Classification of general audio data for content-based retrieval

Article *in* Pattern Recognition Letters · April 2001

DOI: 10.1016/S0167-8655(00)00119-7

Ishwar Sethi
Oakland University
216 PUBLICATIONS **3,950** CITATIONS
SEE PROFILE

Nevenka Dimitrova
Philips Cambridge
220 PUBLICATIONS **2,355** CITATIONS
SEE PROFILE

ELSEVIER SCIENCE B.V. [DTD 4.1.0]
JOURNAL PATREC ARTICLE No. 2700
PAGES 01-12 DISPATCH 12 December 2000 PROD. TYPE: FROM Disk
PATREC 2700 A

ELSEVIER

Pattern Recognition Letters 000 (2000) 000–000

Pattern Recognition Letters

www.elsevier.nl/locate/patrec

Classification of general audio data for content-based retrieval

Dongge Li [a], Ishwar K. Sethi [b,*], Nevenka Dimitrova [c], Tom McGee [c]

[a] *Department of Computer Science, Wayne State University, Detroit, MI 48202, USA*
[b] *Intelligent Information Engineering Laboratory, Department of Computer Science and Engineering, Oakland University, Rochester, MI 48309-4478, USA*
[c] *Image Processing and Network Architecture Department, Philips Research, 345 Scarborough Road, Briarcliff Manor, NY 10510, USA*

Abstract

In this paper, we address the problem of classification of continuous general audio data (GAD) for content-based retrieval, and describe a scheme that is able to classify audio segments into seven categories consisting of silence, single speaker speech, music, environmental noise, multiple speakers' speech, simultaneous speech and music, and speech and noise. We studied a total of 143 classification features for their discrimination capability. Our study shows that cepstral-based features such as the Mel-frequency cepstral coefficients (MFCC) and linear prediction coefficients (LPC) provide better classification accuracy compared to temporal and spectral features. To minimize the classification errors near the boundaries of audio segments of different type in general audio data, a segmentation–pooling scheme is also proposed in this work. This scheme yields classification results that are consistent with human perception. Our classification system provides over 90% accuracy at a processing speed dozens of times faster than the playing rate. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: Audio classification; Audio segmentation; Content-based retrieval; Mel-frequency cepstral coefficients; Pooling

1. Introduction

There are several groups engaged in research and development of methods for automatic annotation of images and videos for content-based indexing and retrieval. The need for such methods is increasingly becoming important as the desktop PC and the ubiquitous TV converge into a single infotainment appliance bringing an unprecedented access to terabytes of video data via the Internet. Although most of the existing research in this area is image-based, there is a growing realization that image-based methods for content-based indexing and retrieval of video need to be supplemented with audio-based analysis. This has led to several efforts related to the analysis of audio tracks in videos, particularly towards the classification of audio segments into different classes to represent the video content (Patel and Sethi, 1996, 1997; Saraceno and Leonardi, 1997; Liu et al., 1998).

The advances in automatic speech recognition (ASR) are also leading to an interest in classification of general audio data (GAD), i.e., audio data from sources such as news and radio broadcasts, and archived audiovisual documents. The motivation is that by performing audio classification as a preprocessing step, an ASR system can employ appropriate acoustic model for each homogenous segment of audio data representing a single class,

[*] Corresponding author.
E-mail address: isethi@oakland.edu (I.K. Sethi).

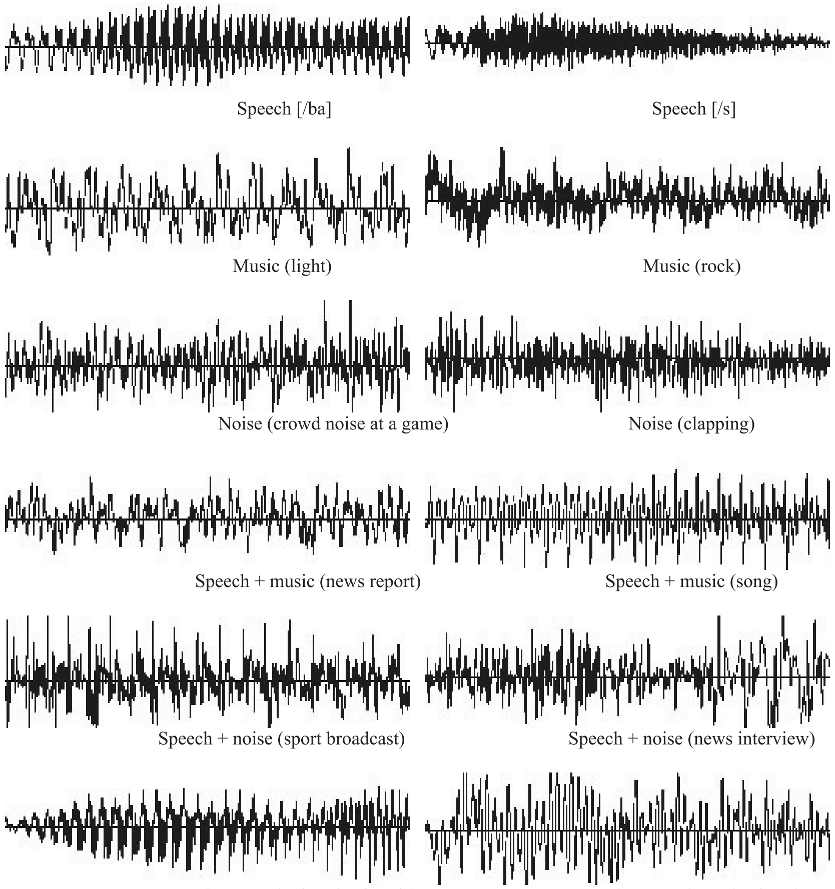
PII: S0167-8655(00)00119-7

resulting in an improved recognition performance (Spina and Zue, 1996; Gopalakrishnan et al., 1996).

