CHARACTERIZATION OF INFRASONIC ACOUSTIC SIGNALS

Sumeet Kumar

Visvesvaraya National Institute of Technology, Nagpur, Maharashtra, 440010

Prof. Subrat Kar

Dept of Eectrical Engineering, Indian Institute of Technology Delhi Hauz Khas,
New Delhi-110 016

CHARACTERIZATION OF INFRASONIC ACOUSTIC SIGNALS

Sumeet Kumar

Visvesvaraya National Institute of Technology, Nagpur, Maharashtra, 440010

Prof. Subrat Kar

Dept of Eectrical Engineering, Indian Institute of Technology Delhi Hauz Khas, New Delhi-110 016

Abstract

The human-animal conflict is one of the most serious conservation problems in Asia and Africa today. The unwanted confrontation of humans and wild animals takes the lives of many every year. For species that occupy large ranges or live in habitats that prevent receiving information through visual cues, acoustic signaling is often a primary means of communication, e.g., whales coordinating movements across oceans, birds singing to attract mates in thick forest habitat, and amphibians using long-ranging vocalizations to establish territories and attract mates. Asian (Elephas maximus) and African (Loxodonta africana and Loxodonta cycotis) elephants, which live in highly fluid fission-fusion societies, have been shown to use infrasonic vocal signals, also called rumbles, to communicate and facilitate social interactions across large distances. This qualifies the elephant as a perfect mode species for acoustic observation as it is possible to detect elephants by their rumbles even if they are out of sight. The rumbles, often have fundamental frequencies in the infrasonic range (fundamental range from 5 to around 30 Hz with harmonics sometimes extending to 600 Hz or higher) and can travel as far as 6 miles, with time duration strongly varying from 0.2s to more than 10s. Our aim is to make a model of an acoustic early warning system around the residential areas nearby elephants populated regions and to monitor their migration patterns. There are different methods to achieve this, like different statistical model classifiers such as Adaptive Boosting, SVM, Random Forest, which are based on their training over a large pre-collected dataset. We have opted for an alternative method, using Mel frequency cepstral coefficients (MFCC) features and Dynamic Time Warping (DTW) matching. Using MFCC, cepstral coefficients are extracted from every frame of the spectrogram. Then for feature matching, DTW is used to find the distance between feature vectors of input audio frame and a template frame. Using Raven lite 2.0 rumbles in a bunch of data were manually selected and saved to find the MFCC coefficients. The MATLAB software is used to implement MFCC and DTW on the processed data. In our proposed method training and testing data required is significantly less, therefore, we have used freely available online resources for elephant voice data; as getting vocalization data of free-ranging elephants is a very challenging task. This method has been popularly used for Automatic Speech Recognition (ASR) of human speech and it can prove useful in an early prediction of the movement of animals.

Keywords or phrases: Elephas maximus, Loxodonta africana, Loxodonta cycotis, MFCC, DTW

Abbreviations

Abbreviations

MFCC	Me frequency cepstra coefficients
DTW	Dynamic Time Warping

1 INTRODUCTION

1.1 Background

The low-frequency vocal signals can carry farther than calls in higher frequency bands, such as trumpets or roars, and therefore may be more useful in communicating over large distances or through thick vegetation that could more easily dampen high-frequency signals. Similar vocal signals are frequently used by African forest elephants (Loxodonta cycotis), which live in the dense rain forests of Central Africa and occupy ranges of up to 2000l square km. Forest elephants often exhibit coordinated social behaviors, such as synchronized gatherings near resources in forest clearings, which may be made possible by these infrasonic rumbles. Forest elephant populations have declined dramatically in recent years, largely due to illegal poaching for ivory. There is a critical need to closely monitor key forest elephant populations and better understand their distribution and habitat use. However, forest elephants largely inhabit closed-canopy forest where direct observation is difficult and aerial census methods, not applicable. Traditional methods for estimating population sizes of large mammal species, in particular, line-transect methods, are highly cost and labor intensive, limiting their frequency of use and rendering them unsuitable for monitoring seasonal shifts in habitat use and behavior. Furthermore, forest elephants are known to alter behavior and spatial distribution in response to anthropogenic presence and activity, meaning that traditional survey methods could be disruptive and therefore are poorly suited for an endangered population. Passive acoustic monitoring is a highly suitable method of monitoring forest elephants, as it is non-invasive and less labor-intensive and as such could facilitate more frequent tracking of population trends and behavior patterns than traditional methods.

