Segmentation of Audio Data Based on the Binary Images of the Audio Samples

S.R. Subramanya

National University (California)

111 PUBLICATIONS **369** CITATIONS

SEE PROFILE

Ilker Ersoy

University of Missouri

37 PUBLICATIONS **361** CITATIONS

SEE PROFILE

Segmentation of Audio Data Based on the Binary Images of the Audio Samples

S.R. Subramanya
Dept. of Computer Science
University of Missouri–Rolla
Rolla, MO 65409
subra@umr.edu

Ilker Ersoy
Dept. of Computer Science
University of Missouri–Rolla
Rolla, MO 65409
ersoy@umr.edu

Abdou Youssef
Dept. of EE and CS
George Washington University
Washington, DC 20052
youssef@seas.gwu.edu

Abstract

Audio databases are beginning to be developed to cope with the phenomenal increases in the generation and use of audio data in several applications. Efficient indexing of data facilitates fast and accurate retrievals for content-based queries, which are crucial from a user point of view. Segmentation of audio data is a common operation done during the analysis of audio data, to derive the indices. This paper proposes a novel scheme to segment audio data, based on the segmentation of the binary image derived from the audio samples. The viability of the proposed scheme is confirmed by the experimental results.

1 Introduction

The phenomenal increases in the generation and use of audio data in several computer applications have necessitated the development of audio databases with newer features such as *content-based queries* and *similarity searches*([9, 3, 7]). Example applications of audio databases are in digital libraries, entertainment industry, forensic laboratories, virtual reality, and several others. Of interest to users are easy-to-use queries, with fast and accurate retrievals for content-based queries. The latter is dependent on a good indexing scheme, where the indices are structured with respect to the *audio features* extracted from the data. The kind of features is a design choice which depends on the level of analysis.

A scheme for indexing audio data based on transforms was proposed in [7] which handled audio data of any kind, and of any duration, by considering short windows of signal (blocks) and application of short-term transforms. However, semantic units of units of data, such as segments, are expected to provide better indices which facilitate accurate retrievals. Segments of audio data could be used as indices either directly or as 'units' from which indices could be derived by further processing. The indices so derived are expected to provide more accurate data retrievals. However, segmentation of audio data is a hard problem. Most audio segmentation schemes handle only speech data. They use special characteristics in the speech signal. In this paper we propose a scheme for segmentation of any kind of audio data (not necessarily speech).

The next section briefly describes the use of segmentation in the indexing process. Section 3 describes the proposed scheme for mapping audio waveform to a binary image and using the segmentation of the derived binary image to obtain the audio segmentation. Section 4 presents the experimental results, followed by conclusions.

2 Segmentation of Audio Data for Indexing

An audio database supporting content-based retrievals should have the indices structured with respect to the *audio features* which are extracted from the data. The kind of features is a design choice which depends on the level of analysis. Audio being a *non-stationary* signal, the indices derived by a global analysis of data may not adequately capture the finer, local variations in the data. Indices based on semantic units of data are expected to provide accurate retrievals to content-based queries, which is one of the motivating factors for segmenting audio data for the purposes of indexing.

Segmentation is one of the common operations done during the analysis of audio/multimedia data. Segmentation essentially groups data elements (image pixels, audio samples, video frames and pixels) which are similar to each other, and in addition, close to one another. Within an audio data segment, the signal characteristics do not have marked variations, while the boundaries between two consecutive segments have rapid changes in the signal. The segments are usually variable-sized units. A *segment* of audio data roughly denotes a meaningful unit of data, which could be used

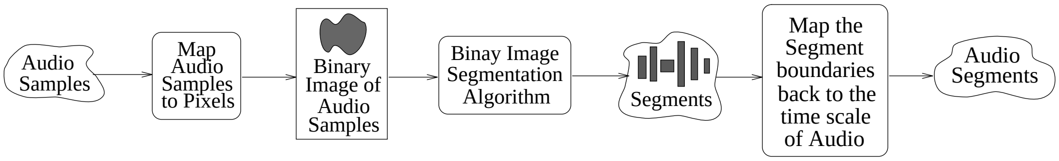
Audio Samples
Map Audio Samples to Pixels
Binary Image of Audio Samples
Binay Image Segmentation Algorithm
Segments
Map the Segment boundaries back to the time scale of Audio
Audio Segments

Figure 1: Outline of the proposed audio data segmentation scheme.