Many audio classification schemes have been investigated in recent years. These schemes mainly differ from each other in two ways: the choice of the classifier, and the set of the acoustical features used. The classifiers that have been used in current systems include Gaussian model-based classifiers (Spina and Zue, 1996), neural network-based classifiers (Liu et al., 1998; Hansen and Womack, 1996), decision trees (Zhang and Kuo, 1999), and the hidden Markov model-based (HMM-based) classifiers (Zhang and Kuo, 1999; Kimber and Wilcox, 1996). Both the temporal and the spectral domain features have been investigated. Examples of the features used includes short-time energy (Zhang and Kuo, 1999; Li and Dimitrova, 1997; Wold and Blun et al., 1996), pulse metric (Pfeiffer et al., 1996; Fischer et al., 1995), pause rate (Patel and Sethi, 1996), zero-crossing rate (Saraceno and Leonardi, 1997; Zhang and Kuo, 1999; Scheirer and Slaney, 1997), normalized harmonicity (Wold and Blun et al., 1996), fundamental frequency (Liu et al., 1998; Zhang and Kuo, 1999; Li and Dimitrova, 1997; Wold and Blun et al., 1996; Pfeiffer et al., 1996), frequency spectrum (Fischer et al., 1995), bandwidth (Liu et al., 1998; Wold and Blun et al., 1996), spectral centroid (Liu et al., 1998; Wold and Blun et al., 1996; Scheirer and Slaney, 1997), spectral roll-off frequency (SRF) (Li and Dimitrova, 1997; Scheirer and Slaney, 1997), and band energy ratio (Patel and Sethi, 1996; Liu et al., 1998; Li and Dimitrova, 1997). Scheirer and Slaney (1997) evaluated various combinations of 13 temporal and spectral features using several classification strategies. They report a classification accuracy of over 90% for a two-way speech/music discriminator, but only about 65% for a three-way classifier that uses the same set of features to discriminate speech, music, and simultaneous speech and music. Hansen and Womack (1996), and Spina and Zue (1996) have investigated the cepstral-based features, which are widely used in the speech recognition domain. In Hansen and Womack (1996), the autocorrelation of the Mel-cepstral (AC-Mel) parameters are suggested as suitable features for the classification of stress conditions in speech. In Spina and Zue (1996), Spina and Zue used fourteen mel-frequency cepstral coefficients (MFCC) to classify audio data into seven categories. The categories in their work are: studio speech, field speech, speech with background music, noisy speech, music, silence, and garbage, which covers the rest of audio patterns. They tested their algorithm on an hour of NPR radio news and achieved 80.9% classification accuracy.

While many researchers in this field place considerable emphasis on the development of various classification strategies, Scheirer and Slaney (1997) concluded that the topology of the feature space is rather simple. Thus, there is very little difference between the performances of different classifiers. In many cases, the selection of features is actually more critical to the classification performance. Our work in this paper is motivated by the above observation of Scheirer and Slaney. We have investigated a total of 143 classification features for the problem of classifying continuous GAD into seven categories. The seven audio categories used in our system consists of silence, single speaker speech, music, environmental noise, multiple speakers' speech, simultaneous speech and music, and speech and noise. The environmental noise category refers to noise without foreground sound. The simultaneous speech and music category includes both singing and speech with background music. Example waveforms for the seven categories are shown in Fig. 1. To make the task of feature evaluation easier, an audio toolbox was developed for feature extraction. Our classifier parses a continuous bit-stream of audio data into different non-overlapping segments such that each segment is homogenous in terms of its class. Since the transition of audio from one category into another can cause classification errors, we also suggest a segmentation-pooling scheme as an effective way to reduce such errors.

The paper is organized as follows. Section 2 describes our auditory toolbox and the extraction of features. The framework for the classification of continuous GAD is presented in Section 3. The comparison of different sets of features and the evaluation of the suggested system is given in


Speech [/ba]

Speech [/s]

Music (light)

Music (rock)

Noise (crowd noise at a game)

Noise (clapping)

Speech + music (news report)

Speech + music (song)

Speech + noise (sport broadcast)

Speech + noise (news interview)

Speech + speech (female+female)

Speech + speech (male+female)

Fig. 1. Short segments from six of the seven categories. The seventh category is silence.

Section 4. Finally, summarizing remarks and potential applications are given in Section 5.

2. Auditory toolbox

To make our work easily reusable and expandable and to facilitate our experiments on different feature extraction designs in this research, we have developed an auditory toolbox. In its current implementation, it has more than two dozens of tools. Each tools is responsible for a single basic operation that is frequently needed for the analysis of audio data. By using the toolbox, many of the troublesome tasks related to the processing of streamed audio data, such as buffer management and optimization, synchronization between different processing procedures, and exception handling, become transparent to users. Operations that are currently implemented in our toolbox include frequency-domain operations, temporal-domain operations, and basic mathematical operations such as short-time averaging, log operations, windowing, clipping, etc. Since a common communication agreement is defined among all of the tools, the results from one tool can be shared with other types of tools without any limitation. Tools within the toolbox can thus be organized in a very flexible way to accommodate various applications and requirements.

