MERL – A MITSUBISHI ELECTRIC RESEARCH LABORATORY
http://www.merl.com

Audio-Visual Event Detection based on Mining of Semantic Audio-Visual Labels

King-Shy Goh, Koji Miyahara, Regunathan Radhakrishan, Ziyou Xiong, and Ajay Divakaran

TR-2004-008 March 2004

Abstract

Removing commercials from television programs is a much sought-after feature for a personal video recorder. In this paper, we employ an unsupervised clustering scheme (CM Detect) to detect commercials in television programs. Each program is first divided into Ws-minute chunks, and we extract audio and visual features from each of these chunks. Next, we apply k-means clustering to assign each chunk with a commercial/program label. In contrast to other methods, we do not make any assumptions regarding the program content. Thus, our method is highly content-adaptive and computationally inexpensive. Through empirical studies on various content, including American news, Japanese news, and sports programs, we demonstrate that our method is able to filter out most of the commercials without falsely removing the regular program.

Publication History:

1. First printing, TR-2004-008, March 2004

Audio-Visual Event Detection based on Mining of Semantic Audio-Visual Labels

King-Shy Goh, Koji Miyahara, Regunathan Radhakrishan, Ziyou Xiong, Ajay Divakaran
Mitsubishi Electric Research Labs, Cambridge, MA, USA.
{goh, miyahara, regu, zxiong, ajayd }@merl.com

ABSTRACT

Removing commercials from television programs is a much sought-after feature for a personal video recorder. In this paper, we employ an unsupervised clustering scheme (CM_Detect) to detect commercials in television programs. Each program is first divided into W_s-minute chunks, and we extract audio and visual features from each of these chunks. Next, we apply k-means clustering to assign each chunk with a *commercial/program* label. In contrast to other methods, we do not make any assumptions regarding the program content. Thus, our method is highly content-adaptive and computationally inexpensive. Through empirical studies on various content, including American news, Japanese news, and sports programs, we demonstrate that our method is able to filter out most of the commercials without falsely removing the regular program.

Keywords: Commercial Detection, Unsupervised Clustering, Audio Classification, Motion Activity.

1. INTRODUCTION

Many consumers crave for an automatic way for their video recorders to remove commercials while retaining only the interesting segments of their recorded program. However, the problem of detecting commercials (or interesting program segments) is difficult. The content is diverse, which in turn makes it difficult to formulate a universal model that is applicable to all commercials. An effective detection system needs to be content-adaptive, and it needs to have low computational cost. In this paper, we formulate the detection problem as a "usual" versus "unusual" binary classification problem. Our motivation comes from fact that commercials are relatively rare compared to the regular parts of a program, and they are usually quite different in terms of visual and audio characteristics. With this problem formulation, we do not need to define the concept of *commercial* or *program* a priori.

Our commercial detection scheme (CM_Detect) consists of three components. We first extract visual and audio features from the video content, then we divide the content into small chunks and compute a departure value for each chunk. This departure value measures the dissimilarity between each chunk and a longer video sequence that surrounds it. If the chunk is vastly different, we will get a high value and we can consider this chunk to be "unusual". The departure values form a feature vector for each chunk (data points), and we perform $k-$means clustering on the feature vectors to produce a *program* group and a *commercial* group.

In the remaining parts of the paper, we first discuss some previous work that has been done for commercial detection. In Section 2, we describe the CM_Detect scheme in details, and present our empirical results in Section 3. Finally, we present concluding remarks and avenues for further investigation in Section 4.

1.1. Related Works

Most current systems apply one or a combination of the following techniques:

- Detect lapses in close captioning capture above a certain threshold.
- Discover the occurrence of black frames and audio silence.
- Search for changes in rate of scene and motion change.
- Detect sound volume changes since volume is generally higher in commercials.

In ComDetect,[1] the video signal is monitored round-the-clock so that any known commercial or video clip of interest can be detected. The system is flexible in that no watermarks or embedded information have to be inserted into the transmitted video signal. However, it requires a database of pre-recorded commercials or video clips of interest. While the database can be customized for different users, it may require constant updating as new commercials appears frequently, and old ones are taken off the air.

Lienhart et al.[2] identify potential commercials by using a set of features common to every commercial block. The features includes restricted temporal length of commercial blocks, black frames, sound volume, motion using edge change ratio and motion vector length, and rate of scene changes measured by frequency of hard cuts and fades. The candidates can be compared with a database of known commercials to achieve a higher detection accuracy and a lower false alarm rate.

Sanchez et al.[3] first analyze the color contents of regions in consecutive frames and measure the variations between frames with their associated color histograms. When discontinuity between consecutive frames is detected, a global variation of the scene is computed. This method is used to decomposed a video sequence into shots, and using the first frame as the shot's representative, this frame is then compared to a database of commercial frames to identify out commercial shots. Potential shortcomings of this method are the heavy computations involved in the shot detection process, and the inherent assumption that all video sequences can be decomposed into shots. Its reliance on a database of known commercials also requires frequent database updating.

Hauptmann and Witbrock[4] detect commercials as part of a larger system used for story segmentation for news video. They use black frames and rate of scene changes to estimate the beginning and end of a commercial segment.

