AUDIO FEATURE EXTRACTION AND ANALYSIS FOR SCENE SEGMENTATION AND CLASSIFICATION

Zhu Liu and Yao Wang
Polytechnic University
Brooklyn, NY 11201
{zhul,yao}@vision.poly.edu

Tsuhan Chen
Carnegie Mellon University
Pittsburgh, PA 15213
tsuhan@ece.cmu.edu

Abstract

Understanding of the scene content of a video sequence is very important for content-based indexing and retrieval of multimedia databases. Research in this area in the past several years has focused on the use of speech recognition and image analysis techniques. As a complimentary effort to the prior work, we have focused on using the associated audio information (mainly the nonspeech portion) for video scene analysis. As an example, we consider the problem of discriminating five types of TV programs, namely commercials, basketball games, football games, news reports, and weather forecasts. A set of low-level audio features are proposed for characterizing semantic contents of short audio clips. The linear separability of different classes under the proposed feature space is examined using a clustering analysis. The effective features are identified by evaluating the intracluster and intercluster scattering matrices of the feature space. Using these features, a neural net classifier was successful in separating the above five types of TV programs. By evaluating the changes between the feature vectors of adjacent clips, we also can identify scene breaks in an audio sequence quite accurately. These results demonstrate the capability of the proposed audio features for characterizing the semantic content of an audio sequence.

1. Introduction

A video sequence is a rich multimodal information source, containing speech, audio, text (if closed caption is available), color patterns and shape of imaged objects (from individual image frames), and motion of these objects (from changes in successive frames). Although the human being can quickly interpret the semantic content by fusing the information from different modalities, computer understanding of a video sequence is still in a quite primitive stage. With the booming of the Internet and various types of multimedia resources, there is a pressing need for efficient tools that enable easier dissemination of audiovisual information by the human being. This means that multimedia resources should be indexed, stored and retrieved in a way similar to the way that a human brain processes them. This requires the computer to understand their contents before all other processing. Other applications requiring scene understanding include spotting and tracing of special events in a surveillance video, active tracking of special objects in unmanned vision systems, video editing and composition, etc.

The key to understanding of the content of a video sequence is scene segmentation and classification. Research in this area in the past several years has focused on the use of speech and image information. These include the use of speech recognition and language understanding techniques to produce keywords for each video frame or a group of frames [1, 2], the use of image statistics (color histograms, texture descriptors and shape descriptors) for characterizing the image scene [3-5], detection of large differences in image intensity or color histograms for segmentation of a sequence into groups of similar content [6, 7], and finally detection and tracking of a particular object or person using image analysis and object recognition techniques [8]. Another related work is to create a summary of the scene content by creating a mosaic of the imaged scene with trajectories of moving objects overlaying on top [9], by extracting key frames in a video sequence that are

representative frames of individual shots [10], and by creating a video poster and an associated scene transition graph [11].

Recently several researchers have started to investigate the potential of analyzing the accompanying audio signal for video scene classification [12-15]. This is feasible because, for example, the audio in a football game is very different from that in a news report. Obviously, audio information alone may not be sufficient for understanding the scene content, and in general, both audio and visual information should be analyzed. However, because audio-based analysis requires significantly less computation, it can be used in a preprocessing stage before more comprehensive analysis involving visual information. In this paper, we focus on audio analysis for scene understanding.

Audio understanding can be based on features in three layers: low-level acoustic characteristics, intermediate-level audio signatures associated with different sounding objects, and high level semantic models of audio in different scene classes. In the acoustic characteristics layer, we analyze low level generic features such as loudness, pitch period and bandwidth of an audio signal. This constitutes the pre-processing stage that is required in any audio processing system. In the acoustic signature layer, we want to determine the object that produces a particular sound. The sounds produced by different objects have different signatures. For example, each music instrument has its own "impulse response" when struck. Basketball bouncing is different from a baseball hit by the bat. By storing these "signatures" in a database and matching them with an audio segment to be classified, it is possible to categorize this segment into one object class. In the high level model-based layer, we make use of some *a prior* known semantic rules about the structure of audio in different scene types. For example, there is normally only speech in news report and weather forecast, but in a commercial, usually there is always a music background, and finally, in a sports program there exists a prevailing background sound that consists of human cheering, ball bouncing

and music sometimes. Saraceno and Leonardi presented a method for separating silence, music, speech and noise clips in an audio sequence [12], and so did Pfeiffer, *et al.* in [13]. These can be considered as low-level classification. Based on these classification results, one can classify the underlying scene based on some semantic models that govern the composition of speech, music, noise, etc. in different scene classes.

In general, when classifying an audio sequence, one can first find some low-level acoustic characteristics associated with each short audio clip, and then compare it with those pre-calculated for different classes of audio. Obviously classification based on these low-level features alone may not be accurate, but the error can be addressed in a higher layer by examining the structure underlying a sequence of continuous audio clips. This tells us that the very first and crucial step for audio-based scene analysis is to determine appropriate features that can differentiate audio clips associated with various scene classes. This is the focus of the present work. As an example, we consider the discrimination of five types of TV programs: commercials, basketball games, football games, news and weather reports. To evaluate the scene discrimination capability of these features, we analyze the intra- and inter-class scattering matrices of feature vectors. To demonstrate the effectiveness of these features, we apply them to classify audio clips extracted from above TV programs. Towards this goal, we explore the use of neural net classifiers. The results show that an OCON (One Class One network) neural network can handle this problem quite well. To further improve the scene classification accuracy, more sophisticated techniques operating at a level higher than individual clips are necessary. This problem is not addressed in this paper. We also employ the developed features for audio sequence segmentation. Saunders [16] presented a method to separate speech from music by tracking the change of the zero crossing rate, and Nam and Tewfik [14] proposed to detect sharp temporal variations in the power of the subband signals. Here, we propose to use the changes in the feature vector to detect scene transitions.

The organization of this paper is as follows. In Section II, we describe all the audio features we have explored. Analysis of the feature space is presented in Section III. In Sections IV and V, we show the applications of the developed features for scene classification and segmentation. Experimental results are provided within each section. Finally, Section VI concludes the paper by summarizing the main results and presenting remaining research issues.

II. Audio Feature Analysis

There are many features that can be used to characterize audio signals. Generally they can be separated into two categories: time-domain and frequency-domain features. In this section, we describe several audio features that we have explored. In our experiment, the audio signal is sampled at 22KHz, and divided into clips of one second long. Feature analysis is conducted on each clip (i.e. a feature vector is calculated for each clip). These clip-level features are computed based on frame-level features, which are calculated over overlapping short intervals known as frames. Each frame contains 512 samples shifted by 128 samples from the previous frame. Figure 1 illustrates the relation between clips and frames.

Features Derived from Volume Contour

The volume distribution of an audio clip reveals the temporal variation of the signal's magnitude, which is important for scene classification. Here, we use the root mean square (RMS) of the signal magnitude within each frame to approximate the volume of that frame. Specifically, the volume of the *n*-th frame is calculated by:

$$v(n) = \sqrt{\frac{1}{N}\sum_{i=0}^{N-1} s_n^2(i)},$$

where $s_n(i)$ is the *i*-th sample in the *n*-th frame audio signal and *N* is the frame length.

Note that the volume of an audio signal depends on the gain of the recording and digitizing system. Therefore, the mean volume of a clip does not necessarily reflect the scene content. On the other hand, the temporal variation of the volume in a clip does. Figure 2 gives the waveforms and Figure 3 shows the volume contours of three audio clips: a commercial, a basketball game and a news report. The characteristics of the football sequence are similar to basketball and so are the weather forecasts to news reports. From these plots, we know that the volume of these three audio clips have different distributions. To measure the temporal variation of an audio clip's volume, we define two time domain features based on volume distribution. The first one is the *volume standard deviation* (VSTD), which is the standard deviation of the volume over a clip, normalized by the maximum volume in the clip. The second one is the *volume dynamic range* (VDR) defined as $VDR = (\max(v) - \min(v))/\max(v)$, where $\min(v)$ and $\max(v)$ represent the minimum and maximum volume within an audio clip. We have found that, in sports programs, there is a nearly constant level of the background sound, and therefore the volume does not change a lot. On the other hand, in news and weather reports, there are silent periods between speech, so the VSTD and VDR are much higher. Obviously these two features are correlated, but we found that they do carry some independent information about the scene content.

To detect frames that are silent, we compare the volume and zero crossing rate (ZCR - the times that an audio waveform crosses the zero axis) of each frame to some preset thresholds. If both volume and ZCR are less than the thresholds, the frame is declared to be silent. The use of ZCR in addition to the volume can prevent the low energy unvoice speech frames from being classified to be silent. In our simulation, the signal waveform is digitized to a range of -127 to 128 and the maximum volume is about 40. Based on the distribution of the volume and ZCR of the audio signals in our database (see below), we found that a threshold of 3 for volume and a threshold of 50 for

ZCR worked well. From the result of silence detection, we calculate the *non-silence-ratio (*NSR), which is the ratio of the non-silent interval to the entire clip. We found that this ratio varies quite significantly in different video sequences. In news reports there are regular pauses in the reporter's speech; on the other hand in commercials there are always some background music which results in a higher NSR.