1.2 Statement of the Problems

Passive acoustic monitoring and acoustic detection of forest elephants.

1.3 Objectives of the Research

1.3.1 Overall objective

Processing audio files in Raven lite 2.0 to understand rumble spectrogram features in the infrasonic range.

Implementing MFCC for feature extraction in MATLAB.

Implementing template matching using DTW in MATLAB.

1.4 Scope

The present work focuses only on implemention of MFCC and DTW on the rumbles annotated in an audio signal using spectrogram analysis. We can use it to characterise other animals with particular vocal features by changing some spectral parameters and frequency range of mel-fiter banks.

2 LITERATURE REVIEW

2.1 Information

Passive acoustic monitoring of forest elephants is necessarily different from previous analytical work on the problem of fundamental frequency estimation and harmonic characterization described above in that most of the time a fundamental frequency contour is expected and analysis is performed on short sound streams. Here, the technical challenge is acoustic object detection in the context of information retrieval, where target sound objects occur occasionally but not frequently, and therefore an approach based on feature extraction and classification is better suited for this problem.

Venter and Hanekom (2010) developed a novel method for detecting rumbles by estimating the fundamental frequency of adjacent windows in the sound stream and determining whether this value was constant over time.

Another method proposed by Wijayakuasooriya (2011) uses linear predictive coding to measure formant frequencies in the sounds stream and then classifies each sound segment using a hidden Markov model. However, although the aforementioned methods are technically rigorous they have not been sufficiently tested on large datasets in which rumbles are rare.

A third detection-classification method was put forth by Zeppelzauer et al. (2015), which uses several stages of spectrogram pre-processing before extracting cepstral features from the sound stream and then classifying each time frame using a support vector machine. This method appears to perform well in the presence of wind and rain noise and was tested more extensively than the former techniques, but was designed to detect rumbles from savannah elephants in a single reserve in South Africa.

Therefore a more comprehensive research is required in this area for a generalise detection of rumbles and not restricted to a region.

2.2 Summary

Using Raven Lite 2.0 rumbles in a bunch of data were manually selected and saved to find the MFCC coefficients. The MATLAB software is used to implement MFCC and DTW on the processed data. Using MFCC, cepstra coefficients are extracted from every frame of the annotated audio signal. Then for feature matching, DTW is used to find the distance between feature vectors of input audio frame and a reference template frame.

3 METHODOLOGY

3.1 Methods

The audio file of elephant rumbles was collected from open source websites. The audio file was annotated and a selection table was made which stores the start time, end time, lowest frequency, highest frequency of the selected rumble. these tabes were used to select the samples calculated from the start and end time in the audio file used to calculate the MFCC coefficients.

3.2 Feature Extraction: MFCCs

The first step is to extract features i.e. identify the components of the audio signal that are good for identifying the linguistic content and discarding the other stuff which carries information like background noise, emotion, etc. The main point to understand about speech is that the sounds generated are filtered by the shape of the vocal tract including the tongue, teeth, etc. This shape determines what sound comes out. If we can determine the shape accurately, this should give us an accurate representation of the phoneme being produced. The shape of the vocal tract manifests itself in the envelope of the short time power spectrum, and the job of MFCCs is to accurately represent this envelope.

