A TWO PHASE METHOD FOR GENERAL AUDIO SEGMENTATION

Jessie Xin Zhang[1], *Jacqueline Whalley*[1], *Stephen Brooks*[2]

[1] School of Computing and Mathematical Sciences, Auckland University of Technology, Auckland 1010, New Zealand {jwhalley, jessie.zhang@aut.ac.nz}
[2] Faculty of Computer Science, Dalhousie University, 6050 University Avenue, Halifax, Nova Scotia, Canada B3H 1W5 {sbrooks@cs.dal.ca}

ABSTRACT

This paper presents a model-free and training-free two-phase method for audio segmentation that separates monophonic heterogeneous audio files into acoustically homogeneous regions where each region contains a single sound. A rough segmentation separates audio input into audio clips based on silence detection in the time domain. Then a self-similarity matrix, based on selected audio features in the frequency domain to discover the level of similarity between frames in the audio clip, is calculated. Subsequently an edge detection method is used to find regions in the similarity image that determine plausible sounds in the audio clip. The results of the two phases are combined to form the final boundaries for the input audio. This two-phase method is evaluated using established methods and a standard non-musical database. The method reported here offers more accurate segmentation results than existing methods for audio segmentation. We propose that this approach could be adapted as an efficient pre-processing stage in other audio processing systems such as audio retrieval, classification, music analysis and summarization.

Index Terms— Audio segmentation, similarity map, edge detection

1. INTRODUCTION

Segmentation plays an important role in audio processing applications, such as content-based audio retrieval recognition and classification, and audio database management. Audio segmentation is a process that divides an audio file into its composite sounds. Each segment or clip should consist of a single sound that is acoustically different from other parts of the audio file. An accurate segmentation process can identify appropriate boundaries for partitioning given audio streams into homogeneous regions.

Although there are many approaches to audio segmentation they are focused on a narrow type of audio such as speech/music separation, speaker recognition and music structure extraction. These methods work well for specific tasks but are not generalisable even, for example, across different genres of music. A method to segment any given audio file that contains different types of sounds remains an open and important problem.

2. RELATED WORK

Several methods have been developed for audio segmentation. Chen identifies two types of segmentation approaches namely, classification-dependent segmentation (CDS) and classification-independent segmentation (CIS) [1]. CDS methods are problematical because it is difficult to control the performance [1].

CIS approaches can be further separated into time-domain and frequency-domain depending upon which audio features they use, or supervised and unsupervised approaches depending on whether the approach requires a training set to learn from prior audio segmentation results. CIS may also be defined as model-based or non-model based methods.

The work of Panagiotakis and Tziritas [2] is a typical time-domain approach and uses the root-mean-square (RMS) and zero crossing rate (ZCR) to discriminate speech from music. Tzanetakis and Cook [3] presented a general methodology for temporal segmentation based on multiple features. In model-based approaches, Gaussian mixture models **(**GMM) [4], [5], Hidden Markov Models (HMM) [6], Bayesian [7], and Artificial Neural Networks (ANN) [8] have all been applied to the task of segmentation. Examples of an unsupervised audio segmentation approach can be found in [7] and [9]. These unsupervised approaches test the likelihood ratio between two hypotheses of change and no change for a given observation sequence. On the other hand, the systems developed by Ramabhadran et al. [6] and Spina, and Zue [4] must be trained before segmentation. These existing methods are limited because they deal with limited and narrow classes such as speech/music/noise/ silence.

Audio segmentation methods based on a similarity matrix, have been employed for broadcasting news, which is relatively acoustic dissimilar, and for music to extract structures or music summarizations. The accuracy evaluation of these methods was undertaken with specific input audios and has not been previously reported for use with audio files in a non-music/non-speech database. This

978-1-4244-4291-1/09/$25.00 ©2009 IEEE

paper introduces a two phase unsupervised model-free segmentation method that works for general audio files. In this paper, we discuss the process by which we developed and evaluated an efficient CIS method that can determine segment boundaries without being supplied with any information other than the audio file itself.

