Physics 640
November 4, 2011

Project 5
Week 6 & 7: Time-Frequency Analysis and Fractals
Project 5: Pick one option below for this project

a) Develop a Matlab code that allows you to recognize a vowel from a recording (wav file) using the Wigner transform.
b) Write a Matlab code to demonstrate the advantage of using wavelet transform versus Fourier transform in signal recovery
    Matlab code to calculate the <u>Wigner function</u>; and <u>wavelet</u> subroutines will be available
c) Work on the Mandelbrot Fractal Fortran Code, then visualize using Matlab, make a movies to show the "self-similarity" zooming in effects of fractals

Option a)
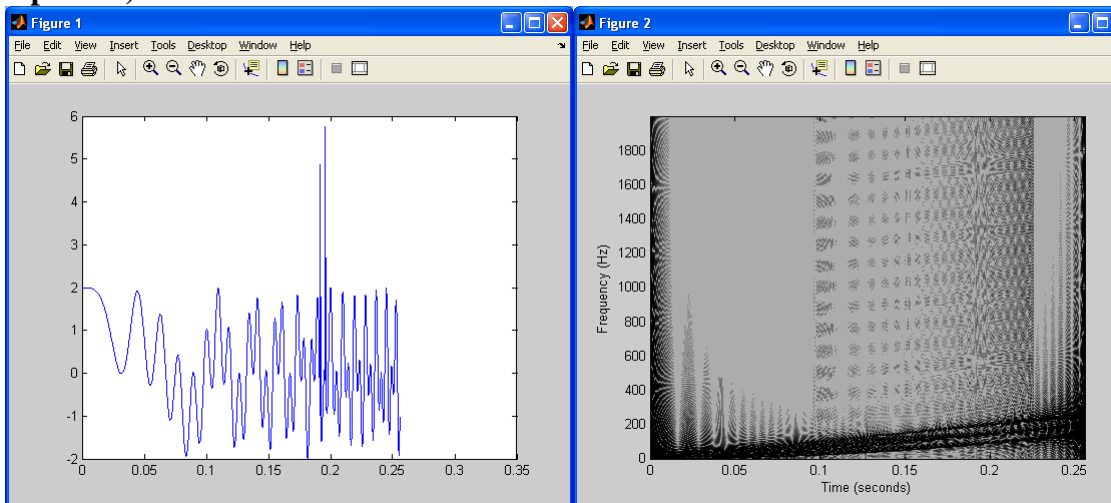
Chirp signal with two perturbations in time   Wigner transform showing linear dependence of frequency w.r.t. time

Start by recording one vowel to use as a gold standard. Should be able to determine if a given wav file contains that same vowel or not by doing correlation between the two Wigner function 2D plots ("corr2")

Human voice processing is very intensive, will need to use a very short piece of your recording.

How do we get the wav file ?