3.2.1 Motivation behind the MFCCs steps:

An audio signal is constantly changing, so to simplify things we assume that on short time scales the audio signal doesn't change much (when we say it doesn't change, we mean statistically i.e. statistically stationary, obviously the samples are constantly changing on even short time scale). This is why we frame the signal into 100ms frames. If the frame is much shorter we don't have enough samples to get a reliable spectrum to estimate, if it is longer the signal changes too much throughout the frame. The next step is to calculate the power spectrum of each frame. This is motivated by the human cochlea (an organ in the ear) which vibrates at different spots depending on the frequency of the incoming sounds. Depending on the location in the cochlea that vibrates (which wobbles small hairs), different nerves fire informing the brain that certain frequencies are present. Our periodogram estimate performs a similar job for us, identifying which frequencies are present in the frame. The periodogram spectral estimate still contains a lot of information not required for vocal identification. In particular, the cochlea can not discern the difference between two closely spaced frequencies. This effect becomes more pronounced as the frequencies increase. For this reason, we take clumps of periodogram bins and sum them up to get an idea of how much energy exists in various frequency regions. This is performed by our Mel filterbank: the first filter is very narrow and gives an indication of how much energy exists near 0 Hertz. As the frequencies get higher our filters to get wider as we become less concerned about variations. Once we have the filter bank energies, we take the logarithm of them. This is also motivated by human hearing: we don't hear loudness on a linear scale. Generally to double the perceived volume of a sound we need to put 8 times as much energy into it. This means that large variations in energy may not sound a that different if the sound is loud, to begin with. This compression operation makes our features match more closely what humans actually hear. The final step is to compute the DCT of the log filter bank energies. There are two main reasons this is performed. Because our filter banks are overlapping, the filter bank energies are quite correlated with each other. The DCT decorrelates the energies which mean diagonal covariance matrices can be used to mode the features in e.g. a DTW classifier.

3.2.2 What is Mel-Scale?

The Mel scale relates perceived frequency, or pitch, of a pure tone to its actual measured frequency. Humans are much better at discerning small changes in pitch at low frequencies than they are at high frequencies. Incorporating this scale makes our features match more closely what humans hear.

The formula for converting from frequency to Mel scale is:

$$M(f) = 1125*\ln(1 + f/700) \tag{1}$$

To go from Mels back to frequency:

$$M^{-1}(m) = 700 * (exp(m/1125) - 1) \tag{2a}$$

3.2.3 MFCCs Implementation Steps

We have the audio signal of varied length recorded at a sampling rate (fs) is also varied. We now proceed with the following steps:

STEP 1: Divide the total Time Domain Speech Signal into FRAMES of width 100 ms (= fs*0.1 samples) with an overlap between consecutive frames of about 50 ms (=0.05*fs samples). If speech doesn’t divide into an integral value, pad extra zeros to the signal or cut down very few samples to the end of the signal. We finally would extract a set of 8 coefficients per frame after the entire MFCC process.

STEP 2: Now, to each frame multiply with a smoothing function such as a Hamming Window, and take the Discrete Fourier Transform of it (for each frame). We have used fft() function of MATLAB for this. Now, compute the Periodogram Estimate of the frame by taking the modulus squared of the components of the above taken Fourier Transform and divide it by the number of samples per frame.

STEP 3: Compute the Mel-spaced filter bank. This is a set of triangular filters that we apply to the periodogram power spectral estimate from step 2. To get the filter banks shown in the below figure we first have to choose a lower and upper frequency. Good values are 0.8Hz for the lower and 400Hz for the upper frequency. Using the Frequency to Mel-scale formula, convert the upper and lower frequencies to Mels. Now, take 10 additional points spaced linearly between the lower and upper Mel-scale values (as we need 8 filter banks). Convert these 10 Mel-linearly spaced points back to Frequencies using the Mel-to-Frequency conversion formula.