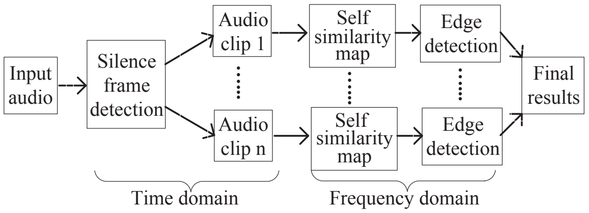

Fig. 1. Frame work for audio segmentation method.

3. THE TWO PHASE SEGMENTATION METHOD

The first stage of the segmentation method roughly separates audio by using silence detection in the time domain. After this initial segmentation, further segmentation in the frequency domain is performed. This is done by taking advantage of the fact that sounds tend to be homogeneous, in terms of a sounds audio features. We make the assumption that any abrupt change in the audio features indicates the start of a new sound clip. This more sophisticated approach is performed on segments from initial segmentation to detect subtle changes based on more complex acoustic features until each audio clip contains a single sound.

3.1. Time Domain Silence Segmentation

The first phase of our segmentation finds the start and end of audio clips based silence periods. Root Mean Square (RMS) [2] and Spectral Power (SP) [10] are widely used audio features in silence detection. If the signal in a frame is less than a pre-determined threshold [10], it is regarded as a silence frame. We have found that neither RMS nor SP are suitable when the sound's duration is less than 10ms, (e.g. hand claps in Fig.2).

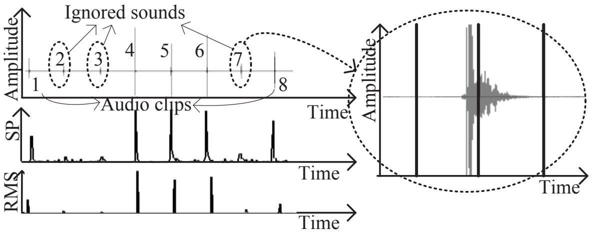

Fig. 2. (L) Wave shape of audio file "Hands clapping" and its SP, RMS; (R) Enlarged wave shape of a sound in "Hands clapping".

The signal in Fig 2 contains 8 hand claps, each lasting ~ 11ms. Using both RMS and SP, only 5 are detected. The remaining 3 clearly audible sounds that are visually identifiable in the time-amplitude signal shapes are not detected. When sound 1 and sound 7 are compared (the top image in Fig.2 (L)) the SP values of the two sounds are similar if we manually mark their boundaries. However sound 7 is not identified as signal in the input audio. This is because when the input audio is split into frames, the Sound 7 is cut into two frames (Fig.2 (R)) which contain short limited sample points, resulting in two frames being detected as silence. Neither RMS nor SP works for short duration signals such as this hand clapping example.

In order to cope with very short sounds we propose a new segmentation method as follows. The maximum amplitude value of each frame is used as a parameter to label silence and signal frames with an empirical threshold T. The threshold T adapts to each audio file, as follows:

$$T = \begin{cases} 0.015 \times \max(A), if\ mean(A) > 0.075 \times \max(A) \\ 0.05 \times \max(A),\ if\ mean(A) \leq 0.075 \times \max(A) \end{cases} \quad \mathbf{(1)}$$

where A is the absolute amplitude of the signal. Adapting the threshold using the mean value of the absolute amplitudes of an input sound is necessary to minimize the effects of noise signals within audio signals of variable SP. This results in higher thresholds that eliminate most of the noise in a signal but may also result in the removal of the edges of a fading signal. This tradeoff between loss of signal edges and elimination of noise is necessary to achieve a good segmentation.

We separate the given audio into frames, each lasting 2.5ms. Such a short frame size is not appropriate for accurate audio feature extraction. But in our method audio features are not required in this phase. We have found that a 2.5ms frame leads to accurate silence detection for signal boundaries using the maximum amplitude.

