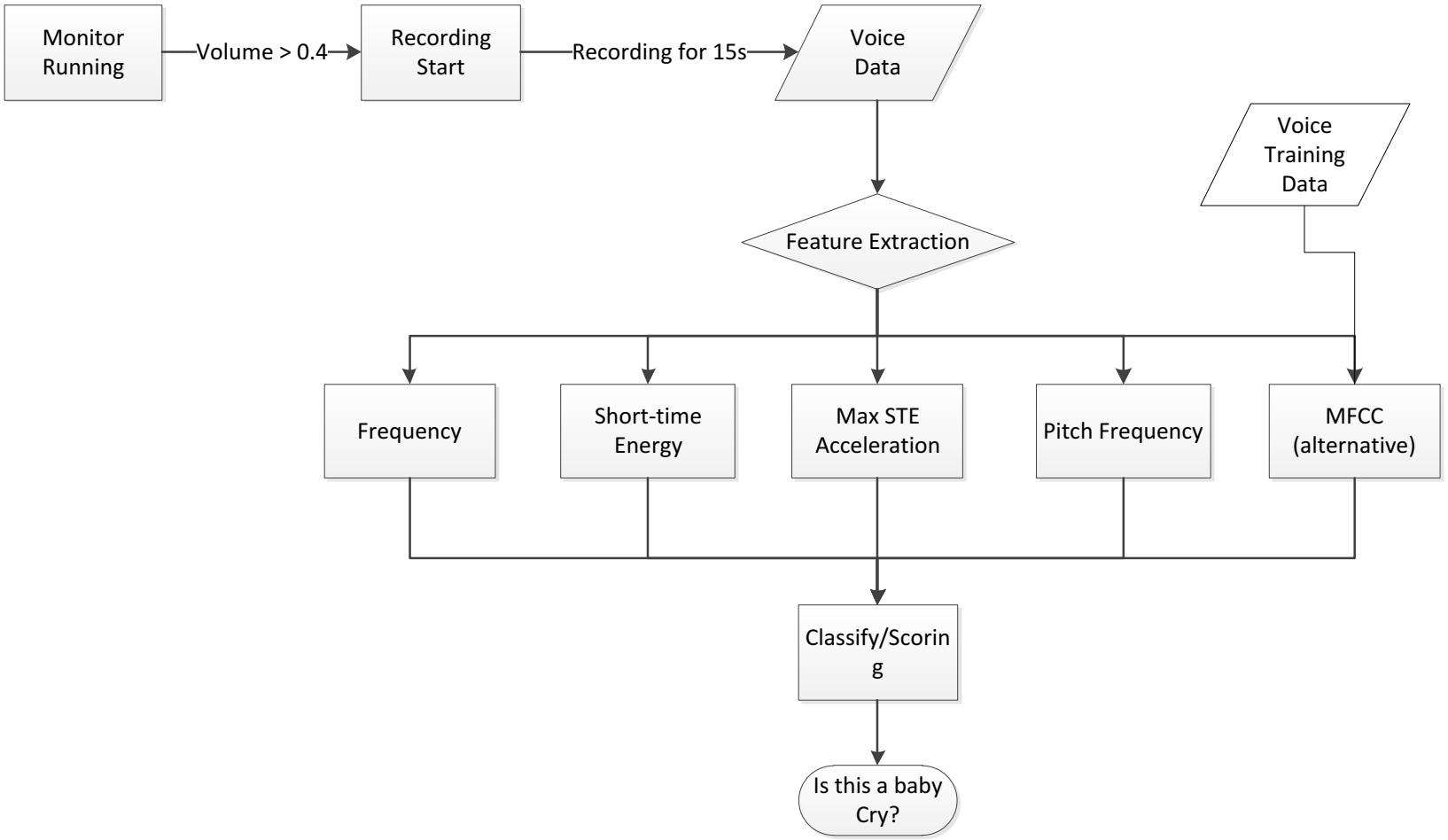
Baby Cry Detector Report

Zhijian Wang

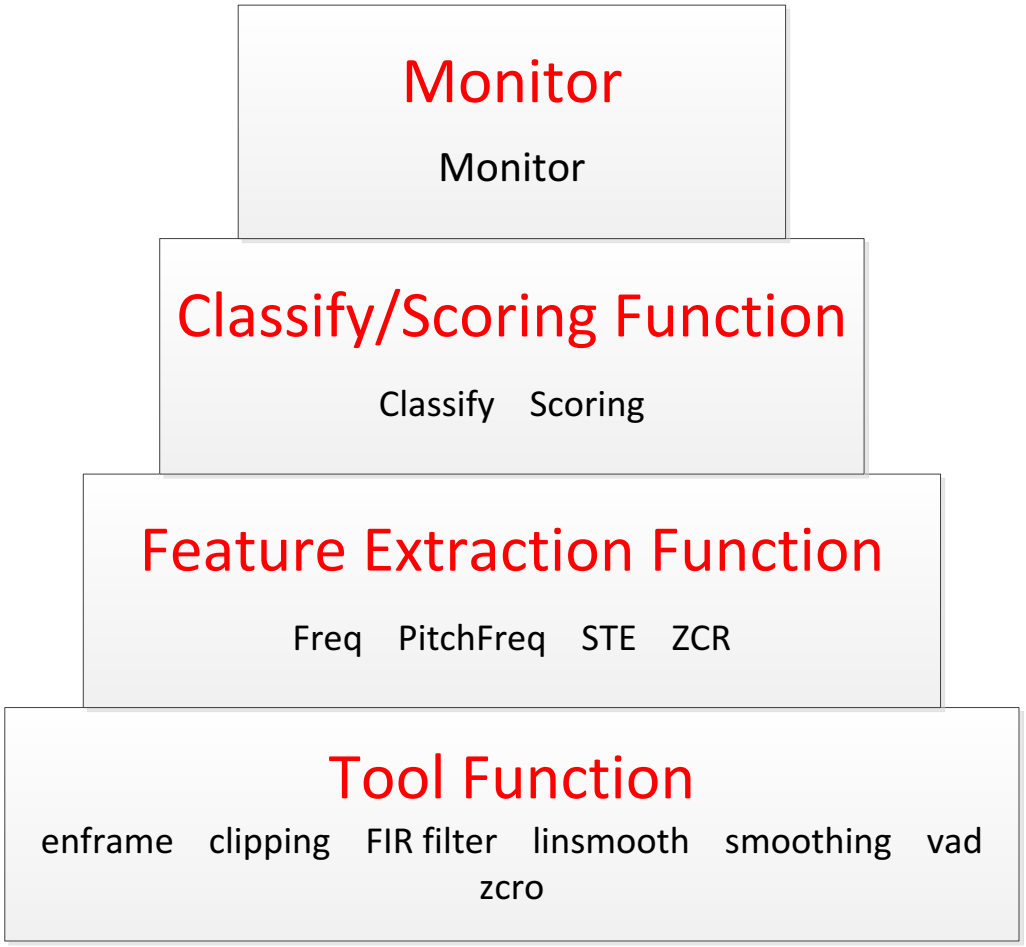
~~Zhijian.Wang@intel.com~~

wangzhijian22@gmail.com

Flowchart


Monitor Running
Volume > 0.4
Recording Start
Recording for 15s
Voice Data
Voice Training Data
Feature Extraction
Frequency
Short-time Energy
Max STE Acceleration
Pitch Frequency
MFCC (alternative)
Classify/Scoring
Is this a baby Cry?

