ECE4095 Trustable Computing Systems Lab 2
February 8, 2013

Keyboard Acoustic Emanations

The goal of this lab is to try and detect keyboard presses by analyzing the acoustic signal generated by a key press. The lab uses a technique described in the paper "Keyboard Acoustic Emanations" by Asonov and Agrawal in the 2004 IEEE Symposium on Security and Privacy. The lab will use MATLAB to collect the audio, process it, and then use a neural network to classify the signals.

1. Record the Signals

Connect a microphone to the computer with MATLAB and place it next to the target keyboard that you are trying to intercept signals from. The keyboard should ideally not be the same keyboard that is connected to the MATLAB computer.

In MATLAB, issue the following command:

```
>> r = audiorecorder(44100,16,1);
```

This will create a recorder object that records one channel of data at 44.1KHz with 16-bit samples

```
>> record(r);
```

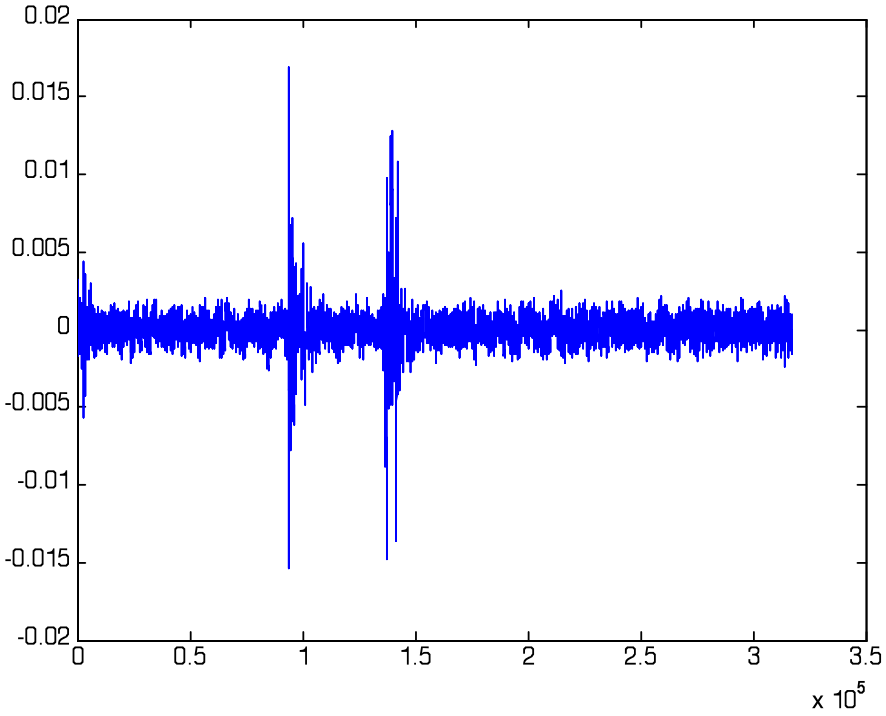
This will start the recording. You can now type on the target keyboard that you want to record. To start, just press two keys relatively slowly.

```
>> stop(r);
```

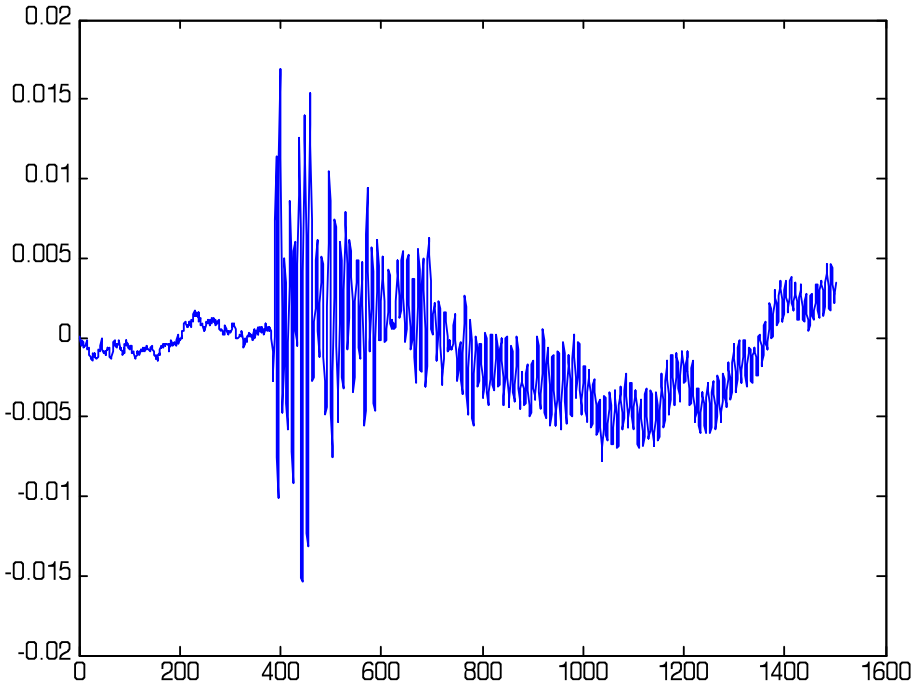
This will stop the recording. Once you have captured the recording, you will need to extract the raw data into a MATLAB array as follows:

```
>> y=getaudiodata(r,'double');
>> plot(y);
```

The plot may look something like the following:



There are peaks that correspond to key presses. Plot subarrays of y to zoom in on just the peaks. They will look something like the following:



Copy this subarray into a new array.

```
>> a=y(16500:17500);
```

Do the same for the other key press as well.

2. **Process the Signals**

Now, for each array, we want to do an FFT to find the frequency spectrum for the key presses.

```
>> N=512;
>> Xa=fft(a,N);
>> Xb=fft(b,N);
```

Now, we have two 512-point FFTs (`Xa` and `Xb`) for the two keys.

3. **Classify the Signals**

Using these FFTs, we can create a neural network that can be used to guess key presses. MATLAB has a neural network toolbox that can be used for pattern recognition and classification.

```
>> X = [Xa Xb];
>> T = [0 1];
>> net = newpr(X, T, 20);
```

The `X` array is the set of input vectors to the neural net, `T` is the set of target outputs. `X` is a 512x2 array with 2 input vectors each with 512 data points. The target array has two values that correspond to the 2 input vectors. A 0 indicates the corresponding input vector is key 'a' and a 1 indicates the input vector is for key 'b'. If you collect more input data – i.e. more recordings of the two keys – you can add them as input vectors to `X`. The corresponding values will need to be added to the target vector as well. The more inputs you have, the more accurate the neural network.

Now, train the network.

```
>> net = train(net, X, T);
```

This, will bring up a new window that shows the neural network training progress.

Now that the network has been trained, you can use the neural network to evaluate a new input FFT.

```
>> sim(net, Xb)

ans =
    0.9999 + 0.00001
```

A value close to 1 indicates that the key was probably the 'b' key.

4. Lab Assignment

Since neural networks work best with multiple inputs, collect 5 samples/FFTs for each key. That will give you a 512x10 `X` array and a 1x10 `T` array. Make sure that the target array values match up with the appropriate columns in the inputs array. Once you've trained the network, use two more samples of each key to test the neural network.

What to hand in:

1. Show two plots of the acoustic signal – one for each key.
2. Show the four results of your tests. Did the neural network correctly identify the keys? What level of confidence do you have in the identification?
3. Write a paragraph on using acoustic emanations as an attack. Comment on the effectiveness and practicality. Are there ways you could improve the attack? Can you think of any other systems that might provide useful acoustic emanations. How about countermeasures to the attack.