Audio Segmentation

Ashutosh Kulkarni, Deepak Iyer, Srinivasa Rangan Sridharan
Stanford University, Stanford.
{ashuvk, ideepak, rangan}@stanford.edu

Abstract

In this paper, we propose a novel algorithm to segment an audio piece into its structural components. The boundaries of the homogeneous regions are decided based on various time and frequency domain features. The algorithm has been designed in 2 stages. In the first stage, a vocal/non-vocal/silence classification is done using multinomial softmax regression. The second stage uses a hidden Markov model to 'smooth' the previous output as well as enforce the time dependent structuring. The training and testing was done specific to songs of the popular English rock group, The Beatles with an average error window of 600 ms.

1 Introduction

The internet has proved to be, by far the most effective medium for music distributors to reach out to music-lovers and enthusiasts by providing them with immediate access to music across geography, genre, artists and eras. Domains like amazon.com, iTunes.com are popular names in this context. A key component of their marketing strategy in this process is to allow a hearing of the songs before the users finalize their purchase. Any optimization of this process could help enhance user experience and thereby help bring in greater returns for distributors.

One such optimization in online music retailing would be to allow users to have direct access to different segments of a song (like the intro, verse, lead, outro). In this paper, we present an approach to solve the problem of segmenting a song into these homogeneous regions.

Most existing algorithms differ from our approach in the sense that we do not consider the full range of the frequency spectrum; and instead narrow in on the region where the information content is maximum and can be used best for classification. This, when coupled with a hidden Markov model which not only tries to negate the classification error of the previous algorithm, but also incorporates musical rules which are time-dependent makes our model capture almost all rules from the audio domain.

This paper is structured as follows: Section 2 gives an insight into the various feature vectors that were tried, while section 3 deals with the algorithm development and section 4 gives a detailed description of the final algorithm and the results. Concluding remarks, acknowledgements and references follow in Section 5.

2 Feature Vectors

(Henceforth, the term 'frame' refers to a sequence of 1024 audio samples. This frame size is popular in audio analysis due to the tradeoff it provides between the audio content and computational complexity)

There can be numerous metrics which can be used as feature vectors in audio analysis, a few of which we have incorporated into our model depending on their relevance to the problem at hand; which was classifying a frame into the classes we aimed in the first step of the algorithm. These feature vectors were calculated, as mentioned earlier, on frames of size 1024 samples. This section gives a brief description of the various feature vectors that were tried in the design of the algorithm.

2.1 Mel Frequency Cepstral Coefficients (MFCCs)

A spectrum is a positive real function of a frequency variable associated with a stationary stochastic process, while a cepstrum is the result of considering the spectrum (in mel scale) as the process. An MFCC

representation mathematically put, is the Discrete Cosine Transform (DCT) of the mel scale representation of the frequency spectrum and simply put, is the spectrum of a spectrum. MFCCs are very effective in representing the audio content as a whole. Their disadvantage is that they do not represent the perceived loudness accurately. This, in fact, worked in our favor because we desire an amplitude-independent analysis. Additionally, the number of MFCCs is a tradeoff between resolution and computational complexity. Hence, we narrowed down on 13 coefficients.

2.2 Spectral Flux

Spectral flux is a measure the rate of change of the power spectrum of a signal and is obtained by comparing the power spectrum of the current frame against previous frames. We took average spectral flux over 15 frames to be our feature vector, because most instrumental onset times are lower than this time. The spectral flux is expected to be larger for non-vocal content because of the onset and decays of instruments.

2.3 Zero Crossing Rate

It is the rate of sign changes of a signal in time, i.e. the high frequency content. Human vocals are known to exhibit a high zero crossing rate, because speech can be modeled as a random process.

2.4 Average Energy

Energy is the square of the amplitude. The energy of a frame gives an average of the total frequency content in a frame. This metric differs for vocal and instruments like the drums. Hence, it was selected as a feature vector.

2.5 Centroid

The centroid is a weighted mean in an algebraic sense. In audio analysis, it is that frequency which divides the spectrum into 2 parts with equal energy.

2.6 Rolloff

A percentage rolloff can give more information about the frequency distribution than the centroid. We tried various percentage rolloffs to try to capture the distribution.

Apart from these we also tried using the average energy over the past few frames. We settled upon MFCCs, spectral flux, zero crossing rate and average energy as our feature vectors because of their relevance to the problem and their ability to effectively incorporate most of the features.

