AUDIO SCENE SEGMENTATION USING MULTIPLE FEATURES, MODELS AND TIME SCALES

Hari Sundaram *Shih-Fu Chang*
Dept. Of Electrical Engineering, Columbia University,
New York, New York 10027.
Email: {sundaram, sfchang}@ctr.columbia.edu

ABSTRACT

In this paper we present an algorithm for audio scene segmentation. An audio scene is a semantically consistent sound segment that is characterized by a few dominant sources of sound. A scene change occurs when a majority of the sources present in the data change. Our segmentation framework has three parts: (a) A definition of an audio scene (b) multiple feature models that characterize the dominant sources and (c) a simple, causal listener model, which mimics human audition using multiple time-scales. We define a correlation function that determines correlation with past data to determine segmentation boundaries. The algorithm was tested on a difficult data set, a 1 hour audio segment of a film, with impressive results. It achieves an audio scene change detection accuracy of 97%.

1. INTRODUCTION

This paper deals with the problem of segmenting audio into semantically consistent chunks of data. This is an important problem for several reasons: (a) Segmenting the audio data into coherent chunks is the first step towards generating semantics of the sound. (b) Many algorithms for summarizing video [3,10] rely exclusively on video data. Organizing video data into semantically coherent units is made difficult due to the presence of multiple camera angles and abrupt scene changes. We believe that the associated audio track possess long term temporal coherence that will be of immense help in generating meaningful video summaries.

There has been prior work done dealing with the problem of sound segmentation [7,8,9,11]. Broadly, in each of these papers the authors use a few features (e.g. energy, cepstra) to classify the audio data into several predefined classes such as speech, music environmental sounds etc. However, we believe that the following issues still need to be addressed:

1. **Fixed time scale:** In prior work [9,11] the authors determine the features of interest at fixed time-scale. The features are extracted at the granularity of a *frame.* The frame length differs with implementation (e.g 100ms in [9,11], 2.4 sec. in [7]).
2. **Short term memory:** The existing algorithms examine the difference between the existing and the *next* frame. This is really the idea of very shot-term memory span.
3. **Restricted Consistency:** In [9,11] a change is detected when there is a significant change in the values of any of the features in the current frame. This is simply the idea that : A section of audio is considered to be a segment if all the feature *values* are held constant. We refine this idea as: A section of audio is considered a segment if it is consistent with respect to a certain *property*.

Our paper seeks to address these issues. We define an audio scene in terms of a few dominant sources of sound. Then we develop a causal algorithm by defining a simple model of a listener. A listener has two variable parameters: (a) An analysis window that examines the most recent data (the attention span) and (b) the total amount of data stored (memory). We then extract a number of features for the data in the current attention span. For each feature, we propose to represent each feature in terms of three models: extract three attributes: periodicity, randomness and envelope characteristics. Then we compute correlations with past data and hence determine the optimal threshold for scene segmentation. In this paper, we focus on envelope behavior for audio segmentation with good results.

The rest of the paper is structured as follows: In the next section we discuss the scene and the listener models. In section 3 we discuss features and the models that we develop to represent them. In section 4 we present our scene change detection algorithm. In section 5 we discuss the our experiments and finally in section 6, we present our conclusions.

2. WHAT IS A SCENE?

In this section we define characteristics of a sound scene and also the models that we assume in order to segment the data. We model the scene as a collection of sound sources. We further assume that the scene is *dominated* by a few of these sources. These dominant sources are assumed to possess stationary properties that can be characterized using a few features. For example, if we are walking away from a tolling of a bell, the envelope of the energy of the sound of the bell will decay quadratically. A scene change is said to occur when the majority of the dominant sources in the sound change.

2.1 The Listener model

In order to segment sound into scenes, we need to use a simple causal model of a listener. The causality assumption stems from a desire to mimic the process of human audition [1]. In our model of a listener, two parameters are of interest:

(a) Memory: This is the net amount of information (T_m) used by the listener to come to a decision about a scene change.

(b) Attention span: The attention span (T_{as}) is the most recent data in with the memory of the listener. This is the data that

is used by the listener to compare against the contents of the memory in order to decide if a scene change has occurred. In contrast to our model, the leaky memory buffer algorithm described in [3] does not have a notion of an attention-span.

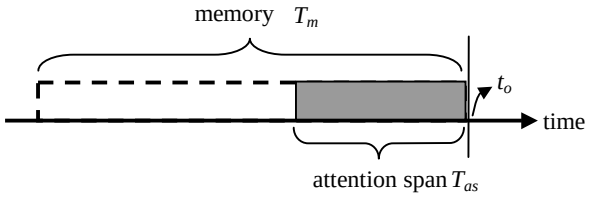

Figure 1: The attention span (T_{as}) is the most recent data in the buffer. The memory (T_m) is the size of the entire buffer. Clearly, $T_m \geq T_{as}$.