In,[5] Divakaran et al. employed the same feature extraction framework as this paper, and performed peak detection on the histogram distance metric to single out the commercials. While the method is able to detect most of the commercials, it also falsely labels some of the regular program as commercials. In this paper, we improve upon the ad-hoc threshold based method of fusion of the motion and audio features proposed in.[5]

2. COMMERCIAL DETECTION SCHEME (CM_Detect)

While the audio and visual characteristics of different commercials are diverse, we observe that they share a common characteristic — commercials are relatively rare, and they are somewhat different compared to the main programs. Thus, we can model commercial detection as a general two-class learning problem, "*usual*" versus "*unusual*" events (*program* versus *commercial*). Moreover, we know that in a program, the commercials appear together in a continuous segment rather than scattered individually throughout the program. We can make use of this time-line information to reduce the number of false alarms (regular program wrongly detected as commercials).

Our CM_Detect scheme comprises three main components:

1. **Feature extraction**. Given a video sequence, we extract audio and visual feature from it.
2. **Departure from stationarity**. We compare a short subsequence of the video with a longer subsequence that surrounds it to determine if the short sequence is "unusual" or not.
3. **Clustering**. We perform unsupervised clustering to partition the data (video sequence) into two groups, *commercial* or *program*.

2.1. Feature Extraction

For each television program (or video sequence), we extract two major types of features, audio and visual. The audio features consist of semantic audio labels derived from audio classification, while the visual features are generated from MPEG-7 motion activity descriptors. We divide a video sequence into N chunks of W_s-minute long segment and we extract the following audio and visual labels.

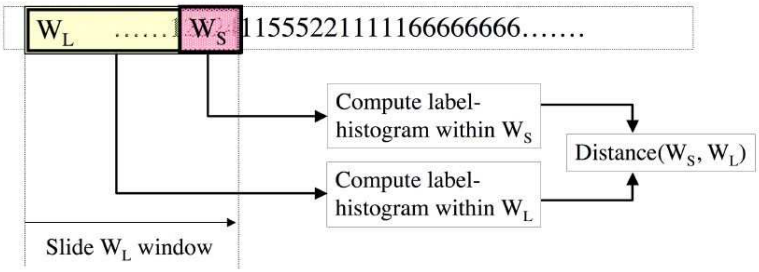


Figure 1. Departure from stationarity for detecting *unusual* events.

Audio Labels from Classification

In,[6] an audio classification framework is introduced for sports highlights extraction. Hidden Markov Models (HMMs) using Mel Frequency Cepstral Coefficients (MFCCs) are used to classify an audio signal of duration 30ms into one of seven labels — *silence*, *applause*, *ball-hit*, *cheering*, *music*, and *music with speech*. The signals are then aggregated to to produce one audio label for each W_s-minute video chunk.

Visual Labels From Motion Activity Descriptors

The motion activity descriptors are introduced in MPEG-7 to capture the "intensity of action", or the "pace" of a video sequence. For example, suppose we have a a sequence showing a golfer swinging his club, followed by the camera pan that traces the golf balls' trajectory, and another sequence showing an anchor-man reading news. The viewers might consider the golf sequence to be of "high action", whereas they will find the news sequence to be of "low action". Using motion activity descriptor, we can quantify such perception effectively. In,[7] it has been shown that by quantizing the variance of the motion vector magnitude to one of five levels — *very low*, *low*, *medium*, *high* and *very high* — a decent representation for actions can be obtained. The quantized value forms the visual label for each chunk of video sequence.

2.2. Departure from Stationarity

The idea of departure from stationarity is first introduced in[8] to measure the amount of "usual" characteristics in a sequence. The implementation of departure in this paper differs slightly and Figure 1 illustrates the scheme. The steps involved are:

1. Define two windows of different sizes. We form a global window that consists W_L minutes of video chunks, and a local window with just one video chunk.
2. Compute histogram. For each window, we compute one histogram from the audio labels, and a second histogram from the visual labels.
3. Compute dissimilarity. To compare the histograms from the two windows, we compute a dissimilarity value that that is based on the Kullback-Liebler distance.
4. Shift global window by W_s.
5. Check for overlaps between the two windows. If the shift results in no overlaps, we proceed to step 6. If overlapping occurs, we return to step 2.
6. Tally dissimilarity values. For each local window W_s, there are $\frac{2\times W_L}{W_s}$ dissimilarity values. We pick the maximum of these values as the distance between the global and the local window.
7. Return to step 1 for the next local window until we reach the end of the video sequence.

In addition to the distance values from the labels, we also use the percentage of "speech with music" present in the audio signal of each W_s chunk. The reason for this inclusion is based on empirical observations of commercials. We realized that often, commercial content contains a large amount of human speech set to some background music. Thus, the feature vector associated with each video chunk consists of three elements: audio departure, "speech with music" percentage, and motion departure.