It is known that speech has a characteristic energy modulation peak around 4Hz syllabic rate [17]. We define the frequency component of the volume contour around 4Hz (FCVC4) as

$$FCVC4 = \frac{\int_0^{\infty} W(\omega)\left|C(\omega)\right|^2 d\omega}{\int_0^{\infty} \left|C(\omega)\right|^2 d\omega}$$

where C(ω) is the Fourier transform of volume contour of the current clip and W(ω) is a triangular window function centered at 4Hz. Clips composed of speech tend to have higher FCVC4 than those composed of music or noise.

Features Derived from Pitch Contour

Pitch is the fundamental period of an audio waveform, and is an important parameter in the analysis and synthesis of speech signals. In an audio signal, we can still use pitch as a low-level feature to characterize the periodicity of waveforms in different audio signals. Among the many available pitch determination algorithms, we choose the one that uses the short time *Average Magnitude Difference Function* (AMDF) to determine the pitch of each frame. The AMDF is defined as

$$\gamma(l) = \frac{\sum_{i=0}^{N-l-1} \left|s_n(i+l) - s_n(i)\right|}{N-l}.$$

Figure 4 shows the AMDF of an audio frame. The algorithm for pitch detection from AMDF follows that in [18]. The idea is to find the first valley point in the AMDF from left to right in the search range. Here a valley is defined as a local minimum that satisfies additional constraints in terms of its value relative to the global minimum and its curvature. For example, the AMDF in Figure 4 has two valleys, and the pitch period is the distance between the origin and the first valley. Normally such valleys exist in voice or music frames and vanish in noise or unvoice frames. Here we set the search range to be from 2.3 ms to 15.9 ms, which is the pitch range of normal human speech. After computing the pitch of each frame, we obtain a pitch contour for the entire clip. For the intervals for which no pitch is found, the pitch is assumed to be zero. The pitch contour obtained by this method may not match perfectly the real pitch contour. In some short time intervals, the pitch value diverges from the real contour, sometimes twice or half of the real pitch. A median filter is applied to this contour to smooth out falsely detected pitches. Figure 5 gives the pitch tracks of the same three audio clips as used in Figure 2. In the commercial clip, there exists music background with overlapping notes and the detected pitch at a particular frame depends on which note is stronger. Therefore, the pitch track stays flat for short intervals and there exist both high and low pitch periods. In the basketball clip, since there is significant background noise, the pitch track is very rough, rarely with a smooth region. In the news clip, the pitch track is smooth and lasts relatively long. The intervals between two smooth tracks correspond to silence/unvoice period. We have found that the pitch level is primarily influenced by the speaker (male or female) in the scene rather than by the scene content. On the other hand, the dynamics of the pitch contour appears to reveal scene content more. We therefore use the *pitch standard deviation* (PSTD) as another audio feature.

Based on the pitch estimation results, we can also detect which frame corresponds to voice or music since continuous voice or music frames usually yield a smooth pitch track. The *voice-or-*

music ratio (VMR) is used as another audio feature, which is defined as the ratio of the length of the voice or music frames to the entire audio clip. A frame is considered to be voice or music if its associated pitch track is longer than 50ms. Note that some of the music frames with too high or too low pitch are missed in our pitch detection algorithm because the search range is set according to the human speech. If there is no pitch detected in a frame that is not silent, the frame is considered as noise or unvoice. The *noise-or-unvoice* ratio (NUR) is defined as the ratio of the length of the noise or unvoice frame to the entire audio clip.

Frequency Domain Features

To obtain frequency domain features, we first calculate the spectrogram of an audio clip, which is a 2D plot of the short-time Fourier transform (over each audio frames) along the time axis. Let $S_i(\omega)$ represents the short-time Fourier transform of the *ith* frame. We define the *frequency centroid*, $C(i)$, and *bandwidth*, *B(i)* of this frame as:

$$C(i) = \frac{\int_0^\pi \omega |S_i(\omega)|^2 d\omega}{\int_0^\pi |S_i(\omega)|^2 d\omega}, \qquad B^2(i) = \frac{\int_0^\pi (\omega - C(i))^2 |S_i(\omega)|^2 d\omega}{\int_0^\pi |S_i(\omega)|^2 d\omega}.$$

Similar features have been proposed for audio classification in [19]. Figure 6 shows the spectrogram of the three audio clips given in Figure 3. Figure 7 and Figure 8 show the contours of the frequency centroid and bandwidth computed based on the spectrograms. The zero regions in the contour correspond to silent frames. From these figures, we can see that the basketball clip's frequency centroid is high and has bigger dynamic range, on the other hand, the news clip has low frequency centroid and bandwidth during the voice period and high centroid and bandwidth during the unvoice period. In the commercial clip, there is a continuous music background, so the frequency centroid and bandwidth contours are quite smooth.

Since the energy distribution in different frequency bands varies quite significantly among different audio signals, we also use ratios of the energies in different subbands to the total energy as frequency domain features, which are referred to as *subband energy ratios*. Considering the perceptual property of human ears, we divide the entire frequency band into four subbands, each consists of the same number of critical bands which represent cochlear filters in the human auditory model [20]. Specifically, the frequency ranges for the four subbands are 0-630Hz, 630-1720Hz, 1720-4400Hz and 4400-11025Hz. Figure 9 shows the 4 subband energy ratio contours of the three audio clips given in Figure 3. The four contours in the commercial clip are rather smooth, on the other hand, the contours in basketball clip vary a lot. The energy ratio of subband 1 in the news clip is much higher than those of the other subbands. Considering that the four subband ratios sum to 1, we use the first three ratios as features.

Since the frame with a high energy has more influence on the human ear, when we compute the clip-level features from the above five frame-level features, we use a weighted average of frame-level features, where the weighting for a frame is proportional to the energy of the frame. This is especially useful when there are many silent frames in a clip because the frequency features in silent frames are almost random. By using energy based weighting, their detrimental effects are removed.

To summarize, we have developed twelve clip-level audio features: 1) non-silence-ratio (NSR), 2) volume standard deviation (VSTD), 3) volume dynamic range (VDR), 4) frequency component of the volume contour around 4Hz (FCVC4), 5) pitch standard deviation (PSTD), 6) voice-or-music ratio (VMR), 7) noise-or-unvoice ratio (NUR), 8) frequency centroid (FC), 9) frequency bandwidth (BW), 10-12) energy ratios of subbands 1–3 (ERSB1-3).

III. Feature Space Evaluation

We have collected audio clips from TV programs containing the following five scene classes: news reports with one or two anchor men/women, weather forecasts with male/female reporters, TV commercials, live basketball games and live football games. For each scene class, we collected 1400 audio clips from different TV channels. These data were randomly divided into two sets: training and testing data sets. The training data set includes 400 clips from each scene class and the remaining 1000 clips in each class form the testing data set. All the analysis results reported in this section are computed from the training data.

1. Mean and Variance Analysis

To see how the above features differ among separate scene classes, Table 1 lists the mean feature vectors for the five different scene classes, obtained by averaging the features extracted from all clips in the same class in the training data. We can see that for most features, news and weather reports have similar values. The football and basketball games also have similar values for most features. Features 2, 3, 8, and 12 (VSTD, VDR, FC and ERSB3) differ significantly among commercial, basketball/football, and news/weather. On the other hand features 5, 6 and 7 (PSTD, VMR and NUR) are different between news and weather report. This makes sense because the speech styles in the two programs are different: normally the speed of speech in a weather report is faster than that in news report and there are several speakers in news report while only one in weather report. Features 8 and 9 (FC and BW) differ between basketball games and football games. Although both contain a high level of background sound, their frequency structures are not the same, with one being indoor and the other being outdoor. Besides, in basketball games there exist a lot of high frequency audio components which are caused by the friction between the shoes of the players and the floor.

Table 2 gives the standard deviation of individual features within each class. From the table we can see that the temporal variations of these features in different types of TV programs are different. For example, the range of NSR in news and weather is bigger than that in the other three programs, while the range of FC and BW in news and weather is smaller than those in the other three.

2. Clustering Analysis

In order to evaluate the linear separability of different classes under the feature space generated by the twelve proposed features, we performed a clustering analysis of the training data using the Euclidean distance measure. The intention of this study is to see whether the five audio classes form linearly separable clusters in the feature space, so that they can be separated using a simple nearest neighbor classifier.

Since we do not know how many clusters one audio class may have, and nor do we know whether certain audio classes overlap in the feature space, we use an automatic clustering method to find the number of clusters and their centroids in the feature space. There are two popular clustering algorithms: K-Means and Iterative Self-Organizing Data Analysis Techniques Algorithm (ISODATA) [21]. With the K-Means algorithm, we must provide the expected number of clusters, while with ISODATA, the number is determined by dynamic merging and splitting of clusters based on certain criteria. Without knowing the exact number of clusters in our training data, we use the ISODATA algorithm. By restricting the maximum number of clusters to be 40 and requiring each cluster to contain no less than 10 members, this method results in thirteen clusters.