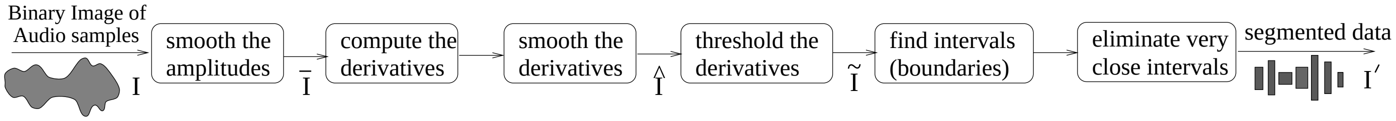

Figure 2: Steps in the segmentation of the binary image derived from audio samples.

as indices either directly, or as ‘metadata’ from which indices could be derived by further processing. It is desirable for the scheme for segmenting audio to be both fast and accurate.

Almost all of the audio segmentation schemes deal with speech data. Among the numerous schemes developed for the segmentation of speech, we mention only a few schemes below. Automatic segmentation schemes of speech proposed in [5, 8] are based on acoustical features of speech. The algorithm of [6] uses both energy and zero crossing thresholds to detect the endpoints of speech. In [4], algorithms based on hidden markov models (HMM) for dividing audio into segments corresponding to different speakers (or acoustic classes) are presented. Each speaker is modeled using an HMM consisting of states corresponding to the acoustic patterns produced by the speaker. No phonetic knowledge is used. Speaker segmentation is performed using the *Viterbi algorithm.* In *rhythmic* speech (for example recitation of verses, hymns, etc.), prosodic features need to be taken into consideration for segmentation. The scheme proposed in [2] considers both prosodic and acoustical features of speech for rhythmic speech segmentation.

3 The Proposed Scheme

The proposed scheme is based on the observation that the plot of the samples of audio data can be treated as a binary image, which exhibits well defined segments. These segments in the image (plot) correspond fairly well to segments of the audio data perceived by a human listener, when the audio is played on a speaker. This *visual-aural correspondence* is the basis for the development of this scheme, which assures the correctness of the segmentation. The resulting binary image can be segmented by a very simple algorithm. This simplicity of the segmentation of the binary image (plot of audio samples), compared to the segmentation of raw audio data, offers considerable speedup in the segmentation, and is the motivation for the proposed scheme. Figure 1 shows the outline of the proposed scheme. The audio samples are plotted on a suitable scale, determined empirically. Too few samples per unit width results in the plot with finer variations of the audio signal, and loss of higher level perception. Too many samples per unit width results in losing finer variations. The plot is thereafter treated as a binary image I, and considered for segmentation. The steps in the segmentation of the binary image is shown in Figure 2.

The image is first passed through a low-pass filter (to remove noise and jagged edges) to get an approximate envelope of the image contour, $\bar{I}$. The derivatives of this envelope are then computed, which correspond to the sharp changes in the contour (this is equivalent to high-pass filtering). The result is then passed through a low-pass filter to remove spurious spikes, and the resulting image is denoted by $\hat{I}$. Then the mean and standard deviation are determined, and all values below the threshold of the sum of mean and standard deviation are zeroed. The image at this stage, denoted by $\tilde{I}$, shows demarkations between the ‘segments’. Subsequently, the segments which are too close together (below a threshold) are merged to get I'. This is then mapped back to the audio plot to obtain the audio segments.

3.1 Quantification of segmentation error

The segments obtained by using the proposed scheme (referred to as *automated segments*) are compared with segments obtained by manual segmentation (referred to as *manual segments.* Manual segmentation is done by carefully listening to the audio clips (using audio tool) and marking the positions (in time) where there are aurally significant changes in the audio

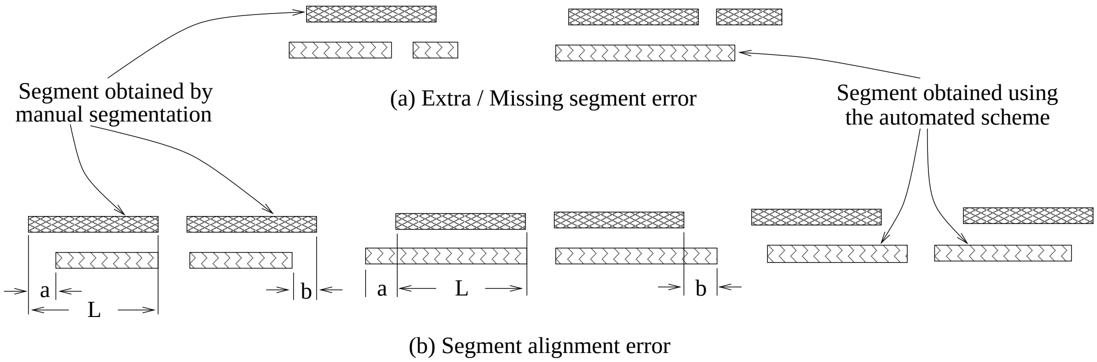
Segment obtained by manual segmentation
(a) Extra / Missing segment error
Segment obtained using the automated scheme
a
L
b
a
L
b
(b) Segment alignment error

