gabor filter wavelets, the Mexican hat wavelet and a step-function (Shannon) window, we can see the variation of the spectrogram related to the time and frequency domain.

1. **Theoretical Background**

**Gabor transformation:**

The Gabor transform, named after Dennis Gabor, is a special case of the short-time Fourier transform. It is used to determine the sinusoidal frequency and phase content of local sections of a signal as it changes over time. The function to be transformed is first multiplied by a Gaussian function, which can be regarded as a window function, and the resulting function is then transformed with a Fourier transform to derive the time-frequency analysis. (wiki)

**Gaussian window function:**

Here, a is the window width and b is the translation parameter.

**the Mexican hat wavelet:**

In mathematics and numerical analysis, the Ricker wavelet, also known as **the Mexican hat wavelet** is the negative normalized second derivative of a Gaussian function.

here controlling the wavelet window width.

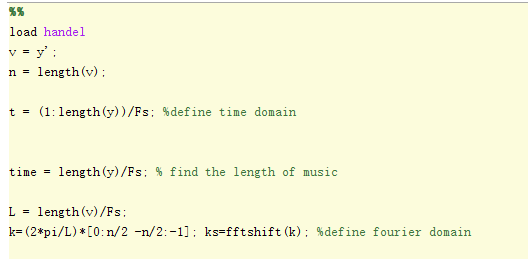
**Step-function (Shannon) window:**

It is the function of real intervals.

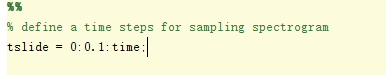
here can control the wavelet window width.

1. **Algorithm Implementation and Development**

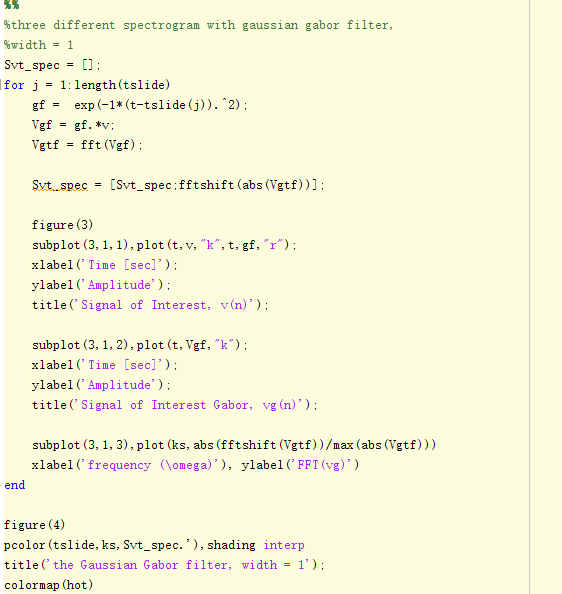
First, load Handel music signal and create a Fourier domain for its frequency.



Define the time steps for spectrogram sampling.

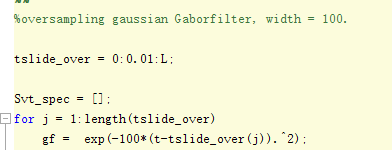


We use the Gaussian Gabor filter with width = 1 to sampling the spectrogram.

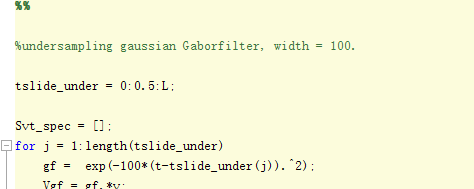


Then we change the width of the Gaussian Gabor filter to 100 and 1000 respectively.

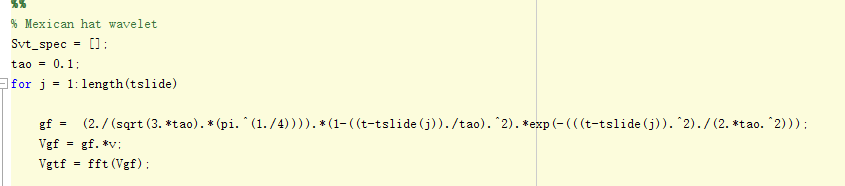
Now we oversample the Gaussian Gabor Filter by decreasing the time interval of time steps to 0.01, of the normal value(0.1).