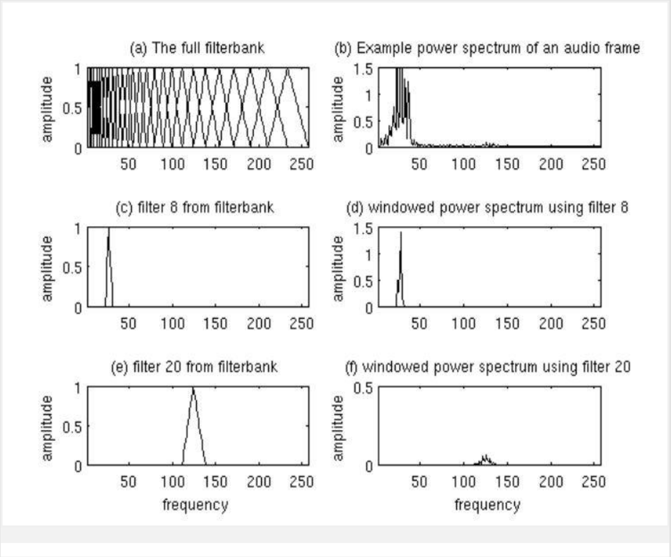
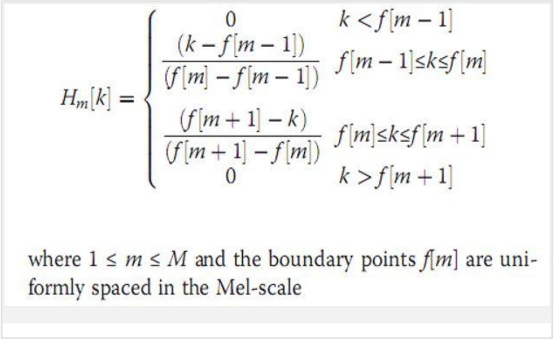


Fig 1 pot of me-fiter bank and windowed power spectrum.

Now we create our filter banks. The first filter bank will start at the first point, reach its peak at the second point, then return to zero at the 3rd point. The second filter bank will start at the 2nd point, reach its max at the 3rd, then be zero at the 4th etc.



$$H_m[k] = \begin{cases} 0 & k < f[m-1] \\ \dfrac{(k - f[m-1])}{(f[m] - f[m-1])} & f[m-1] \le k \le f[m] \\ \dfrac{(f[m+1] - k)}{(f[m+1] - f[m])} & f[m] \le k \le f[m+1] \\ 0 & k > f[m+1] \end{cases}$$

where $1 \le m \le M$ and the boundary points $f[m]$ are uniformly spaced in the Mel-scale

Fig 2 formula for calculating filter bank shown in the paper by Shen et al. [3] in EURASIP Journal (2012)

STEP 4: To caculate filter bank energies we multiply each filter bank with the power spectrum, then add up the coefficents. Once this is performed we are left with 8 numbers that give us an indication of how much energy was in each filter bank.

STEP 5: Take the log of each of the 16 energies from step 4. This leaves us with 16 log filter bank energies.

STEP 6: Take the Discrete Cosine Transform (DCT) of the 8 log filter bank energies to give 8 cepstral coefficents. Out of these, the first MFC Coefficient is just the sum of log signal energies, and therefore we remove it before processing to make our recognition loudness-Variations Robust.
The resulting features (8 numbers for each frame) are called Mel Frequency Cepstral Coefficients.

3.3 Pattern Matching- Dynamic Time Warping (The DTW Algorithm):

Now, once we got the MFC Coefficients of Speech Train Samples and Test Sample, we need an Algorithm for Classification designed especially for Speech Applications. The best match (lowest distance measured) is based upon dynamic programming. This is the DTW Algorithm. Note that here each speech train sample or test sample is associated with a set 8 coefficients per frame and the number of frames being num_frame . So, we have num_frame (8-dimensional) time sequence of vectors for each of base template and test audio files,(the number of frames in base and test files may not be same as they are of different time intervals and have different sampling frequency. Therefore match the frames available in the base template with the bigger annotated time duration with some hoping, so as to have a precise match) and we need to Match them! The distance between “Two Sequences of Vectors” is highly nontrivial and is achieved by DTW with extra advantages. The algorithm and description are mentioned below: We have not a single feature vector for each sample, but a

‘set of feature vectors’ that must be matched.
Since the feature vectors could possibly have multiple elements, a means of calculating the local distance is required. The distance measured between two feature vectors is calculated using the Euclidean distance metric. Therefore the local distance between feature vector x of signal 1 and feature vector y of signal 2 is given by,

$$d(x,y) = \sqrt{\sum_i (x_i - y_i)^2} \tag{2 a}$$

The input test audio file will be matched against the base template in the system's repository. The best matching is the one for which there is the lowest distance path aligning the input pattern to the template. A simple global distance score for a path is simply the sum of local distances that go to make up the path. To make the algorithm and reduce excessive computation we apply certain restriction on the direction of propagation. The constraints are given below.