Table 3 lists the mean feature vector of each cluster. Table 4 shows the distribution of each cluster in different audio classes. We can see that some clusters belong to a single class, such as cluster 3 and 11 which belong to the commercial class, while other clusters are shared by different classes, e.g. cluster 6 and 8 are shared by news and weather report.

The fact that the five scene classes are not linearly separable under the twelve clip-level features is not surprising. For example, in a TV commercial, there are periods of pure speech, periods of pure background music and periods of mixtures. The pure speech period also appears in news/weather report. In order to tell that a particular speech clip is in a TV commercial rather than a new report, one must look at several neighboring clips and make the classification based on high-level semantic analysis. Because of this inherent ambiguity, it is doubtful that additional clip-level features can make all different scene classes linearly separable.

3. Intra-cluster and Inter-cluster Scattering Analysis

In order to evaluate how discriminative each feature is and whether certain features are correlated with each other, we also calculated the intra-class and inter-class scattering matrices [21] of the feature vector consisting of the twelve features. The intra-class scattering matrix reveals the scattering of samples around their respective class centroids, and is defined by

$$S_{\text{intra}} = \sum_{i=1}^{N} P(\omega_i) E\{(X - M_i)(X - M_i)^T \mid \omega_i\},$$

where $P(\omega_i)$ is the *a priori* probability of class ω_i, X is the sample feature vector, M_i is the mean feature vector (centroid) of class ω_i, N is the number of classes. On the other hand, the inter-class scattering matrix is defined as:

$$S_{\text{inter}} = \sum_{i=1}^{N} P(\omega_i)(M_i - M_0)(M_i - M_0)^T \text{ , where } M_0 = \sum_{i=1}^{N} P(\omega_i) M_i \text{ .}$$

The intra-class and inter-class scattering analysis is appropriate only if there exists a single cluster for each class. This is however not the case with our feature space. Therefore, we applied the above analysis to the 13 clusters found by the ISODATA algorithm. This analysis is therefore used to see how useful each feature is in differentiating among clusters. Because of the overlapping of certain clusters among several classes, a feature that cannot discriminate among clusters is likely to fail to

differentiate one class from the other. On the other hand, a feature that can separate different clusters may not be able to differentiate all the classes. Table 5 and Table 6 give the intra-cluster and inter-cluster scatter matrices determined from the training data. When calculating these matrices, and when performing classification (to be discussed in Section IV), all features are normalized by the maximum values of respective features in the training set.

The diagonal entries in these two matrices characterize the intra- and inter-cluster separability of individual features. If the diagonal entry in the intra-cluster scattering matrix is small while that in the inter-cluster matrix is large, then the corresponding feature has good cluster separability. The off-diagonal entries in these two matrices reveal the correlation between different features. We can use these measures to eliminate highly correlated features and reduce the dimensionality of the feature space.

From Table 5 we can see that there exists high correlation between some feature pairs, such as VSTD and VDR, FC and BW, ERSB1 and ERSB2, etc. Here, the correlation is measured by $S(i,j)/\sqrt{S(i,i)S(j,j)}$, where *S(i,j)* is the entry at the *i*-th row and *j*-th column of the intra-cluster scattering matrix. Based on this table we can reduce the dimension of the feature space by proper transformations. Table 7 compares the diagonal entries of the intra-cluster and inter-cluster scattering matrices. We can see that FCVC4, FC, BW, ERSB1 and ERSB3 have good cluster separability (with an inter-to-intra ratio greater than 2.5), while NSR, PSTD and VSTD have poor cluster separability.

IV. Audio-Based Scene Classification

As described in Section I, there are three layers at which we can fulfill scene classification. In the audio characteristics layer, we consider the features of individual clips independently. Classification

based on the clip-level features alone is difficult, because the feature space is not linearly separable. If we use a simple nearest neighbor classifier using the Euclidean distance, we will not get satisfactory results. This has been proven by a preliminary experiment. In this simulation, we calculate the mean feature vector for each of the five classes based on the feature vectors obtained from the training data consisting of 400 clips from each class (results are previously given in Table 1). Then for the testing data set that consists of 1000 clips for each class, we use a nearest neighbor classifier. That is, each clip is classified by calculating the distance of its feature vector to the mean feature vectors of the five classes and identifying the class to which the distance is the shortest. The classification results using this method for the testing data are shown in Table 8, which are quite poor. The reason is that there are more than one cluster for each audio class in the feature space and not all these clusters are closer to the centroid of this class than those of other classes.

Because of above reasons, the conventional nearest neighbor classifier is not effective for our classification task. Artificial neural networks have been used successfully as pattern classifiers in many applications for their ability to implement nonlinear decision boundaries and their capability to learn complicated rules from training data [22, 23]. Conventional multi-layer perceptron (MLP) use the *all-class-in-one-network* (ACON) structure, which is shown in Figure 10. But such a network structure has the burden of having to simultaneously satisfy all the desired outputs for all classes, so the required number of hidden units tends to be large. Besides, if one wants to adapt the network to new training data or add new classes, all the weights need to be re-computed. On the other hand, in the *one-class-one-network* (OCON) structure, one subnet is designated for recognizing one class only [24]. The structure is illustrated in Figure 11. Each subnet is trained individually using the back-propagation algorithm so that its output is close to 1 if the input pattern belongs to this class, otherwise the output is close to 0. Given an input audio clip, it is classified to the class whose

subnet gives the highest score. An advantage of the OCON structure is that one can accommodate a new class easily by adding a subnet trained for that class.

We have applied both ACON (1 hidden layer with 14 neurons) and OCON (5 subnets, 1 hidden layer with 7 neurons in each subnet) structures to perform audio classification. The weights are trained using the training data. The classification results for the testing data are shown in Table 9 and Table 10. Both classifiers can accurately distinguish among commercials, basketball game, football game, and news/weather reports. But the separation of the news from weather reports is less successful. This is not surprising because they contain primarily speech. To distinguish these two classes, some high level correlation information between successive clips that reflects the flow of the conversation may be necessary. As for the comparison of OCON and ACON structures, the classification accuracy using OCON is slightly higher than using ACON. Here the classification accuracy is defined as the average of the correct classification rates of all classes, *i.e.* the diagonal entries in Table 9 and Table 10 respectively. Using more neurons in the hidden layer of the ACON classifier may improve its performance.

The classification results reported in this section are obtained by using all the twelve features described in Section II. We have also tested the performance of the OCON classifier when the three features involving pitch calculation (PSTD, VMR and UNR) are eliminated. From the analysis in Section III.3, these three features do not have good cluster separability. Also, pitch calculation is time consuming. The classification results obtained from using the remaining nine features are given in Table 11. It can be seen that the classification accuracy is about the same as that obtained using twelve features. This confirms that the three pitch-based features are not very useful and that the reduced feature set is sufficient for audio classification. Theoretically, using a reduced feature set should never give better classification results than using the original feature set. The somewhat improved performance for the classification of certain classes by the reduced feature set (compare

Table 9 and Table 11) may be due to the better convergence performance achievable by the reduced feature space when the neural net classifier is trained.

V. Scene Segmentation Using Audio Features

To perform scene analysis in a video sequence, one common approach is to first segment the sequence into shots so that each shot contains the same type of scene, and then classify each shot into one scene type. Usually, scene segmentation is accomplished by detecting significant changes in the statistics of the underlying visual and audio signals. In this section, we consider scene segmentation based on audio information only. Note that an audio segment belonging to the same scene class may be segmented to different shots. For example, if in the middle of a commercial the background music is changed, then the sequence may be segmented into two shots, although they may both be classified as commercial in a later classification stage.

Speech segmentation is a fundamental processing in speech recognition, where a speech signal is segmented into pieces containing voice, unvoice and silence. Segmentation of the audio signal is quite different from that of pure speech. In speech, the length of each segment is very short and the onset and offset of the segment should be precisely determined. On the other hand, in our task, we want to track the semantic content of an audio sequence. Normally, the audio signal with the same scene content will last from several seconds to several minutes. Because a scene transition usually occurs over a relatively long period so that one scene gradually changes to another, exact localization of the transition time is difficult to achieve and is usually not necessary. It is sufficient for most practical applications if a transition is detected shortly after the new scene is stabilized.