This idea is illustrated in figure 1. t_o is the current instant.

In summary, the memory is a first-in-first-out buffer holding data of duration T_m sec.; the attention-span contains the most recent T_{as} sec. of data. In our framework, the data in memory is broken up into overlapping chunks (each chunk is T_{as} long). Each chunk of data is further broken down to *frames* (100 ms. duration) and a value is determined for each feature at each frame instant.

3. FEATURES AND MODELS

In this section we shall describe the features that were used in the segmentation algorithm. We also describe the models used for each feature.

3.1 Features

We use ten different features in our algorithm Features are extracted per frame for the duration of the analysis window.

1. Cepstral Flux: The norm of the difference between cepstra [6] of successive frames: $\|C_i - C_{i+1}\|$.
2. Multi-channel cochlear decomposition: A 10 dimensional vector is derived from the output of a Patterson model [4] of the cochlea.
3. Cepstral vectors: Liftered, 18 dimensional cepstra from each frame for the duration of the analysis window.

We also use [7,8,9,11] (a) low energy fraction (b) zero crossing rate (c) spectral flux (d) energy (e) spectral roll off point. In addition, we also use the variance of the zero crossing rate and the variance of the energy as additional features.

3.2 Models

Given a particular feature f and a finite time-sequence of values, we wish to determine three attributes: (a) periodicity (b) envelope behavior (c) randomness For the sake of definiteness, let us assume that the feature is the zero-crossing rate.

Periodicity: A simple method to determine the periodic components is to use the FFT. A direct spectral analysis will reveal multiple frequencies simultaneously. However, the FFT generates a lot of spurious maxima. We can eliminate the spurious maxima by using a simple threshold on the ratio of the spectral peak to the median spectral energy.

Envelope: We wish to determine *gross* properties of the envelope of the feature. We force fit the envelope into signals of the following types: Constant. linear, quadratic, exponential, hyperbolic and sum of exponentials. Each model is force-fit into being monotonic (increasing/decreasing). This is not done for the sum of exponentials.

Two additional types are also used: (a) A "cup" quadratic (b) A "hat" quadratic. All the quadratic fits are obtained using a constrained least square minimization. All the other envelope fits are obtained using a robust curve fitting procedure using the Tukey bi-square influence function [2]. We pick the fit that minimizes the least median error. Our simple envelope models have the advantage that they allow us to assign semantic labels (increasing/decreasing/monotonic/linear etc.) to the envelope.

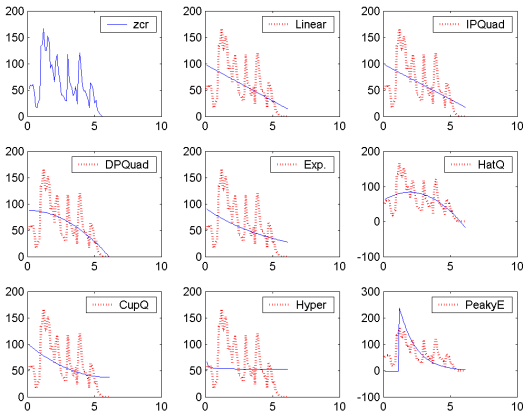

Figure 2: The different envelope fits for the zero crossing rate feature for the duration of one window.

Randomness: Given a time series we wish to test the hypothesis that this sequence was generated by a Gaussian noise source $N(\mu,\sigma)$. Using the mean and standard deviation of the data we can use the chi-square test [2] to decide if the data corresponds to the hypothesized distribution. We reject the hypothesis at the 1% confidence level.

It is immediately apparent that this kind of model analysis is easily extended to all the scalar variables. However, the vector variables (cepstra and the cochlear output) and the aggregate variables (variance of the zero-crossing rate and the spectral roll off point) are retained in the raw form.

4. DETECTING A SCENE CHANGE

Let us examine the case where a scene change occurs just to the left of the listeners attention span (figure 3). First, for each feature, we do the following:

1. Place an analysis window of length T_{as} (the attention-span length) at t_o and compute a sequence of feature values for each frame (100 ms duration) in the window.
2. Determine the optimal envelope fit for these feature values.

3. Shift the analysis window back by Δt and repeat steps 1. and 2. till we have covered all data in the memory.

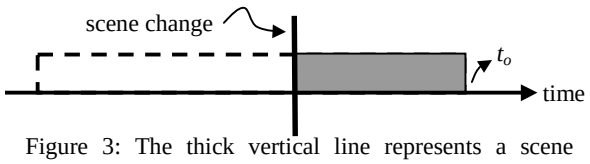

Figure 3: The thick vertical line represents a scene change. Here the scene change occurs just left of the listeners attention span.