Once all the frames have been marked, the start and end points for each audio clip are detected. Given frame f_i if f_{i-1} and f_{i-2} are silent then f_i is a clip start frame f_s and if $f_{i+1}$and f_{i+2} are silent then f_i is a clip end frame f_e. The clip is then defined as $C \in \{f_s, f_{s+1},...f_e\}$. If the silence between two audio clips C_i and C_{i+1} is short (less than 0.01 × *input_audio_duration* for general sound) these two audio clips are combined. If an audio clip C_i is very short (less than 5 ms in duration), it is hard to perceive audibly and therefore does not need to be regarded in further analysis and is discarded as noise. If a result sound clip lasts more than 5ms but less than 16ms, the frame in front of it and the frame after it are added to it to form a signal clip. The result of the first phase is a set of sound clips (signal frames).

Fig. 3 gives an example of segmentation results based on silence detection. There are 2 signals in the input audio. The first audio is a telephone's touch tones, (marked in gray) and the second is a violin (marked in black). Using our adaptive threshold based on amplitude the silence periods between the telephone touch tones are correctly detected.

But if 2 sounds are seamlessly connected with no silence or very short silence in between, such as clip 7 in the telephone/violin audio, silence detection does not work. It is necessary to undertake a second segmentation refinement phase to deal with contiguous adjacent signals.

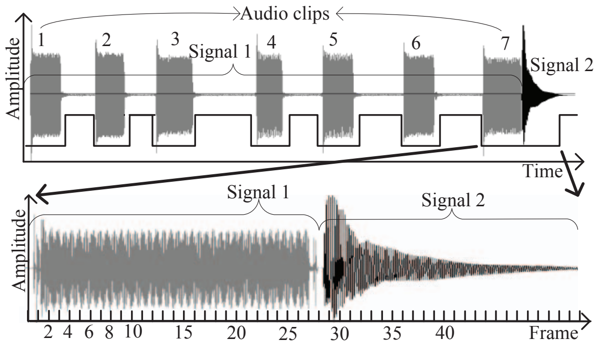

Fig. 3. Audio segmentation result for a mixed sound audio file.

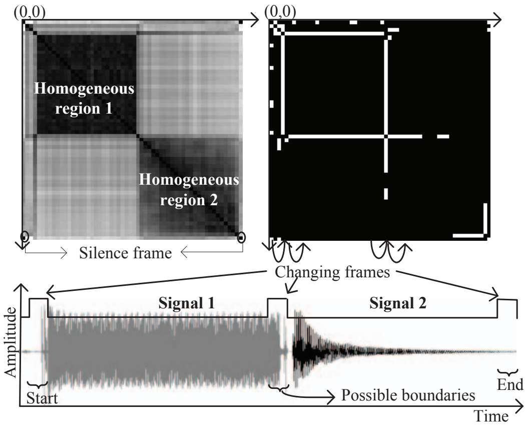

Fig. 4. Top: (L) Similarity map image; (R) Edge detection of similarity map image; Bottom: Segmentation result based on similarity map image.

3.2. Frequency Domain Similarity Segmentation

For any resultant audio clips from the first phase, if it needs further segmentation, a sophisticated segmentation approach is used to detect if there are subtle changes in acoustic features.

We adapt Foote's similarity map [11] into a general sound segmentation procedure. Foote first introduced the similarity map to identify significant changes in music. A checkerboard kernel is used to calculate cross-similarity values between different regions along the diagonal of the similarity map. Kernel size limits the accuracy, a small kernel tends to be sensitive and on a short time scale is capable of finding detailed changes. On the other hand, a large kernel is suitable for those audio files that have few segments but each segment contains several sounds. In a similar vein, Cooper [12] used a 2D similarity matrix for music summarization. Because it is impossible to find a kernel size that suits all kinds of sounds, we developed an edge detection method using image processing techniques, instead of a kernel comparison, based on a similarity map that finds changes in audio properties.

Generally speaking, accurate audio feature extraction requires sounds of sufficient duration (such as the standard frame size 16ms in [13]). We perform further segmentation on an audio clip if it is longer than 160ms (10 frames). For the audio clips shorter than 160ms, there is little possibility of them containing more than one distinguishable sound. In the second phase we process any audio clips with duration of 160ms or longer that were generated in phase 1 and separate them into 16ms non-overlapping frames.

