CONTENT-BASED INDEXING AND RETRIEVAL-BY-EXAMPLE IN AUDIO

*Zhu Liu**

Electrical Engineering Department
Polytechnic University
Brooklyn, NY 11201
zhul@vision.poly.edu

Qian Huang

AT&T Labs - Research
100 Schulz Drive
Red Bank, NJ 07701
huang@research.att.com

ABSTRACT

This paper presents a system for audio content-based indexing and retrieval-by-example. An unsupervised method to automatically index the audio content is proposed. To realize a retrieval system where the on-line query can be performed efficiently, a new parametric distance metric is used to allow faster identification of the audio segments that are similar to the given query example. Preliminary experimental results show that the proposed new approach can serve as an effective means in content based indexing and retrieval in audio streams.

1. INTRODUCTION

With the rapid development of multimedia and network technology, more and more digital media is generated. With such huge amount of data, efficient means to index the content for future retrieval have to be made available. In addition, various ways for users to search and to browse the data of interests are also necessary. Efforts have been made in the literature along both directions, some in single media domain [1, 2, 3, 4] and some in multimedia domain [5, 6, 7]. This work addresses the issues in (1) automatic index generation based on acoustic content in an unsupervised way and (2) efficient search and retrieval based on given audio query examples.

Audio content analysis for information indexing and retrieval is a relatively new field that has attracted more attention in recent years. Saunders [2] classified radio broadcast into two classes of event: speech and music based on the statistical analysis of zero crossing rate (ZCR) and audio energy. In [4], Wold et al. proposed to classify the audio into 10 different audio content: animal, bells, crowds, laughter, machine, instrument, male speech, female speech, telephone, and water. The instrument is further classified into 7 subclasses. Although such supervised classification can be useful when there are a fixed number of known categories, it is not adequate to index general audio content.

For unconstrained browsing and query, it is often impossible to plan ahead in terms of what can or can not be retrieved. For example, a user may want to find the audio clips that sounds similarly to an audio sample in hand. In this case, to retrieve the desired clips, the audio data in the database has to be segmented (but not necessarily labeled), each segment has to be matched against the sound of interest, then the clips that are similar to the given example can then be returned to the user. Here, the intention is to find similar clips but not to understand what the clips are. Therefore, another line of the research effort is to segment an audio stream into homogeneous audio events in an unsupervised manner. Such segmentation provides a set of primitives so that higher level of grouping and clustering can be further performed. Another advantage is that this allows a free form browsing and query-by-example. The work described in this paper is our effort along this line. The contribution of this work is two folds. One is that we propose a way to succinctly represent homogeneous audio segments and the other is to develop a useful metric so that the similarity among audio segments can be measured effectively from their succinct representations. They are two important aspects because they together directly facilitate both indexing as well as retrieval needs.

The goal of our work is to provide users the means of organizing the audio stream, to derive some useful structure of the data, and then to allow users to quickly browse or query about the content they need. Our strategy is that we first of all obtain a set of audio events via unsupervised segmentation. Then each homogeneous segment is represented by a Gaussian Mixture Model (GMM). To meet the query needs, we further propose a parametric distance metric so that the acoustic similarity between different audio events can be measured. To further structure the data based on content, an unsupervised clustering method is applied to group the segments that possess similar acoustic properties into the same cluster. A query system is then built to (1) present the content structure, (2) provide an search/browsing interface, and (3) enable query by audio example.

This paper is organized as follows. In Section 2 we describe the audio features and segmentation algorithm. In section 3 we propose a new metric for difference measure of GMM and present the clustering method based on the new metric. Query mechanism is illustrated in section 4. In section 5, we show some of the preliminary results of the content based audio query system. Finally section 6 concludes the paper.

2. AUDIO SEGMENTATION

Segmentation is the first step to explore the content structure of an audio stream. Parallel to the scene cut in video domain, the objective is to identify the boundaries of changes

*The original work was done when the author worked at AT&T Labs - Research as an intern student.