There are two considerations in developing segmentation algorithms: the types of audio features based on which a change index is calculated, and the way such indices are calculated and used to locate scene changes. First, we should choose audio features that well describe the statistic

characteristics of individual audio clips. Second, we should find a proper way to compare the features of current and previous audio clips. Saunders [16] used several features derived from the zero crossing rate to separate speech and music, and Nam and Tewfik [14] used energy ratios in five subbands to segment audio input. Here, we explore the use of the distances between feature vectors of adjacent clips for this task. From the intra-cluster and inter-cluster scattering analysis of the twelve features presented in Sec. III.3, we know that the three features involving pitch (PSTD, VMR and NUR) do not have good class discrimination capability. Since the computation of pitch is rather time-consuming, we exclude these three features when computing the feature vector difference. In [14], the difference between the feature vectors of two successive clips (80 msec. long) is computed for measuring scene change. This method is sensitive to short interference, such as the whistling in a basketball game. Here we adopt a different strategy, which compares the feature vector of the current clip to some previous clips and following clips. For a clip to be declared as a scene change, it must be similar to all the neighboring future clips, and different from all the neighboring previous clips. Based on this criterion, we propose using the following measure:

$$Scene-change-index = \frac{\left\| \frac{1}{N}\sum_{i=-N}^{-1} f(i) - \frac{1}{N}\sum_{i=0}^{N-1} f(i) \right\|^2}{\sqrt{\left(c + \mathrm{var}(f(-N),\ldots,f(-1))\right)\left(c + \mathrm{var}(f(0),\ldots,f(N-1))\right)}}$$

where *f(i)* is the feature of the *i*-th clip, with *i*=0 representing the current clip, *i*>0 a future clip, *i*<0 a previous clip, $\|\cdot\|$ is the *L-2* norm, var(…) is the average of the squared Euclidean distances between each vector and the mean feature vector of the N clips considered, and *c* is a small constant to prevent division by zero. When the feature vectors are similar within previous *N* clips and following *N* clips, respectively, but differ significantly between the two groups, a scene break is declared. The selection of the window length *N* is critical: If *N* is too large, this strategy may fail to

detect scene changes between short audio shots. It will also add unnecessary delay to the processing. Through trials-and-errors, we have found that $N=6$ give satisfactory results.

Figure 12(a) shows the content of one testing audio sequence used in segmentation. This sequence is digitized from a TV program that contains seven different semantic segments. The first and the last segments are both football games, between which are TV station's logo shot and four different commercials. The duration of each segment is also shown in the graph. The sequence in Figure 13(a) is obtained by manually combining 5 segments of TV programs, a commercial, a basketball game, a weather report, a football game and news report, each 25 seconds long. Figure 12(b) and Figure 13(b) show the scene-change-indices computed for these two sequences. Scene changes are detected by identifying those clips for which the scene-change-indices are higher than a threshold, D_{min}, and are at least T_{min} seconds away from a previous detected scene change. We have used D_{min}=3, and N =6, which have been found to yield good results through trial-and-error. In these graphs, mark "o" indicates real scene changes and "*" detected scene changes. All the real scene changes are detected using this algorithm. Note that there are two falsely detected scene changes in the first segment of the first sequence. They correspond to the sudden appearance of the commentator's voice and the audience's cheering.

Table 12 summarizes the segmentation results for ten test sequences (including the above two) using the proposed the algorithm. These sequences cover various combinations of different TV programs, including football and commercials, news and commercials, basketball and commercials, news and weather report and commercials, etc. Totally there are 44 real scene changes. The algorithm detects 42 of them. One missed scene change happens between two smoothly connected commercial programs and the other one happens between the news and weather report. There are 36 false alarms given by the algorithm, most of them happen during the sports programs and commercials, where audio discontinuities actually occurred. To remove these false detections, we

can make use of the visual information in the video sequence For example, the color histogram in a sports segment usually remains the same. By requiring that both color histogram and audio features experience sharp changes, we can eliminate falsely detected scene changes based on the audio features alone.

VII. Conclusions

The primary goal of this study is to identify low-level features that are effective in characterizing the semantic content of an audio sequence. Towards this goal, we developed and analyzed twelve features. Through feature space analysis we have found that FCVC4, FC, BW, ERSB1 and ERSB3 have good scene discrimination capability while the three features involving pitch calculation (PSTD, VMR and NUR) have poor discrimination capacity. Using all twelve features, an OCON neural network classifier was successful (82.5% on average) in separating four kinds of TV programs: commercial, basketball game, football game, news report/weather forecast. The discrimination of news report from weather forecast was less accurate (less than 70%). Similar classification results were obtained with a reduced feature set containing nine features (excluding the three pitch-related features). Using the reduced feature vector consisting of the above nine features, a scene-change-index function was developed. From the experiment conducted, it promises to be an effective tool for scene segmentation. Because these nine features are easy to compute, clip-level scene classification and segmentation using these features can be achieved at quite low computation cost.

Our feature space analysis has indicated that some features are less effective than others, and that several useful features are highly correlated. Use of feature space reduction techniques to deduce a more efficient set of features that can retain the discrimination capability of these twelve features is a topic of our future studies. There are also other features that we like to explore and compare with

the ones presented here. In the present study, the effectiveness of the proposed features is evaluated by examining their capability in discriminating five common types of TV programs. However, because these are low-level features calculated over short audio-clips (one second long), we believe they are useful fundamental statistical features for any audio analysis task. Of course, analysis of a general audio sequence acquired in a noisy environment is more difficult and further study is needed to validate the effectiveness of the features presented here for other applications. The classification and segmentation results reported here are meant to show the promise of using audio features for scene analysis. Better classification and segmentation results should be obtainable with more optimized algorithms.

In the present study, classification is accomplished for each clip based on the features of that clip alone. To achieve more accurate classification, such clip-wise classification results should be refined by considering high level semantic models. For example, in a news program, although there is only speech, the reporters are alternated frequently. On the other hand, during the weather forecast, usually the speaker talks for a much longer time period. This requires one to look at the classification results for several clips at a time.

For both the classification and segmentation tasks, ideally, one should make use of both visual and audio information. In general, visual-based analysis involves much more computation than audio-based one. Using audio information alone can often provide a good initial solution for further examination based on visual information. A challenging problem is how to combine the results from audio and visual signal analysis for understanding the semantic content of a video sequence. These are some of the interesting and important problems for future study.

ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under Grant No. IRI-9619114 and by the New York State Center for Advanced Technology in Telecommunications at Polytechnic University, Brooklyn, New York.

REFERENCES

[1] M. A. Smith and T. Kanade, "Video Skimming and Characterization through the Combination of Image and Language Understanding Techniques," *Carnegie Mellon University Technical Report CMU-CS-97-111*, Feb. 1997

[2] Y. Chang, W. Zeng, I. Kamel and R. Alonso, " Integrated Image and Speech Analysis for Content-Based Video Indexing," *Proc. of the 3rd IEEE International Conference on Multimedia Computing and Systems*, pp. 306-313, 1996

[3] M. Flickner, *et al.*, "Query by image and video content: The QBIC system," *IEEE Computer*, Vol. 28, No. 9, pp. 23-32, September 1995.

[4] S. W. Smoliar and H. Zhang, "Content-Based Video Indexing and Retrieval," *IEEE Multimedia Magazine*, Vol. 1, No. 2, pp. 62-72, Summer 1994.

[5] J. R. Smith and S. -F. Chang, "SaFe: A General Framework for Integrated Spatial and Feature Image Search," *Proc. IEEE 1st Multimedia Signal Processing Workshop*, pp. 301-306, June 1997.

[6] H. J. Zhang, A. Kankanhalli, and S. W. Smoliar, "Automatic partitioning of full-motion video," *Multimedia Systems*, Vol. 1, No. 1, pp. 10-28. 1993.

[7] H. J. Zhang, *et. al*, "An Integrated System for Content-Based Video Retrieval and Browsing," *Pattern Recognition*, Vol. 30, No. 4, pp.643-658, 1997.

[8] J. D. Courtney, "Automatic Video Indexing via Object Motion Analysis," *Pattern Recognition*, Vol. 30, No. 4, pp.607-625, 1997.

[9] M. Irani, P. Anandan, J. Bergern, R. Kumar, and S. Hsu, "Efficient Representations of Video Sequences and Their Applications," *Signal Processing: Image Communication*, Vol. 8, pp. 327-351, 1996.

[10] B. Shahraray and D. C. Gibbon, "Pictorial Transcripts: Multimedia Processing Applied to Digital Library Creation," *Proc. IEEE 1st Multimedia Signal Processing Workshop*, pp. 581-586, June 1997.

[11] M. M. Yeung and B. -L. Yeo, "Video Visualization for Compact Presentation and Fast Browsing of Pictorial Content," *IEEE Trans. Circuits and Systems for Video Technology*, Vol. 7, No. 5, pp. 771-785, Oct. 1997.

[12] C. Saraceno and R. Leonardi, "Audio as a Support to Scene Change Detection and Characterization of Video Sequences," *Proc. of ICASSP'97*, Vol. 4, pp. 2597-2600,1997.

[13] S. Pfeiffer, S. Fischer and W. Effelsberg, "Automatic Audio Content Analysis," *Proc. ACM Multimedia'96*, pp. 21-30, 1996

[14] J. Nam and A. H. Tewfik, "Combined Audio and Visual Streams Analysis for Video Sequence Segmentation," *Proc. of ICASSP'97*, Vol. 3, pp. 2665-2668, 1997.

[15] Y. Wang, J. Huang, Z. Liu, and T. Chen, "Multimedia Content Classification using Motion and Audio Information," *Proc. of IEEE ISCAS' 97*, Vol. 2, pp.1488-1491, 1997.

