Duel-Tone Multi-Frequency Detection Using Modified Goertzel Algorithm

John Clapham & Alex Topping, Department of Electrical Engineering, Bradley University
Peoria, Illinois 61625 U.S.A

Using duel-tone multi-frequency (DTMF) signaling has become a standard within the telecommunications industry. DTMF signaling has many advantages over the standard signaling scheme. The DTMF signaling scheme provides accuracy in detection methods, even with corruption of data. The DTMF scheme can be seen in Figure 1.

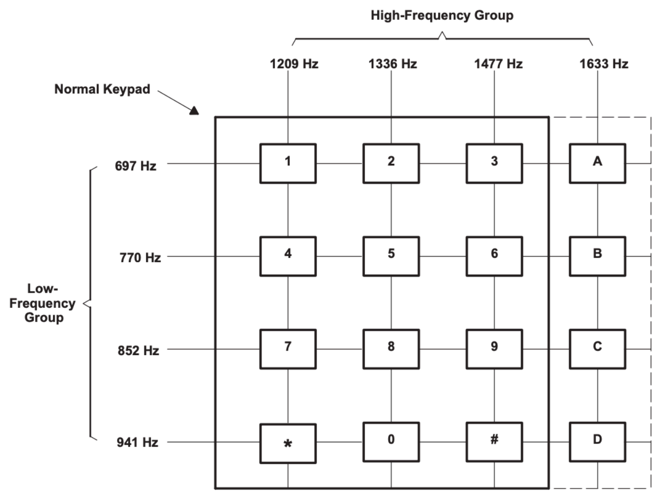


Figure 1: DTMF Keypad Scheme

Figure 1 shows both a high frequency and a low frequency for each digit on the keypad. It is shown that the high frequency group remains between 1KHz and 2KHz, where as the low frequency group is within the range of 600Hz to 1KHz. These frequencies were selected so that the DTMF signal can be distinguished accurately even in the presence of noise or corruption.

Three datasets were provided by Dr. Yufeng Lu of Bradley University that contains data of ten dialed digits. These datasets were recorded using MATLAB, and have been sampled at 44.1KHz, with 16 bits per sample.

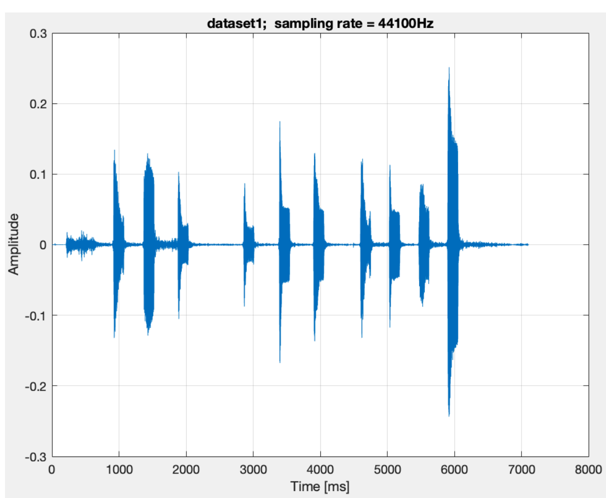
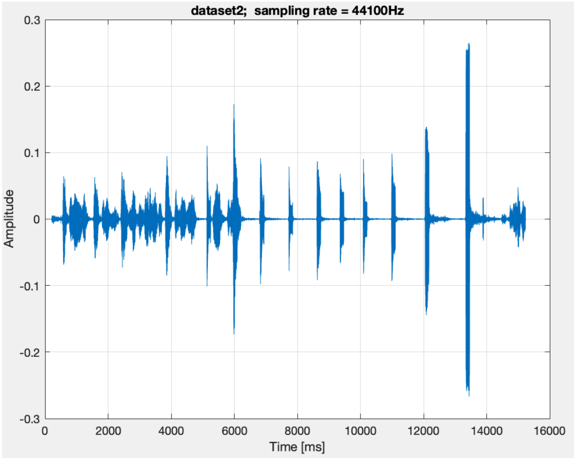


Figure 2: Dataset 1 - Clean Signal



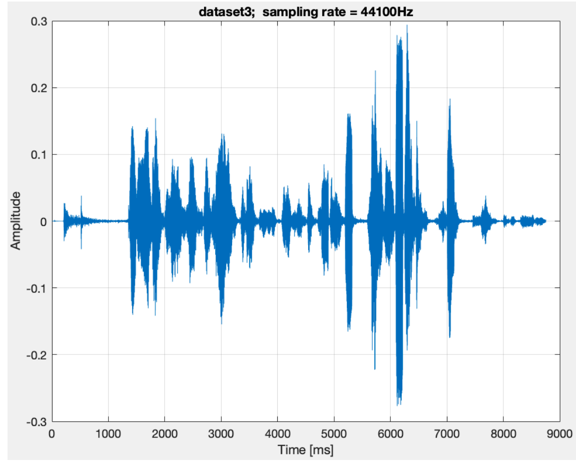


Figure 3: Dataset 2 - Light Vocal Noise

Figure 4: Dataset 3 - Heavy Vocal Noise

It can be seen in Figure 2, that there have been 10 digit tones detected within about 6 seconds. There is little noise shown throughout the recording. The 10 dialed digits remain consistent over all three datasets, however, Figure 3 and Figure 4 show data corruption within the dialed digits. This posses a problem for accurately detecting both the high and low frequencies needed to find the associated number that had been dialed.