An audio feature vector is then extracted for each frame. The audio features are: Total Spectrum Power, Bandwidth, Brightness, Pitch and an 8 order Mel Frequency Cepstral Coefficient (MFCC). This feature set was selected as it has been shown to give the most accurate classification result and can therefore best represent the difference between classes. Details of the computation of these features can be found in [13].

A similarity matrix of these audio vectors is then calculated, where each element represents a distance of two frames. This similarity matrix is normalized and is represented by a gray scale similarity map (image) for the audio clip. If there is a single sound in the audio clip, the similarity map is homogeneous. Otherwise a similarity map illustrates the differences between the audio clips' frames. By applying Sobel edge detection [14] on the similarity map, a heterogeneous image can be separated into homogeneous regions. The edge pixels of the regions represent the boundaries in the audio clip. In this way, the audio clip is then separated into homogeneous sound regions.

For a homogeneous audio clip, the similarity distances are similar before normalization, but are extended in the similarity map. To represent the audio clip correctly, two silence frames are added at the front and the end of the audio clip. After adding a silence frame, the distances between the signal frames in the audio clip are compared with silence frames. Homogeneous audio clips will not be extended. The similarity is measured and compared using 5 different distance metrics, as discussed in section 4.

We use the audio clip in Fig. 3 (lower image) as an example; its similarity map is the left image of the top row in Fig. 4. The similarity map has zeros along its diagonal because the distance is zero from each frame to itself. There are 2 homogeneous blocks in the similarity map that indicate two homogeneous sounds in the audio clip. Fig. 4Top(R) gives the resultant image from performing Sobel edge detection on the self-similarity map (Fig. 4Top(L)). The changing frames are clearly evident from the result edge detection image (Fig. 4Top(R)). The final segmentation result for the phone and violin mixed audio clip is illustrated in Fig. 4 (bottom image). We can see that the edges of similarity map highly illustrate the boundaries of the sounds.

4. SEGMENTATION METHOD EVALUATION

To fully evaluate this novel segmentation method we employ a widely used standard non-musical sound database called MuscleFish [13]. MuscleFish contains 16 classes and

410 audio files. When contrasted with previously reported segmentation evaluations this database introduces some significant segmentation issues. Some classes in MuscleFish contain many audio files that are very similar in acoustic audio features. Moreover, the database has a broad range of classes including, music, speech, animal sounds and everyday sounds (e.g. thunderstorm and hand clapping).

For segmentation evaluation, we generated 10 groups, of 100 audio files. Each file is generated by conjoining two randomly selected files from the MuscleFish database. As the audio files in MuscleFish contains up to 7 sounds, the generated audio files may contain up to 14 segments. To evaluate how our segmentation method works, we compared the two phase method with a Euclidean similarity map, a cosine angle and silence detection (Table 1). To fully evaluate the use of Euclidean distance in our method, we also tested the two phase method using a cosine angle in the second phase. After extracting the audio features for all the frames, for each feature, the values of all audio frames are normalized to [0, 1]. This ensures each feature has the same weight in the Euclidean distance calculation.

A tolerance frame before and after the true boundary is integrated for evaluation. This small tolerance is introduced to allow frames for what contains more than one signal. If two signals belong to the same class, we regard any segmentation result as "correct" because they are similar in acoustic features.

Table 1 shows the segmentation accuracy using different methods. Our two-phase method with Euclidean distance in a similarity map gives the most accurate segmentation result with an average of 91.9%. The result using silence detection gives the worst average result (81%).

When comparing the result of using the cosine angle and the Euclidean distance, the latter gives better accuracy regardless of the approach (two-phase method or the similarity map alone). Compared with using a similarity map alone, our two-phase method gives improved accuracy (see Fig. 5).