- Matching paths cannot go backwards in time.
- Every frame in the input must be used in a matching path.
- Local distance scores are combined by adding to give a gobal distance.

This algorithm is known as Dynamic Programming (DP). When applied to template-based speech recognition, it is often referred to as Dynamic Time Warping (DTW). DP is guaranteed to find the lowest distance path through the matrix while minimizing the amount of computation. The DP algorithm operates in a time-synchronous manner: each column of the time-time matrix is considered in succession (equivalent to processing the input frame-by-frame) so that, for a template of length N, the maximum number of paths being considered at any time is N. If D(i,j) is the global distance up to (i,j) and the local distance at (i,j) is given by d(i,j):

$$D(i,j) = \min[D(i-1,j-1), D(i,j-1),D(i-1,j)] + d(i,j) \tag{2 a}$$

Given that D(1,1) = d(1,1) (this is the initial condition), we have the basis for an efficient recursive algorithm for computing D(i,j). The final global distance D(n, N) gives us the overall matching score of the template with the input. The input word is then recognized as the word corresponding to the template with the lowest matching score.

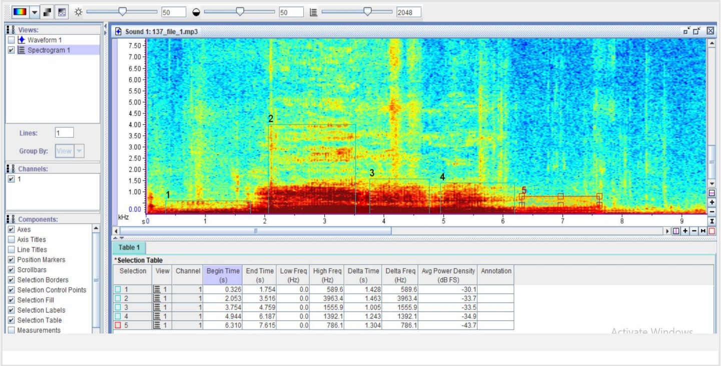
We have implemented the above DTW algorithm using dtw() in MATLAB.

4 RESULTS AND DISCUSSION

4.1 Purpose

We have tested our program with a total of 15 annotated audio signals and rumbles positively detected in the frames. We checked for a threshold distance over the euclidian distances found by dtw() function. Signals having elephant rumbles gave upon matching, a distance lesser than the threshold 270 for frames including the rumbles.

4.1.1 Annotated spectrogram on Raven Lite 2.0 :



Sound 1: 137_file_1.mp3

Views:
- ☐ Waveform 1
- ☑ Spectrogram 1

Lines: 1

Group By: View

Channels:
- ☑ 1

Components:
- ☑ Axes
- ☐ Axis Titles
- ☐ Line Titles
- ☑ Position Markers
- ☑ Scrollbars
- ☑ Selection Borders
- ☑ Selection Control Points
- ☑ Selection Fill
- ☑ Selection Labels
- ☑ Selection Table
- ☐ Measurements

Table 1

*Selection Table

Selection	View	Channel	Begin Time (s)	End Time (s)	Low Freq (Hz)	High Freq (Hz)	Delta Time (s)	Delta Freq (Hz)	Avg Power Density (dB FS)	Annotation
1	1	1	0.326	1.754	0.0	589.6	1.428	589.6	-30.1	
2	1	1	2.053	3.516	0.0	3963.4	1.463	3963.4	-33.7	
3	1	1	3.754	4.759	0.0	1555.9	1.005	1555.9	-33.5	
4	1	1	4.944	6.187	0.0	1392.1	1.243	1392.1	-34.9	
5	1	1	6.310	7.615	0.0	786.1	1.304	786.1	-43.7	

Fig 3 Annotated audio file for testing.(Selection table is saved)