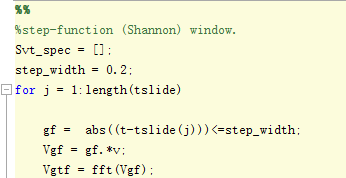
Then we undersample the Gaussian Gabor Filter by increasing the time interval of time steps to 0.5, 5 times of the normal value(0.1).



We use a new wavelet, the Mexican hat wavelet, to sample the spectrogram. Using to control the width.

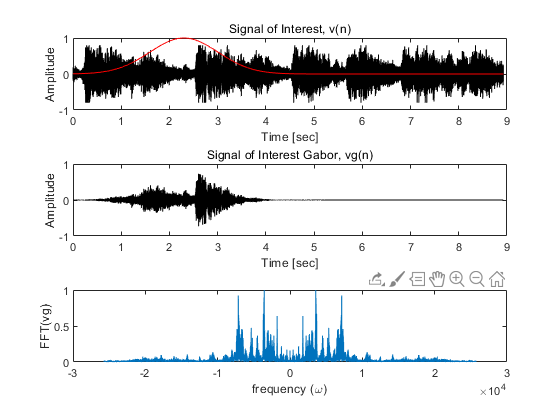
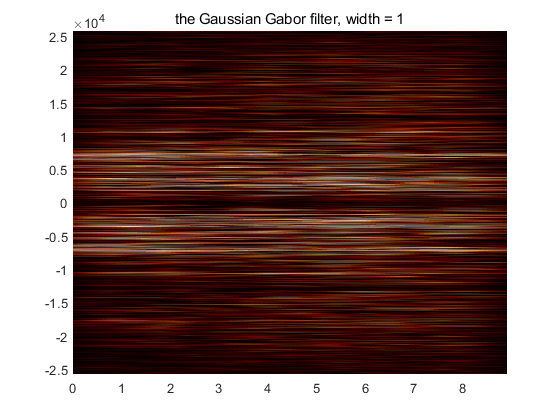


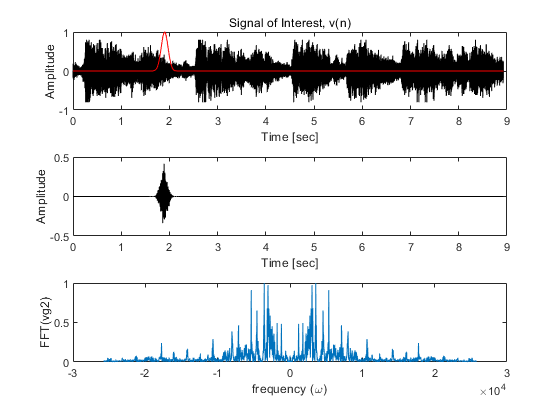
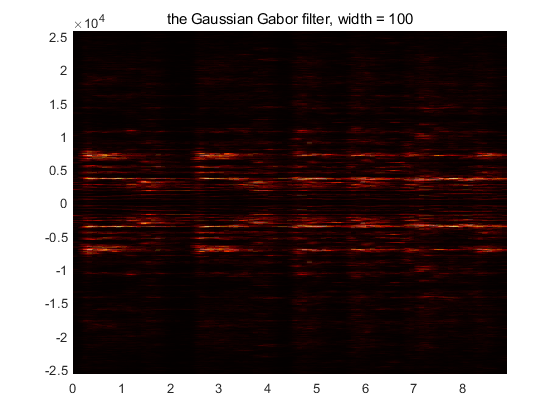
Then, we use a step-function (Shannon) window to sample the spectrogram.