Figure 3: Classes of segmentation errors.

data. Note that the basic objective of the segmentation is to use the segments in content-based retrievals and not in any kind of 'recognition'. Any deviation of the automated segments from the manual segments is considered a *segmentation error*. Thus the manual segments are used as the basis for determining the 'goodness' of the proposed scheme. We identify two classes of segmentation errors: $\mathcal{E}_1$, the *extra/missing segment error*, and $\mathcal{E}_2$, the *segment alignment error*. In the former class, the number of automated segments is more/less than the manual segments, and in the latter case, an automated segment could be misaligned with the corresponding manual segment by a factor of $\pm a$ and/or $\pm b$. These are shown in Figure 3. The errors are quantified using the following formulae:

$$\mathcal{E}_1 = \frac{\text{No. of extra segments} + \text{No. of missing segments}}{\text{Total number of manual segments in the data}}$$

$$\mathcal{E}_2 = \frac{1}{2}\frac{\sum_{i=1}^{N}(\mid a_i \mid + \mid b_i \mid)/L_i}{N}$$

where N is the number of segments in the data, and L_i is the length of segment i, a_i and b_i are the amount of misalignments at the segment boundaries (see Fig. 3).

4 Experimental Results

The proposed scheme was implemented and tested on several audio data consisting of speech, music, and other sounds. The audio files were Sun/NeXT `.au` files. The μ-law encoded files were converted to linear encoding, and the plots of these samples were treated as the binary images to be segmented.

The snapshots of the data during the segmentation process of three different audio data: (1) speech (`quota.au`), (2) music (`gatchaman7`), and (3) chirps of a bird (`bird-1.au`), are given in Figures 4, 5, and 6, respectively. In each of these figures the top leftmost figure is the original signal, the top middle figure represents $\bar{I}$, the result of passing the image I through the low-pass filter; the top right figure is the image $\hat{I}$ obtained after taking the derivative of $\bar{I}$ and then smoothing it; the bottom left figure is the image $\tilde{I}$ resulting from thresholding of $\hat{I}$; the bottom middle figure shows I', the final segments (after merging segments in $\tilde{I}$ which are too close); the bottom right figure shows the segments obtained by manual segmentation.

The segmentation errors of a few representative files are tabulated in Table 1, where N is the number of (manual) segments in the data, N_e and N_m are the number of extra and missing segments in the automated segments.

				Errors	
File	N	N_e	N_m	$\mathcal{E}_1$	$\mathcal{E}_2$
Bach-Toccata	16	2	4	0.37	0.0911
Figaro	8	11	0	1.38	0.0782
RRE2	22	1	3	0.18	0.0839
anyway	13	1	0	0.08	0.1044
gatchaman.1	11	13	0	1.18	0.0428
gatchaman1	13	4	0	0.31	0.0253
laugh	6	10	0	1.67	0.1084
quota	14	4	1	0.36	0.0833
CavalryCall	25	1	2	0.12	0.0928
data-amusing	9	8	0	0.89	0.0531
gatchaman6	8	4	0	0.50	0.0182
gatchaman7	19	4	2	0.32	0.0719
sample3	20	4	3	0.35	0.1016
jerry	17	7	1	0.47	0.0885
sample2	22	4	3	0.32	0.2473
bird-1	18	0	2	0.11	0.0655

