Audio-Based Radio and TV Broadcast Monitoring

Bruno Oliveira
IBOPE Pesq. de Mídia Ltda.
Al. Santos, 2101
São Paulo, Brazil
brunotc@gmail.com

Alexandre Crivellaro
IBOPE Pesq. de Mídia Ltda.
Al. Santos, 2101
São Paulo, Brazil
acrivellaro@ibope.com.br

Roberto M. César Jr.
IME - USP
Rua do Matão, 1010
São Paulo, SP
roberto.cesar@vision.ime.usp.br

ABSTRACT

This paper describes a scalable real-time audio fingerprinting system developed by IBOPE Midia for radio and TV broadcast monitoring. A special temporal feature extraction strategy based on the Short-Time Fourier Transform has been designed. When given an input stream to analyse, the system matches it against the database and automatically recognizes instances of the previously registered samples within the input stream. The algorithm exploits the temporal evolution of the signal frequency spectrum in order to identify patterns and produce the final classification. The database is clusterized in order to provide an efficient and scalable search strategy. The system has been assessed using a database containing 393 distinct commercials. A 41-hour audio stream from three different TV channels has been analysed in less than 3 hours, attaining a 95.4% recognition rate.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous;
I.5 [**Pattern Recognition**]

General Terms

Algorithms, Performance, Theory.

Keywords

Multimedia applications, Short-time Fourier transform, pattern recognition, audio recognition, clustering

1. INTRODUCTION

Audio-based recognition of multimedia content is an important problem in many practical situations [**5**]. This paper focuses on the recognition of commercials, i.e. advertisement broadcasts on radio and television. This is an important problem faced by companies that perform broadcast monitoring to recognize commercials and other features of interest in radio and TV. Apart from audio and video signal processing, this kind of application involves manipulation of possibly huge databases and require real-time responses. These requirements call for the development of accurate and efficient signal processing and pattern recognition techniques.

Different approaches for audio-based multimedia recognition have been recently proposed [**3,4**]. One of the main applications for such technologies is music recognition, and some commercial services are now [**1**].

Nevertheless, such approaches cannot be directly applied to broadcast monitoring, as they fail to address two major problems faced by this kind of application: (1) determining the beginning and end points of each commercial (i.e. *segmentation*). and (2) adequate behavior when confronted with variations of the same commercial with subtle differences. The system described in this paper overcomes these limitations by combining techniques from various sources and adapting them to our purposes, thus providing reliable and efficient audio recognition.

The input audio stream is divided into blocks which undergo the Short-Time Fourier Transform (**STFT**) [**2**] in order to generate fingerprints, which are then submitted to a classifier function which compares them against the fingerprints stored in the database and applies a label to each portion of the input audio. In a later phase, the labels are scanned for consistent patterns which indicate the presence of an occurrence of a known audio sample.

2. SYSTEM OVERVIEW

The system takes an audio stream as input and classifies each audio block of the stream based on Fourier features [**2**]. The audio stream is taken as a single-channel real-valued signal *f(t)*. The input stream is divided in blocks, called *frames*. The parameter *Δt* defines the distance between the beginning of one frame and the beginning of the next. At the instant corresponding

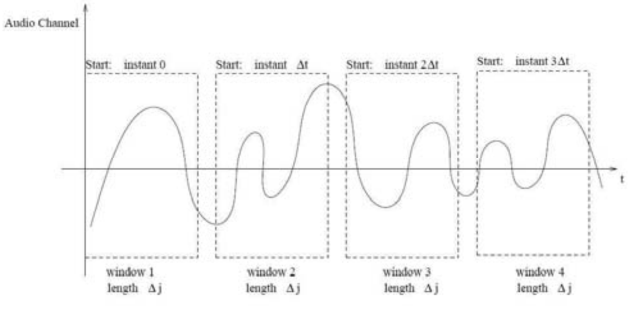

Figure 1: Basic diagram of the recognition algorithm: the input is divided into frames that are labeled according to the database.