0-7803-6536-4/00/$10.00 (c) 2000 IEEE

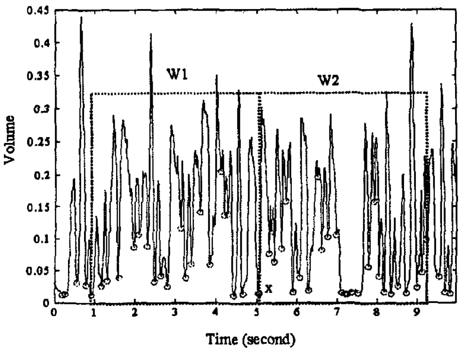


Figure 1: Illustration of Audio Segmentation.

in terms of some acoustic properties. Mel-frequency cepstral coefficient (MFCC) is widely used in speech domain and it provides a smoothed version of spectral that considers the non-linear human hearing property. The degree of smoothness depends on the order of MFCC being employeed. Two properties of MFCC, one being that the first coefficient is proportional to the audio energy and the other being that there is no correlation among different coefficients, make MFCC attractive. In our study, we employed 13 order MFCC features.

The segmentation algorithm consists of two steps: splitting and merging. During splitting, we identify possible scene change boundaries. During merging, neighboring scenes are merged if their contents are similar. In the first step, low energy frames, which are local minimum points on the volume contour, are located as boundary candidates. Figure 1 shows the volume contour of an audio file, where all low energy frames are indicated by a circle. For each boundary candidates, the difference between its neighbors (both left and right) is computed. The definition of neighbors is illustrated in figure 1, where for frame X, two dotted rectangular windows $W1$ and $W2$ are the neighbors of X and each with length of L seconds. If the difference is higher than certain threshold and it is the maximum in surrounding range, we declare that the corresponding frame is a scene boundary.

Kullback Leibler distance (KLD) [8] is adopted to measure the difference. Although the standard KLD does not possess the properties of symmetry and triangular inequality of a distance metric, it can be easily overcome by extending the original KLD to $D_{KL}(G,F) + D_{KL}(F,G)$. The integration is realized numerically, making it computational expensive. However, in some special cases, it can be simplified. For example, when $G(m_1, \sigma_1)$ and $F(m_2, \sigma_2)$ are both one dimensional Gaussians, ignoring the constant multiple, the extended KLD between G and F can be simplified as a computation directly from the model parameters

$$D_P(G,F) = \frac{\sigma_1^2}{\sigma_2^2} + \frac{\sigma_2^2}{\sigma_1^2} - 2 + \left(\frac{\sigma_1^2 + \sigma_2^2}{\sigma_1^2\sigma_2^2}\right)(m_1 - m_2)^2$$

In our initial segmentation, we adopt such extended KLD to measure the difference. Since different audio features we used are independent, the overall distance is simply the summation of KL Distance of each dimension.

The above described splitting process yields, in general, over-segmentation. A merging step is necessary to group similar neighboring segments together to form homogeneous audio events. This is done by comparing the statistical properties of adjacent segments. The KL distance based on the mean and standard deviation vectors of adjacent segments is computed. If it is lower than a threshold, the two segments are grouped and the corresponding feature vector are updated.

To precisely and, at the same time, concisely represent each homogeneous segment, Gaussian Mixture Model (GMM) is chosen to approximate the feature distribution of an audio segment. A GMM model consists of a set of weighted Gaussian's [9]. For an audio segment, a model is generated by fitting the model with the features of that segment. The derived model parameters serve as the representation of the segment.

3. AUDIO CLUSTERING

After the segmentation, each boundary point is where the transition from one homogeneous audio event to another. While this may describe the primitive structure of the audio content, more can be done to identify its higher level structure. For example, all segments that possess similar acoustic properties should be semantically grouped together. We achieve that using clustering approach. For example, for an audio stream containing a dialog, the speech of the same speaker distributed at different time instances can be grouped as one cluster identified as the speaker.

Two technical problems have to be solved in order to cluster the segments based on their representations. One is how to measure the dissimilarity between two segments based on their model parameters. The other is how to cluster based on such dissimilarity values. There is no existing metric in the literature that can effectively measure the dissimilarity between two mixture PDF's. So, we propose a new distance metric, in its closed form solution, that measures the distance between two PDF's of mixture type, directly from their parameters in 3.1. The clustering procedure is then described in section 3.2.