-Plug the microphone into the back of the computer, then use Microsoft Sound Recorder (under Accessories/Entertainment). Use "Save As" to save it into a .WAV file.
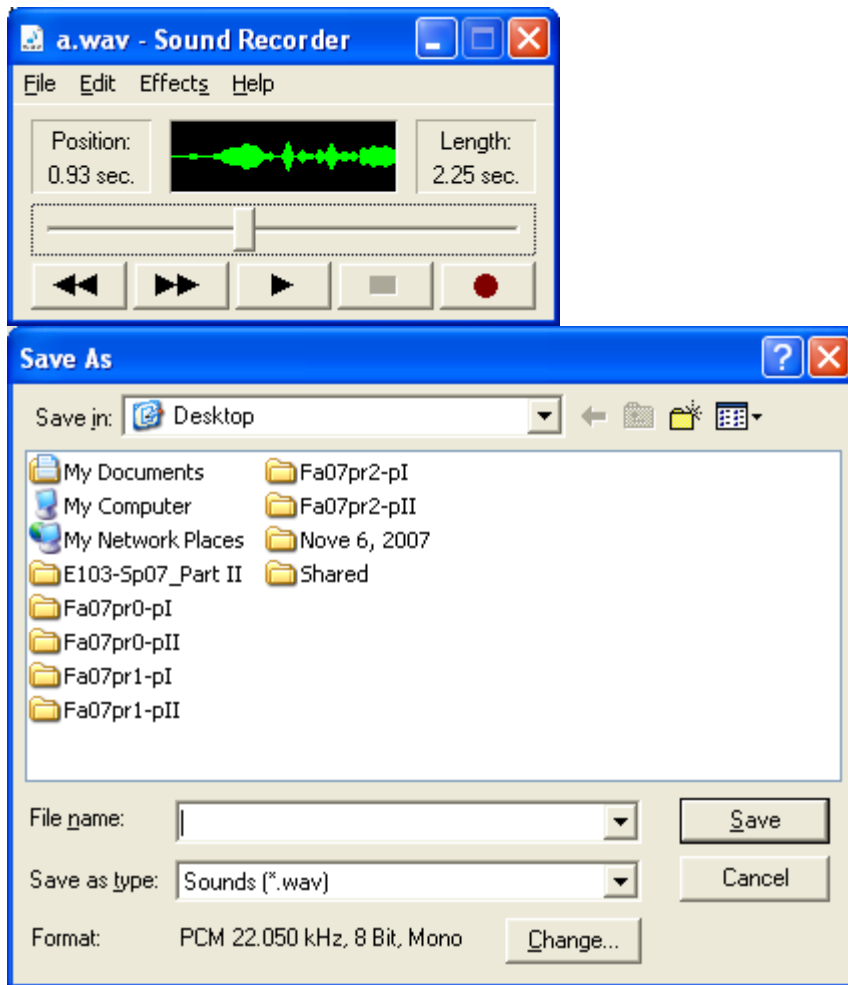
You can do some manual editing of the sound sequence by using the sliding bar and select "Delete Before/After Current Position"

What reference files should we use?

To identify a vowel between a,e,i,o,u, record at least one file per vowel, then perform the Wigner transform on each vowel. When you are given a new file, perform the Wigner transform, then use "corr2" to find max correlation with your five references. Suppose the new file has highest correlation with a reference file containing an "e", then you say the new file contains a recording of an "e"
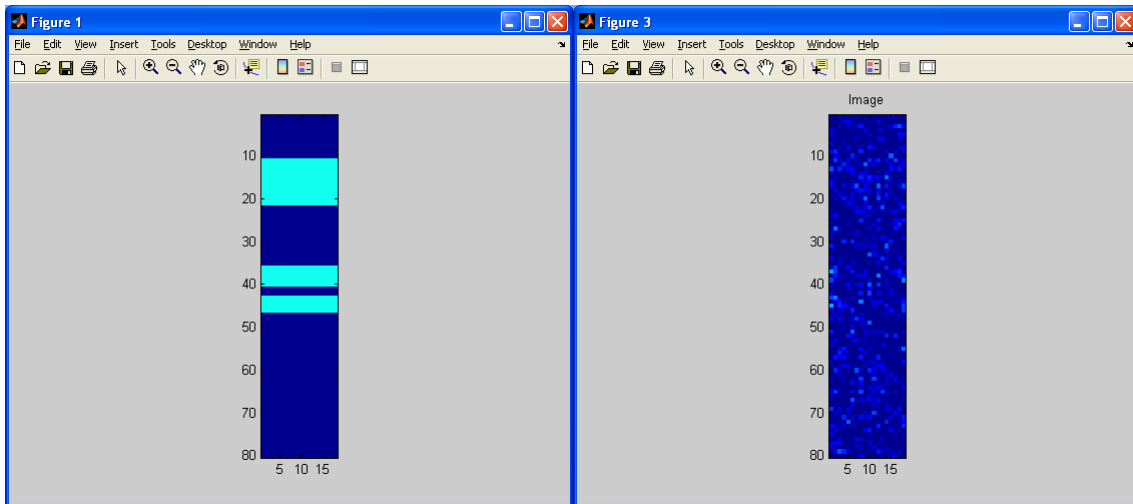
How to perform the Wigner transform?