We show in Fig. 2 the arrangement of tools we used for the extraction of six sets of acoustical

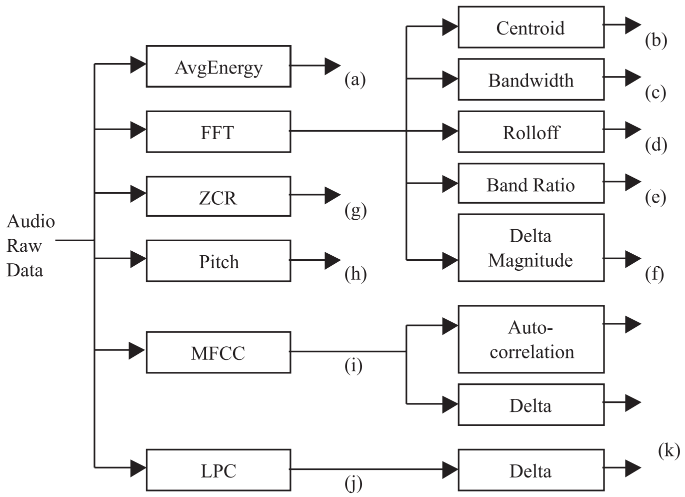


Fig. 2. The organization of tools for audio feature extraction.

features, including MFCC, linear prediction coefficients (LPC), delta MFCC, delta LPC, autocorrelation MFCC, and several temporal and spectral features. The definitions or algorithms adopted for these features are given in the Appendix A

Based on the above acoustical features, many more features that are used in the classification of audio segments can be further extracted by analyzing acoustical features of adjacent frames. According to our experimental results, these features, which correspond to the characteristics of audio data over a longer term, e.g., 600 ms, are more suitable for the classification of audio segments. The features used for audio segment classification include:

- The means and variances of acoustical features over a certain number of successive frames centered on the frame of interest.
- Pause rate: The ratio between the number of frames with energy lower than a threshold and the total number of frames being considered.
- Harmonicity: The ratio between the number of frames with a valid pitch value and the total number of frames being considered.
- Summations of energy of MFCC, delta MFCC, automation MFCC, LPC, and delta LPC.

3. Classification framework for GAD

Our audio classification system, as shown in Fig. 3, consists of four processing steps: feature extraction, pause detection, automatic audio segmentation, and audio segment classification. Feature extraction is implemented using our toolbox described in the previous section. During the run time, acoustical features that are used in the succeeding three procedures are extracted frame by frame along the time axis from the input audio raw data (in our case, the PCM WAV-format data sampled at 44.1 kHz). Pause detection is responsible for separating the input audio clip into silence segments and signal segments. Here, the pause means a time period that is judged by a listener to be a period of absence of sound, other than one caused by a stop consonant or a slight hesitation (Brady, 1965). It is thus very important for a pause detector to generate results that are consistent with the perception of human beings.

Many of the previous studies on audio classification were performed with audio clips containing data only from a single audio category. However, a “true” continuous GAD contains segments from many audio classes. Thus, the classification performance can suffer adversely at places where the underlying audio stream is making a transition from one audio class into another. We call this loss

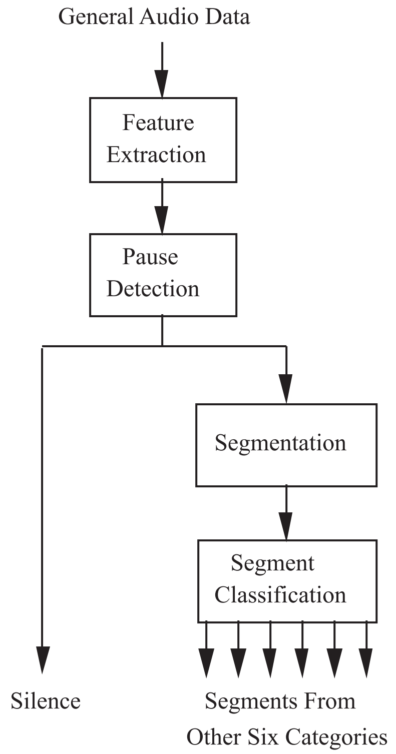


Fig. 3. Block diagram for the classification of GAD.

in accuracy the *border effect*. This loss in accuracy due to the border effect is also reported in (Spina and Zue, 1996; Scheirer and Slaney, 1997). In order to minimize the performance losses due to the border effect, we use a segmentation–pooling scheme. The segmentation part of the segmentation–pooling scheme is used to locate the boundaries in the signal segments where a transition from one type of audio to another type is taking place. This part uses the onset and offset measures, which indicate how fast the signal is changing, to locate boundaries in the signal segments of the input. The result of the segmentation processing is to yield smaller homogeneous signal segments. The pooling component of the segmentation-pooling scheme is used at the time of classification. It involves pooling of the frame-by-frame classification results to classify a segmented signal segment. In the following three subsections we will discuss the algorithms adopted in our pause detection, audio segmentation, and audio segment classification.