3.1. Parametric Distance Metric for GMM

Suppose $G(\mathbf{x})$ and $H(\mathbf{x})$ are two GMM's,

$$G(\mathbf{x}) = \sum_{i=1}^{N} \mu_i g_i(\mathbf{x}), \quad H(\mathbf{x}) = \sum_{k=1}^{K} \gamma_K h_k(\mathbf{x}),$$

where $G(\mathbf{x})$ is a mixture of N element gaussians $g_i(\mathbf{x})$, $H(\mathbf{x})$ is a mixture of K element gaussians $h_k(\mathbf{x})$, and μ_i and γ_k are corresponding weights that satisfy $\sum_{i=1}^{N} \mu_i = 1$ and $\sum_{k=1}^{K} \gamma_k = 1$. Denote the distance between any two element gaussians g_i and h_k by $d(g_i, h_k)$, the overall distance between G and H is defined as,

$$D_M(G,H) = \min_{\mathbf{w}=[w_{ik}]} \sum_{i=1}^{N} \sum_{k=1}^{K} w_{ik} d(g_i, h_k), \quad s.t. \quad (1)$$

$$w_{ik} \geq 0, \quad 1 \leq i \leq N, 1 \leq k \leq K \quad (2)$$

$$\sum_{i=1}^{N} w_{ik} = \gamma_k, 1 \leq k \leq K, \quad \sum_{k=1}^{K} w_{ik} = \mu_i, 1 \leq i \leq N \quad (3)$$

In the definition, any component g_i in one mixture PDF can interact with any other component h_k in the other mixture PDF via weighted element distance $w_{ik} d(g_i, h_k)$. The degree of interaction is inversely proportional to the element distance and proportional to the mixture weights μ_i and γ_k. The weights w_{ik} are ultimately determined through optimization with respect to the given constraints in (2, 3).

The solution is clearly posed as a linear programming (LP) problem and it is a closed form solution. Efficient algorithms exist to solve an LP problems, e.g., simplex tableau method. We have a total of $N \times K$ parameters (w_{ik}'s) and $N + K$ equality constrains, where only $N + K - 1$ of them are independent. By the optimization theory, at most $N + K - 1$ of the $N \times K$ parameters will not vanish. The above problem has solution because (1) we can easily find a feasible vector that satisfy all the constrains: $w_{ik} = \mu_i \times \gamma_k$ and (2) the upper bound for the objective function exists: $\max_{ik} d(g_i, h_k)$.

This new metric is proposed as a general framework for mixture type PDF's by constructing the overall distance from element distances. We demonstrated in [10] that if the element distance satisfies the three properties of a distance metric, the overall distance does as well. Its generality is therefore due to that the element distance measure is left unspecified. Depending on different application needs, appropriate element distance measures, which may be even non-parametric, can be plugged in and the overall distance between two PDF's can be computed using the same framework. Furthermore, this metric can also be used for different types of mixture PDF functions such as GMM (in this paper) and Gammar Mixture Model.

3.2. Agglomerative Hierarchical Clustering

Using the metric proposed above, the distance between any pair of audio segments represented by their GMMs can be efficiently computed. For an audio stream containing N segments, a dissimilarity matrix DM of size $N \times N$ can be generated for the entire audio stream, where $DM(i,j)$ is the distance between the i^{th} and j^{th} segments. Based on this matrix, a clustering operation is performed that ultimately group similar audio events into the same clusters.

The details of agglomerative hierarchical clustering algorithm can be found in [11]. Each segment is initially treated as a cluster on its own. In each iteration, two clusters with an minimum dissimilarity value are merged. This procedure continues until the minimum cluster dissimilarity exceeds a preset threshold. After the clustering, the cluster index for each segment is saved in databases for future retrieval purpose.

In general, the clustering can also be performed across different audio streams. This is especially useful in some of the scenarios such that the speech segments of certain speaker (e.g. Clinton) on different broadcasts or similar commercial on different channels can be clustered together so that users can easily retrieve such content based clusters.

4. AUDIO QUERY BY EXAMPLE

Traditionally, query is often based on text. With the fast development of multimedia applications, not only the demand has grown out of needs in text, but also the manual annotation is no longer feasible. Query based on acoustic characteristics is one alternative to text based retrieval. For example, to retrieve some audio clips based on text, one has to know exactly how this clip is labelled. But there are many cases where users know only how the content (speakers, music, or songs) sounds like but not what semantics it has been identified previously. Therefore, retrieval by audio example is an alternative to conventional text based retrieval. The user can simply provide a sample audio stream and ask to retrieve the audio segments that posess similar acoustic properties.