[16] J. Saunders, "Real-time Discrimination of Broadcast Speech/Music," *Proc. of ICASSP'96*, Vol. 2, pp. 993-996, 1996.

[17] E. Scheirer and M. Slaney, "Construction and Evaluation of a Robust Multifeature Speech/Music Discriminator," *Proc. of ICASSP'97*, Vol. 2, pp.1331-1334,1997.

[18] W. Hess, *Pitch Determination of Speech Signals*, Springer-Verlag, 1983.

[19] E. Wold, *et al.*, "Content-Based Classification, Search, and Retrieval of Audio," *IEEE Multimedia Magazine*, Vol. 3, No. 3, pp. 27-36, Fall 1996.

[20] N. Jayant, J. Johnston and R. Safranek, "Signal Compression Based on Models of Human Perception," *Proceedings of IEEE*, Vol. 81, No. 10, pp. 1385-1422 October 1993.

[21] K. Fukunaga, *Introduction to statistical pattern recognition*, Academic Press, 1972.

[22] B. Kosko, *Neural Networks for Signal Processing*, Englewood cliffs, NJ Prentice Hall, 1992.

[23] R. P. Lippman, "An Introduction to Computing with Neural Nets," *IEEE ASSP Magazine*, pp. 4-22, April 1987.

[24] S. H. Lin, S. Y. Kung, and L. J. Lin, "Face Recognition/ Detection by Probabilistic Decision-Based Neural Network," *IEEE Trans. Neural Networks*, Vol. 8, No. 1, pp. 114-132, Jan. 1997.

List of Figures

List of Tables

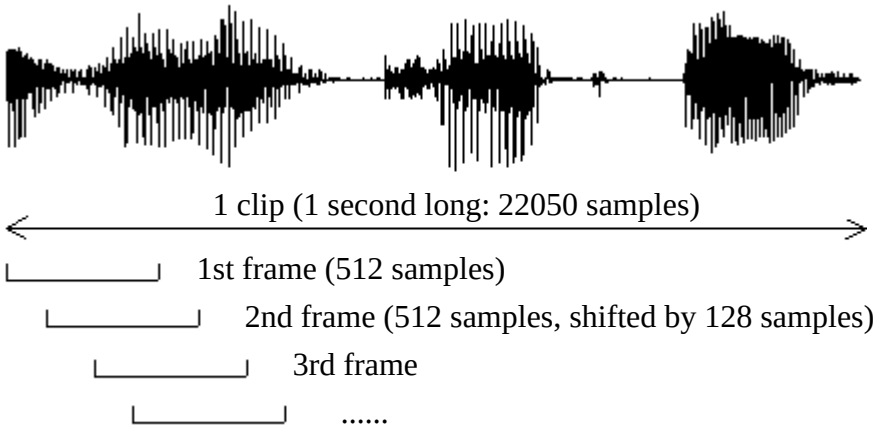
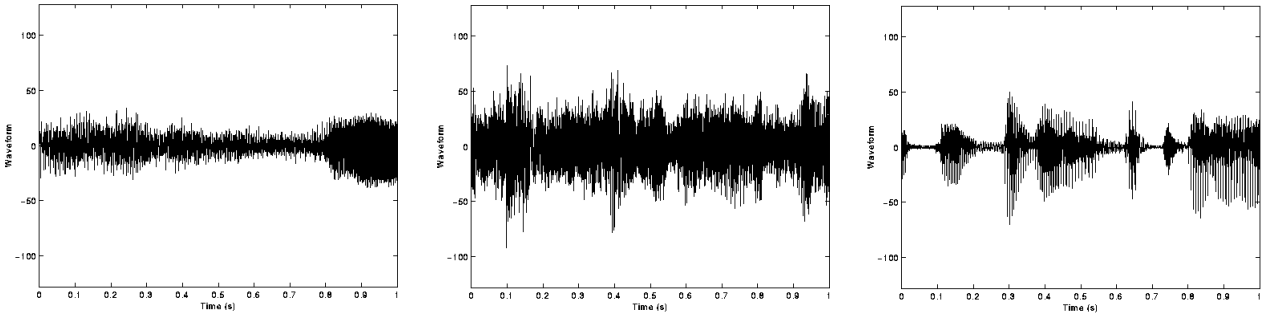
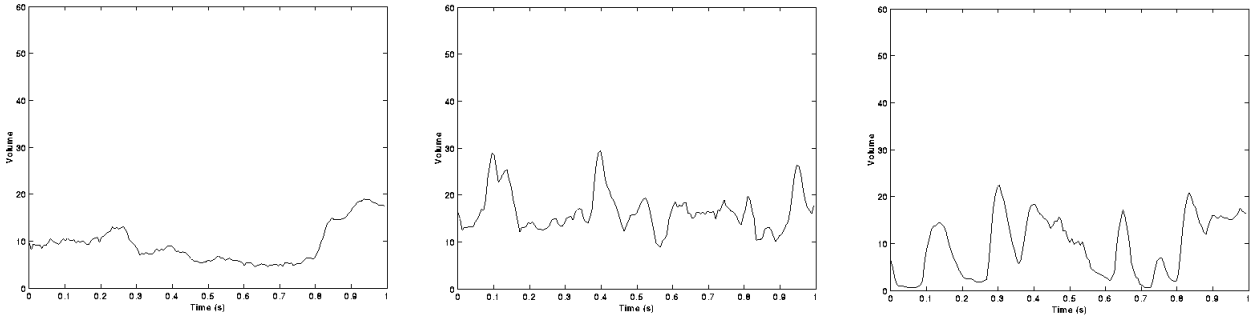


Figure 1 Clip and frames used in feature analysis



(a) commercial (b) basketball (c) news

Figure 2 Waveforms of three audio clips



(a) commercial (b) basketball (c) news

Figure 3 Volume contours of three audio clips

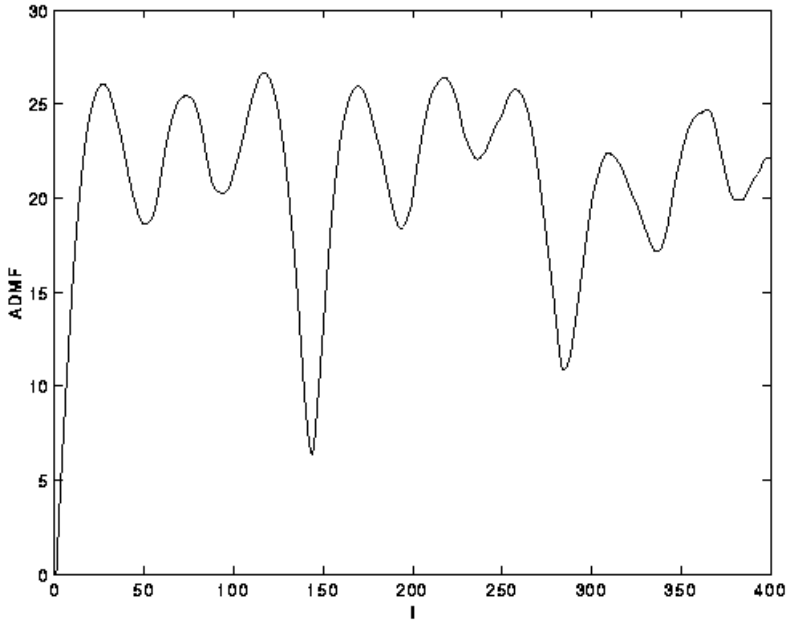
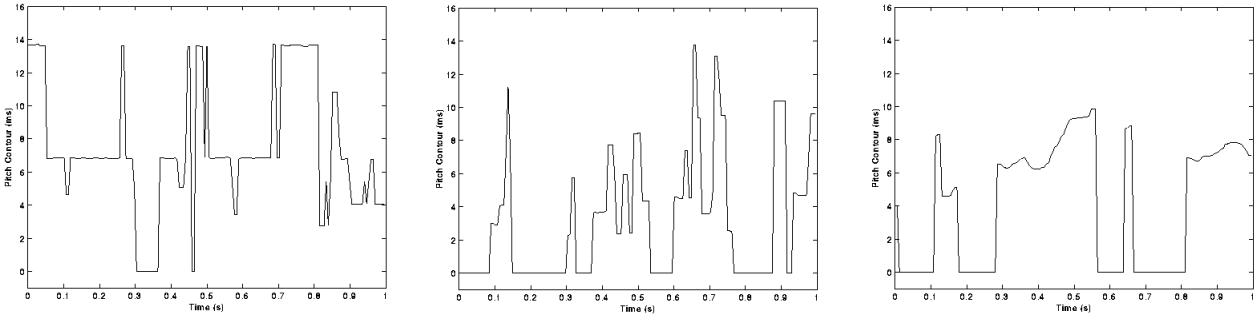
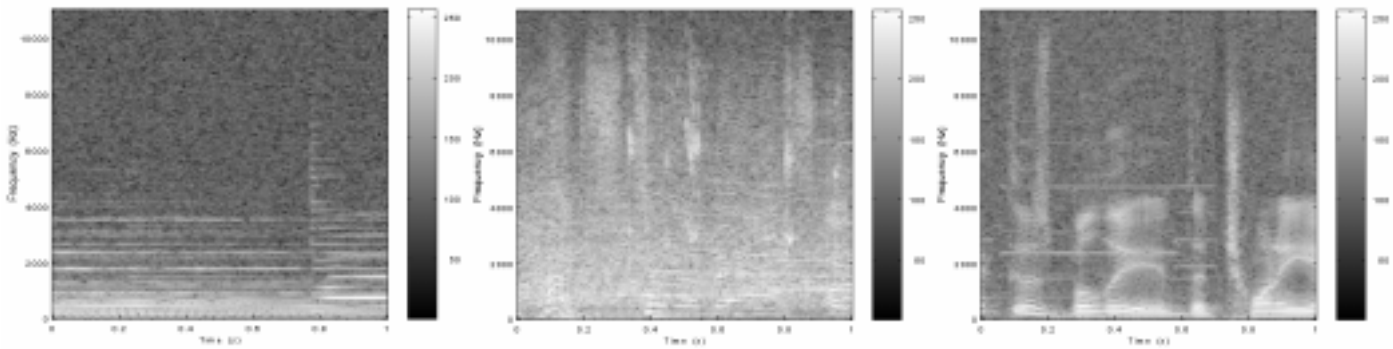