We now have a sequence of envelope fits for each feature. In order to detect the scene change, we need to define a local correlation function. This correlation function determines the correlation between the data in the attention span and the past data in memory. The correlation function C_f for each feature *f* is then defined as follows:

$$C_f(m\,\Delta t) = 1 - d(f(t_o, t_o - t_{as}), f(t_o + m\,\Delta t, t_o + m\,\Delta t - t_{as}))$$

where, $f(t_1, t_2)$ represents the envelope fit for feature *f* for the duration $[t_1,t_2]$. Clearly, $m \in [0..\text{-}N]$, where $N \equiv (T_m - T_{as})/\Delta t$. Δt is the duration by which the analysis window is shifted. and *d* is the Euclidean metric on the envelopes. We do acknowledge that more sophisticated predictor models could be used to compute the distance between the envelope models rather than the simple Euclidean metric that we use at the moment.

For the vector and the aggregate data, we do not compute the distance between the windows using envelope fits but use a L^2 metric on the raw data. In our experiments we use $\Delta t = 1$ sec.

We expect that when the scene change is just to the left of the attention span, the correlation function will decay rapidly as a function of *t*. However, in the absence of any scene change, the correlation function ought to be flat (this depends on the metric on the envelopes). This must be so since within a scene we expect the dominant source properties to be stationary. Hence we model the correlation decay as a decaying exponential: $C_i(t) = \exp(b_i t)$, $t < 0$ where C_i is the correlation model for feature *i*, and b_i is the exponential decay parameter

The scene decision function $D(t_o)$ at any instant t_o is defined as follows:

$$D(t_o) = \begin{cases} \text{scene change}: & \sum_i b_i > T \\ \text{no change} & : \text{otherwise} \end{cases}$$

where b_i is the exponential decay rate from the correlation function and where T is the optimal threshold. Note, we do not at present, incorporate the periodicity and the randomness estimates in the decision.

5. EXPERIMENTS

In this section we present experimental results using our listener model and the feature set. We shall first describe the hand-labeled data and then discuss the experimental results. The experiments were carried on the first one hour of a classic science fiction film: *Blade Runner*. The data is complex with non-trivial scene changes. For example, a typical scene change sequence is: ambient music → street sounds → conversation → sounds in a bar.

5.1 Understanding the hand-labeled data

The first hour of the film was hand labeled into coherent, semantically consistent scenes in two ways: by looking at the video along with the sound (video scenes) and by listening to the audio alone (audio scene). Note that a video scene is a complex semantic unit, comprising many shots. For example, in a scene involving two characters who are engaged in a conversation, we will have the camera switching from one person to the other.

The table below shows the strong agreement between two kinds of labeled data. A video and an audio scene are said to agree if they can be cross validated. There seem to be 10 "extra" sound scenes. These scenes are actually correct; they reflect a change of mood (or theme) within the same video scene.

Type	No.
Video Scenes	28
Audio Scenes	33
Scene Agreement	23

Table 1 The audio scene breaks were labeled *without* watching the video while the video scene breaks were obtained by watching the film with the audio. Note the strong scene agreement.

A comparison of the audio scene labels with video scene labels (on scenes that agree) revels a consistent location ambiguity. This ambiguity is positive (μ = +2.87 sec., σ=5.26 sec.) . There were two reasons for this:

- During a video scene transition, the sound from the previous video scene continues over into the next video scene for a few seconds.
- There is genuine ambiguity when listening to the audio data; the listener needs to wait for a few seconds before concluding that there has been a scene change.

This implies that a long latency period (in the order of seconds rather than milliseconds as in [7,8,9,11]) is to be expected while identifying scenes when only using audio data.

5.2 The Scene change detector results

The scene change detector was evaluated against the hand-labeled *audio data* rather than using the hand-labeled video data. The reason for this is as follows: the positive location ambiguity observed in identifying the scene change is modeled in the attention-span parameter in our listener model. Also, the extra scenes in the audio are important: they often convey a significant change in the mood (or the theme) of the film.

The different attention spans will clearly cause the detectors to have different time resolutions. We label claim a scene to be correctly detected if it is identified with a location error proportional to the attention span.

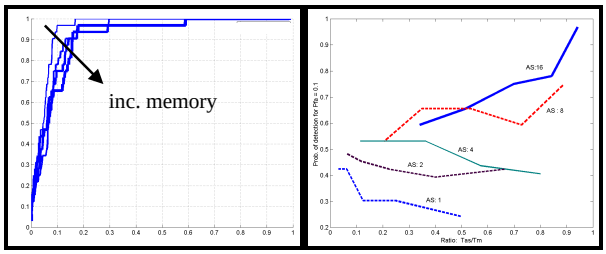

Figure 4: (a) The probability of detection against prob. of false alarms for fixed attention span but different memory lengths {T_{as}: 16 sec. T_m: 17, 19, 23, 31, 47 sec.}. (b) The Prob. of detection against different T_{as}/T_m ratios, for a fixed probability of false alarm (P_{fa}=0.1). The plot is grouped using constant attention span lengths. Starting from the top, T_{as}: {16,8,4,2,1 sec.}.