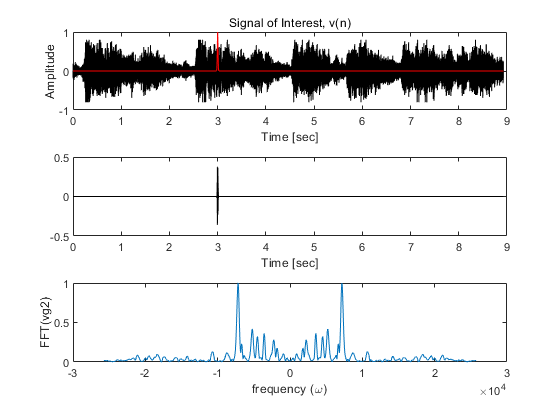
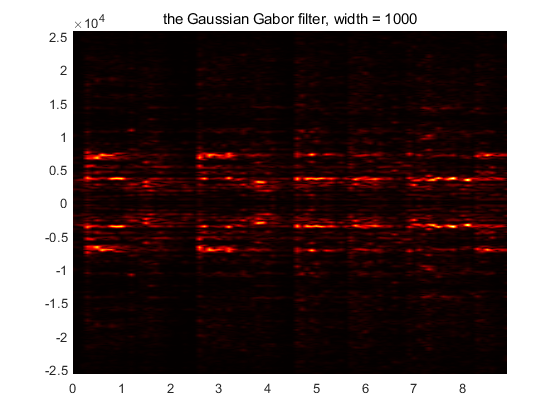


1. **Computational Results**

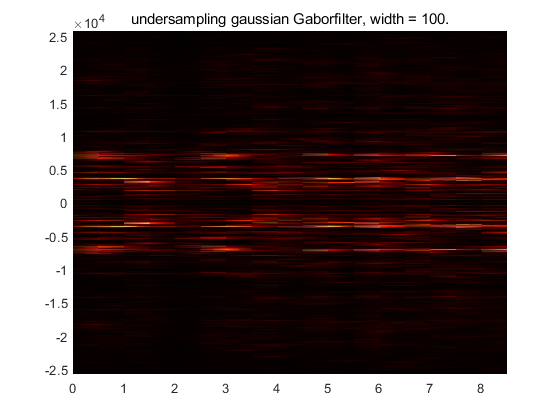
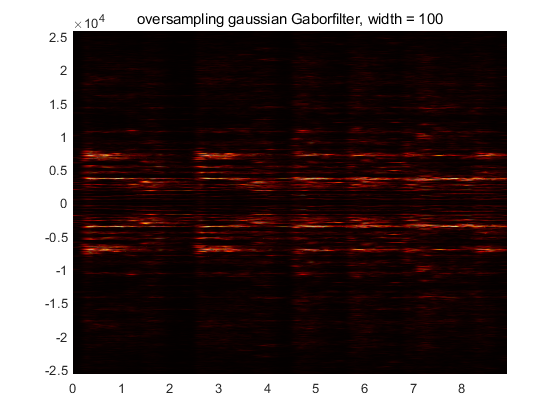
For three different width of the Gaussian Gabor Filter:



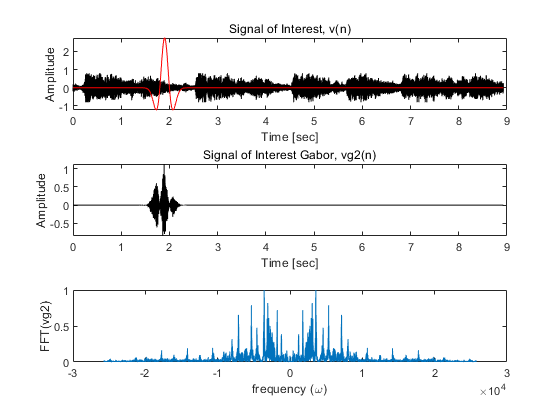
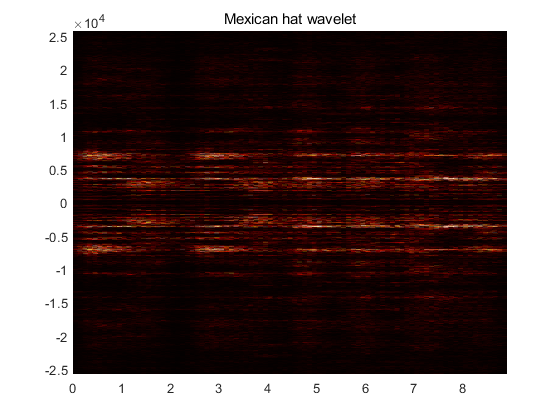