The Modified Goertzel Algorithm can be implemented in order to detect DTMF effectively and accurately, even with corruption of the DTMF. The Goertzel algorithm is used to examine the energy levels at all eight frequencies starting from 697Hz to 1633Hz as shown in Figure 1. The two highest energy levels will correspond to both a high frequency and a low frequency. This will determine the digit that had been dialed.

The Modified Goertzel Algorithm is described as follows.

$$K = N * (fi/fs) \tag{1}$$

K is calculated eight times with fs = sampling frequency, and fi = each low and high frequency. N is carefully chosen to be the filter length.

Then, eight vk(n) values are calculated for each K value using a recursive difference equation.

$$vk(n) = 2*\cos(2*pi*K/N) *vk(n-1)-vk(n-2)+x(n) \tag{2}$$

where,
n = 0,1,2,…N
vk(-1) and vk(-2) are assumed to be zero.

After, with the same K value as used for each vk(n), eight different yk(N) values are found.

$$yk(N) = vk(N)-Wkn*vk(N-1) \tag{3}$$

where,
Wkn = exp(-2*pi*k/N)

Finally, in order to obtain the energy levels of all eight outputs of yk(N), squaring is needed.

Energy Levels = (yk(N))^2 (4)

Equation 4 will have eight different energy levels for each frequency of the DTMF scheme. There will be two maximum energy levels out of all eight calculated energy levels. The two max energies will correspond to a high frequency and a low frequency, thus differentiating between each possible digit. Figure 5 shows the block diagram of the implemented algorithm.

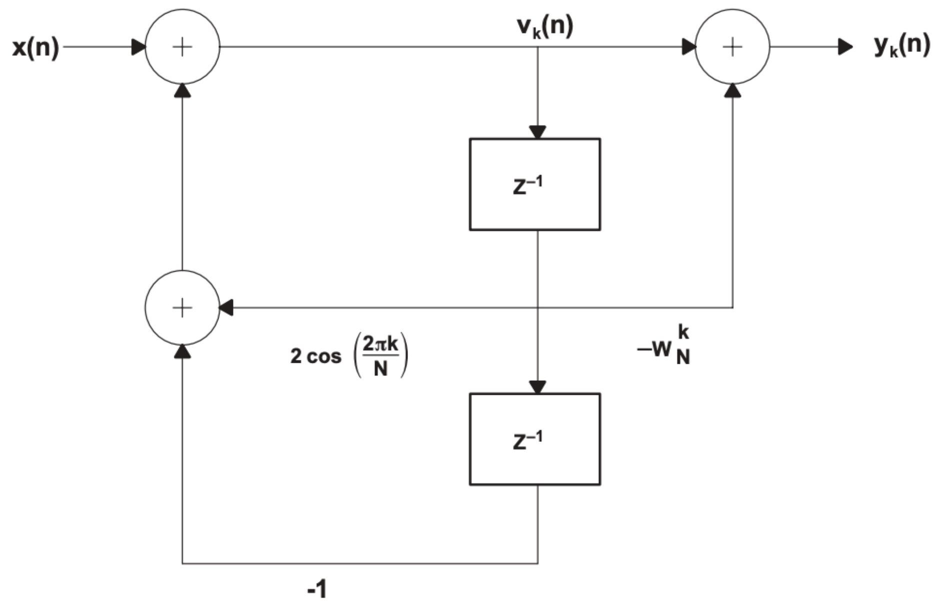


Figure 5: Goertzel Algorithm Block Diagram

This works well even with noise introduced. This is because even with corruption to each digit dialed, the calculated energies are only for the eight different possible frequencies of the DTMF scheme. So with whatever frequencies that have been introduced as noise, they can be effectively filtered out, and only the expected frequencies are calculated for. Even if the noise adds energy to each frequency, it will add to the maximum value as well, retaining the information needed to determine the digit that had been dialed.

It is expected that if noise is introduced specifically at the eight frequency values (thus overwhelming the max energy levels at unexpected frequencies) this method would not be effective in determining the digits dialed.

The Goertzel Algorithm was implemented as a function titled DTMFfinder.m. Once the energy levels are found for each digit, the energy levels are passed through a function titled numberfinder.m. This function finds the two maximum energy levels and compares them to the high and low DTMF. The digit is then found and returned to the main script (ECE301_DTMF.m). The location of each digit was found through observation and broken into segments manually.

A main issue that was faced was finding the correct value of N to calculate K values (shown in Equation 1). Choosing different N values seems to effect the output entirely. Eventually, the N value was set to the same length as the window used to segment Dataset 1. N was chosen to be 6000. Table 1 shows the calculated K values when N = 6000, with a sampling frequency of 44.1KHz.

Sampling Rate = 44.1 Hz, N = 6000	
Frequency	**K**
697	95
770	105
852	116
941	128
1209	164
1336	182
1477	201
1633	222

Table 1: Calculated K values for N = 6000

Eventually, the outputs for datasets 1 through 3 were found to be the expected output. This is shown in Figure 6.

```
number1 =

     3     0     9     6     7     7     2     7     3     4


number2 =

     3     0     9     6     7     7     2     7     3     4


number3 =

     3     0     9     6     7     7     2     7     3     4

>>
```

Figure 6: Numbers Found for Datasets 1 through 3.

A possible improvement that could be made would be to segment the digits automatically by creating another function that detects the start and end of each digit. This would be difficult when dealing with noise in the system, however if this is done, the Goertzel Algorithm could be implemented in real time.