The Matlab code provided does not start with a subroutine name and defined arguments as with the "perform_wavelet_transform"
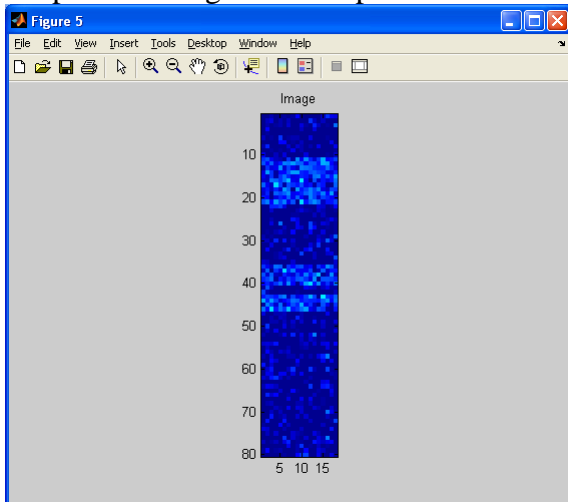```
%Program wvd.m
```
In this case you can embed your code within this same file. You will need to replace the defined signal f with the input from your WAV file.

**Option b)**

2D pattern or signal with a perturbation in the lower band    Masked by added noise



Recovered pattern using wavelet transform

The code provided, "perform_wavelet_transform.m", can perform direct and inverse 2D wavelet transforms. You would need to create your own 2D signal, add noise and do the analysis to recover the signal . Write your own Matlab code, when wavelet transform is needed, call in this subroutine.