The results are shown in figure 4. Figure 4a shows the prob. of detection against prob. of false alarms for fixed attention span but different memory lengths {T_{as}: 16 sec. T_m: 17, 19, 23, 31, 47 sec.}. Figure 4b shows a plot of prob. detection against different T_{as}/T_m ratios with a fixed probability of false alarm (0.1). The highest scene detection rate of 97% was achieved with T_{as} = 16 sec. and T_m=17 sec. The results clearly indicate three trends:

- For longer attention spans {T_{as}:8,16 sec.} the probability of detection increases with increase in the T_{as}/T_m ratio (figure 4b).
- Shorter attention spans {T_{as}:1, 2, 4 sec.} perform poorly, with performance only improving marginally with increase in the memory T_m (i.e. the buffer size) (figure 4b).
- For a fixed attention span (T_{as} = 16 sec.) the detector performance decreases with increasing the memory (T_m) (figure 4a).

The observation of *increase* in probability of detection with *increase* in the T_{as}/T_m ratio seems surprising since this indicates that a longer memory is a deterrent to scene change detection. Note, however, that the attention-span parameter is long (16 sec. in the best result). There are two possible explanations for this phenomena: (a) The correlation function depends on Δt, the duration by which we shift back our analysis window when computing the correlation. In our experiment Δt= 1 sec. which is perhaps too large. (b) It is also possible that the exponential model for the correlation is too simple to adequately capture the observed correlation behavior.

6. CONCLUSIONS

We have described a framework for segmentation of audio scenes. We define an audio scene as a collection of sound sources. This is used in conjunction with a listener model that has two parameters: (a) attention-span and (b) memory. The detection algorithm works as follows: We extract different features for each chunk of data. Then we determine the optimal envelope fits for each feature. Then, by determining the correlation amongst the envelopes, we determine segmentation boundaries. We observe that the detector performance increases in two cases: (a) with increase in the attention span (b) with an increase in the ratio of the attention span length to the memory.

The algorithm achieves a segmentation detection accuracy of 97% at a false alarm probability of 10%. Our results are preliminary and we believe that more sophisticated models (e.g. a more sophisticated memory/attention-span model.) will improve the performance of the scene change detector.

The strong agreement between the audio scene and the video scene boundaries is an important observation. Since exiting techniques [3,10] to summarize video data at the semantic level only use image data, we believe that the use of the audio scene change detection algorithm offers an excellent avenue of improving existing video summarization algorithms.

7. REFERENCES

[1] A. S. Bregman *Auditory Scene Analysis: The Perceptual Organization of Sound*, MIT Press, 1990.

[2] F. R. Hampel et. al. *Robust Statistics: The Approach Based on Influence Functions*, John Wiley and Sons, 1986.

[3] J.R. Kender B.L. Yeo, *Video Scene Segmentation Via Continuous Video Coherence*, CVPR '98, Santa Barbara CA, Jun. 1998.

[4] R. Patterson et. al. *Complex Sounds and Auditory Images*, in Auditory Physiology and Perception eds. Y Cazals et. al. pp. 429-46, Oxford, 1992.

[5] S. Pfeiffer et. al. *Automatic Audio Content Analysis*, Proc. ACM Multimedia '96, pp. 21-30. Boston, MA, Nov. 1996,

[6] L. R. Rabiner B.H. Huang *Fundamentals of Speech Recognition*, Prentice-Hall 1993.

[7] John Saunders *Real Time Discrimination of Broadcast Speech/Music*, Proc. ICASSP '96, pp. 993-6, Atlanata GA May 1996.

[8] Eric Scheirer Malcom Slaney *Construction and Evaluation of a Robust Multifeature Speech/Music Discriminator* Proc. ICASSP '97, Munich, Germany Apr. 1997.

[9] S. Subramaniam et. al. *Towards Robust Features for Classifying Audio in the CueVideo System*, Proc. ACM Multimedia '99, pp. 393-400, Orlando FL, Nov. 1999.

[10] S. Uchihashi et. al. *Video Manga: Generating Semantically Meaningful Video Summaries* Proc. ACM Multimedia '99, pp. 383-92, Orlando FL, Nov. 1999.

[11] T. Zhang C.C Jay Kuo *Heuristic Approach for Generic Audio Segmentation and Annotation*, Proc. ACM Multimedia '99, pp. 67-76, Orlando FL, Nov. 1999.