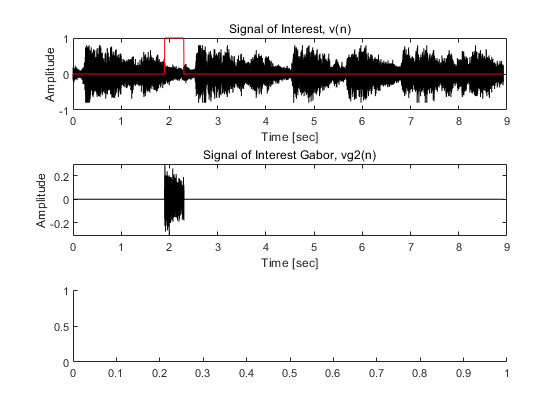
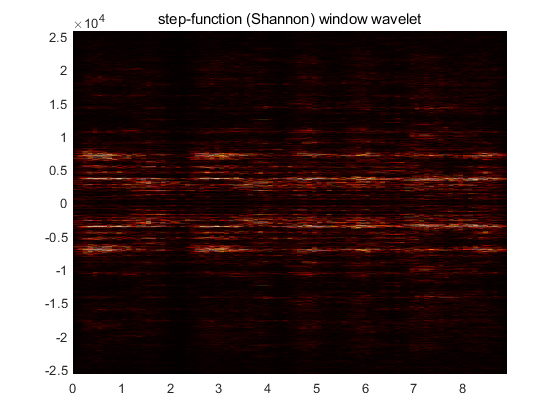


For oversampling and undersampling the Gaussian Gabor Filter, width = 100:



Mexican hat wavelet and step-function wavelet:





1. **Summary and Conclusions**

When the only variable of the Gaussian Gabor Filter is width, the spectrogram will tell us many things. When we see a huge wide window Gaussian Gabor Filter, we lose much information on the time domain, however, it sampled more information on the frequency domain. On the contrary, when we see a small window width Gaussian Gabor Filter, we lose much information on the frequency domain, however, it sampled more information on the time domain. So, a proper window width is critical to get a clear spectrogram and analyze information. Oversampling will create more accuracy in both time and frequency domain.

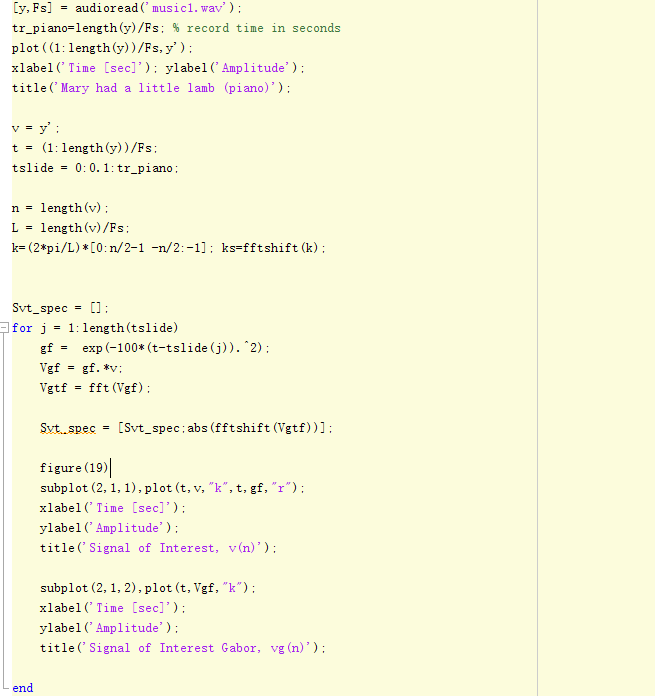
Using different wavelets can acquire more information in the frequency domain.