3.1. Pause detection

A three-step procedure is implemented for the detection of pause periods from GAD. Based on the features extracted by our toolbox, the input audio data is first marked frame-by-frame as a signal or a pause frame to obtain raw boundaries. The frame-by-frame classification is done using a decision tree algorithm. The decision tree is obtained based on the hierarchical feature space partitioning method due to Sethi and Sarvarayudu (1982). In Fig. 4, we show the partitioning result for a two-dimensional feature space and its corresponding decision tree for pause detection. Since the results obtained in the first step are usually

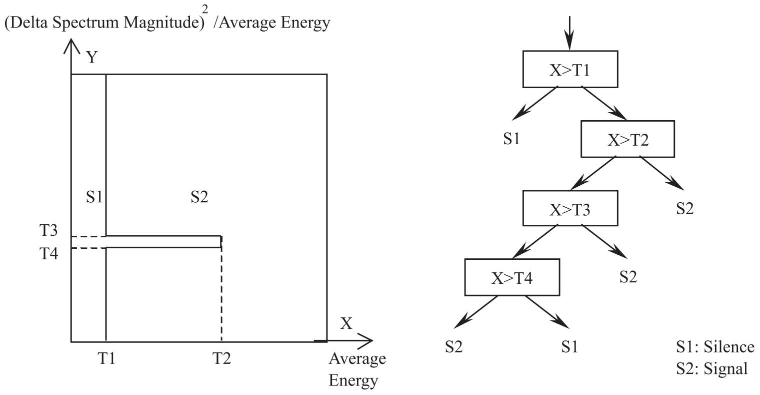


Fig. 4. Two-dimensional partitioned space and its corresponding decision tree.

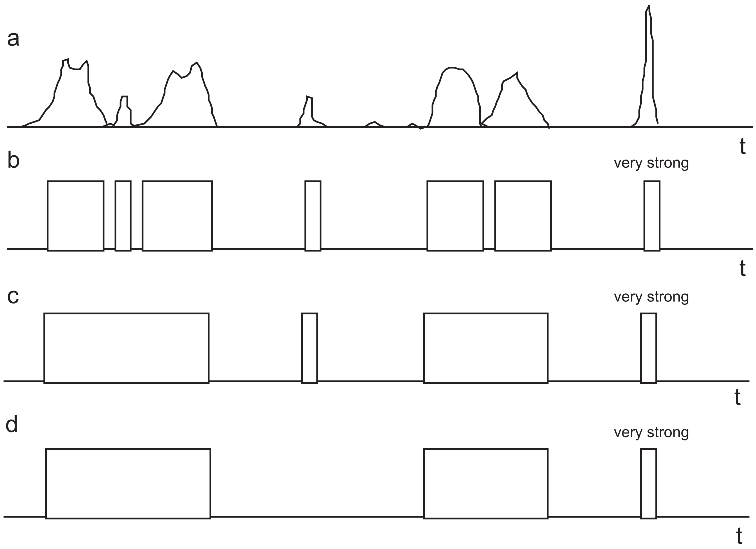


Fig. 5. Pause detection steps: (a) short time energy of input signal; (b) initial result for candidate signal segments; (c) result after fill-in step; (d) result after throwaway step.

sensitive to unvoiced speech and slight hesitations, a fill-in process and a throwaway process are then applied in the succeeding two steps to generate results that are more consistent with the human perception of pause.

During the fill-in process, a pause segment, that is, a continuous sequence of pause frames, with a length less than the fill-in threshold is relabeled as a signal segment and is merged with the neighboring signal segments. During the throwaway process a segment labeled signal with a strength value smaller than a threshold is relabeled as a silence segment. The strength of a signal segment is defined as:

$$\text{Strength} = \max\left(L, \sum_{i}^{L} \frac{s(i)}{T_1}\right), \tag{1}$$

where L is the length of the signal segment and T_1 corresponds to the lowest signal level shown in Fig. 4. The basic idea of defining segment strength, instead of using the length of the segment directly is to take signal energy into account so that segments of transient sound bursts will not be marked as silence during the throwaway process (Brady, 1965). Fig. 5 illustrates the three steps of our pause detection algorithm.

3.2. Automatic audio segmentation

The pause detection stage yields two kinds of segments: silence and signal. The silence segments do not need any further processing because they are already classified. The signal segments, however, need additional processing to mark the transition points, i.e., locations where the category of the underlying signal changes, before classification. To locate transition points, our segmentation scheme uses a process of two successive steps: break detection and break merging. During the break detection step, a large detection window over the signal segment is moved and the average energy of different halves of the window at each sliding position is compared. This permits us to detect two types of breaks:

$$\begin{cases} \text{onset break}: & \text{if } \overline{E}_2 - \overline{E}_1 > \text{TH}_1, \\ \text{offset break}: & \text{if } \overline{E}_1 - \overline{E}_2 > \text{TH}_2, \end{cases}$$

where $\overline{E}_1$ and $\overline{E}_2$ are average energy of the first and the second halves of the detection window, respectively. The onset break indicates a potential change in audio category because of increased signal energy. Similarly, the offset break implies a change in the category of the underlying signal