4.1.2 Created mel filter bank with 8 triangular filter :

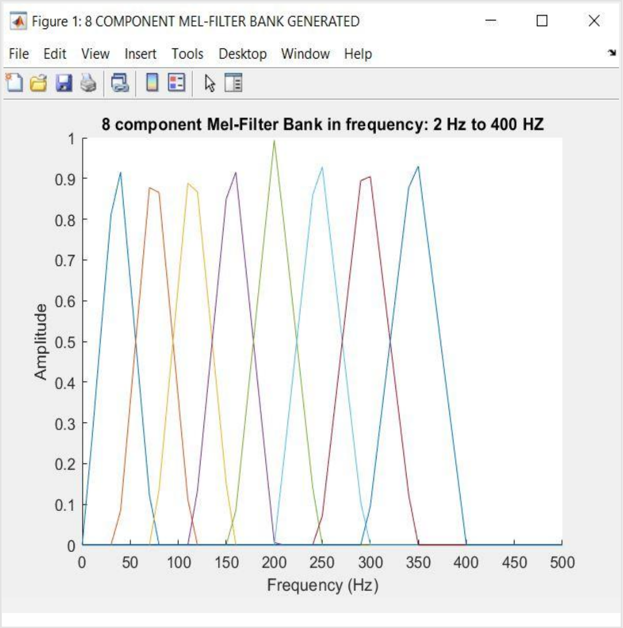


Figure 1: 8 COMPONENT MEL-FILTER BANK GENERATED
File Edit View Insert Tools Desktop Window Help
8 component Mel-Filter Bank in frequency: 2 Hz to 400 HZ
Amplitude
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Frequency (Hz)
0
50
100
150
200
250
300
350
400
450
500

Fig 4 8 component Mel-Filter bank in frequency range : 2-400 hz.

4.1.3 Feature matching result that is the euclidean distance due to DTW :

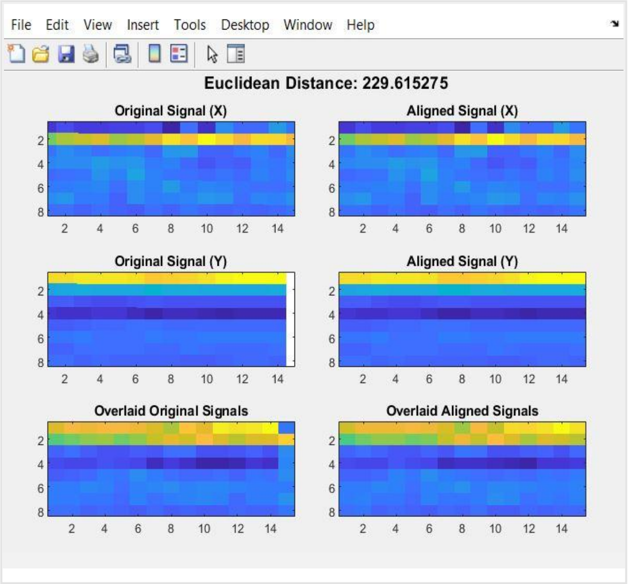


Fig 5 plot of DTW matching of base template(original signal Y) & test template(original signal X) along with the euclidean distance.

5 CONCLUSION

MFCCs provides a very powerful way to extract efficient Feature Vectors for performing Speech/Voice Recognition. Then a special method of DTW is applied to perform Feature Matching. These state-of-the-art methods have been discussed above.

ACKNOWLEDGEMENTS

I am grateful to the Science Academies' for giving me this wonderful opportunity. Financial assistance from the Indian Academy of Sciences (IAS) is gratefully acknowledged. I would like to thank my guide, Prof. Subrat Kar of IIT Delhi for his constant encouragement and support in building this project. I equally thank my student mentor Manohar Sir for his help and making me learn new concepts more easily.

REFERENCES

1. S. Davis and P. Mermelstein, IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 28 – 4, p.357- 366 (1980)

2. H. Sakoe and S. Chiba, IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. Assp-26 (1978)

3. L. Shen, C. Yeh and S. Hwang, EURASIP Journal on Audio, Speech and Music Processing 2012, 2012:28

4. L. Rabiner and B. Juang, Fundamentals of Speech Recognition, First Edition, ISBN-13: 9780130151575

5. https://github.com/bsvineethiitg/Digit-Speech-Recognition

6. http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/

7. https://hearinghealthmatters.org/waynesworld/2012/the-sounds-of-africa/

8. https://www.elephantvoices.org/elephant-communication/acoustic-communication.html