System Structure

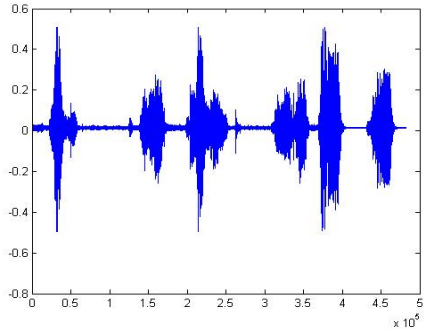


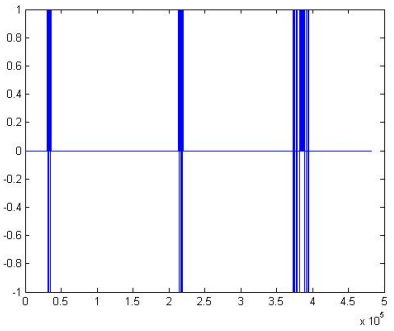
Tool-function#1 enframe

- Split the voice data into frames
- Input:
  - original voice data
  - frame length
  - frame shift/increment
- Output:
  - arrays of frames

Tool-function#2 clipping

- Pre-processing function for audio signals, there're two common clipping method: center clipping and 3-level clipping. Pitch frequency calculation can be accelerated by 10 times after clipping.
- 3-level clipping:
  - $f(x) = \begin{cases} 1, & x > x_L \\ 0, & -x_L \leq x \leq x_L \\ -1, & x < x_L \end{cases}$
  - x_L is normally 60%~70% of the maximum
- Input:
  - original voice data
  - clipping method (center or 3-level)
- Output:
  - clipped voice data





Tool-function#3 FIR

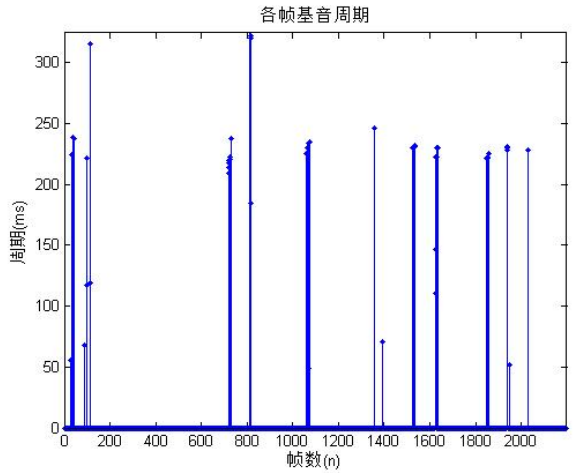
- FIR low pass filter.
- Since human voice (include baby cry) frequency is always below 3000Hz, we only care about those voice signals going through a low pass filter with a cutoff frequency of 3000Hz. This is why the sampling frequency of the phone is 8KHz.
- Input:
  - original voice data
  - cutoff frequency
- Output:
  - filtered voice data

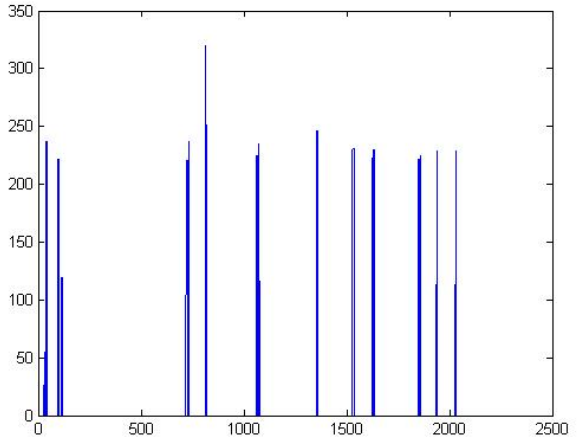
Tool-function#4 linsmooth

- A linear smoothing function to eliminate the ‘wild points’ of a dataset.
- In our system, we use another kind of smoothing method called median smoothing due to their efficiency. They can also be combined to form other smoothing methods. I didn’t do a result-accuracy experiment on this smoothing thing, so this linear smoothing function is still provided here for future work, research or optimization.
- Input:
  - original dataset
  - smoothing window length
  - Window type (hanmming as default)
- Output:
  - smoothed dataset