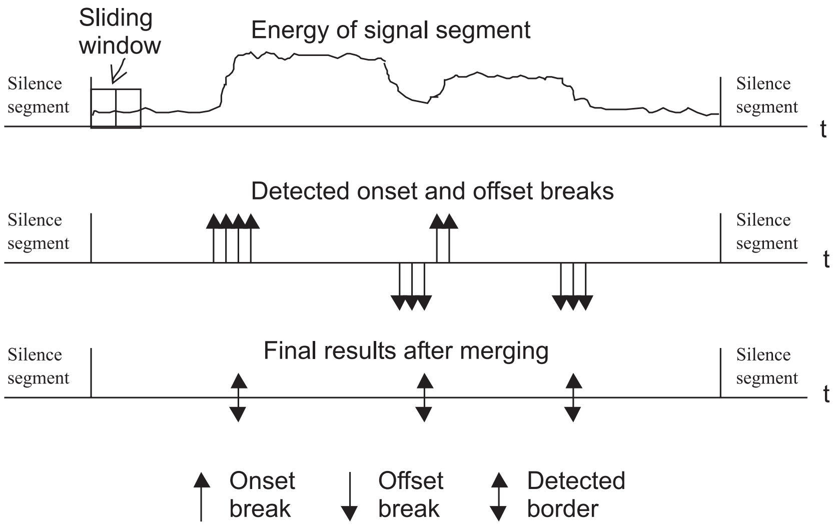


Fig. 6. An illustration of the segmentation process.

because of lowering of energy. Since the break detection window is slid along the signal, a single transition in the audio category of the underlying signal can generate several consecutive breaks. The merger of such a series of breaks is accomplished during the second step of our segmentation process. During this step, adjacent breaks of the same type are merged into a single break. An offset break is also merged with its immediately following onset break, provided the two are close to each other. This is done to bridge any small gap between the end of one signal and the beginning of another signal. Fig. 6 provides an illustration of the segmentation process through the detection and merger of signal breaks.

3.3. Audio segment classification

In order to classify an audio segment, first we classify each and every frame of the segment. Next, the frame classification results are integrated to arrive at a classification label for the entire segment. The integration is performed by a pooling process which counts the number of frames assigned to each audio category. The category most heavily represented in the counting is taken as the audio classification label for the segment. The features used to classify a frame not only come from that frame but also from other frames as mentioned earlier in Section 2. The classification is performed using a Bayesian classifier under the assumption that each category has a multidimensional Gaussian distribution. The classification rule for frame classification can be expressed as

$$c^* = \arg \min_{c=1,2,\ldots,C} \{D^2(\boldsymbol{x}, \boldsymbol{m}_c, \boldsymbol{S}_c) + \ln(\det \boldsymbol{S}_c) - 2\ln(p_c)\}, \tag{2}$$

where C is the total number of candidate categories (in this case, C is 6), c^* the classification result, and $\boldsymbol{x}$is the feature vector of the frame being analyzed. The quantities $\boldsymbol{m}_c$, $\boldsymbol{S}_c$, and p_c represent the mean vector, covariance matrix, and probability of class c, respectively, and $D^2(\boldsymbol{x}, \boldsymbol{m}_c, \boldsymbol{S}_c)$ represents the Mahalanobis distance between $\boldsymbol{x}$and $\boldsymbol{m}_c$. Since $\boldsymbol{m}_c$, $\boldsymbol{S}_c$, and p_c are unknown, these are determined using the maximum a posteriori (MAP) estimator (Duda and Hart, 1973).

4. Performance evaluation

4.1. Comparison of different feature sets

We collected a large number of audio clips from various types of TV programs, such as talk shows, news, football games, weather reports, advertisements, soap operas, movies, late shows, etc. These audio clips were recorded from four different stations (i.e., ABC, NBC, PBS, and CBS) and stored as 8-bit, 44.1 kHz WAV-format files. Care was taken to obtain a wide variation in each category. For example, musical segments of different types of music were recorded. We selected half an hour of our corpus as training data and another hour as testing data. Both training and testing data were then manually labeled with one of the seven categories once every 10 ms. According to Brady (1965) and Agnello (1963), a minimum duration of 200 ms was imposed on silence segments to exclude *intraphase pauses* that are normally not perceived by listeners. Further, the training data was used to estimate the parameters of the classifier.

To investigate the suitability of different feature sets, 68 acoustical features, including eight temporal and spectral features, and 12 each of MFCC, LPC, delta MFCC, delta LPC, and autocorrelation MFCC features, were extracted every 20 ms from the input data. For each of these 68 features, we computed mean and variance over adjacent frames centered around the frame of interest. Thus, a total of 143 classification features, 68 mean values, 68 variances, pause rate, harmonicity, and five summation features, were computed every 20 ms.

We show in Fig. 7 the relative performance of different feature sets on the training data. These results are obtained based on an extensive training and testing on millions of promising subsets of features. The accuracy in Fig. 7 is the classification accuracy at frame level. Furthermore, frames near segment borders are not included in the accuracy calculation. The frame classification accuracy of Fig. 7 thus represents the classification performance that would be obtained if the system were presented segments of each audio type separately. From Fig. 7, we notice that different feature sets perform unevenly. We also see that temporal and spectral features do not perform very well. In our experiments, both MFCC and LPC achieve much better overall classification accuracy than temporal and spectral features. With just 8 MFCC features, we obtain 85.1% classification accuracy using the simple MAP Gaussian classifier; it rises to 95.3%, when the number of MFCC features is increased to 20. This high classification accuracy indicates a very simple topology of the feature space and further confirms Scheirer and Slaney's conclusion for the case of seven audio categories. The effect of using a different classifier is thus expected to be very limited.