**Part II**

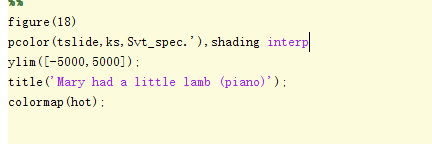
1. **Introduction and Overview**

We will use the same sampling skill from part I to find two music, which the same pattern played by two different musical instruments. By analyzing their spectrogram, we will see each musical scale lying on their frequency. Then, determined the scale by checking the datasheet

1. **Algorithm Implementation and Development**



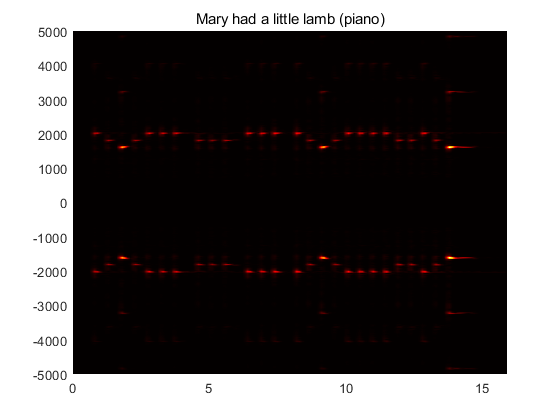
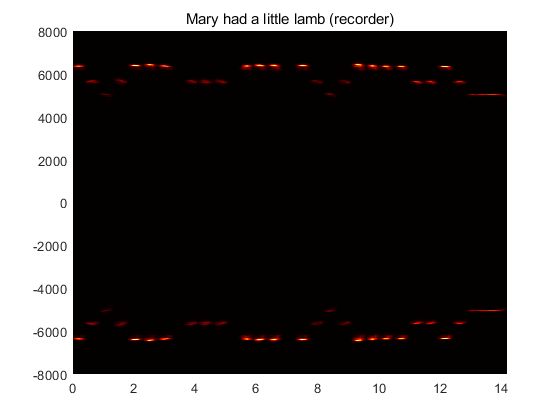
We first import our music to Matlab by using function audioread. Then we analyze it by using the Gaussian Gabor filter with width 100.



We draw a spectrogram of this music.

Same procedural for music2.

1. **Computational Results**



1. **Summary and Conclusions**

We see that the timbre of that two pieces of music are different since the frequency range of mucic1 is much smaller than the one of music2.

**Appendix A.** MATLAB functions used and brief implementation explanation

|  |  |
| --- | --- |
| length(v) | Find the length of variable v, in the float. |
| fftshift(v) | rearranges a Fourier transform v by shifting the zero-frequency component to the center of the array. |
| pcolor(v), shading | creates a pseudocolor plot using the values in matrix C. shading changes the shading model. |
| plot(X,Y) | plot(X,Y) creates a 2-D line plot of the data in Y versus the corresponding values in X. |
| colormap(map) | Colormap(map) sets the colormap for the current figure to one of the predefined colormaps. |
| [y,Fs] = audioread(filename) | [y,Fs] = audioread(filename) reads data from the file named filename, and returns sampled data, y, and a sample rate for that data, Fs. |

**Appendix B**. MATLAB codes

clear all; close all; clc

load handel

v = y';

n = length(v);

t = (1:length(y))/Fs; %define time domain

time = length(y)/Fs; % find the length of music

L = length(v)/Fs;

k=(2\*pi/L)\*[0:n/2 -n/2:-1]; ks=fftshift(k); %define fourier domain

define a time steps for sampling spectrogram

tslide = 0:0.1:time;

%three different spectrogram with gaussian gabor filter,

%width = 1

Svt\_spec = [];

for j = 1:length(tslide)

gf = exp(-1\*(t-tslide(j)).^2);

Vgf = gf.\*v;

Vgtf = fft(Vgf);

Svt\_spec = [Svt\_spec;fftshift(abs(Vgtf))];

figure(3)

subplot(3,1,1),plot(t,v,"k",t,gf,"r");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest, v(n)');

subplot(3,1,2),plot(t,Vgf,"k");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest Gabor, vg(n)');

subplot(3,1,3),plot(ks,abs(fftshift(Vgtf))/max(abs(Vgtf)))

xlabel('frequency (\omega)'), ylabel('FFT(vg)')

end

figure(4)

pcolor(tslide,ks,Svt\_spec.'),shading interp

title('the Gaussian Gabor filter, width = 1');

colormap(hot)