Table 1: Segmentation errors

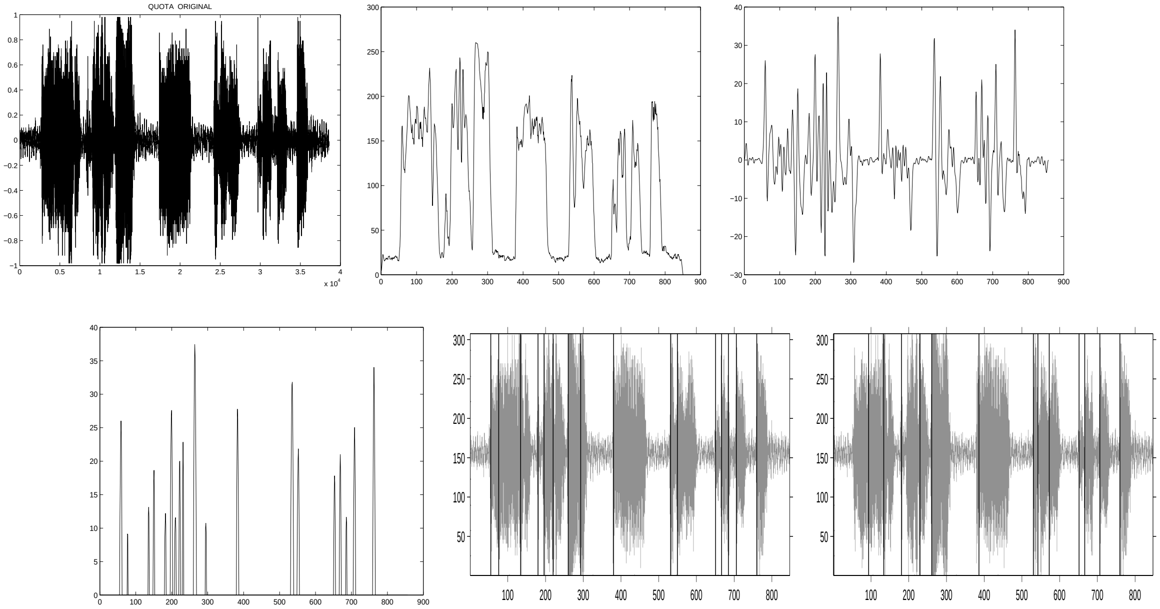

Figure 4: Snapshots in the segmentation of data 1 (speech).

5 Conclusions

Segmentation is one of the common operations done during the analysis of audio/multimedia data. Audio data segments are used in the indexing of audio data for content-based retrievals in audio databases. This paper proposed a novel scheme to segment audio data, based on the segmentation of the binary image corresponding to the plot of the audio samples. The good correspondence between the audio segments and the segments in the binary image of the audio sample plot, together with the simplicity of the binary image segmentation compared to audio segmentation, makes the proposed scheme both fast and accurate. The viability of the proposed scheme was confirmed by the experimental results.

References

[1] Cheng, J-C. and Don, H-S. 'Segmentation of Bilevel Images: A Morphological Approach', *Pattern Recognition: Architectures, Algorithms and Applications*, 1991, pp141–185. McGraw-Hill, 1993.

[2] Essa, O. 'Using Prosody in Automatic Segmentation of Speech', *Proc. 36th ACM Southeast Conf.*, pp44–49, April 1998.

[3] Ghias, A. et al. 'Query by humming', *Proc. ACM Multimedia Conf.*, 1995.

[4] Kimber, D. and Wilcox, L. 'Acoustic Segmentation for Audio Browsers', *Proc. Interface Conference*, Sydney, July 1996.

[5] Leung, H.C. and Zue, V.W. 'A Procedure for Automatic Alignment of Phonetic Transcriptions with Continuous Speech', *Proc. ICASSP-84*, pp2.7.1–2, 1984.

[6] Rabiner, L.R. and Sambur, M.R. 'An Algorithm for Determining the Endpoints of Isolated Utterances', *Bell Systems Technical Journal*, Vol.54, pp297–315, Feb. 1975.

[7] Subramanya, S.R. *et. al.* 'Transform-Based Indexing of Audio Data for Multimedia Databases', *IEEE Int'l Conference on Multimedia Systems*, Ottawa, June 1997.

[8] Van Hemert, J.P. 'Automatic Segmentation of Speech', *IEEE Trans. Signal Processing*, Vol.39, pp1008–12, April 1991.

[9] Wold, E. *et. al.* 'Content-based classification, search and retrieval of audio data', *IEEE Multimedia Magazine*, 1996.