Figure 4 The ADMF of one speech frame



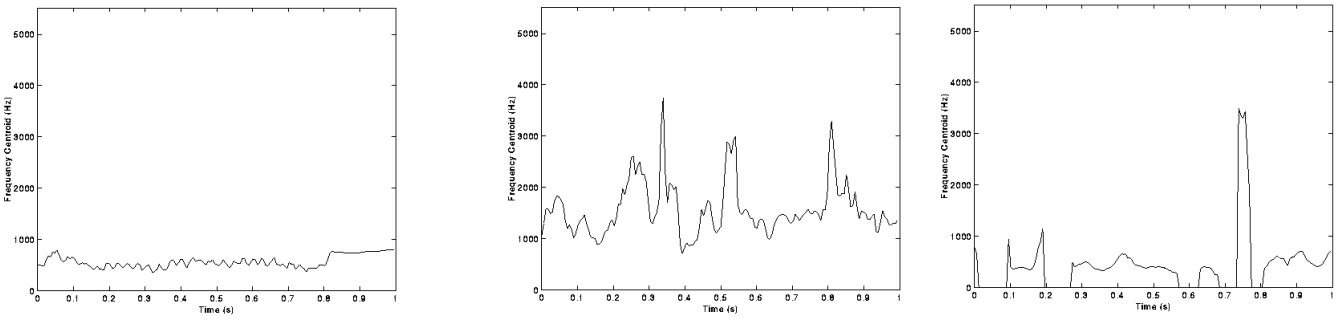
(a) commercial (b) basketball (c) news

Figure 5 Pitch contours of three audio clips



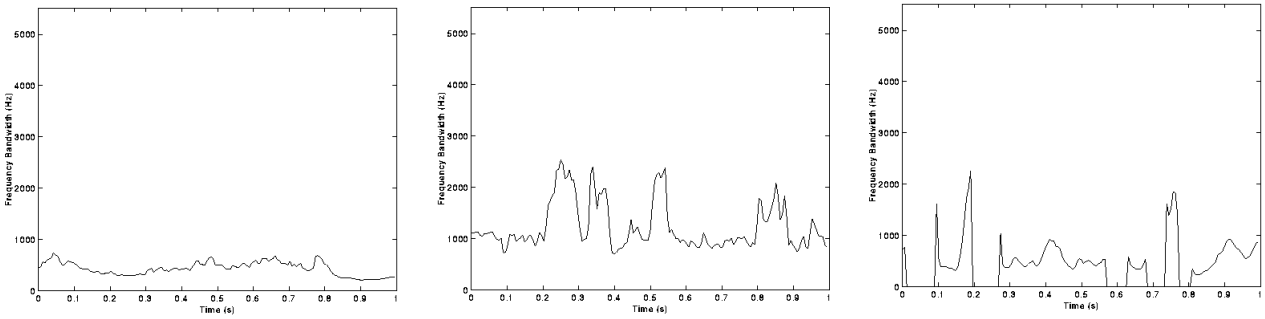
(a) commercial (b) basketball (c) news

Figure 6 Spectrograms of three audio clips



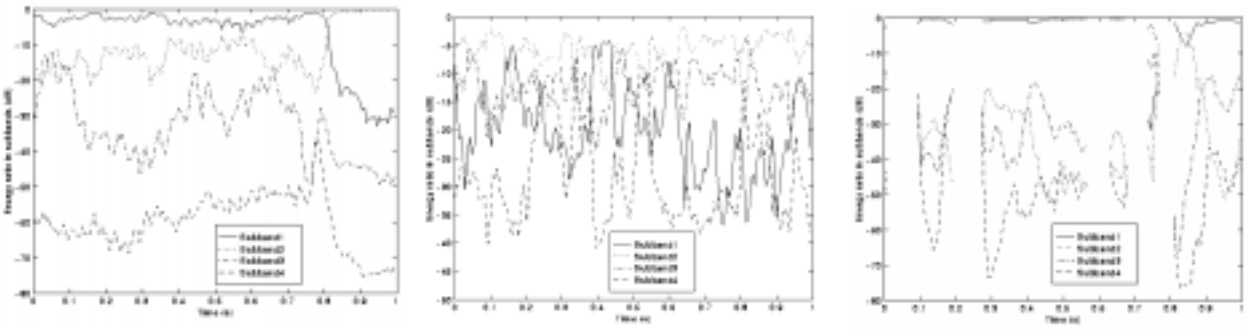
(a) commercial (b) basketball (c) news

Figure 7 Contours of frequency centroid of three audio clips



(a) commercial (b) basketball (c) news

Figure 8 Contours of bandwidth of three audio clips



(a) commercial (b) basketball (c) news

Figure 9 Energy ratio in 4 subbands of three audio clips

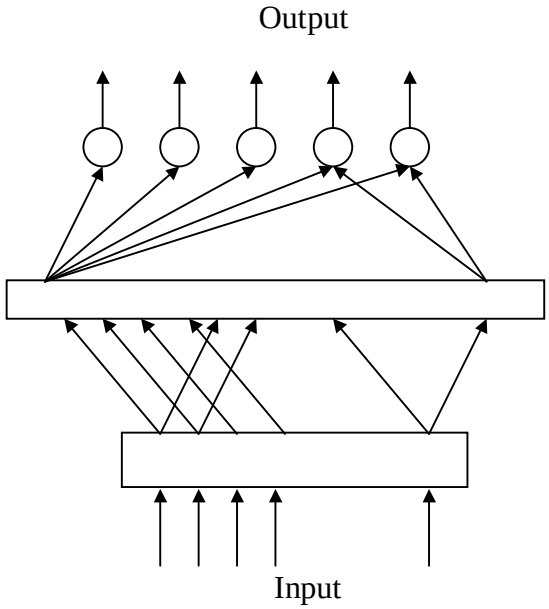


Figure 10 Structure of the ACON neural net classifier

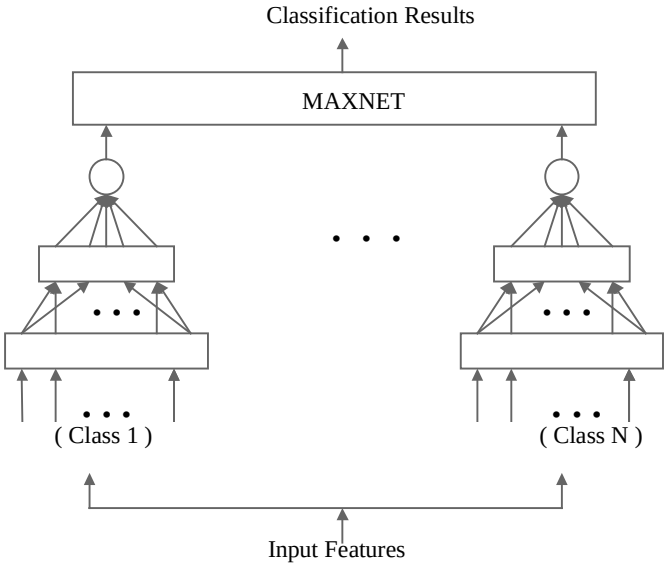
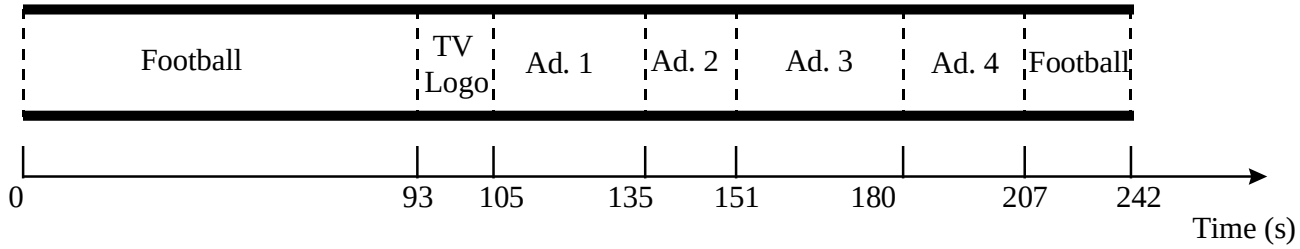
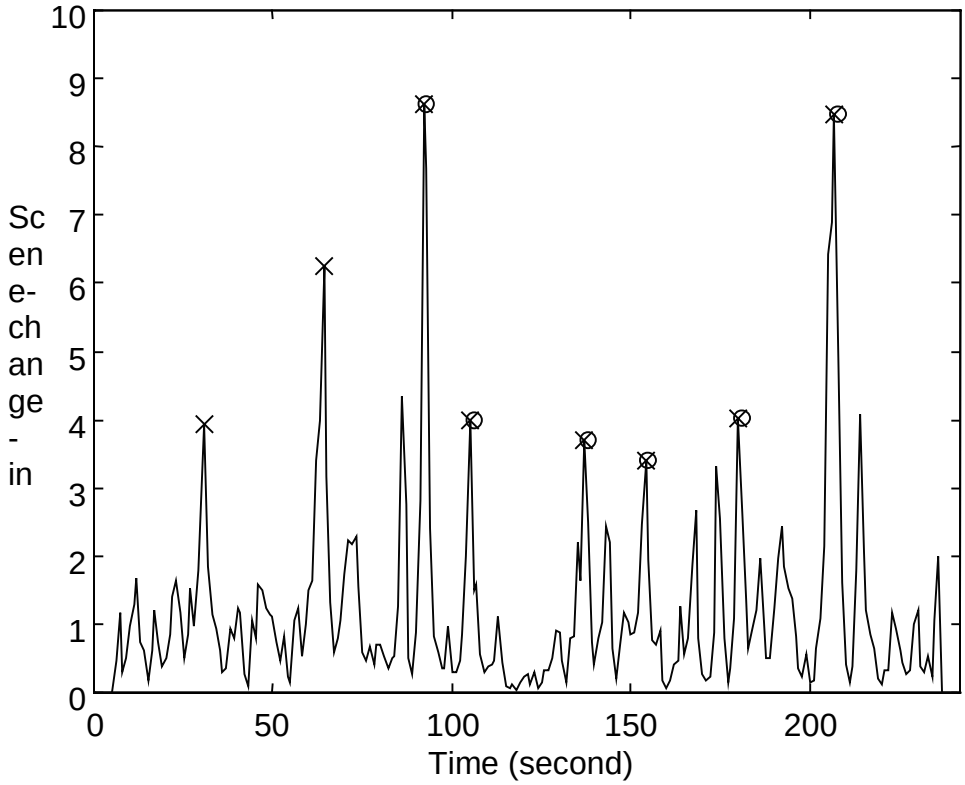


Figure 11 Structure of the OCON neural net classifier

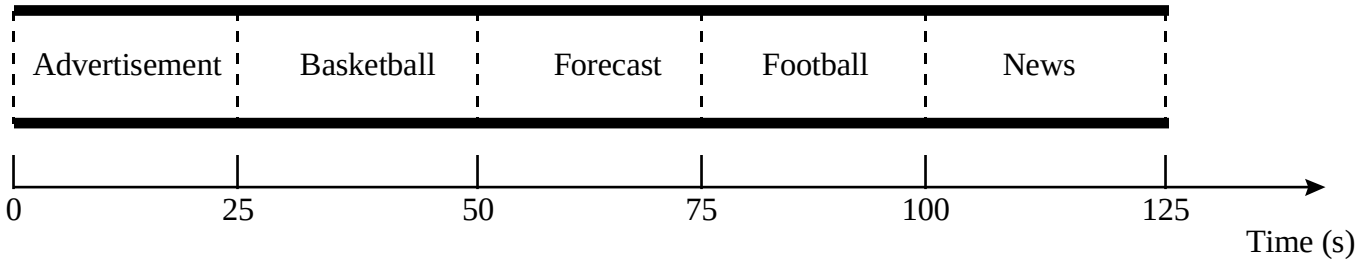


(a) Semantic contents of first testing sequences

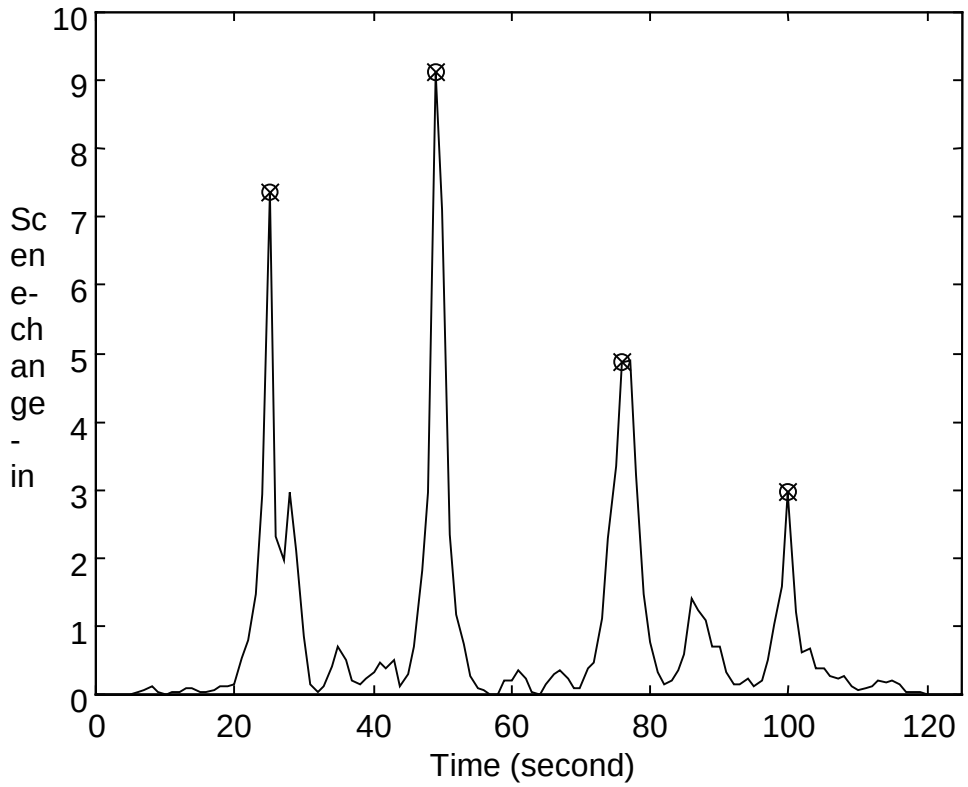


(b) Scene-change-index

Figure 12 Content and Scene-change-index calculated for the first sequence



(a) Semantic contents of the sequence



(b) Scene-change-index

Figure 13 Content and Scene-change-index calculated for the second sequence