Tool-function#5 smoothing

- Post-processing function for pitch frequency feature extraction.
- Use this smoothing function to eliminate the wild points.
- In detail, I use the composition of 3-point median smoothing method and 5-point median smoothing method.
- Input:
  - original pitch frequency dataset
- Output:
  - smoothed pitch frequency dataset





Tool-function#6 vad

- Voice Activity Detection (Endpoint detection) function
- This function is used to find the start point and the end point of the voice signal.
- I don’t use this in this system, but still provided here in case of future work ,research or optimization.
- Input:
  - original voice data
  - frame length
  - frame shift/increment
- Output:
  - voice activity start point
  - voice activity end point

Tool-function#7 zcro

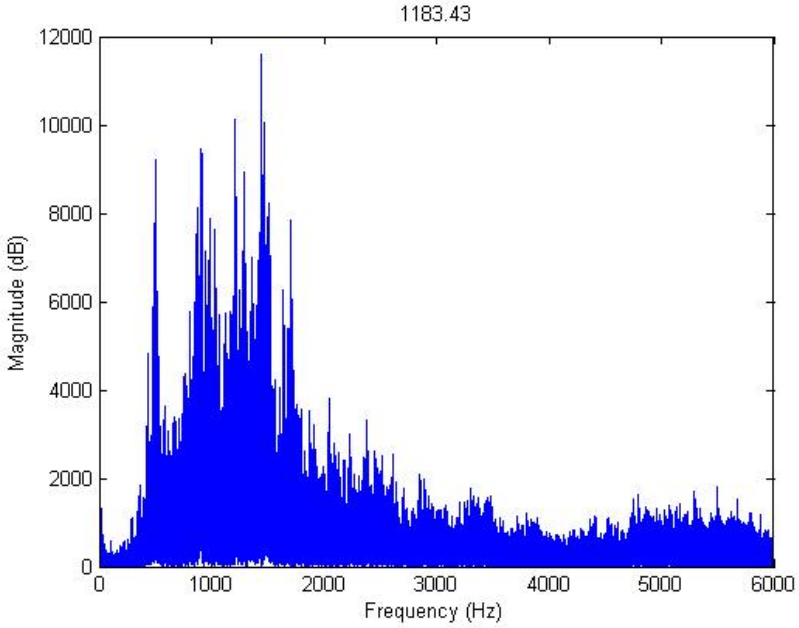
- Function for zero-crossing rate calculation.
- This function is used to calculate the zero-crossing rate of each signal frame.
- Input:
  - voice frames
- Output:
  - zero-crossing ratio of each frame

Feature Extraction functions

- Names of the feature extraction functions in this system all start with a capital letter. They have the same input, and one feature-value as output (the STE has two feature-values).
- They are:
  - Freq : For frequency-related feature extraction
  - PitchFreq: For pitch-frequency-related feature extraction
  - STE: For short-time-energy-related feature extraction
  - ZCR: For zero-crossing-ratio-related feature extraction
- Input:
  - original voice data
  - sampling frequency of the voice signal (In the monitor module, we record with a frequency of 8KHz)
  - sampling bits of the voice signal
- Output:
  - one or two value/values for the corresponding feature

Feature#1 Frequency

- Human voice is mainly made up of the frequency range from 85Hz ~ 255Hz. Sometimes the soprano can arrive 700-1000Hz. For baby cry, the most of the voice stays over 1000Hz.
- Use frequency spectrum to see the distribution of the voice frequency.



Feature#1 Frequency

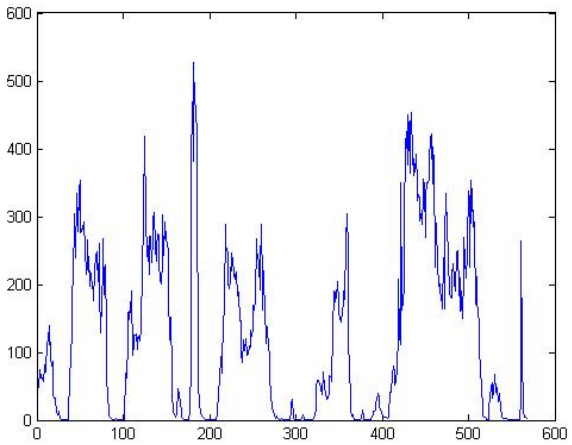
- We want a value that can illustrate in which frequency band the voice data mostly stay.
- For all frequency whose magnitude > 0.25*magnitude , we get a value:

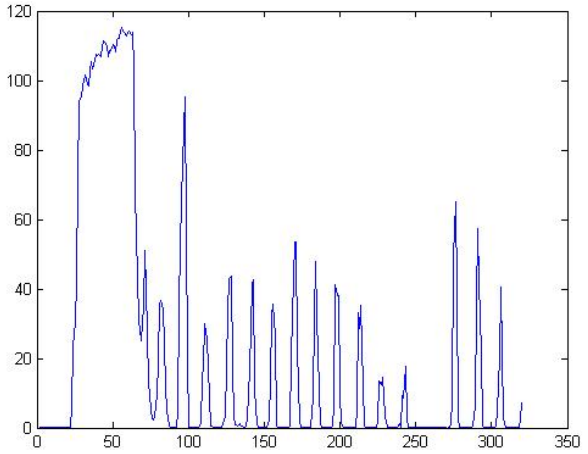
$$feature = \frac{\sum_1^n (freq_n * mag_n)}{\sum_1^n mag_n}$$

- As the title of the last picture says, the feature of frequency of this baby cry is 1183.43Hz.

Feature#2 Short-Time Energy

- Energy of 浊音 signals is much bigger than that of 清音 signals. A Baby's cry is usually the former one.
- This feature is not so important, because adults can produce 浊音 with very high energy. And baby cry's energy is not so high sometimes.





Feature#3 Max STE Acceleration

- Due to the burst of the baby cry, the STE of the cry signal can arrive a very high value in just one or two frames.
- Here are the max-accs of 20 baby cry samples. Some baby cry samples have a low value of max-acc because the cry is very smooth.

3.867008	61.02453	34.52153	327.1174	349.7301	493.1078	348.7071	426.5815	
626.6556	292.4846	213.7207	31.01905	12.46664	15.4839	96.94944	47.65147	
16.93027	81.86182	17.52431	57.89369					

- Here are the max-accs of 5 adult voice samples. Some adult voice samples have a high value of max-acc because of the instantaneous bursts.

7.852295	333.1889	4.116091	12.79636	17.88202	49.82367	41.89862	

Feature#4 Pitch Frequency

- Pitch Frequency is one of the most important feature of human voice. For ref, http://en.wikipedia.org/wiki/Pitch_detection_algorithm
- The fundamental frequency of speech can vary from 40 Hz for low-pitched male voices to 600 Hz for children or high-pitched female voices.
- As I mentioned in the previous page, we use 3-level clipping to reduce the calculation of pitch frequency.
- After clipping, we use short-time-autocorrelation methods to calculate the pitch frequency of each frame.

Feature#4 Pitch Frequency

- For discrete signals $x(n)$, its autocorrelation function is :

$$R(k) = \sum_{-\infty}^{\infty} x(m)x(m+k)$$

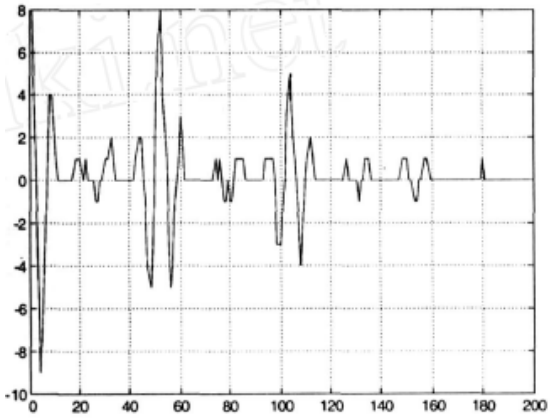
- For voice signals, its short-time-autocorrelation function is:

$$R_n(k) = R_n(-k) = \sum_{m=-\infty}^{\infty} [x(m)x(m-k)][w(n-m)w(n-m+k)]$$

- Autocorrelation methods need at least two pitch periods to detect pitch. This means that in order to detect a fundamental frequency of 40 Hz, at least 50 milliseconds (ms) of the speech signal must be analyzed. In this system, we use 30ms as a frame for analyze.