Data

Data collection in itself is a tricky problem in the domain of music because the feature vectors are entities which are not directly visible. In order to keep the genre consistent over training and testing and to avoid boundary conditions due to song arrangements, we limited our study to the sound tracks of The Beatles. The training set was created by editing the soundtracks and creating pieces which contain purely instruments, vocals and silence. Structural components like outro, intro and bridge were not incorporated in the training stage because of a lack of distinction between them in any domain other than time. The training set requires a wide range of samples. An important thing to note here is that in audio, "wide range" is not mathematically but aesthetically. For e.g. in the training set for vocals, we would need samples with vocals in the low pitch, medium pitch and high pitch. Also, tracks with different levels of vocal to instrument amplitude ratios further add to the variety. Hence the training set of soundtracks was decided by careful analysis of all the soundtracks to give us a wide range of training samples. The data used for training of the Vocal and Non-vocal parts consisted of over 3000 frames. These frames were created from our database of songs by selecting small, representative clips (aesthetically) of audio. For Silence, 100 frames were used.

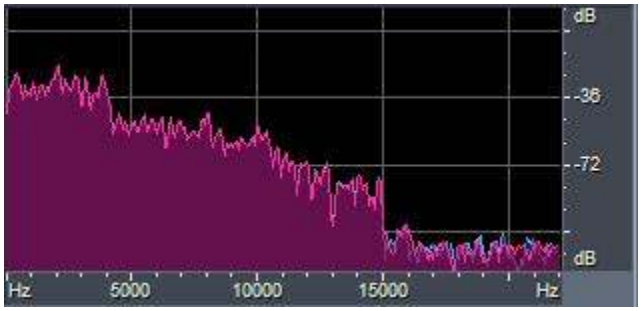
3. Algorithm Development

This section deals with the first step in the algorithm. Here, we classify each frame of the song as belonging to one of three classes – Non-vocal (**N**), Vocal (**V**) or Silence(**S**). These classes were chosen based on the fact that they help characterize the components that we seek to find in the song. For e.g.: The intro part of a song consists purely of Non-vocal content while the verse usually contains a good mix of both Vocal and Non-vocal content. The Silence class represents portions of the songs which can be ignored and is usually found to be at either the start or the end of the song

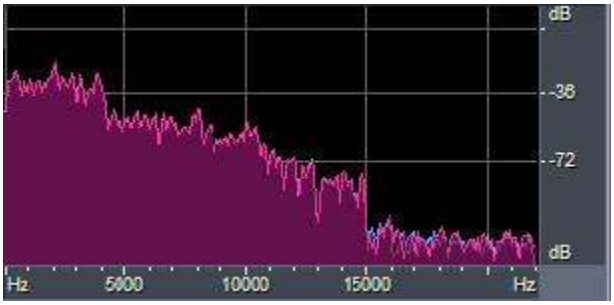
The problem of classifying each frame of the song as Vocal, Non Vocal or Silence was tried using various algorithms like multinomial softmax regres-

sion and support vector machines (with different kernels like linear, polynomial, weighted linear, weighted polynomial, rbf) but we selected multinomial softmax regression because of its performance over the other classifiers. Each frame of the song was thus classified based on the maximum probability class returned by the algorithm. The feature vectors mentioned in the previous section were used with all the classifiers.

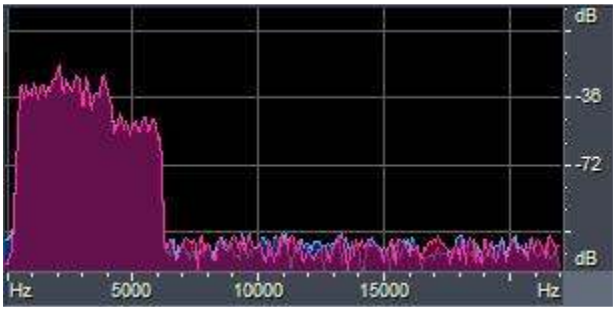
Existing algorithms compute the feature vectors based on the whole frequency spectrum (fig a). This is the basic initial approach we took. The results were far from satisfactory largely because of equal important being given to all parts of the spectrum. Also, the maximum vocal frequency formants in case of singing occur at around 3000 Hz. The next obvious approach was to boost the vocal range (around 300-3000 hz) in the spectrum (see fig. b). This approach did not work even though vocals were boosted because of the influence of other frequencies in the spectrum being higher. Hence, we decided to apply a steep band-pass filter on the training data-sets with a lower cut-off frequency of 400 Hz and various higher cut-off frequencies like 6000Hz (fig. c), 3000Hz (fig. d) and 2000Hz (fig. e). The primary reason for choosing the lower cut-off frequency at 400 Hz is to avoid using the base frequencies which occur uniformly throughout. The higher cut-off frequency of 3000 Hz gave the best results because it modeled the vocal range best. The performance errors were based on finding the error in milliseconds between the segment boundaries produced by the model and those obtained manually.