%width = 100

Svt\_spec2 = [];

for j = 1:length(tslide)

gf = exp(-100\*(t-tslide(j)).^2);

Vgf = gf.\*v;

Vgtf = fft(Vgf);

Svt\_spec2 = [Svt\_spec2;abs(fftshift(Vgtf))];

figure(5)

subplot(3,1,1),plot(t,v,"k",t,gf,"r");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest, v(n)');

subplot(3,1,2),plot(t,Vgf,"k");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest Gabor, vg2(n)');

subplot(3,1,3),plot(ks,abs(fftshift(Vgtf))/max(abs(Vgtf)))

xlabel('frequency (\omega)'), ylabel('FFT(vg2)')

end

figure(6)

pcolor(tslide,ks,Svt\_spec2.'),shading interp

title('the Gaussian Gabor filter, width = 100');

colormap(hot)

%width = 1000

Svt\_spec3 = [];

for j = 1:length(tslide)

gf = exp(-10000\*(t-tslide(j)).^2);

Vgf = gf.\*v;

Vgtf = fft(Vgf);

Svt\_spec3 = [Svt\_spec3;abs(fftshift(Vgtf))];

figure(7)

subplot(3,1,1),plot(t,v,"k",t,gf,"r");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest, v(n)');

subplot(3,1,2),plot(t,Vgf,"k");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest Gabor, vg2(n)');

subplot(3,1,3),plot(ks,abs(fftshift(Vgtf))/max(abs(Vgtf)))

xlabel('frequency (\omega)'), ylabel('FFT(vg2)')

end

figure(8)

pcolor(tslide,ks,Svt\_spec3.'),shading interp

title('the Gaussian Gabor filter, width = 1000');

colormap(hot);

%oversampling gaussian Gaborfilter, width = 100.

tslide\_over = 0:0.01:L;

Svt\_spec = [];

for j = 1:length(tslide\_over)

gf = exp(-100\*(t-tslide\_over(j)).^2);

Vgf = gf.\*v;

Vgtf = fft(Vgf);

Svt\_spec = [Svt\_spec;abs(fftshift(Vgtf))];

figure(9)

subplot(3,1,1),plot(t,v,"k",t,gf,"r");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest, v(n)');

subplot(3,1,2),plot(t,Vgf,"k");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest Gabor, vg(n)');

subplot(3,1,3),plot(ks,abs(fftshift(Vgtf))/max(abs(Vgtf)))

xlabel('frequency (\omega)'), ylabel('FFT(vg2)')

end

figure(10)

pcolor(tslide\_over,ks,Svt\_spec.'),shading interp

title('oversampling gaussian Gaborfilter, width = 100');

colormap(hot);

%undersampling gaussian Gaborfilter, width = 100.

tslide\_under = 0:0.5:L;

Svt\_spec = [];

for j = 1:length(tslide\_under)

gf = exp(-100\*(t-tslide\_under(j)).^2);

Vgf = gf.\*v;

Vgtf = fft(Vgf);

Svt\_spec = [Svt\_spec;abs(fftshift(Vgtf))];

figure(11)

subplot(3,1,1),plot(t,v,"k",t,gf,"r");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest, v(n)');

subplot(3,1,2),plot(t,Vgf,"k");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest Gabor, vg(n)');

subplot(3,1,3),plot(ks,abs(fftshift(Vgtf))/max(abs(Vgtf)))

xlabel('frequency (\omega)'), ylabel('FFT(vg2)')

end

figure(12)

pcolor(tslide\_under,ks,Svt\_spec.'),shading interp

title('undersampling gaussian Gaborfilter, width = 100.');

colormap(hot);

Mexican hat wavelet

Svt\_spec = [];

tao = 0.1;

for j = 1:length(tslide)

gf = (2./(sqrt(3.\*tao).\*(pi.^(1./4)))).\*(1-((t-tslide(j))./tao).^2).\*exp(-(((t-tslide(j)).^2)./(2.\*tao.^2)));

Vgf = gf.\*v;

