International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-4 Issue-6, August 2015

Analyze Features Extraction for Audio Signal with Six Emotions Expressions

Salwa A. Al-agha, Hilal H. Saleh, Rana F. Ghani

***Abstract*—Audio feature extraction plays an important role in analyzing and characterizing audio content. Auditory scene analysis, content-based retrieval, indexing, and fingerprinting of audio are few of the applications that require efficient feature extraction. The key to extract strong features that characterize the complex nature of audio signals is to identify their discriminatory subspaces. The audio information analysis for emotion recognition generally comprises linguistic and paralinguistic measurements. The linguistic measurement conforms to the rules of the language whereas paralinguistic measurement is the meta-data; i.e. related to how the words are spoken based on variations of pitch, intensity and spectral properties of the audio signal. This paper presents a technique for analyzing the features which extracted from recording audio signals in time domain and frequency domain by using statistical methods.**

***Index Terms*— Audio Signals, Audio Feature Analysis, Feature Extraction, Emotion Expression, MFCC, Pitch Extraction**

I. INTRODUCTION

Emotion recognition is an important research field of pattern recognition. Emotion takes a significant role in human communications, and has an effect on perception and decision making [1].Communication is an important capability, not only based on the linguistic part but also based on the emotional part. In the field of HCI, emotion recognition from computer is still a challenging issue, especially when recognition is based solely on voice, which is the basic mean of human communication. It is an important preparation for automatic classification and recognition of emotions to select a proper feature set as a description to the emotional speech. The efficiency of Speech emotion recognition (SER) system is highly dependent upon naturalness of database used in the system. SER is not an easy task as it requires a set of successive operation such as voice activity detection, feature extraction, training & classification. In scientific world, everything is going digital, and emotion detection in speech processing is one of the burning arenas in this filed. Many different researchers have tried their approach in this filed but accuracy is the major factor of the processing [2]. Starting in the 1930s, quantitative studies of vocal emotions have had a longer history than quantitative studies of facial expressions.

Manuscript published on 30 August 2015.
***** Correspondence Author (s)
Kavin. R*, Department of EEE, Sri Krishna College of Engineering and Technology, Coimbatore, India.
Elamcheren. S, Department of EEE, Sri Krishna College of Engineering and Technology, Coimbatore, India.
Dr. S. Sheebarani Gnanamalar, Department of EEE, Sri Krishna College of Engineering and Technology, Coimbatore, India.

Traditional as well as most recent studies in emotional contents in speech have used prosodic information which includes the pitch, duration and intensity of the utterance [3]. Many researchers have used speech signals to recognize emotions of people. Zhao L put forward an emotion recognition method based on speech by extracted energy, fundamental frequency and formant frequencies from each speech frame, and calculated their statistics, such as mean and variance [1]. This paper, present the analysis of features which extracted from recording audio signals (paralinguistic signals) by using different statistical functions in time domain and frequency domain with four basic emotions \ voice signals (anger, happy, sad, surprise).

II. AUDIO FEATURES

Sound features can be defined as mathematical algorithms which can be implemented, either by software or hardware, to extract useful information from the signal that is not obvious from the raw data. Features are extracted either from the time (temporal) domain or the frequency (spectral) domain. In time domain the signal is analyzed with respect to time, while in the frequency domain the signal is analyzed with respect to frequency. Spectrograms are sometimes also used to extract features that carry spectral as well as temporal information.

The various audio features can be categorized into two broad domains as illustrated in Fig. 1[4].

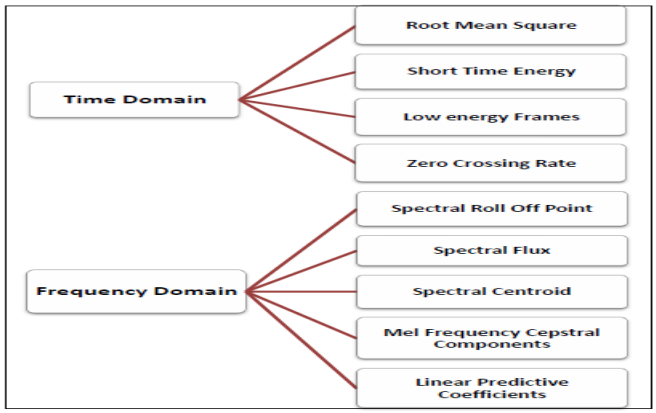


Fig.1 Audio features with examples from each category.

A. Time Domain Features

This section describes some of the temporal (time domain) features and define the structure of the audio signals:
Audio features are extracted at two levels:

1. Short-term (frame) level which last about 10 to 30 milliseconds within which theaudio signal is more stationary.

2. Long-term (track) level which go from 1 second up to tens of seconds. Such longintervals are also referred to in the literature as a "clip" or "window".

Both levels are illustrated in Fig. 2 a long track that has a stream of N samples is splitinto a number of frames M. Each frame would then contain R samples. While someapplications consider a certain amount of overlap between consequent frames, other applications do not.

For example, when sampling at fs = 44.1 kHz, a track of 5 seconds would contain N = 5 × 44100 = 220500 samples. If each track is split into 20 millisecond frames, then each frame would contain R = 20 × 10^{-3}× 44100 = 882 samples and there will be M = 250 frames in each 5 seconds track [5].

B. Root Mean Square (RMS)

The root mean square (RMS) of a signal is an indication of the power content in the signal. The RMS power of a particular audio frame can be found using equation (1):

$$RMS = \sqrt{\frac{\sum_{n=1}^{N} X^2(n)}{N}} \quad (1)$$

Where N is the number of samples in each frame and X(n) is the audio signal at sample n.The RMS is a feature that is used at the frame level instead of the track level. The RMS power is sometimes loosely called by different names; e.g. volume, loudness, and energy[6].

A. Short-time energy (STE)

The energy of any signal X(n) is a very fundamental characteristic, it can be computed using equation (2),[7]:

$$STE = \frac{1}{N}\sum_{n=1}^{N} X(n)^2 \quad (2)$$

Where N is the number of samples in each frame and X(n) is the audio signal at sample n.

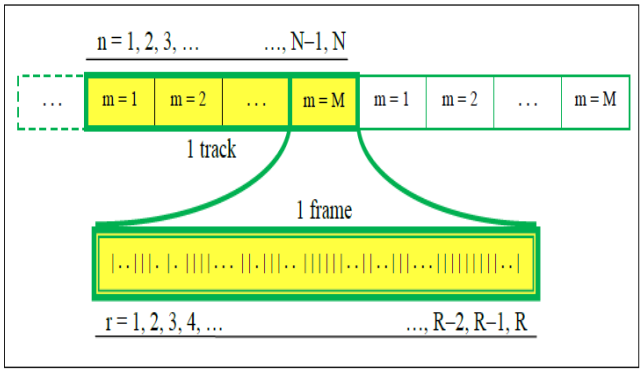


Fig. 2 Stream of tracks that is split into M frames; each frame has R samples.