To give an idea about how each category fares, we provide in Table 1 the results for the three most important feature sets when using the best 16 features. These results show that the MFCC not only performs best overall but also has the most

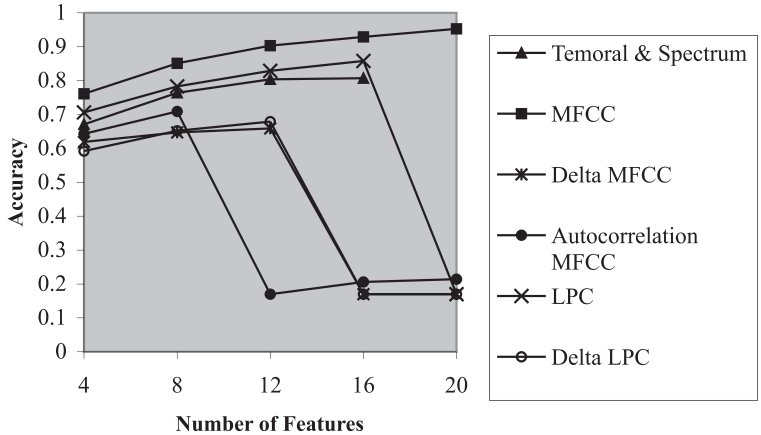


Fig. 7. Comparison of different feature sets for frame classification accuracy.

Table 1
Classification performance using the best 16 features in each set

Feature Set	Classification accuracy					
	Noise	Speech	Music	Speech + noise	Speech + speech	Speech + music
Temporal & Spectrum	93.2	83	75.1	66.4	88.3	79.5
MFCC	98.7	93.7	94.8	75.3	96.3	94.3
LPC	96.9	83	88.7	66.1	91.7	82.7

even performance across the different categories. This further suggests the use of MFCC in applications where just a subset of audio categories is to be recognized.

We also conducted a series of additional experiments to examine the effects of parameter settings. Only minor changes in performance were detected using different parameter settings, for example a different windowing function, or varying the window length and window overlap. No obvious improvement in classification accuracy was achieved when increasing the number of MFCC or using a mixture of features from different features sets.

4.2. Segment classification performance

To see how well the classifier performs on the test data, we used the remaining 1-h of the data as test data. Using the set of 20 MFCC features, the frame classification accuracy of 85.3% was achieved. This accuracy is based on all of the frames including the frames near borders of audio segments. Compared to the accuracy on the training data, we see that there is about 10% drop in accuracy when the classifier deals with segments from multiple classes. The experiments are carried out on a Pentium II PC with 266 MHz CPU and 64 M of memory. For 1 h of audio data sampled at 44.1 kHz, it took 168 s of processing time, which is roughly 21 times faster than the playing rate.

Next, the pooling process was applied to determine the classification label for each segment as a whole. As a result of the pooling process, some of the frames, mostly the ones near the borders, had their classification label changed. Comparing to the known frame labels, the accuracy after the pooling process was found 90.1%, which represents an increase of about 5% from the accuracy without pooling.

An example of difference in classification with and without the segmentation–pooling scheme is shown in Fig. 8. The horizontal axis in the figure represents time. The different audio categories correspond to different levels on the vertical axis. A level change represents a transition from one category into another. The figure shows that the segmentation–pooling scheme is effective in correcting scattered classification errors and eliminating trivial segments. Thus, the segmentation–pooling scheme can actually generate results that are more consistent with the human perception by reducing degradations due to the border effect.

5. Conclusion

The problem of the classification of continuous GAD has been addressed in this paper and an audio classification system, which is able to classify audio segments into seven categories, has been presented. With the help of our auditory toolbox, we have tested and compared a total of 143 classification features. Our results confirm the observation due to Scheirer and Slaney that the selection of features is more important for audio classification. We also conclude that the cepstral-based features such as MFCC, LPC, etc., provide a much better accuracy and should be used for audio classification tasks irrespective of the number of audio categories desired.

A segmentation–pooling scheme is also proposed in this work as an effective way to reduce the border effect and to generate classification results