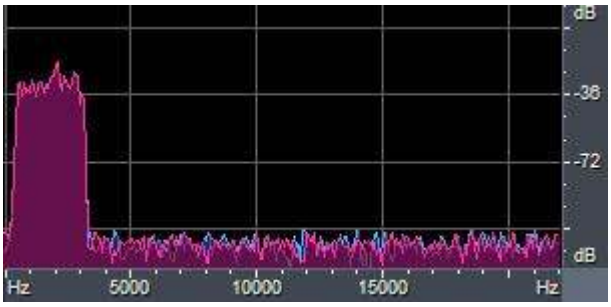
a) Unaltered Spectrum



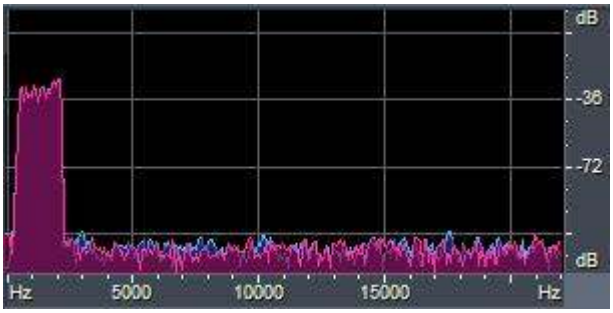
b) Vocal boost



c) 400-6000 Hz Band pass

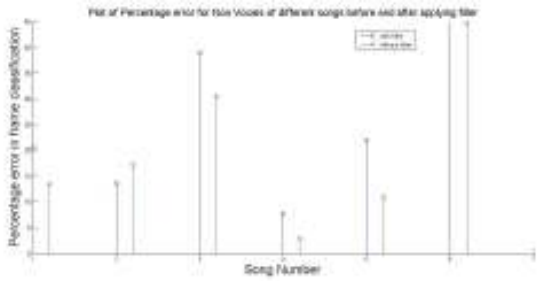


d) 400-3000 Hz Band pass



e) 400-2000 Hz Band pass

The results for an unaltered spectrum vs. a band pass of 400-3000 Hz are as follows:



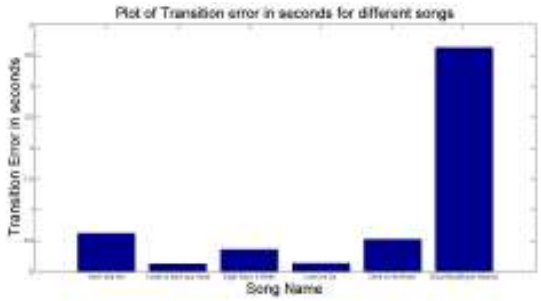
Non-vocal error

4. Algorithm

As mentioned in the previous section, a steep band pass filter with cut off frequencies of 400 and 3000 Hz is used on the training data-sets before computation of the feature vectors. The output of this stage is a sequence of a combination of the classes Vocal, Non-vocal and Silence.

Now given a time sequence, where each member of the sequence represents the class of the corresponding frame, the next step is to obtain the structures in the song. This is achieved by using a Hidden Markov Model, where the hidden states were "**Instrumental**" and "**Verse**", with each state "emitting" a Non-Vocal, Vocal or Silence frame. The output of the HMM therefore represents the sequence of states which are most likely to produce the sequence of frames that were observed.

Once the sequence of states is obtained, the **Instrumental** portions of the song are renamed as intro, outro or lead depending on whether they occur at the beginning, end or between two verses in the song. Such a procedure is followed because the distinguishing feature between the intro, outro and lead is along the time scale rather than in any other easily discernible feature.



Error in ms for different songs.

5. Future work

The most important issue we intend to work on to make the algorithm commercially is to expand the range of songs which are classified by the algorithm. The decent result for songs of different genres currently is because the model hasn't been trained for different arrangements in songs. We would like to expose our model to many more arrangements of songs and aesthetic variety of data so that it can work on any generic sample.

Acknowledgements

We would like to thank Asst. Prof. Ge Wang and Kyogu Lee from the Stanford University Center for Computer Research in Music and Acoustics (CCRMA) for their assistance and support right from the nascent stages of the project. We would also like to express our gratitude towards Prof. Roger Dannenberg of Carnegie Mellon University for his insightful guidance. We are thankful to Prof. Andrew Ng and the TAs of the course who were always available and ready to help.

References

[1] R. Dannenberg and M. Goto, "Music *Structure Analysis from Acoustic Signals*", Music Structure 16 April 2005.

[2] A. L. Berenzweig and D. P. W. Ellis, "*Locating singing voice segments within music signals*", Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp.119-122, New York, Oct. 2001.

[3] Tong Zhang, "*Semi-Automatic Approach for Music*", HPL – 2003.

[4] Goffredo Haus, Luca A. Ludovico, "*Music Segmentation: An XML-oriented Approach*", LIM-DICO University of Milan.
[5] Kristoffer Jensen, "*Multiple Scale Music Segmentation Using Rhythm, Timbre and Harmony*", EURASIP Journal on Advances in Signal Processing, Volume 2007, Article ID 73205.
[6] Tom Zhang and Jay Kuo, "Content based classification and retrieval of audio", Integrated Media Systems and Electrical Engineering systems, University of Southern California.