C. Percentage of Low Energy Frames (LEF)

This is a feature that can be extracted in the time domain. Splitting a sound track (e.g. 1 second long) into small frames (e.g. 20 ms) would produce a number of frames (e.g. 50 frames). The RMS power of each frame is then computed using equation (1). The mean of RMS is then computed. Low energy frames are then labeled. A low energy frame is a frame whose RMS is less than 50% of the mean RMS; i.e. 0.5× RMS_{ave}, can be found in equation (3) and (4)[4].

$$RMS_{thresh} = 0.5 \times RMS_{mean} \quad (3)$$

$$LEF = (RMS < RMS_{thresh}) \quad (4)$$

Where $RMS_{mean}$is average of RMS.

D. Modified Low Energy Ratio (MLER)

The modified feature was named "Modified Low Energy Ratio" (MLER); and was computed using equation (5):

$$MLER = \frac{1}{2M}\sum_{m=1}^{M}[Sgn(lowthres - E(m)) + 1] \quad (5)$$

$$lowthres = \delta\frac{1}{M}\sum_{m=1}^{M} E(m) \quad (6)$$

Where M is the total number of frames in a track, E(m) is the short time energy of the m_{th} frame, and δ is a control coefficient which decides how low E(m) needs to be so that the frame is considered as "low energy".

Sgn(x) is by convention defined in equation (7):

$$Sgn(x) = \begin{cases} +1 & ,x > 0 \\ 0 & ,x = 0 \\ -1 & ,x < 0 \end{cases} \quad (7)$$

While δ was set to 0.5 in the LEF feature, it is varied in the MLER and a range of [0.05 to 0.12] was recommended [8].

E. Zero Crossing Rate (ZCR)

This feature was first introduced by Kedem[9], who related it to the dominant frequency of a signal. It is a count of the number of times an audio signal crosses the level of zero-amplitude in the time domain. Prior to counting, the mean of the signal is deducted in order to remove any DC component from the signal.

Assuming a discrete signal with a series of samples [S1, …, SN] with zero mean, the following set of equations were used by Kedem to count the number of times the signal crosses the level of zero amplitude:

Putting the mean (zero) as the threshold that the signal crosses,

$$X_t = \begin{cases} 1, & S_t \geq 0 \\ 0, & S_t < 0 \end{cases} \quad ,t = 1, 2, \dots, N. \quad (8)$$

The function:

$$d_t = (X_t - X_{t-1})^2 \quad (9)$$

Is an indicator of whether a crossing occurred (dt = 1) or not (dt = 0). Thus, the ZCR is simply as equation (10):

$$D = \sum_{k=2}^{N} d_k \quad (10)$$

Where N is the number of samples in each frame and St is the audio signal at each frame, dk is the summation of crossing in each frame.

Or as equation (11):

$$ZCR = \frac{1}{N}\sum_{n=2}^{N}\left|sign(x(n)) - sign(x(n-1))\right| \quad (11)$$

Where N is the number of samples in each frame and x(n) is the audio signals at each frame.

The ZCR is not useful when used alone. However, it could be exploited in association with other features. For example, the correlation between the ZCR and the RMS of audio frames. While the ZCR and RMS are independent for music, they are somehow correlated for speech.

➢ Characteristics :

- Noise and unvoiced sound have high ZCR.
- ZCR is commonly used in endpoint detection, especially in detection the start and end of unvoiced sound.
- To distinguish noise/silence from unvoiced sound, usually we add a shift before computing ZCR[6].

F. Energy Entropy (EE)

The energy entropy expresses abrupt changes in the energy level of the audio signal.

In order to calculate this feature, the frames are further divided into K sub-windows of fixed duration. For each sub-window i, the normalized energy σi² is calculated, i.e., the sub-window's energy divided by the whole window's energy. Then, the energy entropy is computed for frame j using equation (12):

$I_j = -\sum_{i=1..k} \sigma_i^2 \, log_2 \, \sigma_i^2$ (12)

Where j is the frame of audio signal, and k is the sub-windows of fixed duration in each frame. The value of the energy entropy is small for frames with large changes in energy level. Therefore, we can detect many violent actions like shots, which are characterized by sudden energy transitions in a short time period[10].

G. Pitch

The fundamental frequency (F0) for any audio signal is called pitch, (no. of fundamental period within a second, the unit used here is Hertz (Hz)). This feature cannot be accurately measured, but can be estimated[11]. Different methods exist that can be used to estimate the pitch of an audio signal:

1. Average Magnitude Difference Function (AMDF)

The original AMDF was proposed in equation (13):

$D(\tau) = \frac{1}{N-\tau-1}\sum_{n=0}^{N-\tau-1}|x(n) - x(n+\tau)|$ (13)

Where x(n) is the speech sample sequence multiplied by a rectangular window of length N, and τ is the lag number. The range of τ is between 0 and N-1, and the constant term outside summation is for normalization. For a periodic or quasi periodic signal with a period of T_p, equation should exhibit minimum at lag $T_p$and minimum peaks with lower degree at its multiple. In general, a rough estimation of pitch is derived by equation (14):

$T_p = \arg_\tau MIN_{\tau=\tau_{min}}^{\tau_{max}} (D(\tau))$(14)

Where τ_{min} and τ_{max} correspond to possible minimum and maximum pitch periods in samples [12].

2. Autocorrelation Function

Related to the time domain feature detector is the autocorrelation method. The autocorrelation of the signal was proposed in equation (15):

$x(n) \otimes x(\tau) = \sum_{i=q}^{q+N-1} x(n) \times x(n+\tau)$ (15)

Where x(n) is the speech sample sequence multiplied by a rectangular window of length N, and τ is the lag number. The range of τ is between 0 and N-1.

The main peak in the autocorrelation function is at the zero lag location (τ = 0). The location of the next peak gives an estimate of the period, and the height gives an indication of the periodicity of the signal. For analog signals this estimate is given by equation (16),[13].

$r(\tau_{max}) = max_\tau \, r(\tau)$ (16)

3. Cepstrum

"Cepstrum" is a play on the word spectrum as one might suspect and is simply a spectrum of a spectrum. The original time signal is transformed using a Fast Fourier Transform (FFT) algorithm and the resulting spectrum is converted to a logarithmic scale. This log scale spectrum is then transformed using the same FFT algorithm to obtain the power cepstrum. The power cepstrum reverts to the time domain and exhibits peaks corresponding to the period of the frequency spacing common in the spectrum, as shown in equations (17), (18):

$C(\tau) = \left|F(log\left|F(x(\tau))\right|^2)\right|^2$ 17)

$C(\tau) = F^{-1}\left(log\left|F(x(\tau))\right|^2\right)$ (18)

Fundamental frequency is estimated in the same way as in the autocorrelation method, as in equation (19), [14]:

$C(\tau_{max}) = max_\tau \, C(\tau)$ (19)

H. Frequency Domain Features

Some researchers have chosen to exploit the spectral features of audio signals. Show the relative level of computational load in comparison to the time domain features. Prior to computing any of the frequency domain features, the signal can be transformed from the time domain to the frequency domain using the Discrete Fourier Transform (DFT). The Inverse Discrete Fourier Transform (IDFT) can be used to transform the signal back to the time domain, but is normally not necessary in the context of feature extraction. Both the DFT and IDFT can be obtained as in equations (20),(21):

DFT:

$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N} \quad k = 0, 1, 2, \ldots\ldots, N-1$ (20)

IDFT:

$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(K)e^{\frac{j2\pi kn}{N}} \quad n = 0, 1, 2, \ldots\ldots, N-1$ (21)

The relative level of complexity is evident from this transform when compared to the time domain features which are directly taken from the time series of samples[7].

