Spring 2016	Report#7 - 02/26/2016	Ahmet Can Ozbek

I. Tasks achieved last week

- Started on classification by SVM and obtained classification results.
- PQRST peak detection of the ECG signal is completed.

II. Feedback and Interaction

- ***Prof. Kuo's F/I***
  - Speaker Identification is similar to our study of ECG biometrics, so those papers should be analyzed further.

III. Report

1. **Removing the noisy segments from the ECG signal**

   In CYBHi dataset some ecg signals have noisy segments, which means that some segments of the ECG recordings are very off from the general amplitude values. From the paper "Dry Contact Fingertip ECG-based Authentication System using Time, Frequency Domain Features and Support Vector Machine", the basic idea of the method to solve this issue is as follows:
   1)The ECG signal is segmented into windows of 1 second duration
   2)For each window, modified coefficient of variance (MCV) is calculated. MCV is basically variance over mean square.
   3)Windows that have MCV higher than a threshold are removed from the signal.
   This implementation is useful now and might also be useful in the future studies of ECG datasets. Some example results can be seen in figure 1 and figure 2.

2. **Identification from ECG signals**

   I am using the methods explained in the paper "Dry Contact Fingertip ECG-based Authentication System using Time, Frequency Domain Features and Support Vector Machine" as a starting point for the identification of individuals from ECG signals. For classification, the feature vector is basically the amplitude values of the ECG signal around the R peaks. From this idea, my implementation and the results I got is as follows:

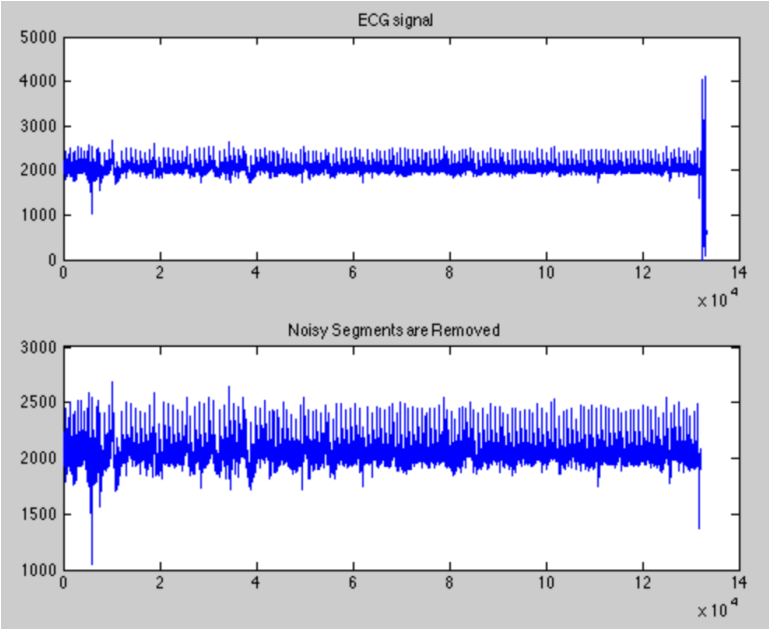

Figure 1: Removal of noisy segments of ECG signal (The noisy part at the end of the signal is removed by this implementation)

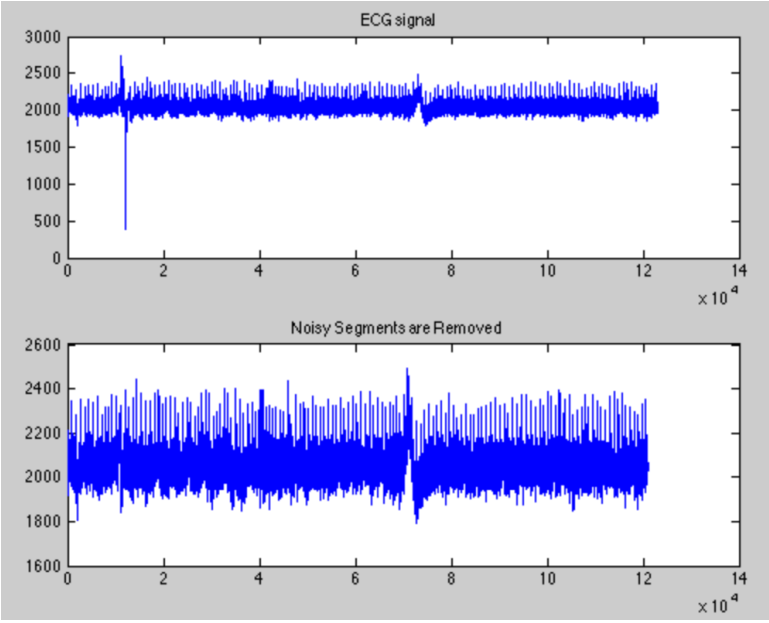

Figure 2: Removal of noisy segments of ECG signal (The noisy part at approximately one fifth of the signal is removed by this implementation)

- Dataset and Training Testing Data Partition:
  I am using CYBHi dataset for this study. This dataset has ECG recording of 63 subjects. In each recording of subjects, first 80 heartbeats are used for training data and the next 15 heartbeats are used for testing data.
- Creating train and test data:
  1)First, noisy segments are removed from the ECG signal by the implementation mentioned in the first part of this report.
  2)ECG signal is filtered by a bandpass filter with cutoff frequncies 0.45Hz and 35Hz
  3)R peaks are detected by Pan Tompkin's algorithm.
  4)For each heartbeat, 250 samples before and 250 samples after the R peak is cropped to be used as a temporal feature vector (R-250, R+250)
  5)For each heartbeat, 80 samples before and 80 samples after the R peak is cropped and taken 50 point FFT to be used as a frequency component feature vector.
  6)Temporal feature vector and frequncy component feature vector is cascaded to make one feature vector for each heartbeat.
  (temporal feature vector + frequency component feature vector = Dimension of the feature vector)
  (250 + 250 + 1) + (50) = 551
  As a result dimensionality of the feature vector for each heartbeat is 551.
  7)The training and testing data is normalized between 0 and 1 for SVM classification.
- SVM classifier and results:
  By using SVM with RBF Kernel with parameters: gamma = 0.02 and C = 1000, the results on training and testing data is as follows:
  *Training Data: Accuracy = **98.7302%** (4976/5040) (classification)
  *Testing Data: Accuracy = **90.582%** (856/945) (classification)

  As we can see from the results, this implementation gives promising results which is more than 90 percent. By fine-tuning the parameters, this result can be improved further. This implementation is done by MATLAB.

IV. Plan for the next week

- Try to get results similar to the papers using feature extraction techniques
- Start on deep learning

V. Milestone

- PQRST peak detection of the ECG signals in CYBHi data is completed.
- A significant accuracy for identification from ECG signal is obtained by implementation.