Conference'05, Month 1–2, 2004, City, State, Country.
Copyright 2005 ACM X-XXXXX-XXX-X/XX/XXXX…$5.00.

to the start of each frame, a window *a(t)* of size *Δt* is extracted. This audio frame *a(t)* is the basic information unit in the process, being considered a sample to be classified. As the system analyses the input audio stream, it assigns labels to each frame *a(t)*. Figure 1 shows a diagram of this process. The sequence of labels is then analysed to produce the final segmentation and recognition of each commercial in the input stream.

3. AUDIO FEATURES

The algorithm exploits the temporal evolution of the frequency spectrum of the audio stream in order to extract its features and thus classifies each frame. The STFT is applied to each audio window (frame) *a(t)* at N_{sj} positions a_k, $k=1...N_{sj}$. Before calculating the transform, the audio within the window is scaled by a Gaussian curve in order to minimize the distortions of the output spectrum normally caused by the division of the input audio into fixed-length windows. If the audio within the window is given as a continuous function *w(t)*, where $-1.0 \leq t \leq 1.0$, the transform changes that input into the function *w'(t)* given as $w'(t) = exp(-t^2 / \sigma)$, where σ is a configurable parameter.

The STFT is divided into N_C spectral bands and the energy within each band, given as $abs(A_k(f))$, is used to form the feature vector that represents *a(t)*. It is worth noting that $abs(A_k(f))$ are invariant to time-shifts [**2**], which is important because it is impossible to know, in advance, where each occurrence of a commercial will appear in the input audio.

Additionally, the resulting feature vector undergoes a series of operations which are targeted at normalizing its values in order to improve matching with feature vectors extracted from other samples. If the feature vector is given as $v = v_1, v_2, \ldots, v_{N_C}$, let $w = ln(v_1), ln(v_2), \ldots, ln(v_{N_C})$. Let us denote the average of w as $\hat{w}$. and its standard deviation by σ_w. Then the transformed feature vector will be $u = u_1, u_2, \ldots, u_{N_C}$, where, for each i, $u_i = (w_i - \hat{w}) / \sigma_w$.

In order to save memory space, the coefficients are quantized and stored as integers, that is, they are stored as $M_k(f) = round(q \cdot abs(A_k(f)))$, where q is a configurable quantization factor. Therefore, each audio block is represented by a feature array defined as:

$$\Gamma_{N_{sj} \times N_C} = \begin{bmatrix} M_1(1) & \cdots & M_1(N_C) \\ M_2(1) & \cdots & M_2(N_C) \\ \cdots & \cdots & \cdots \\ M_{N_{sj}}(1) & \cdots & M_{N_{sj}}(N_C) \end{bmatrix}$$

4. FINGERPRINT DISSIMILARITY MEASURE

The classification of each audio block is carried out by comparing the STFT fingerprint of each block *a(t)* to those stored in the database. A dissimilarity measure is hence defined. Let Γ_e be the fingerprint of an unknown input block and Γ_{bd} the fingerprint of some block stored in the database. The individual elements of these fingerprints are denoted as $\Gamma_{e(k,l)}$ and $\Gamma_{bd(k,l)}$, respectively. The dissimilarity between Γ_e and Γ_{bd} is given by:

$$d(\Gamma_e, \Gamma_{bd}) = \frac{1}{N_{sj}} \sum_k \sum_l (\Gamma_e(k,l) - \Gamma_{bd}(k,l))$$

5. DATABASE FORMATION

The above equation is used by the classifier to label each input block. A database containing all commercials of interest must be formed, i.e. the fingerprint representation Γ_{bd} of all frames of the commercials has to be obtained. A fingerprint generation analogous to that explained above is used. Whenever possible, it is important to set $\Delta t \leq \Delta j$. The smaller Δt is, the higher the correct recognition rate produced by the classifier will be, since more shifted blocks of each commercial will be stored in the database. This is important because there is no guarantee that the commercials to be recognized will be aligned in the same way as the sample used when storing them in the database. Nevertheless, decreasing Δt implies increasing the number of block fingerprints stored in the database, which will increase its size and decrease its performance.