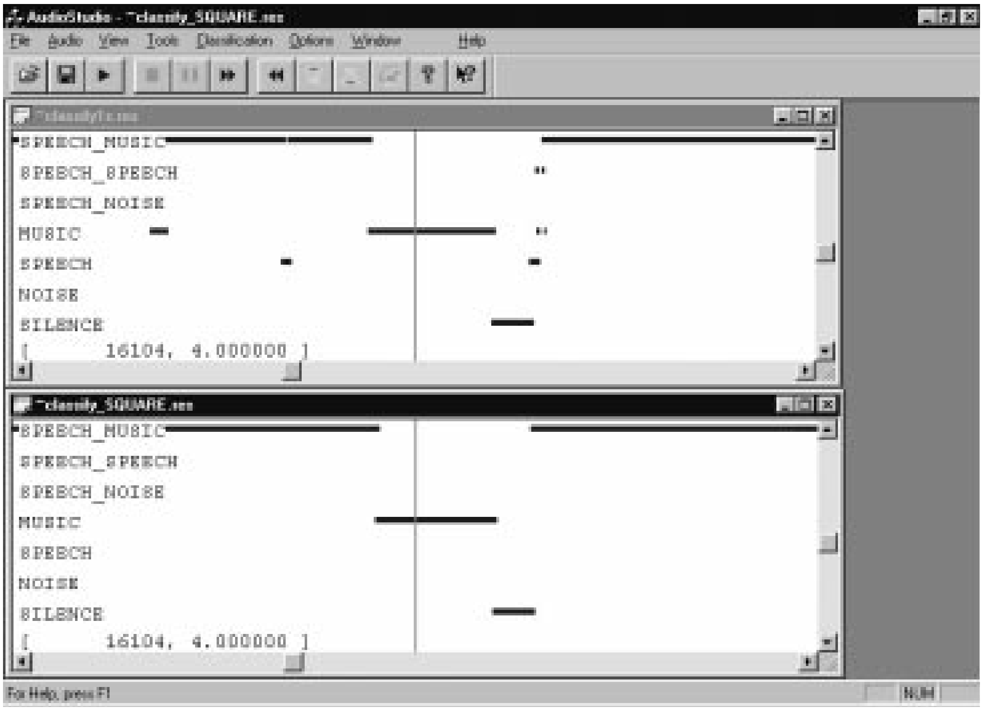


Fig. 8. Classification results: (a) the upper window: results obtained by simply classifying the data frame by frame; (b) the lower window: results obtained from the proposed segmentation–pooling scheme.

that are consistent with the human perception. The experimental results show that the classification system we have built provides about 90% accurate performance with a processing speed dozens of times faster than the playing rate. We expect that the high classification accuracy and processing speed will enable the potential use of audio classification techniques in a wide range of applications, such as video indexing and analysis, automatic speech recognition, audio visualization, video/audio information retrieval, and preprocessing for large audio analysis systems.

Acknowledgements

Dongge Li and Ishwar K. Sethi gratefully acknowledge the support of Philips Research in course of this work. We also wish to thank Nick Mankovich, Ph.D., of Philips Research for his interest in this work.

Appendix A

Here we provide the definitions of different features extracted by our toolbox.

(a) *Short-time average energy*: The tool for calculating short-time average energy is named as AvgEnergy as shown in Fig. 2. The calculation can be expressed as

$$\overline{E}_W = \frac{1}{W}\sum_i s(i)s(i)w(n-i), \tag{A.1}$$

where

$$w(n) = \begin{cases} 1 & 0 < n \leqslant W, \\ 0 & \text{otherwise}, \end{cases}$$

W is the size of the processing window, and $s(i)$ is the discrete time audio signal.

(b) *Spectral centroid*: As shown in Fig. 2, spectral centroid, like the following several spectral features, is calculated based on the short-time Fourier transform, which is performed frame by frame along the time axis. Let $F_i = \{f_i(u)\}_{u=0}^{M}$ represent the short-time Fourier transform of the ith frame, where M is the index for the highest frequency band. The spectral centroid of frame i is calculated as

$$c_i = \frac{\sum_{u=0}^{M} u \cdot |f_i(u)|^2}{\sum_{u=0}^{M} |f_i(u)|^2}. \tag{A.2}$$

(c) *Bandwidth*: Following the definition of spectral centroid given in (A.2), the bandwidth of the FFT of frame i is given as

$$b_i^2 = \frac{\sum_{u=0}^{M} (u - c_i)^2 \cdot |f_i(u)|^2}{\sum_{u=0}^{M} |f_i(u)|^2}. \tag{A.3}$$

(d) *Spectral rolloff frequency*: According to (Li and Dimitrova, 1997), SRF is normally very high for low-energy, unvoiced speech segments and much lower for speech segments with relatively higher energy. Music and noise, however, do not have a similar property, which makes this feature potentially useful for discrimination between speech and other types of audio signals. The definition of SRF is given as

$$\mathrm{SRF}_i = \max\left(h \,\middle|\, \sum_{u=0}^{h} f_i(u) < \mathrm{TH} \cdot \sum_{u=0}^{M} f_i(u)\right), \tag{A.4}$$

where TH is a threshold between 0 and 1. We choose 0.92 for TH in our experiments.

(e) *Band energy ratio*: Although band energy ratio may be defined in different ways, there is essentially not much difference between various definitions. In this work, the band energy ratio (BER) is calculated as

$$\mathrm{BER}_i = \frac{\sum_{u=0}^{h} f_i(u)}{\sum_{u=0}^{M} f_i(u)}, \tag{A.5}$$

where $h = M/4$ for our experiments.

(f) *Delta spectrum magnitude*: As demonstrated in (Scheirer and Slaney, 1997), delta spectrum magnitude is a very suitable feature for a speech/music discriminator. It is given as

$$\Delta F_i = \sum_{u=0}^{M} \left|\,|f_i(u)| - |f_{i+1}(u)|\,\right||. \tag{A.6}$$

(g) *Zero-crossing rate* (ZCR): This feature is a correlate of the spectral centroid. It is defined as the number of time-domain zero-crossings within the processing window.

(h) *Pitch*: The knowledge of pitch contour is used in many applications such as speaker identification, speech analysis, and audio information retrieval. Among the many available pitch detection algorithms, we choose the classical autocorrelation-based pitch tracker due to its robustness (Ghias et al., 1995). To avoid most of the unnecessary time-consuming autocorrelation calculations and to optimize the detection accuracy, a series of modification strategies are adopted in our pitch detection approach. These are described in more detail in (Li and Dimitrova, 1997).