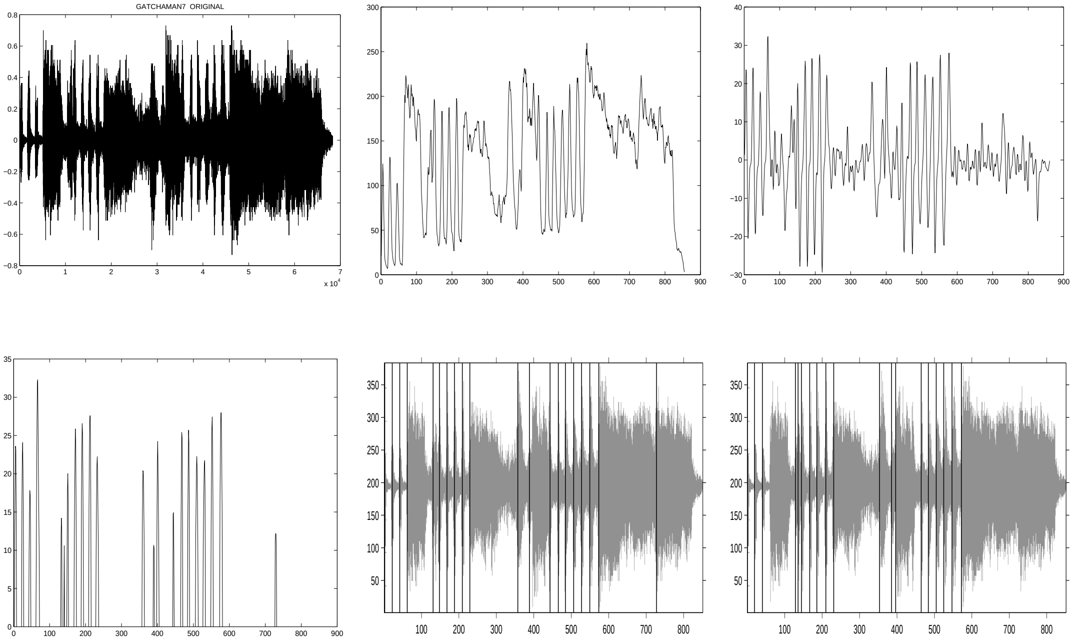

Figure 5: Snapshots in the segmentation of data 2 (music).

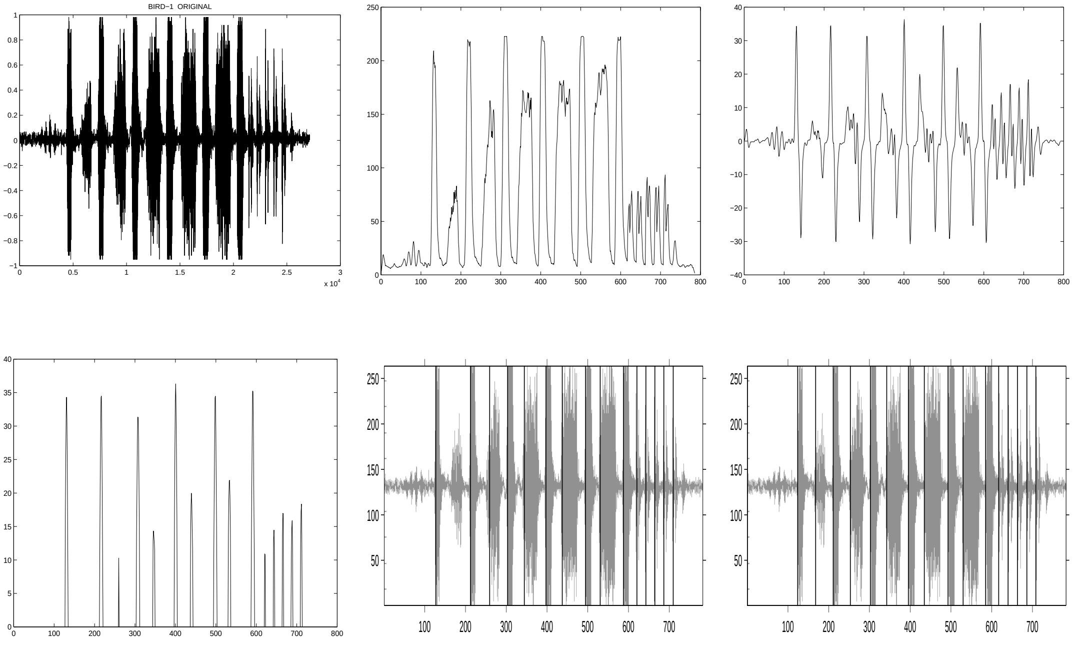

Figure 6: Snapshots in the segmentation of data 3 (bird chirps).