I. Spectral Flux (SF)

This is a frequency-domain measure of the local spectral change between successive frames, and it is defined as in equation (22):

$F_j = \sum_{k=0..s-1} \left(N_{j,k} - N_{j-1,k}\right)^2$ (22)

Where j is each frame, k is audio signals in each frame.

The summation of the differences between adjacent samples in a signal's spectrum for a single frame is known as the spectral flux, as shown in equation (23):

$SF = \sum_k ||X[K]| - |X[K-1]||$ (23)

The average of the spectral flux for all frames is then computed as the feature to be extracted [15].

J. Spectral Centroid (SC)

This feature (sometimes also called brightness) represents the central point in the signal's spectral power distribution in a frame of samples. On average, the SC for speech is low compared to that for music. Also, the SC for voiced speech is lower than for unvoiced speech signals. SC for a frame of an audio signal can be computed as equation (24):

$SC =$

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

$$\frac{\sum_k kX[k]}{\sum_k X[k]} \quad (24)$$

Or Spectral Centroid (correlate with brightness) as equation (25):

$$SC = (\sum_{k=1}^{K} K \times |X(K)|^2)/(\sum_{k=1}^{K}|X(K)|^2) \quad (25)$$

Where k is an index corresponding to a frequency (or a band of frequencies) whose power is X[k] (magnitude). Furthermore, better results were obtained by when using the second moment; i.e. replacing k by K^2as in equation (26),[15], [16]:

$$SC = \frac{\sum_k k^2 X[k]}{\sum_k X[k]} \quad (26)$$

K. Mel Frequency Cepstral Coefficients (MFCC)

Mel Frequency Cepstral Coefficients (MFCCs) are the best known and most popular features, which are based on the known variation of the human ear's critical bandwidths with frequency. This is presented in the Mel-frequency scale, which is a linear frequency space below 1000 Hz and a logarithmic space above 1000 Hz. A popular relation between f(Hz) and mel-frequency scale Fmel is as in equation (27):

$$F_{mel} = 2595 * Log_{10}\left(1 + \frac{f(Hz)}{700}\right) \quad (27)$$

The steps of MFCC feature extraction are:

1. Framing and Windowing

Speech signal is divided into frames after pre-processing operation, and apply a window to each frame, each frame is K samples long, and with adjacent frames being separated by P samples. A commonly used window is the Hamming window, it is calculated as in equation (28):

$$W(k) = 0.54 - 0.46 * \cos\left(\frac{2\pi k}{k-1}\right) \quad (28)$$

2. Fast Fourier Transform (FFT)

The Fast Fourier Transform is a fast implementation of the Discrete Fourier Transform (DFT) which converts N-samples of frames into the frequency spectrum. As in equation (20).

3. Period gram Estimation

The Periodogram-based power spectral estimate for the speech frame is given by equation (29), this calculate by take the absolute value of the complex fourier transform, and square the result.

$$P_i(k) = \frac{1}{N}|X_i(k)|^2 \quad (29)$$

Where N is number of samples in each frame, Xi (k) value of the complex fourier transform for sample Xi(n).

4. Mel Scaled Filter banks

The Mel-scale filter bank implementation includes 40 triangular filters non-uniformly spaced along the frequency axis.

$$MFCC = \sum_{k=1}^{40} PK \cos\left[n\left(K - \frac{1}{2}\right)\frac{\pi}{40}\right], \quad for\ n = 0, 1, 2, \dots, L. \quad (30)$$

Where L is the number of MFCC coefficients and Pk, k = 1, 2, …. , 40, represent the log energy output of the Kth filter.

5. Signal Energy

Furthermore, the signal energy is added to the set of parameters. It can simply be computed from the speech samples S(n) within the time window.

6. Discrete Cosine Transform (DCT)

The spectrum is defined as the inverse Fourier transform of the log magnitude of Fourier transform of the signal. Since the log Mel filter bank coefficients are real and symmetric, the inverse Fourier transform operation can be replaced by DCT to generate the cepstral coefficients. The cepstral coefficients are the DCT of the M filter outputs obtained from equation (31):

$$C(n) = \sum_{k=0}^{M-1} XK \cos\left[n\left(K - \frac{1}{2}\right)\frac{\pi}{M}\right], n =, 2, \dots., L \quad (31)$$

Where L is the number of MFCC coefficients and Xk , k = 0, 1, 2, …. , M-1. Represent the log energy output of the Kth filter.

7. Dynamic Parameters

The voice signal and the frames changes, such as the slope of a formant at its transitions. Therefore, there is a need to add features related to the change in cepstral features over time. 13 delta or velocity features (12 cepstral features plus energy), and 39 features a double delta or acceleration feature are added. Each of the 13 delta features represents the change between frames in the equation (32):

$$d(t) = \frac{c(t+1) - c(t-1)}{2} \quad (32)$$

corresponding cepstral or energy feature, while each of the 39 double delta features represents the change between frames in the corresponding delta features[17], [18].

III. THE PROPOSED TECHNIQUE

In the following, steps for extracting features from recording audio signals by using different statistical functions in time and frequency domain:

A. Framing and Windowing

Framing all of recording signals into 20-40ms frames (this means with 25 ms, the frame length for a 16KHz signal is 0.025 * 16000 = 400 samples (frame size)). Then windowing all of these framing signal (this by window = 0.625 * 25ms = 0.0156), and then extract the step size for each signal (step = floor(window * signal) = 250), this step allows some overlap to the frame. The first 400 sample frame start at sample 0, the next 400 sample frame start at sample 250 etc., until the end of signal file is reached.

B. Audio Feature Extraction

Features will beextractedfor each audio signal which recorded, and for each four basic emotion expression, by using the statistical functions, as following:

1. Calculate the RMS for signal as in equation (1), then compute the further statistical function for RMS (max, min, mean, and median).

2. Calculate the STE for signal as in equation (2), then compute the further statistical function for STE (max, min, mean, median, variance, and standard deviation).