(i) *Mel-frequency cepstral coefficients*: In our implementation, the MFCC are extracted using the DCT of filter-banked FFT spectra (Noll, 1967). The calculations are performed frame by frame on the windowed input data along the time axis. The types of windows that are available include square, and Hamming window.

(j) *Linear prediction coefficients*: The extraction of LPC is implemented using the autocorrelation method, which can be found in (Ramachandran et al., 1995). At each processing step, 12 coefficients are extracted in our experiments.

(k) *Delta MFCC, Delta LPC, and autocorrelation MFCC*: These features provide quantitative measures to the movement of the MFCC or LPC. They have been adopted in some applications in the speech domain. The definitions for these features are given as follows:

$$\Delta\mathrm{MFCC}_i(v) = \mathrm{MFCC}_{i+1}(v) - \mathrm{MFCC}_i(v), \tag{A.7}$$

$$\Delta\mathrm{LPC}_i(v) = \mathrm{LPC}_{i+1}(v) - \mathrm{LPC}_i(v), \tag{A.8}$$

$$\mathrm{ACMFCC}_i^{(l)}(v) = \frac{1}{L} \sum_{j=i}^{i+L} \left(\mathrm{MFCC}_j(v) \cdot \mathrm{MFCC}_{j+l}(v)\right), \tag{A.9}$$

where $\mathrm{MFCC}_i(v)$ and $\mathrm{LPC}_i(v)$ represent the vth MFCC and LPC of frame i, respectively. L is the correlation window length. The superscript l is the value of correlation lag.

References

Patel, N.V., Sethi, I.K., 1996. Audio characterization for video indexing. In: Proc. IS & T/SPIE Conf. Storage and Retrieval for Image and Video Databases IV, San Jose, CA, February, pp. 373–384.

Patel, N.V., Sethi, I.K., 1997. Video Classification using Speaker Identification. In: Proc. IS & T/SPIE Conf. Storage and Retrieval for Image and Video Databases V, San Jose, CA, February, pp. 218–225.

Saraceno, C., Leonardi, R., 1997. Identification of successive correlated camera shots using audio and video information. Proc. ICIP'97 3, 166–169.

Liu, Z., Wang, Y., Chen, T., 1998. Audio feature extraction and analysis for scene classification. J. VLSI Signal Processing (Special issue on multimedia signal processing) October, 61–79.

Spina, M., Zue, V.W., 1996. Automatic transcription of general audio data: preliminary analyses. In: Proc. International Conference on Spoken Language Processing, Philadelphia, PA, October, pp. 594–597.

Gopalakrishnan, et al. P.S., 1996. Transcription of radio broadcast news with the IBM large vocabulary speech recognition system. In: Proc. DARPA Speech Recognition Workshop, February.

Hansen, J.H.L., Womack, B.D., 1996. Feature analysis and neural network-based classification of speech under stress. IEEE Trans. Speech Audio Processing 4(4), 307–313.

Zhang, T., Kuo, C.-C.J., 1999. Audio-guided audiovisual data segmentation, indexing, and retrieval. In: IS & T/SPIE's Symposium on Electronic Imaging Science & Technology – Conference on Storage and Retrieval for Image and Video Databases VII, SPIE 3656, San Jose, CA, January, pp. 316–327.

Kimber, D., Wilcox, L., 1996. Acoustic segmentation for audio browsers. In: Proc. Interface Conference, Sydney, Australia, July.

Li, D., Dimitrova, N., 1997. Tools for audio analysis and classification. Philips Technical Report, August.

Wold, E., Blum, et al. T., 1996. Content-based classification, search, and retrieval of audio. IEEE Multimedia, Fall, 27–36.

Pfeiffer, S., Fischer, S., Effelsberg, W., 1996. Automatic audio content analysis. In: Proc. of ACM Multimedia'96, Boston, MA, pp. 21–30.

Fischer, S., Lienhart, R., Effelsberg, W., 1995. Automatic recognition of film genres. In: Proc. of ACM Multimedia'95, San Francisco, CA, 295–304.

Scheirer, E., Slaney, M., 1997. Construction and evaluation of a robust multifeature speech/music discriminator. In: Proc. ICASSP'97, Munich, Germany, April, pp. 1331–1334.

Brady, P.T., 1965. A technique for investigating on-off patterns of speech. The Bell Syst. Tech. J. 44 (1), 1–22.

Sethi, I.K., Sarvarayudu, G.P.R., 1982. Hierarchical classifier design using mutual information. IEEE Trans. Pattern Recognition Machine Intelligence 4 (4), 441–445.

Duda, R.O., Hart, P.E., 1973. Pattern Classification and Scene Analysis. Wiley, New York.

Agnello, J.G., 1963. A study of intra- and inter-phrasal pauses and their relationship to the rate of speech. Ohio State University Ph.D. Thesis.

Ghias, et al. A., 1995. Query by humming. In: Proc. ACM Multimedia'95, San Francisco, pp. CA, 231–236.

Noll, A.M., 1967. Cepstrum pitch determination. J. Acoust. Soc. Amer. 41 (2).

Ramachandran, R.P., Zilovic, M.S., Mammone, R.J., 1995. A comparative study of robust linear predictive analysis methods with applications to speaker identification. IEEE Trans. Speech Audio Processing 3 (2), 117–125.