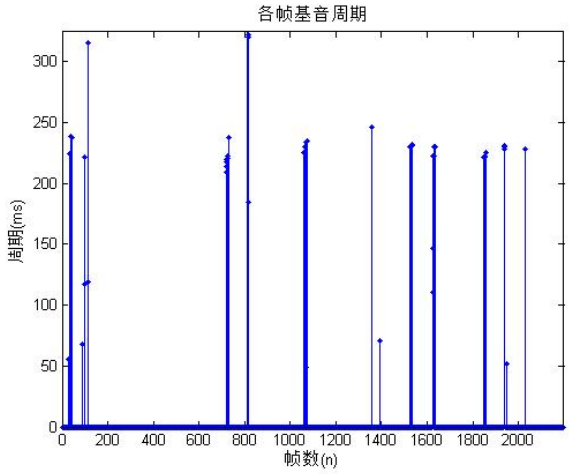
Feature#4 Pitch Frequency

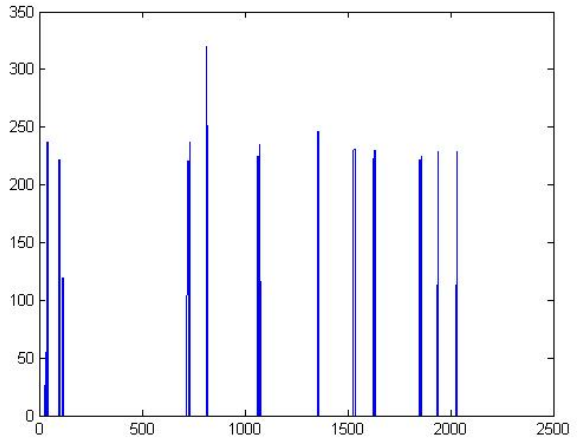
- Pitch frequency estimation. As the pic below shows, the first pitch of this frame arrives at about 50, so the pitch frequency of this frame is (8000/50=160Hz), 8000 is the sampling frequency of the voice signal.
- For every frame, we must discard the first several samples due to signal disturbance. In this system, I discard the samples of the first 10% of each frame, that is 3ms*8KHz=24 samples.
- After getting the pitch of the frame, we need another test. If the pitch value is smaller than 0.25*STE-of-this-frame, then we consider this frame as 清音, and set its pitch frequency to 0.



Feature#4 Pitch Frequency

- Post-processing. As I mentioned in the previous page, after getting the pitch frequency of each frame, we'd better do a smoothing work to eliminate the wild points.





Feature#4 Pitch Frequency

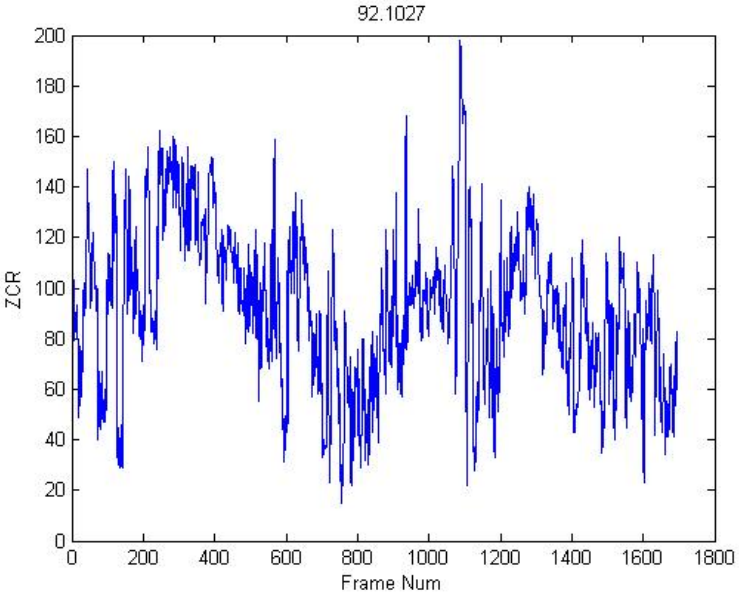
- For this feature, what we care about is, how many frames of the voice signal arrive a pitch frequency over 200.
- In this system, if more than 5 frames of the voice signal arrive a frequency over 200, then this voice has a higher possibility to be a baby cry.