Table 1 Mean Feature Vectors of Different Scene Classes

Feature	1. NSR	2. VSTD	3. VDR	4. FCVC4	5. PSTD	6. VMR
Commercial	0.997	0.195	0.798	0.011	2.289	0.281
Basketball	0.999	0.178	0.719	0.011	2.115	0.254
Football	0.998	0.174	0.702	0.011	2.241	0.297
News	0.898	0.271	0.953	0.027	1.328	0.459
Weather	0.897	0.272	0.966	0.029	1.198	0.505
Feature	7. NUR	8. FC	9. BW	10. ERSB1	11. ERSB2	12. ERSB3
Commercial	0.540	829.4	797.8	0.630	0.245	0.090
Basketball	0.589	1432.5	1176.3	0.353	0.352	0.217
Football	0.520	1161.4	875.2	0.367	0.398	0.213
News	0.299	666.1	436.0	0.650	0.292	0.044
Weather	0.234	664.8	406.5	0.620	0.333	0.035

Table 2 Standard Deviation of Features in Different Scene Classes

Feature	1. NSR	2. VSTD	3. VDR	4. FCVC4	5. PSTD	6. VMR
Commercial	0.013	0.045	0.109	0.007	1.163	0.221
Basketball	0.003	0.045	0.119	0.006	1.087	0.172
Football	0.008	0.059	0.167	0.008	1.131	0.198
News	0.112	0.033	0.033	0.014	0.736	0.170
Weather	0.095	0.031	0.018	0.013	0.562	0.150
Feature	7. NUR	8. FC	9. BW	10. ERSB1	11. ERSB2	12. ERSB3
Commercial	0.272	337.8	310.0	0.205	0.178	0.080
Basketball	0.189	368.7	195.1	0.156	0.138	0.094
Football	0.196	295.5	164.9	0.165	0.166	0.089
News	0.129	219.0	146.5	0.191	0.183	0.037
Weather	0.095	125.2	98.1	0.160	0.156	0.029