Technically, the sample audio is first segmented and clustered based on the procedures previously described. Normally, the sample audio is expected to be short and the entire sample audio stream is one homogeneous audio event. In this case, the sample segment is fitted by a GMM and then the model is compared with those saved in the database. When there are several audio events in the example, there are several alternatives. One is to choose the dominant segment and then make the query based on its GMM. Another possibility is to use all the segments in query without considering their temporal order. Here the query result may be similar to any of the segments in the query example. If temporal order is considered, measures that can characterize both the duration and the order simultaneously are needed. Although dynamic programming can be a good candidate for this kind of optimization, we currently do not use DP approach due to its expensive computation and our requirement to perform efficient on-line query against a large audio database. We instead adopt a simpler method.

Denote the query sequence Q by $(Q_1, Q_2, ..., Q_N)$ and an audio sequence S in database by $(S_1, S_2, ..., S_M)$, where Q has N segments and S has M segments. Without loss of generality, we assume that $M \geq N$. Q_i and S_i are GMMs, representing the corresponding segments. The distance between Q and a portion of S started at t is computed as,

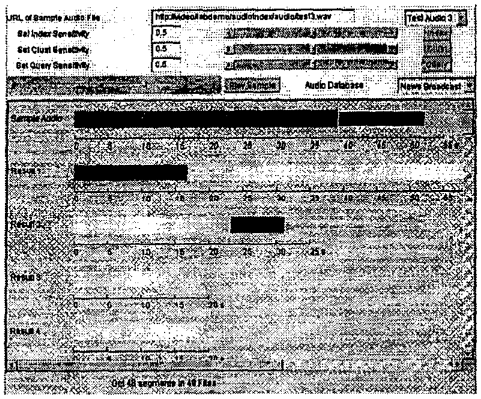
$$D_{Q,S}(t) = \sum_{i=1}^{N} D_M(Q_i, S_{t+i-1}) w(Q_i, S_{t+i-1})$$

where $w(Q_i, S_{t+i-1})$ is a weight related to the duration of Q_i and S_{t+i-1}. If the duration is not considered, w can be assigned to $1/N$. Otherwise, it should reflect the duration effect. When the duration of segment Q_i is T_i^Q and that of S_j is T_j^S, w can be possibly defined by,

$$w(Q_i, S_j) = \frac{T_i^Q}{\sum_{k=1}^{N} T_k^Q} \times (2 - \frac{min(T_i^Q, T_j^S)}{max(T_i^Q, T_j^S)})$$

5. EXPERIMENTAL RESULTS

To evaluate the proposed approach in audio indexing and retrieval, experiments are performed and shown via a content-based audio query system. The system interface is shown in Figure 2 with a web-based client-and-server architecture. Users can load their audio sample by specifying the URL of the audio file. Based on given query sample, the system performs three tasks: segmenting the audio sample, clustering the segments, and querying the database. The indexed audio segments are presented as blocks of different colors in Figure 2. When clustering is performed, segments within the same cluster will be painted with the same color. The query results are painted using a designated yellow color where different hit segments will have different brightness, depending on the similarity scores (the more similar, the brighter the color is). When the query is performed across different audio streams, the order of the hit streams will be returned according to the degree of similarity measured as the minimum distance between the sample segment and any hit segment of an audio stream. As can be seen from



URL of Sample Audio File: http://videolabdemo/audioindex/audiotest3.wav

Test Audio 3

Set Index Sensitivity: 0.5

Set Clust Sensitivity: 0.5

Set Query Sensitivity: 0.6

Audio Database: News Broadcast

Got 48 segments in 48 Files

Figure 2: The interface of audio query system.

the top of Figure 2, the interface offers users the means of adjusting the sensitivity in different tasks based on their application needs.

All the audio data in our database is in the format of mono raw data with 16 KHz sampling rate and 16 bit resolution per sample. In our preliminary experiment, a total of 239 audio sequences (from about 7 hour TV programs) were collected in the database. It includes 50 Tom Brokaw's (an anchor person) speech, 150 pieces of speech of reporters and other anchors, 34 pieces of commercials, and 5 pieces of music. We segment all sequences and a 4 mixture GMM is used to model each segment. All the meta data such as the segment boundaries and the GMM parameters is stored in the database.