6. BLOCK RECOGNITION

Each audio block is classified using the *k-nearest neighbor* classifier (KNN) [**2**]. The similarity measure described previously is applied by the classifier, thus labeling each block in accordance to the *k* previously stored fingerprints that most closely match it.

7. CLUSTERING

The *c-means* clustering algorithm [**2**] is applied to the stored fingerprints, thus partitioning the database into *c* clusters. A cluster prototype is calculated as the mean fingerprint of each cluster. The prototypes are used to guide the classification procedure, allowing the system to compare each block only to the fingerprints belonging to the clusters whose prototypes it most closely compares to, rather than compare it against the entire database, thus allowing for real-time scalable performance.

Recent benchmarking results show that a database containing 108,333 block fingerprints (roughly corresponding to about 400 30-second commercials) can be reclustered in under 15 minutes. The performance gained by reclustering is considerable. A 41-hour audio sample required nearly 20 hours to process without a clustering strategy; by employing the clustering mechanism, however, the processing time has reduced to only 3 hours with no observable loss of precision.

8. AUDIO SEGMENTATION

The last step performed by the system is the recognition and segmentation of each commercial, i.e. the identification of the beginning and end of each commercial. The labels applied by the classifier carry information about which commercial each block is believed to belong to, and what portion of the commercial it is believed to be. Once a sequence of consistent labels is encountered, the algorithm uses the known length of the commercial to generate a hypothesis for its beginning and end points.

The hypotheses are then tested by applying an adapted LCS (Longest Common Subsequence) algorithm, which yields a *score* that measures how closely the sequence of labels matches that of the original commercial. If that score rises above a configurable threshold, the commercial is deemed to have been recognized and is reported as such. If not, the hypothesis is discarded.

9. EXPERIMENTAL RESULTS

Recent tests have been performed on the system and have provided good results, both in terms of performance and reliability. A raw audio sample corresponding to 41 hours of broadcast was extracted from three different Brazilian TV channels and was submitted to the system for recognition on a Pentium 750Mhz machine, and was fully analysed in less than 3 hours, yielding a 95.4% correct recognition rate, and producing less than 1% of false-positives (recognition mistakes). While there is still room for improvement, the performance and reliability attained so far are already considered quite satisfactory for the intended application.

10. CONCLUDING REMARKS

This paper described a scalable real-time audio-based broadcast segmentation and recognition system that has been developed by IBOPE Midia. The system uses STFT fingerprints and other tools to recognize commercials and may be applied to radio and TV broadcasts, and has already been applied on real data and attained satisfactory results.

11. ADDITIONAL AUTHORS

Additional authors: Sérgio D. Fischer (IBOPE Pesquisa de Mídia Ltda. email: **sfischer@ibope.com.br**).

12. REFERENCES

[1] Philips content identification: *Audio fingerprinting*. http://www.research.philips.com/initiatives/contentid/audiofp.html.

[2] L. F. Costa and R. Cesar-Jr. *Shape Analysis and Classification: Theory and Practice*. CRC Press, 2001

[3] J. Haitsma and T. Kalker. *A highly robust audio fingerprinting system*. In Proc. International Symposium on Musical Information Retrieval (ISMIR2002), pages 144–153, 2002.

[4] R. Matushima, D. M. Hiramatsu, R. M. Silveira, W. V. Ruggiero, C. E. M. da Costa, M. M. Monteiro, and C. Hatori. *Integrating mpeg-7 descriptors and pattern recognition: An environment for multimedia indexing and searching*. In Proc. WebMedia & LA-Web 2004 Joint Conference 10th Brazilian Symposium on Multimedia and the Web 2nd Latin American Web Congress, pages 125–132. IEEE Computer Society Press, 2004.

[5] E. Wold, T. Blum, D. Keislar, J. Wheaton, and M. Fish. *Content-based classification, search, and retrieval of audio*. IEEE Multimedia, 3(3):27–36, 1996.