```
y = perform_wavelet_transform(x, Jmin, dir, options);
%
%    'x' is either a 1D or a 2D array.
%    'Jmin' is the minimum scale (i.e. the coarse channel is of size 2^Jmin
%        in 1D).
%    'dir' is +1 for fwd transform and -1 for bwd.
%    'options.wavelet_vm' is the number of Vanishing moment (both for primal and
dual).
%    'options.wavelet_type' can be
%        'daubechies', 'symmlet', 'battle', 'biorthogonal'.
```

How to call a subroutine in Matlab?
-Use m.file name containing the subroutine (without the .m), plus the required arguments (within parentheses). For the perform_wavelet_transform:
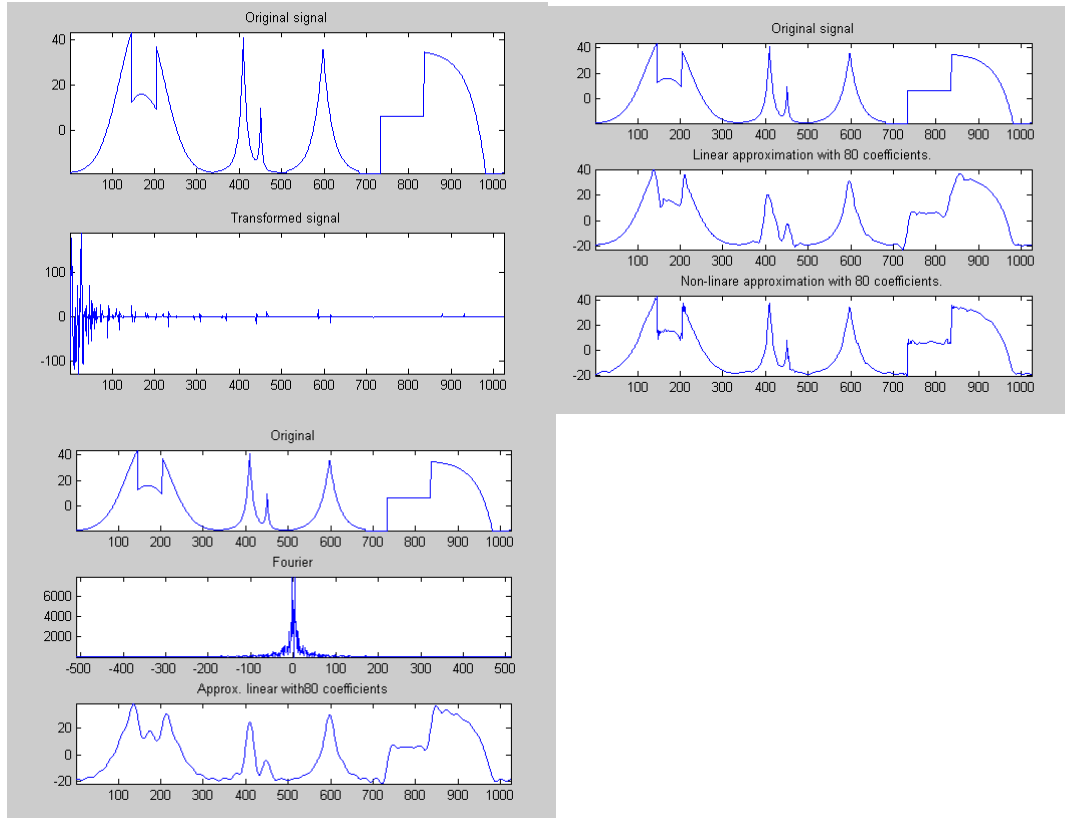a) signal x
b) use Jmin = 5.5
c) "dir" indicates whether you want to do direct (+1), or inverse (-1) wavelet transform

For example: perform_wavelet_transform(x, 5.5, +1) will do the direct wavelet transform on the signal x
Currently this subroutine works with 2D signal x

More information on Project 5 Option b):