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

3. Calculate the LEF for signal as in equation (3, 4).
4. Calculate the MLER for signal as in equation (5, 6).
5. Calculate the ZCR for signal as in equation (11), then compute the further statistical function for ZCR (max, min, mean, median, and max/mean).
6. Calculate the EE for signal as in equation (12), then compute the further statistical function for EE (max, min, mean, median, max/mean, and max/median).
7. Calculate the SF for signal as in equation (23), then compute the further statistical function for SF (max, min, mean, median, and max/mean).
8. Calculate the SC for signal as in equation (26), then compute the further statistical function for SC (max, min, mean, median, and max/mean).
9. Calculate the AMDF for signal as in equation (13, 14), then compute the further statistical function for AMDF (max, min, mean, standard deviation, and range).
10. Calculate the Autocorrelation function for signal as in equation (15, 16), then compute the further statistical function for Autocorrelation (max, min, mean, standard deviation, and range).
11. Calculate the Cepstrum function for signal as in equation (17, 19), then compute the further statistical function for Cepstrum (max, min, mean, standard deviation, and range).
12. Calculate the MFCC function for signal as in equation (30, 32).

IV. EXPERIMENTAL RESULTS AND ANALYSIS

The proposed technique is used for features extraction and analysis for audio signals,which is implemented on recorded a paralinguistic (Meta data: voice without speaking) signals for different adult male and female person with six basic emotion voice, as shown in Fig. 3.

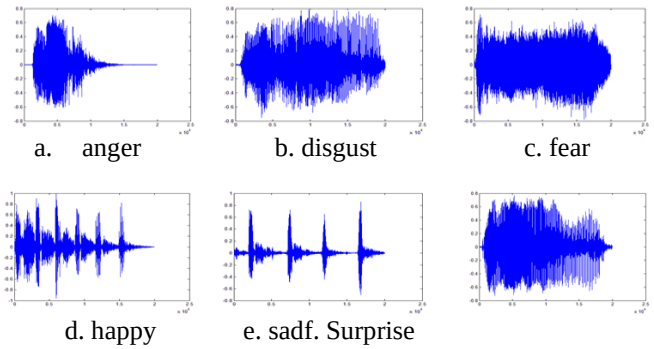


Fig. 3 Voice signals for six basic emotion.

1. Implementing the RMS function, as shown in Fig. (4– a. anger, b. disgust, c. fear) RMS – max, from these features can't recognize any one of expressions (a, b, c) because there is conflict between values. Fig. (5– a. anger, b. disgust, c. fear) RMS – min, from these features can recognize one expression (a.anger from b, c) because there is no conflict between values. Fig. (6– a. anger, b. disgust, c. fear) RMS – mean, canrecognize one expression (a.anger from b, c) because it have low energy than other. Fig. (7– a. anger, b. disgust, c. fear) RMS – median, can't recognize any one of expressions (a, b, c) because there is confusing between values; for 30 voice signals.
2. Implementing the STE function, as shown in Fig. (8– a. happy, b. sad, c. surprise) STE – max, can recognize one feature (c. surprise) have more energy than (a, b), Fig. (9– a. happy, b. sad, c. surprise) STE – min, can recognize one feature (b. sad) have less energy than (a, c). [Fig. (10– a. happy, b. sad, c. surprise) STE – mean, Fig. (11– a. happy, b. sad, c. surprise) STE – median, Fig. (12– a. happy, b. sad, c. surprise) STE – variance, Fig. (13– a. happy, b. sad, c. surprise) STE – Std], can't recognize any features because of the conflict between the values; for 30 voice signals

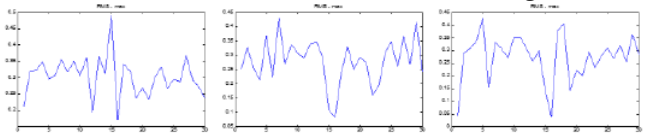


Fig. 4 RMS – max examples of 30 voice signals.

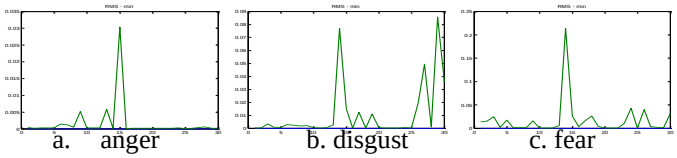


Fig. 5 RMS – min examples of 30 voice signals.

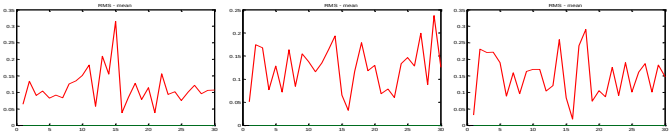


Fig. 6 RMS – mean examples of 30 voice signals.

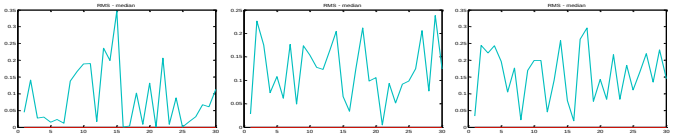


Fig. 7 RMS – median examples of 30 voice signals.

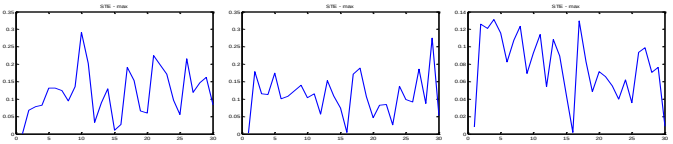


Fig. 8 STE – max examples of 30 voice signals.

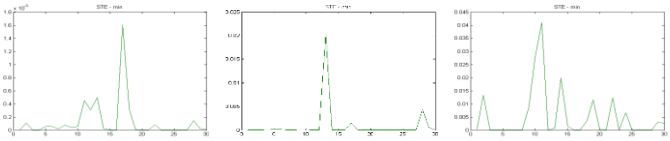


Fig. 9 STE – min examples of 30 voice signals.

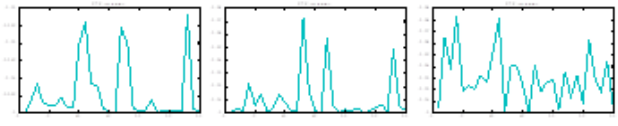
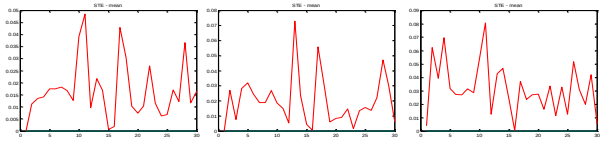


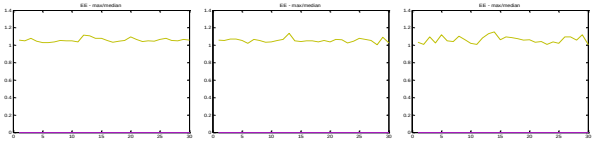
Fig. 10 STE – mean examples of 30 voice signals.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)



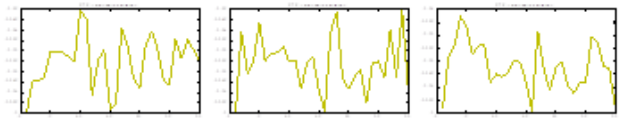
a. happy b. sad c. surprise

Fig. 11 STE – median examples of 30 voice signals.



a. happy b. sad c. surprise

Fig. 12 STE – variance examples of 30 voice signals.



a. happy b. sad c. surprise

Fig. 13 STE – Std examples of 30 voice signals.