Table 3 Mean Feature Vectors of 13 Clusters

Feature / Clusters	1 NSR	2 VSTD	3 VDR	4 FCVC4	5 PSTD	6 VMR	7 NUR	8 FC	9 BW	10 ERSB1	11 ERSB2	12 ERSB3
1	0.990	0.228	0.848	0.017	1.233	0.455	0.426	1061.9	901.5	0.509	0.291	0.167
2	0.960	0.245	0.917	0.014	1.269	0.596	0.253	630.7	456.6	0.702	0.244	0.041
3	0.991	0.190	0.792	0.010	3.287	0.175	0.615	627.0	659.6	0.753	0.175	0.049
4	0.917	0.272	0.954	0.026	1.215	0.547	0.291	767.3	437.9	0.424	0.519	0.045
5	0.997	0.181	0.733	0.010	2.435	0.398	0.350	1084.7	857.2	0.323	0.507	0.143
6	0.815	0.282	0.970	0.049	1.408	0.371	0.252	679.8	413.9	0.583	0.366	0.039
7	0.997	0.190	0.777	0.012	2.051	0.167	0.675	1304.7	1165.2	0.497	0.252	0.166
8	0.900	0.276	0.960	0.028	1.317	0.433	0.299	592.9	407.6	0.765	0.189	0.034
9	0.999	0.189	0.738	0.011	1.824	0.367	0.491	1461.1	1129.0	0.351	0.301	0.290
10	1.000	0.114	0.525	0.006	3.469	0.092	0.634	1303.7	947.9	0.182	0.567	0.217
11	0.997	0.195	0.803	0.012	1.070	0.128	0.802	671.1	794.7	0.797	0.107	0.058
12	1.000	0.147	0.598	0.008	2.286	0.191	0.578	2068.6	1060.9	0.140	0.237	0.564
13	1.000	0.147	0.649	0.009	2.623	0.092	0.773	1671.4	1281.1	0.281	0.318	0.312

Table 4 Distributions of Feature Clusters in Different Scene Classes

Cluster / Class	1	2	3	4	5	6	7	8	9	10	11	12	13
Commercial	23	51	106	16	49	0	54	6	14	2	74	1	4
Basketball	60	2	0	0	57	0	92	0	63	40	2	6	78
Football	102	7	10	4	72	2	25	3	40	93	4	7	31
News	11	94	11	81	1	68	3	123	0	0	8	0	0
Weather	2	95	2	113	0	80	0	108	0	0	0	0	0

Table 5 Intra-Cluster Scattering Matrix

Feature	1	2	3	4	5	6	7	8	9	10	11	12
1	3.872	-0.993	-0.714	-0.846	0.092	2.865	1.467	0.169	0.619	-0.317	0.279	0.399
2	-0.993	8.764	5.725	4.479	-1.232	0.287	-1.439	-0.176	-0.796	1.077	-1.198	-0.929
3	-0.714	5.725	7.507	4.791	0.177	0.564	-1.429	0.277	-0.393	1.633	-2.355	-1.337
4	-0.846	4.479	4.791	26.99	0.818	-1.072	0.371	0.405	-0.215	0.394	-0.707	0.167
5	0.092	-1.232	0.177	0.818	28.27	-5.684	-2.587	-0.334	-0.221	-0.564	0.951	0.370
6	2.865	0.287	0.564	-1.072	-5.684	27.02	-11.99	0.178	-1.826	-0.833	1.235	0.860
7	1.467	-1.439	-1.429	0.371	-2.587	-11.99	18.33	0.202	3.012	1.367	-2.805	0.631
8	0.169	-0.176	0.277	0.405	-0.334	0.178	0.202	7.736	6.174	-4.790	0.855	3.971
9	0.619	-0.796	-0.393	-0.215	-0.221	-1.826	3.012	6.174	11.38	-2.252	-2.274	4.372
10	-0.317	1.077	1.633	0.394	-0.564	-0.833	1.367	-4.790	-2.252	12.51	-15.76	-3.816
11	0.279	-1.198	-2.355	-0.707	0.951	1.235	-2.805	0.855	-2.274	-15.76	26.352	-2.429
12	0.399	-0.929	-1.337	0.167	0.370	0.860	0.631	3.971	4.372	-3.816	-2.429	17.68

Table 6 Inter-Cluster Scattering Matrix

Feature	1	2	3	4	5	6	7	8	9	10	11	12
1	2.953	-5.419	-5.097	-14.84	4.436	-5.155	6.909	4.491	7.347	-3.787	-0.293	9.543
2	-5.419	15.60	15.75	28.08	-17.06	21.78	-19.23	-13.09	-18.35	14.71	-5.430	-27.11
3	-5.097	15.75	16.31	26.34	-17.45	21.83	-18.38	-14.01	-18.42	17.04	-7.962	-29.20
4	-14.84	28.08	26.34	78.94	-25.36	25.10	-33.28	-20.41	-33.08	17.21	1.582	-43.91
5	4.436	-17.06	-17.45	-25.36	28.07	-27.47	18.64	11.13	15.10	-15.57	9.819	22.36
6	-5.155	21.78	21.83	25.10	-27.47	52.87	-40.89	-17.52	-27.88	13.12	3.45	-34.51
7	6.909	-19.23	-18.38	-33.28	18.64	-40.89	37.59	16.28	27.72	-8.685	-9.347	33.15
8	4.491	-13.09	-14.01	-20.41	11.13	-17.52	16.28	19.66	23.17	-21.77	8.319	39.73
9	7.347	-18.35	-18.42	-33.08	15.10	-27.88	27.72	23.17	32.44	-21.00	2.119	46.21
10	-3.787	14.71	17.04	17.21	-15.57	13.12	-8.685	-21.77	-21.00	36.63	-30.77	-41.84
11	-0.293	-5.430	-7.962	1.582	9.819	3.45	-9.347	8.319	2.119	-30.77	41.19	11.59
12	9.543	-27.11	-29.20	-43.91	22.36	-34.51	33.15	39.73	46.21	-41.84	11.59	85.50

Table 7 Diagonal Entries of Intra and Inter-Cluster Scattering Matrices

Feature	NSR	VSTD	VDR	FCVC4	PSTD	VMR
Intra-cluster	3.872	8.764	7.507	26.99	28.27	27.02
Inter-cluster	2.953	15.60	16.31	78.94	28.07	52.87
Inter/Intra Ratio	0.763	1.780	2.173	2.925	0.993	1.957
Feature	NUR	FC	BW	ERSB1	ERSB2	ERSB3
Intra-cluster	18.33	7.736	11.38	12.51	26.35	17.68
Inter-cluster	37.59	19.66	32.44	36.63	41.19	85.50
Inter/Intra Ratio	2.051	2.541	2.851	2.928	1.563	4.836

Table 8 Classification Results Using a Nearest Neighbor Classifier

(unit: 100%)

Data / **Result**	**Input Class**				
	Commercial	Basketball	Football	News	Weather
Commercial	68.0	16.7	16.3	7.4	1.2
Basketball	9.1	59.4	22.8	0.5	0.1
Football	9.1	21.1	53.7	0.4	0.1
News	9.8	2.1	5.3	52.1	39.6
Weather	4.0	0.7	1.9	39.6	59.0

Table 9 Classification Results Using an OCON Neural Net

(Hidden Layer: 7 neurons, unit: 100%)

Data / Result	Input Class				
	Commercial	Basketball	Football	News	Weather
Commercial	74.3	6.0	6.5	2.8	1.2
Basketball	10.2	80.9	13.0	0.2	0.0
Football	7.9	12.2	79.0	2.8	1.0
News	6.3	0.7	1.5	59.0	28.6
Weather	1.3	0.2	0.0	35.2	69.2

Table 10 Classification Results Using an ACON Neural Net

(Hidden Layer: 15 neurons, unit: 100%)

Data / Result	Input Class				
	Commercial	Basketball	Football	News	Weather
Commercial	71.9	7.2	6.6	2.3	0.6
Basketball	9.9	75.9	10.5	0.3	0.0
Football	10.9	16.7	81.2	2.2	0.4
News	6.4	0.2	1.6	64.9	32.4
Weather	0.9	0.0	0.1	30.3	66.6

Table 11 Classification Results Using Nine Features

(OCON, Hidden Layer: 7 neurons, unit: 100%)

Data / Result	Input Class				
	Commercial	Basketball	Football	News	Weather
Commercial	74.5	8.0	10.5	5.2	2.6
Basketball	11.8	79.5	12.1	0.3	0.0
Football	7.2	12.1	75.5	1.1	0.4
News	5.6	0.4	1.9	51.1	23.4
Weather	0.9	0.0	0.0	42.3	73.6

Table 12 Segmentation Results

Sequence	Length (second) (s)	Real scene changes	True scene changes detected	False scene changes detected
1	304	4	4	8
2	242	6	6	2
3	210	3	3	9
4	255	5	4	3
5	243	4	4	2
6	158	2	2	3
7	204	4	3	4
8	155	5	5	2
9	218	7	7	3
10	125	4	4	0
Total	2177	44	42	36