The latest version of 'perform_wavelet_transform.m' is on the website, please download and use this one. This updated version requires a power of 2 length for 1D and 2D data; and 2D data should also be a square.

The figures below show the ability of wavelet transform to recover sharp edges in a 1D signal, over the Fourier transform.


These figures below show the signal with its wavelet transforms in 1D (left) and a 2D (right)
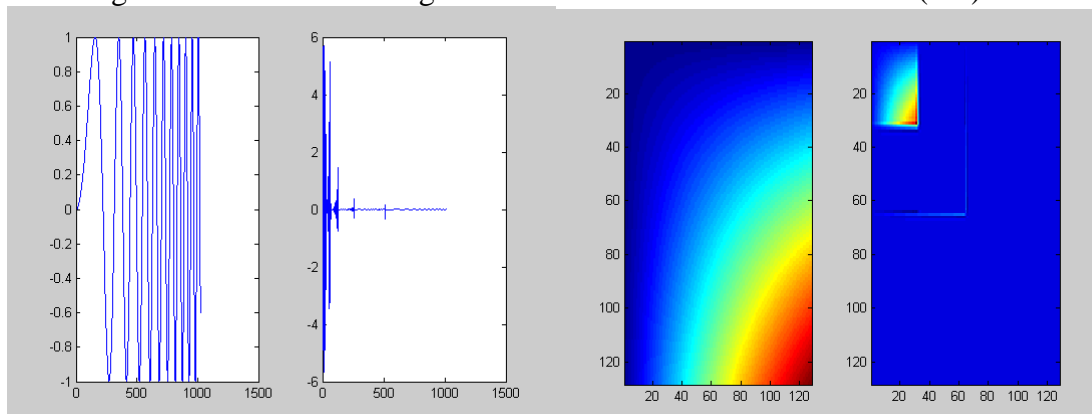
Below, figure 1 shows the original 2D signal with a perturbation (square dot), and the same signal with an added noise (30% of the signal amplitude). Figure 4 shows the original signal, the Fourier and wavelet transform of the same signal with added noise. The perturbation (square dot) is not seen in the Fourier transform but yes in the wavelet transform.
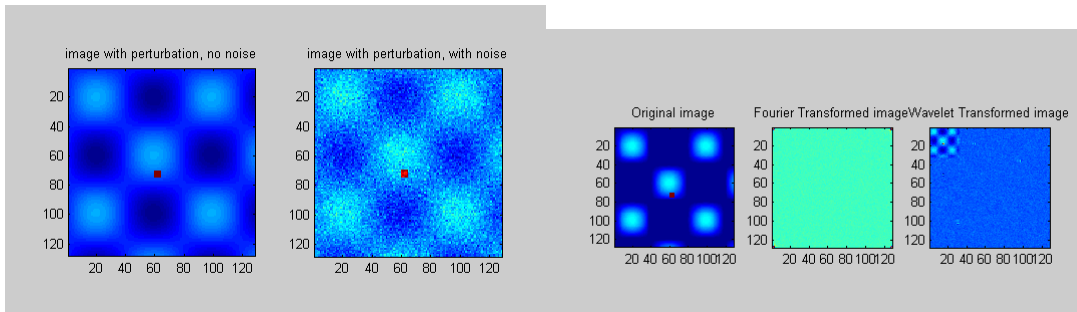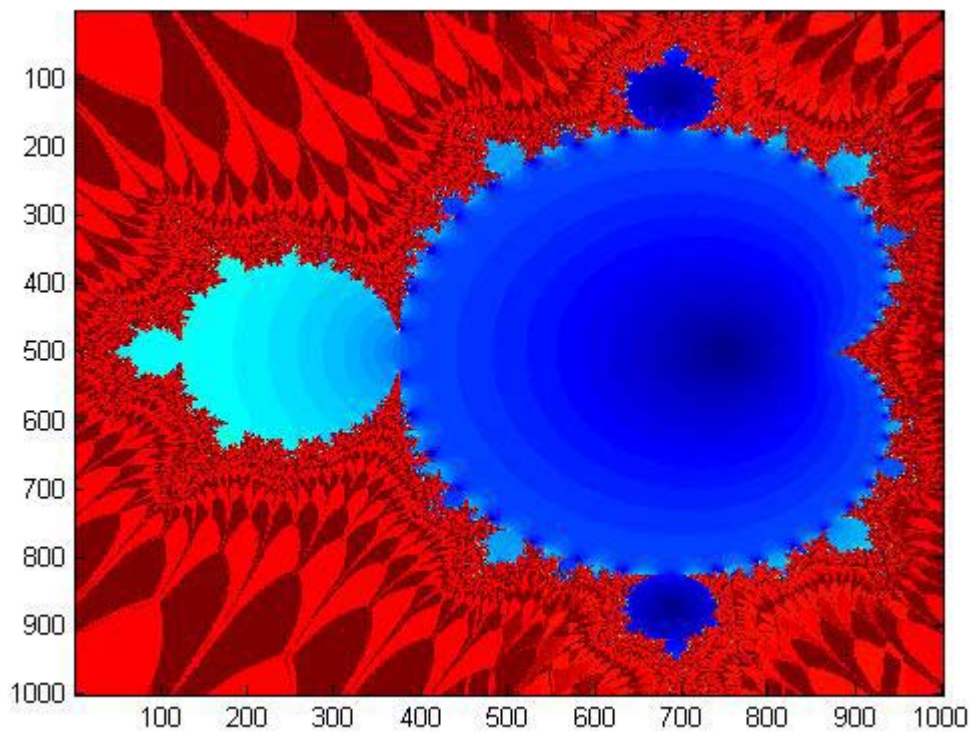
Below are the results of noise elimination using the Fourier transform (Fig 3) and the wavelet transform (Fig 5)
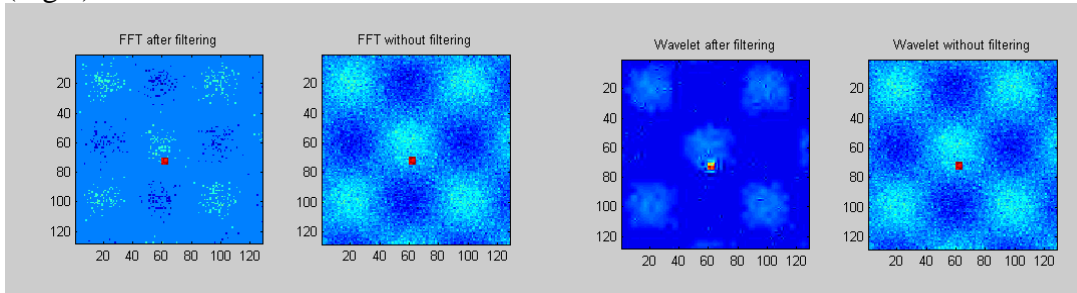
Option c)

The Mandelbrot Fractal

Here is an example code to calculate the Mandelbrot fractal