3. Implementing the LEF function, as shown in Fig. 14, can recognize one feature (happy), which have a group of the successive Farms, which have low energy and consequential intermittently along the signal.

4. Implementing the MLER function, as shown in Fig. 15, can't recognize any features because of the conflicting between the values.

5. Implementing the ZCR function, as shown in Fig. (16– a. anger, b. disgust, c. fear) ZCR – max, can't recognize any features because of the conflict between the values. Fig. (17– a. anger, b. disgust, c. fear) ZCR – min, can recognize one feature (a. anger) have more than one peak. [Fig. (18– a. anger, b. disgust, c. fear) ZCR – mean, Fig. (19– a. anger, b. disgust, c. fear) ZCR – median, Fig. (20– a. anger, b. disgust, c. fear) ZCR – max/mean] can't recognize any features because of the conflict between the values]; for 30 voice signals.

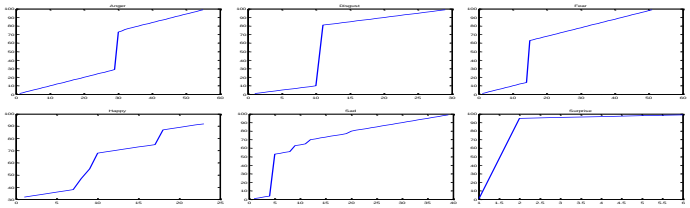


Fig. 14 LEF an example of 1 voice signals.

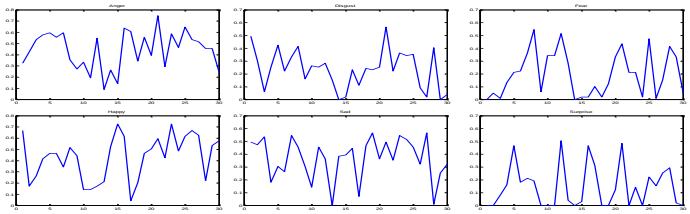
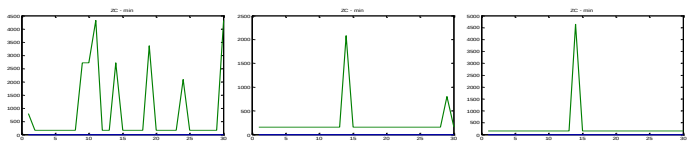
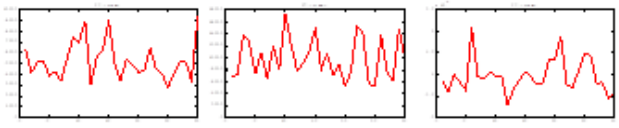


Fig. 15MLER of 30 voice signals.



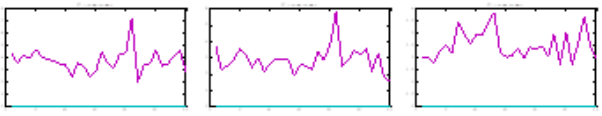
a. angerb. disgust c. fear

Fig. 16 ZCR – max examples of 30 voice signals.



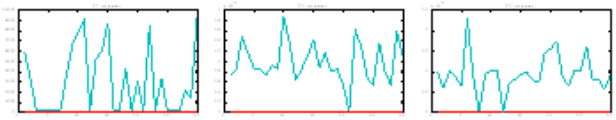
a. anger b. disgust c. fear

Fig. 17 ZCR – min examples of 30 voice signals.



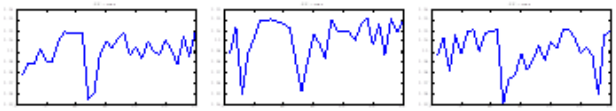
a. anger b. disgust c. fear

Fig. 18 ZCR – mean examples of 30 voice signals.



a. anger b. disgust c. fear

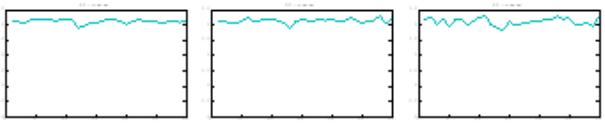
Fig. 19 ZCR – median examples of 30 voice signals.



a. anger b. disgust c. fear

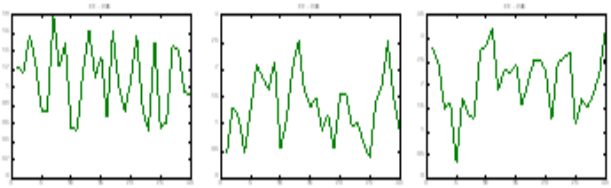
Fig. 20 ZCR – max/mean examples of 30 voice signals.

6. Implementing the EE function, as shown in [Fig. (21– a. happy, b. sad, c. surprise) EE – max, Fig. (22– a. happy, b. sad, c. surprise) EE – min, Fig. (23– a. happy, b. sad, c. surprise) EE – mean, Fig. (24– a. happy, b. sad, c. surprise) EE – median, Fig. (25– a. happy, b. sad, c. surprise) EE – max/mean, Fig. (26– a. happy, b. sad, c. surprise) EE – max/median], can't recognize any features because of the conflict between the values; for 30 voice signals.



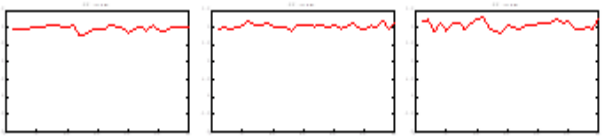
a. happy b. sad c. surprise

Fig. 21 EE – max examples of 30 voice signals.



a. happy b. sad c. surprise

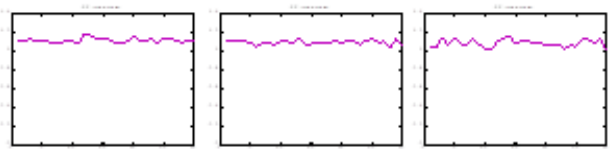
Fig. 22 EE – min examples of 30 voice signals.



a. happy b. sad c. surprise

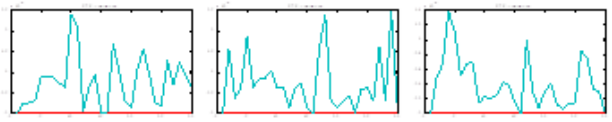
Fig. 23 EE – mean examples of 30 voice signals.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)



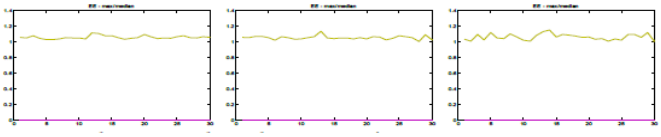
a. happy b. sad c. surprise

Fig. 24 EE – median examples of 30 voice signals.



a. happy b. sad c. surprise

Fig. 25 EE – max/mean examples of 30 voice signals.