To measure the segmentation performance we test on two TV news broadcast sequences, each about 30 minutes. The ground truth of the segment boundaries are generated manually for evaluation purpose. In the ground truth, test sequence one has 157 segments and test sequence two has 159 segments. Four measures are designed to evaluate the performance based on identified event boundaries: 1) Hit Rate (HR) - the ratio of the number of correctly detected boundary points (within two seconds deviation) to the true number of boundary points, 2) False Alarm Rate (FR) - the ratio of the number of falsely detected boundary points to the number of detected boundary points, 3) Mean Difference (MD) - the average difference (in ms) between correctly detected boundaries and the true boundaries, and 4) Standard Deviation of boundary difference (SD). Table 1 gives the results from the two test sequences. The high FR is due to the variation in the background sound.

To test the query performance, we use clean Tom Brokaw's speech as query example. In the database, although there are 50 segments of Brokaw's speech, they have different background sounds, sometimes theme music and sometimes environmental noises. Besides algorithm influence, there are two more factors that affect the performance in a query: the type of background sound and the sensitivity set for the query. In our current experiment, we set the sensitivity to 0.5 and we used clean Brokaw's speech as the query example. Given these experimental conditions, we obtained 30 returns, out of which 24 are Brokaw's speech. It is, therefore, desirable to allow users to adjust the sensitivity so that the hit rate of the query can meet the need of their applications.

Table 1: Audio Segmentation Results.

Sequence	HR (%)	FR (%)	MD (ms)	SD (ms)
1	92.26	20.56	-336	736
2	93.63	22.22	-235	658

6. CONCLUDING REMARKS

This paper proposed a new approach for automatic content based audio indexing. The audio signal is segmented into homogeneous events whose features are characterized using GMM models. A new metric for measuring the similarity between two PDF's of mixture type is described and applied to GMMs. Both audio clustering and on-line query are based on this new metric. An audio retrieval system is presented as a demonstration of the effectiveness of the proposed techniques. Based on our preliminary experimental results, it can be seen that our proposed approach is promising for audio content indexing, description, search, and retrieval.

7. REFERENCES

[1] H. Zhang, A. Kankanhalli, S. Smoliar, "Automatic Partitioning of Full-motion Video," *A Guided Tour of Multimedia Systems and Applications*, IEEE Computer Society Press, 1995.

[2] J. Saunders, "Real-Time Discrimination of Broadcast Speech/Music," *ICASSP'1996*, Vol. 2, pp. 993-996, Atlanta, May 1996.

[3] T. Zhang and C.-C. J. Kuo, "Hierarchical Classification of Audio Data For Archiving and Retrieval," *ICASSP'1999*, Vol. 6, pp. 3001-3004, Phoenix, March 1999.

[4] E. Wold, T. Blum, D. Keislar, and J. Wheaton, "Content-Based Classification, Search, and Retrieval of Audio," *IEEE Multimedia*, Vol. 3, No. 3, pp. 27-36, Fall 1996.

[5] Y.L.Chang and W. Zeng and I. Kamel and R. Alonso, "Integrated Image and Speech Analysis for Content-Based Video Indexing," *Proc. of Multimedia*, pp. 306-313, September 1996.

[6] Q. Huang, Z. Liu, A. Rosenberg, D. Gibbon, B. Shahraray, "Automated Generation of News Content Hierarchy by Integrating Audio, Video, and Text Information," *ICASSP'1999*, Vol. 6, pp. 3025-3028, Phoenix, March 1999.

[7] S. Eickeler and S. Mueller, "Content Based Video Indexing of TV Broadcast News Using Hidden Markov Models," *ICASSP'1999*, Vol. 6, pp. 2997-3000, Phoenix, March 1999.

[8] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley & Sons, Inc. 1991.

[9] D. A. Reynolds and R. C. Rose, "Robust Text-Independent Speaker Identification Using Gausian Mixture Speaker Models," *IEEE Trans. on Speech and Audio Processing*, Vol. 3, No. 1, pp. 72-83, 1995.

[10] Z. Liu and Q. Huang, "A New Distance Measure For Probability Distribution Function of Mixture Types," to appear in *ICASSP'2000*, Istanbul, Turkey, June, 2000.

[11] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*, Prentice-Hall, 1988.

0-7803-6536-4/00/$10.00 (c) 2000 IEEE