Feature#5 Zero-crossing rate

- The zero-crossing rate is the rate of sign-changes along a signal, i.e., the rate at which the signal changes from positive to negative or back.
- ZCR is defined as below:

$$Z_n = \sum_{m=-\infty}^{\infty} |sgn[x(m)] - sgn[x(m-1)]|w(n-m) = |san[x(n)] - sgn[x(n-1)]| * w(n)$$

$$sgn[x(n)] = \begin{cases} 1, & x(n) \geq 0 \\ -1, & x(n) < 0 \end{cases}$$



Feature#5 Zero-crossing rate

- Here are zero-crossing rates of 21 baby cry samples.

33.74628	41.63682	59.18582	61.77363	92.10266	65.18292	58.59253	58.82546
55.54066	87.08642	98.35904	81.80868	80.21596	105.6729	189.2248	62.76613
38.45701	81.29979	50.24396	100.6475	37.61502			

- Here are zero-crossing rates of 8 adult voice samples.

31.76601	17.33648	23.15742	21.41624	19.89316	15.8641	22.86413	33.85861

- We can see that, for human voices, those who have higher values of zero-crossing rate tend to be baby cry voices.
- In this system, we set the boundary to be 50. Of course, this can be accurately determined by more samples and experiments.

Feature#6,7,8 Others

- For future work, research or optimization, the follow feature can be added to the system.
  - Harmonicity Factor
  - Harmonic-to-Average Power Ratio (HAPR)
  - Burst Frequency

Classify/Scoring

- I intended to design a classifier (decision tree, KNN or something like that), but there's not so many instances. In the future work, if many more baby cry wav files or samples are given, we can build a much more accurate classifier for this system instead of just several 'if' clauses. ☺

Monitor

- The monitor monitors the voice activities in the environment. When a voice with a volume over 0.4, the system will start to record for 15ms(0.4 and 15ms can be set manually). Then the voice recorded will be sent to the above-mentioned feature extraction functions to be classified as a baby-cry or no-baby-cry.
- For matlab code, the monitor uses Analog Input. For ref, http://www.mathworks.cn/cn/help/daq/examples/continuous-acquisition-using-analog-input.html
- For matlab code, the real-time voice waveform is shown in a figure.

What can be improved

- Clipping method: Some other clipping methods can be tested to see which is the best, in both efficiency and accuracy aspects.
- Smoothing method: Some other smoothing methods can be tested to see which is the best, in both efficiency and accuracy aspects.
- Feature selection: More voice signal features can be added to the system for classification. As mentioned in page 23.
- Parameters: Some parameters can be set more accurately with more experiments, like threshold for recording, frame length for every feature extraction, pitch frequency cutoff boundary……
- Classifier: Since the baby cry instances is really limited, we've no need to design a classifier like decision tree, KNN or svm. For future work, we can build an accurate classifier for classification (baby cry or not) using those extracted features.
- There are really lots of choices for every step, I've done much to optimize the efficiency and the accuracy, but I'm not so sure if there is some other methods which take both efficiency and accuracy into consideration and can do a better job. So, I think there's really much more we can do in the future.
- So, more……..

Problems I met with

- Too many, if you meet some problems, welcome to communicate with me. Thank you.