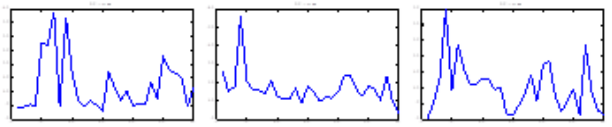


a. happy b. sad c. surprise

Fig. 26 EE – max/median examples of 30 voice signals.

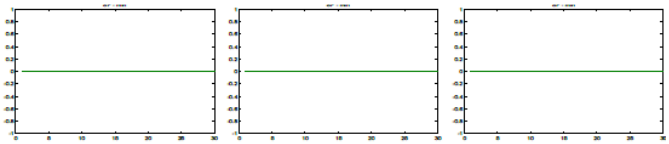
7. Implementing the SF function, as shown in [Fig. (27– a. anger, b. disgust, c. fear) SF – max, Fig. (28– a. anger, b. disgust, c. fear) SF – min, Fig. (29– a. anger, b. disgust, c. fear) SF – mean, Fig. (30– a. anger, b. disgust, c. fear) SF – median, Fig. (31– a. anger, b. disgust, c. fear) SF – max/mean] can't recognize any features because of the conflict between the values; for 30 voice signals.

8. Implementing the SC function, as shown in [Fig. (32– a. happy, b. sad, c. surprise) SC – max. Fig. (33– a. happy, b. sad, c. surprise) SC – min, Fig. (34– a. happy, b. sad, c. surprise) SC – mean, Fig. (35– a. happy, b. sad, c. surprise) SC – median, Fig. (36– a. happy, b. sad, c. surprise) SC – max/mean] can't recognize any features because of the conflict between the values; for 30 voice signals.



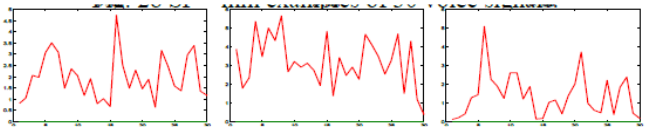
a. anger b. disgust c. fear

Fig. 27 SF – max examples of 30 voice signals.



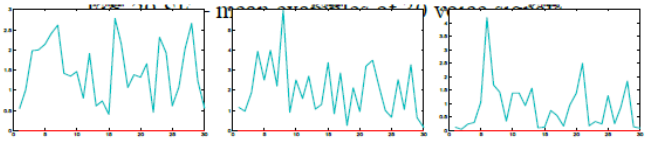
a. anger b. disgust c. fear

Fig. 28 SF – min examples of 30 voice signals.



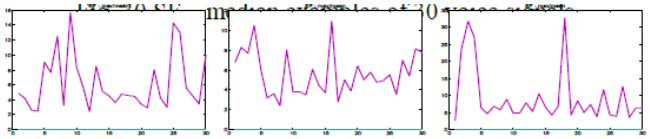
a. anger b. disgust c. fear

Fig. 29 SF – mean examples of 30 voice signals.



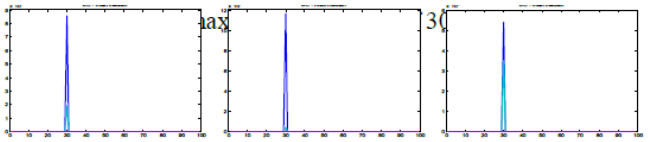
a. anger b. disgust c. fear

Fig. 30 SF – median examples of 30 voice signals.



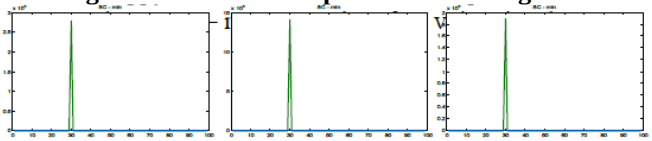
a. anger b. disgust c. fear

Fig. 31 SF – max/mean examples of 30 voice signals.



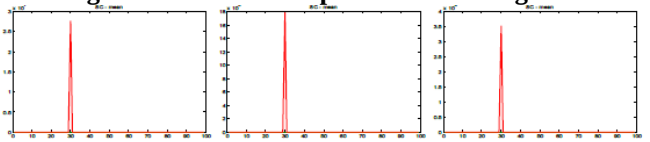
a. happy b. sad c. surprise

Fig. 32 SC – max examples of 30 voice signals.



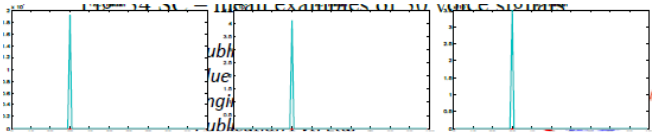
a. happy b. sad c. surprise

Fig. 33 SC – min examples of 30 voice signals.



a. happy b. sad c. surprise

Fig. 34 SC – mean examples of 30 voice signals.

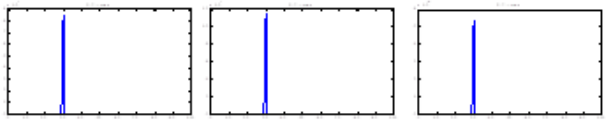


a. happy b. sad c. surprise

Fig. 35 SC – median examples of 30 voice signals.

9. Implementing the AMDF function, as shown in [Fig. (37– a. anger, b. disgust, c. fear) AMDF – max, Fig. (38– a. anger, b. disgust, c. fear) AMDF – min, Fig. (39– a. anger, b. disgust, c. fear) AMDF – mean, Fig. (40– a. anger, b. disgust, c. fear) AMDF – Std, Fig. (41– a. anger, b. disgust, c. fear) AMDF – range] can't recognize any features because of the conflict between the values; for 30 voice signals.

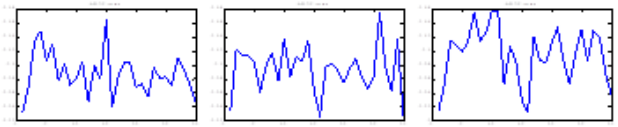
10. Implementing the Autocorrelation function, as shown in Fig. (42– a. happy, b. sad, c. surprise) Auto – max, can recognize one feature (c. surprise) which have maximum values than (a. happy, b. sad). [Fig. (43– a. happy, b. sad, c. surprise) Auto – min, Fig. (44– a. happy, b. sad, c. surprise) Auto – mean, Fig. (45– a. happy, b. sad, c. surprise) Auto – Std, Fig. (46– a. happy, b. sad, c. surprise) Auto – range] can't recognize any features because of the conflict between the values; for 30 voice signals.



a. happy b. sad c. surprise

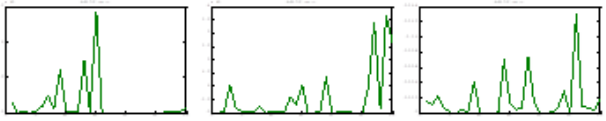
Fig. 36 SC – max/mean examples of 30 voice signals.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)



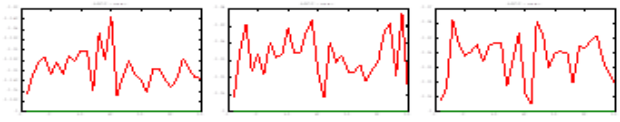
a. anger b. disgust c. fear

Fig. 37 AMDF – max examples of 30 voice signals.