Vgtf = fft(Vgf);

Svt\_spec = [Svt\_spec;abs(fftshift(Vgtf))];

figure(13)

subplot(3,1,1),plot(t,v,"k",t,gf,"r");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest, v(n)');

subplot(3,1,2),plot(t,Vgf,"k");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest Gabor, vg2(n)');

subplot(3,1,3),plot(ks,abs(fftshift(Vgtf))/max(abs(Vgtf)))

xlabel('frequency (\omega)'), ylabel('FFT(vg2)')

end

figure(14)

pcolor(tslide,ks,Svt\_spec.'),shading interp

title('Mexican hat wavelet');

colormap(hot);

%step-function (Shannon) window.

Svt\_spec = [];

step\_width = 0.2;

for j = 1:length(tslide)

gf = abs((t-tslide(j)))<=step\_width;

Vgf = gf.\*v;

Vgtf = fft(Vgf);

Svt\_spec = [Svt\_spec;abs(fftshift(Vgtf))];

figure(15)

subplot(3,1,1),plot(t,v,"k",t,gf,"r");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest, v(n)');

subplot(3,1,2),plot(t,Vgf,"k");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest Gabor, vg2(n)');

subplot(3,1,3),plot(ks,abs(fftshift(Vgtf))/max(abs(Vgtf)))

xlabel('frequency (\omega)'), ylabel('FFT(vg2)')

end

figure(16)

pcolor(tslide,ks,Svt\_spec.'),shading interp

title('step-function (Shannon) window wavelet');

colormap(hot);

clear all; close all; clc

%Part II

figure(17)

[y,Fs] = audioread('music1.wav');

tr\_piano=length(y)/Fs; % record time in seconds

plot((1:length(y))/Fs,y');

xlabel('Time [sec]'); ylabel('Amplitude');

title('Mary had a little lamb (piano)');

v = y';

t = (1:length(y))/Fs;

tslide = 0:0.1:tr\_piano;

n = length(v);

L = length(v)/Fs;

k=(2\*pi/L)\*[0:n/2-1 -n/2:-1]; ks=fftshift(k);

Svt\_spec = [];

for j = 1:length(tslide)

gf = exp(-100\*(t-tslide(j)).^2);

Vgf = gf.\*v;

Vgtf = fft(Vgf);

Svt\_spec = [Svt\_spec;abs(fftshift(Vgtf))];

figure(19)

subplot(2,1,1),plot(t,v,"k",t,gf,"r");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest, v(n)');

subplot(2,1,2),plot(t,Vgf,"k");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest Gabor, vg(n)');

end

figure(18)

pcolor(tslide,ks,Svt\_spec.'),shading interp

ylim([-5000,5000]);

title('Mary had a little lamb (piano)');

colormap(hot);

figure(19)

[y,Fs] = audioread('music2.wav');

tr\_rec=length(y)/Fs; % record time in seconds

plot((1:length(y))/Fs,y);

xlabel('Time [sec]'); ylabel('Amplitude');

title('Mary had a little lamb (recorder)');

v = y';

t = (1:length(y))/Fs;

tslide = 0:0.1:tr\_rec;

n = length(v);

L = length(v)/Fs;

k=(2\*pi/L)\*[0:n/2-1 -n/2:-1]; ks=fftshift(k);

Svt\_spec = [];

for j = 1:length(tslide)

gf = exp(-100\*(t-tslide(j)).^2);

Vgf = gf.\*v;

Vgtf = fft(Vgf);

Svt\_spec = [Svt\_spec;abs(fftshift(Vgtf))];

figure(20)

subplot(2,1,1),plot(t,v,"k",t,gf,"r");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest, v(n)');

subplot(2,1,2),plot(t,Vgf,"k");

xlabel('Time [sec]');

ylabel('Amplitude');

title('Signal of Interest Gabor, vg(n)');

end

figure(21)

pcolor(tslide,ks,Svt\_spec.'),shading interp

ylim([-8000,8000]);

title('Mary had a little lamb (recorder)');

colormap(hot);

[Published with MATLAB® R2019b](https://www.mathworks.com/products/matlab/)