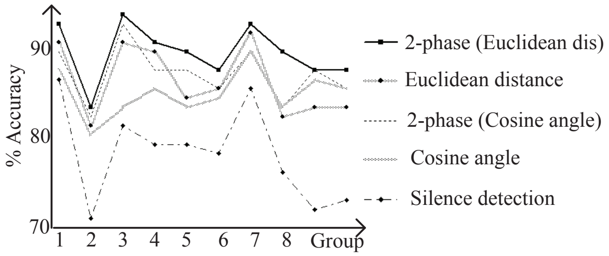

Fig. 5. Segmentation accuracy using various methods.

5. CONCLUSION

This paper provides a model-free and training-free two-phase method for general audio segmentation. From evaluation, because there are a wide variety of classes in the MuscleFish database, we believe our two-phase segmentation method provides a good solution to general audio segmentation problems. As a power segmentation tool, this method outperforms typical CDS methods or model-based approaches reported in the literature. Moreover, by judicious design of the feature set, this method can be tailored for other applications such as music structure analysis and summarization in music field and speaker detection in speech analysis as well as audio retrieval and classification for general sounds.

Table 1. Accuracy of different segmentation methods.

Group	Average accuracy
2-phase Euclidean	91.9%
2-phase Cosine	89.1%
Euclidean	89.6%
Cosine distance	87.6%
Silence detection	81.0%

6. REFERENCES

[1] G. Chen, H. Tan, and X. Chen, "Audio Segmentation via the Similarity Measure of Audio Feature Vectors", Wuhan University Journal of Natural Sciences, Vol. 10, No. 5, pp.833-837, 2005.

[2] C. Panagiotakis, and G. Tziritas, "A Speech/Music Discriminator Based on RMS and Zero-Crossings", IEEE Transactions on Multimedia, pp. 155-166, 2005.

[3] G. Tzanetakis, and P. Cook, "A Framework for Audio Analysis Based on Classification and Temporal Segmentation", EUROMICRO, pp. 2061- 2067, 1999.

[4] M.S. Spina, and V.W. Zue, "Automatic Transcription of General Audio Data: Preliminary Analyses", pp. 594-597, ICSLP 96, 1996.

[5] H. Aronowitz, "Segmental Modeling for Audio Segmentation", Acoustics, Speech and Signal Processing, pp. 393-396, 2007.

[6] B. Ramabhadran, J. Huang, U. Chaudhari, G.Iyengar, and H.J. Nock, "Impact of Audio Segmentation and Segment Clustering on Automated Transcription Accuracy of Large Spoken Archives", Proc. EuroSpeech, pp. 2589-2592, 2003.

[7] S.S. Chen, and P.S. Gopalakrishnan, "Speaker Environment and Channel Change Detection and Clustering via The Bayesian Information Criterion", in *DARPA* speech recognition workshop, pp. 127-132, 1998.

[8] H. Meinedo, and J. Neto, "A Stream-based Audio Segmentation, Classification and Clustering Pre-processing System for Broadcast News using ANN Models", INTERSPEECH, pp. 237-240, 2005.

[9] M.K. Omar, U. Chaudhari, and G. Ramaswamy, "Blind Change Detection for Audio Segmentation", Acoustics, Speech, and Signal Processing, pp. 501-504, 2005.

[10] A. Ganapathiraju, L. Webster, J. Trimble, K. Bsush, and P.Kornman, "Comparison of Energy-based Endpoint Detectors for Speech Signal Processing", Southeastcon, pp. 500-503, 1996.

[11] J. Foote, "Visualizing Music and Audio Using Self-Similarity", *ACM Multimedia*, pp. 77–84, 1999.

[12] M. Cooper, and J. Foote, "Automatic Music Summarization via Similarity Analysis", Proc. IRCAM, pp. 81-85, 2002.

[13] S.Z. Li, "Content-based audio classification and retrieval using the nearest feature line method", IEEE Trans. Speech and Audio Proc., pp. 619-625, 2000.

[14] R. Duda, and P. Hart, *Pattern Classification and Scene Analysis*, John Wiley & Sons Inc, 1973.