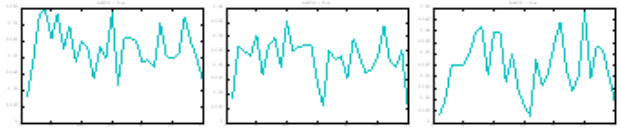
a. anger b. disgust c. fear

Fig. 38 AMDF – min examples of 30 voice signals.



a. anger b. disgust c. fear

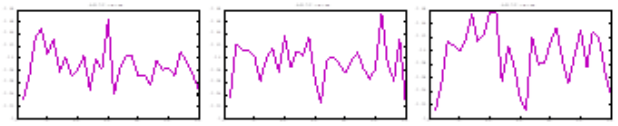
Fig. 39 AMDF – mean examples of 30 voice signals.



a. anger b. disgust c. fear

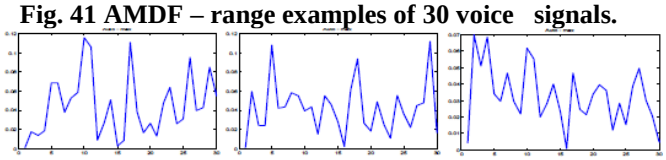
Fig. 40 AMDF – Std examples of 30 voice signals.

11. Implementing the Cepstrum function, as shown in Fig. (47– a. anger, b. disgust, c. fear) Cepstrum – max, can't recognize any features because of the conflict between the values. Fig. (48– a. anger, b. disgust, c. fear) Cepstrum – min, can recognize one feature (a. anger) which have maximum value than (b. disgust, c. fear). [Fig. (49– a. anger, b. disgust, c. fear) Cepstrum – mean, Fig. (50– a. anger, b. disgust, c. fear) Cepstrum – Std, Fig. (51– a. anger, b. disgust, c. fear) Cepstrum – range] can't recognize any features because of the conflict between the values; for 30 voice signals



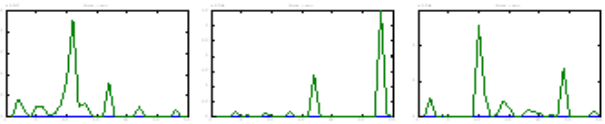
a. anger b. disgust c. fear

Fig. 41 AMDF – range examples of 30 voice signals.



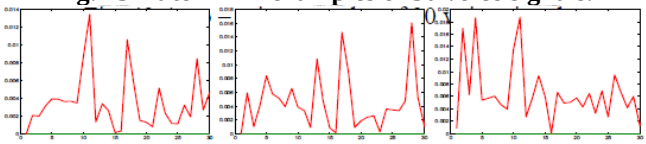
a. happy b. sad c. surprise

Fig. 42 Auto– max examples of 30 voice signals.



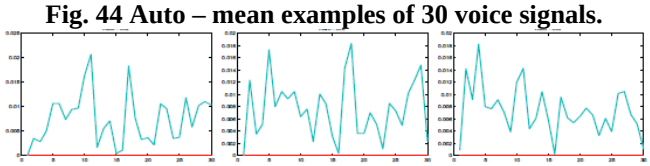
a. happy b. sad c. surprise

Fig. 43 Auto – min examples of 30 voice signals.



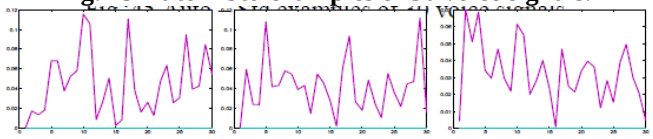
a. happy b. sad c. surprise

Fig. 44 Auto – mean examples of 30 voice signals.



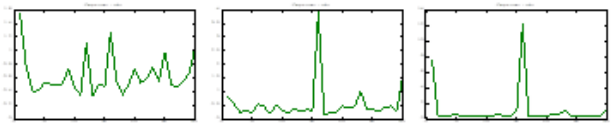
a. happy b. sad c. surprise

Fig. 45 Auto – Std examples of 30 voice signals.



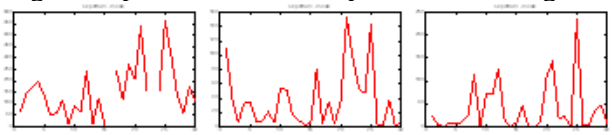
a. happy b. sad c. surprise

Fig. 46 Auto – range examples of 30 voice signals.



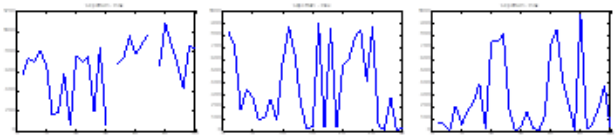
a. anger b. disgust c. fear

Fig. 47 Cepstrum – max examples of 30 voice signals.



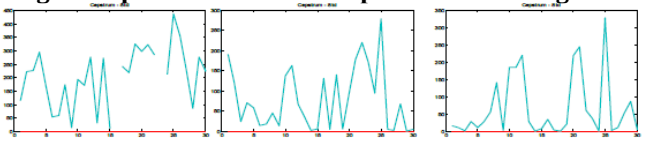
a. anger b. disgust c. fear

Fig. 48 Cepstrum – min examples of 30 voice signals.



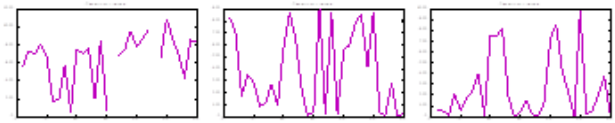
a. Anger b. disgust c. fear

Fig. 49Cepstrum – mean examples of 30 voice signals.



a. anger b. disgust c. fear

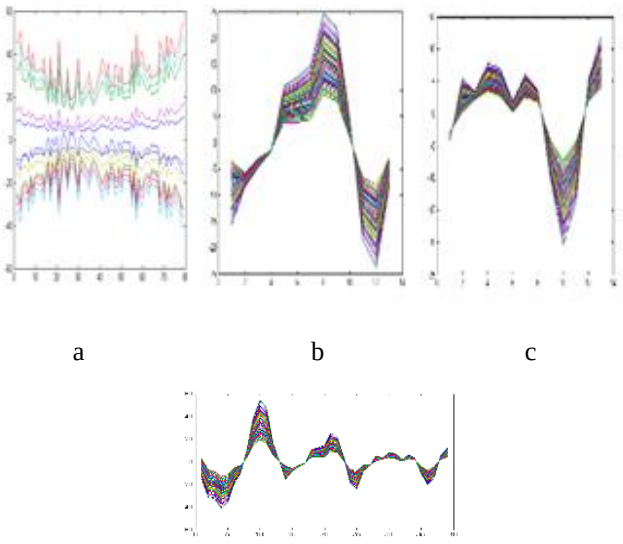
Fig. 50 Cepstrum – Std examples of 30 voice signals.



