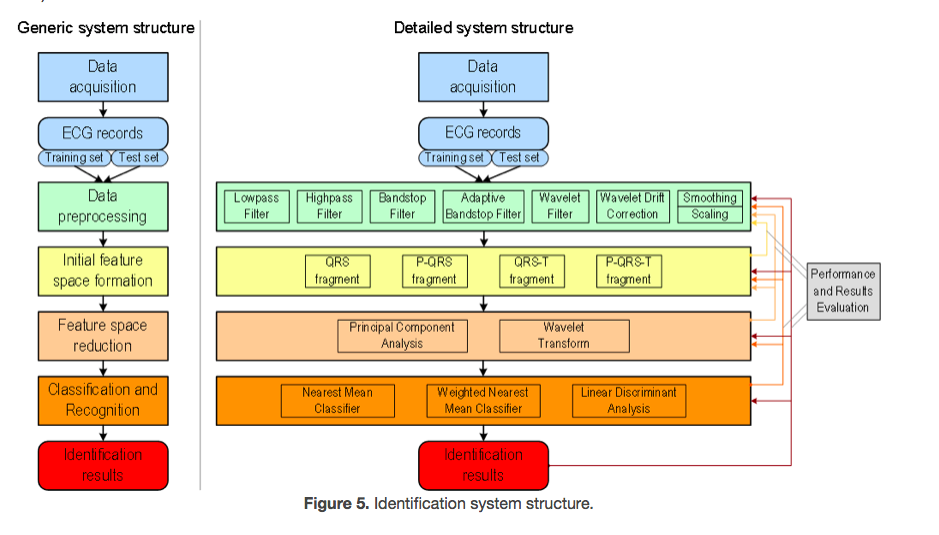
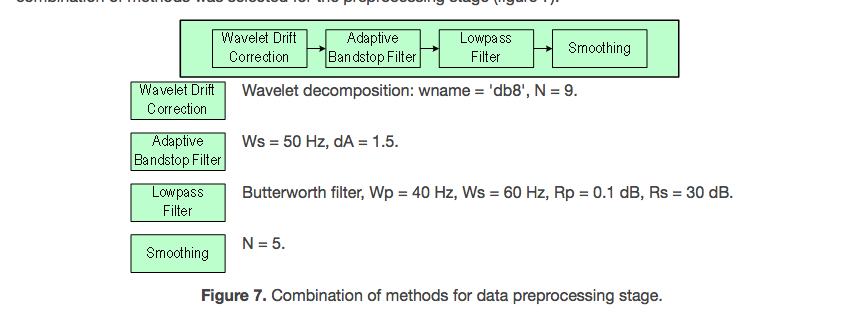
As we’ve discussed PhysioNet is an amazing resource for our project.

<https://physionet.org/pn3/ecgiddb/biometric.shtml> is the paper or part of it at least that’s on ECG as a biometric identification feature.



This is the structure of the system they designed, note that there’s no obvious mention of fiducial point analysis.



Here are the values they used for a bunch of their filters, since our board does some processing we’ll likely have to use different values, and might not need as many filters.

Here’s a blurb on the PCA and wavelet methods:

Two methods of feature space reduction were considered: Principal Component Analysis (PCA) and Wavelet Transform (WT). Principal Component Analysis allows reduction of the initial feature space dimension N to 30 according to the Kaiser criterion, or even to 10 according to the scree test. Use of a Wavelet Transform provides the same space reduction but with slightly poorer PQRTS-fragment classification results (figure 13); furthermore, it is hard to find reasonable criteria for wavelet and decomposition level selection. In a series of experiments with different wavelets, the best results were obtained using Daubechies wavelet ('db3'), and decomposition at level 3.

Note that the graph they included showed somewhat greater accuracy for PCA than Wavelet.

Now for a bunch of links don’t worry about them too much for now:

1. This is the link to the database of signals from this research: [link](https://physionet.org/pn3/ecgiddb/)
2. Here you can get the csv versions if you want: [link](https://www.physionet.org/cgi-bin/atm/ATM?tool=&database=aami-ec13&rbase=&srecord=&annotator=&signal=&sfreq=&tstart=&tdur=&tfinal=&action=&tfmt=&dfmt=&nbwidth=)
3. Link to files to download to enable MATLAB or Octave to read in the files in their original formats (.head and .dat files): [link](https://physionet.org/physiotools/matlab/wfdb-app-matlab/)
4. Hookup guide for the ecg sensor: [link](https://learn.sparkfun.com/tutorials/ad8232-heart-rate-monitor-hookup-guide)
5. Code from guide above: [link](https://github.com/sparkfun/AD8232_Heart_Rate_Monitor/blob/master/Software/Heart_Rate_Display_Arduino/Heart_Rate_Display_Arduino.ino)

Recommended reading: link 4

They found the peaks in each dataset then decided to call them the R peak and selected 80 samples before the peak and 170 (169) samples after the peak then stacked them on each other, adjust for vertical drift then keep the 6 samples closest to the mean for analysis.

This code sample does the sample picking and stacking:

[pks,locs] = findpeaks(ecg1(:,3),ecg1(:,1),'MinPeakProminence',0.45,'Annotate','extents');

for i = 1:length(pks)

    j = find(ecg1(:,1) == locs(i));

    k = j - 80;

    plot(ecg1(1:250,1), ecg1(k:k+249,3))

    hold on;

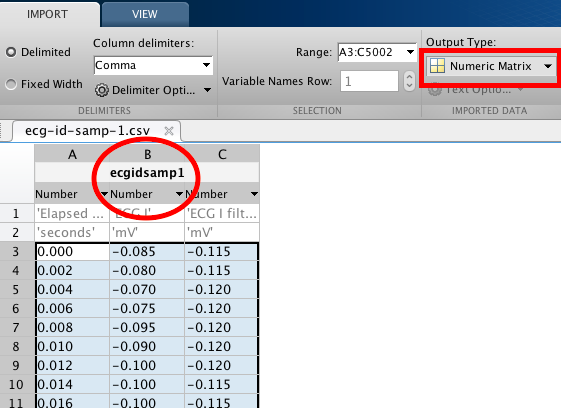
end

You need to import into matlab the data set that I sent along with this as “ecg1”

Step 1:



Step 2: make sure the data type matches the one in the rectangle and change the name in the circle to ecg1



Step 3 copy paste code

Step 4 ????

Step 5 Graph ta-da!