```
      program mandel
C        FORTRAN77 code to generate a Mandelbrot fractal
      implicit none
      integer npts
C     Number of points in side of image
      parameter (npts=1000)
```

```fortran
      real*8 zRe(npts,npts)
      real*8 zIm(npts,npts)
      real*8 kRe(npts,npts)
      real*8 kIm(npts,npts)
      real*8 qRe(npts,npts)
      real*8 qIm(npts,npts)
      integer i,j,k,niter
C        Number of iterations in the Mandelbrot fractal calculation
      niter=51
      do j=1,npts
       do i=1,npts
C        Generating z = 0 (real and imaginary part)
       zRe(i,j)=0.
       zIm(i,j)=0.
C        Generating the constant k (real and imaginary part)
         kRe(i,j)=dble(i)*2.0/(dble(npts)-1.)-1.5
         kIm(i,j)=dble(j)*2.0/(dble(npts)-1.)-1.
         enddo
      enddo


C        Iterating
      do k=1,niter
         do j=1,npts
          do i=1,npts
C        Calculating q = z*z + k in complex space
C        q is a temporary variable to store the result
      qRe(i,j)=zRe(i,j)*zRe(i,j)-zIm(i,j)*zIm(i,j)+kRe(i,j);
      qIm(i,j)=2.*zRe(i,j)*zIm(i,j)+kIm(i,j);
C        Assigning the q values to z constraining between
C         -5 and 5 to avoid numerical divergences
                     zRe(i,j)=qRe(i,j);
                     zIm(i,j)=qIm(i,j);
                     if (zRe(i,j) < -5.) zRe(i,j)=-5.;
                     if (zRe(i,j) > 5.) zRe(i,j)=5.;
                     if (zIm(i,j) < -5.) zIm(i,j)=-5.;
                     if (zIm(i,j) > 5.) zIm(i,j)=5.;
          enddo
         enddo
      enddo
C   You will need to output zRe and zIm to a file here

      end program mandel
```

What to plot for the fractal?

log(sqrt(zRe.*zRe+zIm.*zIm)+1)

Please note the timeline for the Term Project below.

November 2011

S	M	T	W	Th	F	Sat.
		1	2	3	4	5
6	7	8	9	10	11 Term Proj. Selection	12
13	14	15	16	17	18 Midterm Pr 5 due	19
20	21	22	23	24	25	26

December 2011

S	M	T	W	Th	F	Sat.
27	28	29	30	1	2 Term Proj. Progress Report due	3
4	5	6	7	8	9 Term Proj. Presentations	10
11	12	13 Last day of classes	14	15	16 Term Proj. Presentations	17
18	19 Term Proj. Report due	20				