a. anger b. disgust c. fear

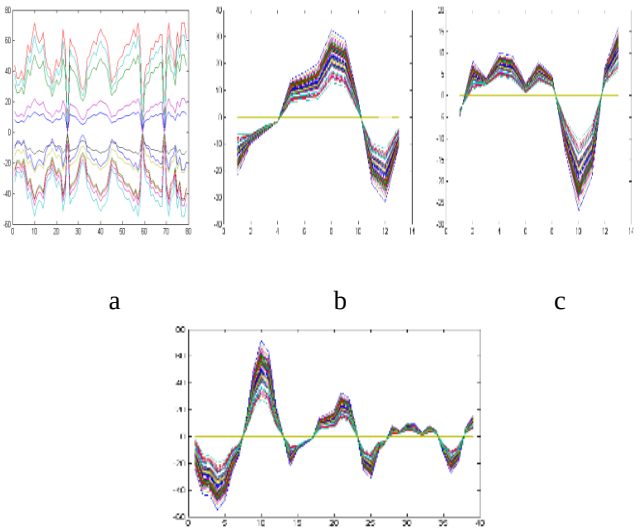
Fig. 51 Cepstrum – range examples of 30 voice signals.

8. Implementing the Mel or MFCC function, as shown in Fig. (52 – a. MFCC, b. first derivative, c. second derivative, d. 39 Mel features), for one person with anger emotion voice. Fig. (53 – a. MFCC, b. first derivative, c. second derivative, d. 39 Mel features), for one person with happy emotion voice as an example of this technique, can recognize between anger and happy from MFCC and Mel.



a b c d

Fig. 52 Mel features for one anger person



a b c d

Fig. 53 Mel features for one happy person

V. CONCLUSIONS

The proposed technique is presented for features extraction and analysis of audio signals, which recorded for 30 people for four basic emotion. In the stage of recording audio signal, it's so easy work by using any recording software in laptop or mobile, but it's so difficult to convince people to register Calculating the features of time domain (RMS, STE, LEF, MLER, ZCR, EE) and these further statistical functions like (max, min, mean, median, etc.) are not time consuming, except Pitch function (AMDF, Autocorrelation, Cepstrum) and these further statistical functions are time consuming. Calculating the features of frequency domain (SF, SC, MFCC) and these further statistical functions like (max, min, mean, median, etc.) are nottime consuming. (RMS – min, RMS – mean, ZCR – min, Cep – min, MFCC) features can be used to recognize anger voice. (LEF, MFCC) features can be used to recognize happy voice. (STE – max, Autocorrelation – max) features can be used to recognize surprise voice. STE – min feature can be used to recognize sad voice.

In future work, these audio features, will be selected and used in the emotion recognition system.

REFERENCES

1. X. Chao, D. Pufeng, F. Zhiyong, M. Zhaopeng, C. Tianyi, and D. Caichao, "Multi-Modal Emotion Recognition Fusing Video and Audio", Applied Mathematics & Information Sciences An International Journal, No. 2, p. 455-462, March 2013.
2. Joshi, "Speech Emotion Recognition Using Combined Features of HMM & SVM Algorithm", International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE), Vol 3, No. 8, pp. 387-393, ISSN: 2277 128X, August 2013.
3. S. Chen, "Joint Processing of Audio-Visual Information for the Recognition of Emotional Expressions in Human-Computer Interaction", Thesis for the degree of Doctor of Philosophy in Electrical Engineering in the Graduate College of the University of Illinois at Urbana- Champaign, 2000.
4. M.Alnadabi, "Speech/Music Discrimination: Novel Features in Time Domain", Thesis for the degree of Doctor of Philosophy in the University of Durham, April 2010.
5. Y. Wang, Z. Liu, and J.-C. Huang, "Multimedia Content Analysis: Using Both Audio and Visual Clues", in IEEE Signal Processing Magazine, p. 12-36, 2000.
6. Panagiotakis, and G. Tziritas, "A Speech-Music Discriminator Based On RMS and Zero-Crossings", IEEE Transactions on Multimedia 7(1), p. 155-166, 2005.
7. J.G. Proakis, and D.G. Manolakis, "Digital Signal Processing: Principles, Algorithms, and Applications", 3rd New Delhi: Prentice-Hall India, 2000.
8. W.Q. Wang, W. GaO, and D.W. Ying, "A Fast and Robust Speech/Music Discrimination Approach", in ICICS-PCM Singapore, 2003.
9. Kedem, "Spectral Analysis and Discrimination by Zero-Crossings", Proceedings ofthe IEEE, 74(11), p. 1477-1493, 1986.
10. Datta, M. Shah, and N. V. Lobo, "Person-on-Person Violence Detection in Video Data", IEEE International Conference on Pattern Recognition, Canada, 2002.
11. K. Ishizuka, and N. Miyazaki, "Speech Feature Extraction Method Using Subband -Based Periodicity and Non periodicity Decomposition", Journal of Acoustical Society of America 120(1), p. 443–452, 2006.
12. Ross, H. Shaffer, A. Cohen, R. Freudberg, H. Manley, "Average Magnitude Difference Function Pitch Extractor", Proceedings of IEEE Transactions Lon Speech and Audio, pp. 353-362, 1974.
13. J. Cioffi, "Limited-Precision Effects in Adaptive Filtering", IEEE Transactions on Circuits and Systems, Vol. 34, No. 7, pp. 821-833, 1987.
14. Sondhi, "New Methods of Pitch Extraction", IEEE Trans. on Audio and Electro Acoustics, Vol. 16, No. 2, pp. 262-266, 1968.
15. Saad, "A Multi-feature Speech/Music Discrimination System", in Nineteenth National Radio Science Conference, Alexandria: URSI, 2002.
16. Scheirer, and M. Slaney, "Construction and Evaluation of a Robust Multi-feature Speech/Music Discriminator", International Conference on Acoustics, Speech, and Signal Processing (ICASSP'97), 1997.
17. R. Hasan, M. Jamil, G. Rabbani, and S. Rahman, "Speaker Identification using Mel Frequency Cepstral Coefficients", 3edInternational Conference on Electrical & Computer Engineering ICECE, ISBN 984-32- 1804-4, p. 565-568, December 2004.
18. M. E. Safi, "Speech Recognition based Microcontroller for Wheelchair Movements", Thesis for the degree of Master of Science in Electronic Engineering in the University of Technology, August 2013.

Salwa A. Alaghais, currently a lecturer in University of Technology Department of Electrical Engineering and a postgraduate student for PhD degree in University of Technology Department of Computer Science. She received her Msc at University of Technology Department of Computer Science in 2003. Her research interest lies in Image and Video processing.

Hilal H. Saleh, is currently a lecturer in University of Technology Department of Computer Science. He received his PhD at Higher Institute Sofia - Bulgariain1981. His research interest lies in Information Systems and Data Security.

Rana F. Ghani is currently a director of the Computer Center of University of Technology and a lecturer in University of Technology Department of Computer Science. She received her PhD at University of Technology Department of Computer Science in 2006. Her research interest lie in Network Security

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)