2.3. Clustering to Detect Commercials

Due to the diversity of commercial content, it would be difficult to train a universal classifier that can detect all types of commercials in all types of programs. We want to make as few assumptions about the semantics of the video content as possible, thus the method of choice should be content adaptive. We also need an algorithm that is simple to integrate into consumer electronics, as well as one that is fast to execute. From the large selection of unsupervised techniques available from machine learning, we pick k-means clustering for its simplicity and sound theoretical basis. With good features, we find that this simple algorithm is able to produce favorable results.

Clustering Scheme

The k-means algorithm first creates k clusters by randomly assigning the data points. The algorithm then iterates the following steps:

1. Compute cluster centroid.
2. For each data point, assign it to the cluster whose centroid is the nearest. The distance function used is L_1.

We halt the iteration when no further changes in the assignment occur, or when we reach the maximum number of iteration specified before we start clustering. Since the initialization of k-means is random, we can end up with different optimal solutions. For this work, we ran a large number of iterations and pick the solution that occurs the most often as the final clustering solution.

Three issues arise when we attempt to use clustering algorithms. The first is how do we decide on the optimal number of clusters (K), the second is which cluster(s) do we choose to be the one containing commercials, and the third is how do we know if the data we want to cluster is indeed separable.

Optimal Number of Clusters

To address the first issue, we employ the aid of various cluster validity indices. After much empirical studies, we find that the Dunn's Index[9] gives a good indication about the number of clusters to specify. The index is define as follows:

$$I_{dunn} = \min_{1\le i\le K}\{\min_{1\le j\le K, j\neq i}\{\frac{\delta(C_i, C_j)}{\max_{1\le l\le K}\Delta(C_l)}\}\}. \quad (1)$$

The denominator $\Delta(C_l)$ can be perceived as the diameter of cluster C_l, and it is defined as $\Delta(C_l) = \max_{x,y\in C_l} d(x, y)$. The numerator of the index $\delta(C_i, C_j)$ is a measurement of the distance between two clusters C_i and C_j, and it is defined as $\delta(C_i, C_j) = \min_{x\in C_i, y\in C_j} d(x, y)$. The term $d(x, y)$ is the L_1-distance between two data points x and y The value of K which maximizes I_{dunn} is chosen to be the optimal.

Selection of *Commercial* Cluster(s)

To address the second issue of commercial cluster selection, our selection criteria makes use of some general characteristics of commercials. First, we know that the content is diverse, thus it would generally have features that vary widely. Second, the number of commercial content relative to the program is small. Lastly, the percentage of "speech with music" audio signal is higher for commercials. We pick the following cluster as the commercial cluster:

$$C_{cm} = \arg\max_{1\le i\le K} \frac{\sigma(C_i) \times_{Pmusp} (C_i)}{n(C_i)}. \quad (2)$$

$\sigma(C_i)$ denotes the standard deviation of the cluster C_i, $P_{musp}(C_i)$ denotes the average percentage of "speech with music" audio signal in that cluster C_i, and $n(C_i)$ denotes the number of data points belonging to cluster C_i. The data points in C_{cm} are considered to be in the *commercials* group, while the remaining clusters C_i where $1 \le i \le k$, and $C_i \neq C_cm$ are combined to form the *program* group.

When the cluster size $n(C_{cm})$ is small, we allow the scheme to select more than one clusters until a specified commercial size is exceeded. Typically, we expect the amount of commercials to be no more than 15% of the entire program duration. The selection criteria of the subsequent clusters is similar to Equation 2. We pick the cluster which gives the next largest $\frac{\sigma(C_i)\times_{Pmusp}(C_i)}{n(C_i)}$ value.

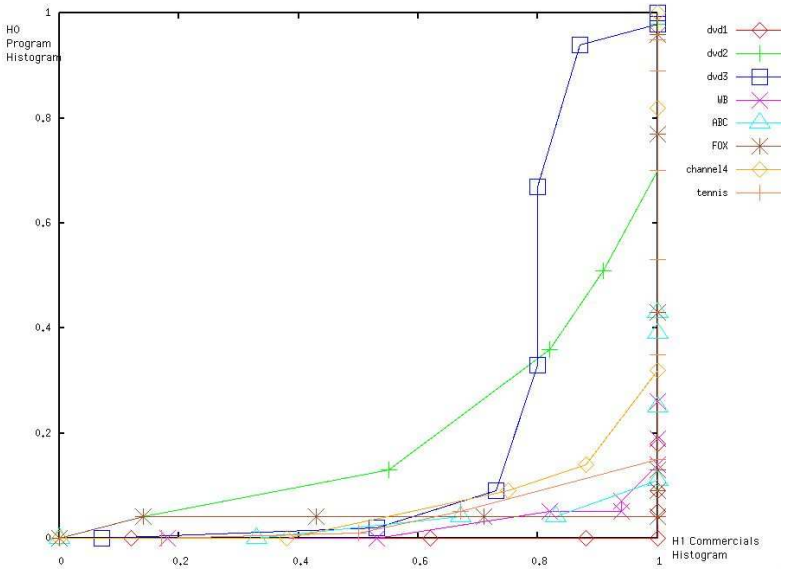


Figure 2. Plot of *commercials* histogram values (H_1) against *program* values (H_0).

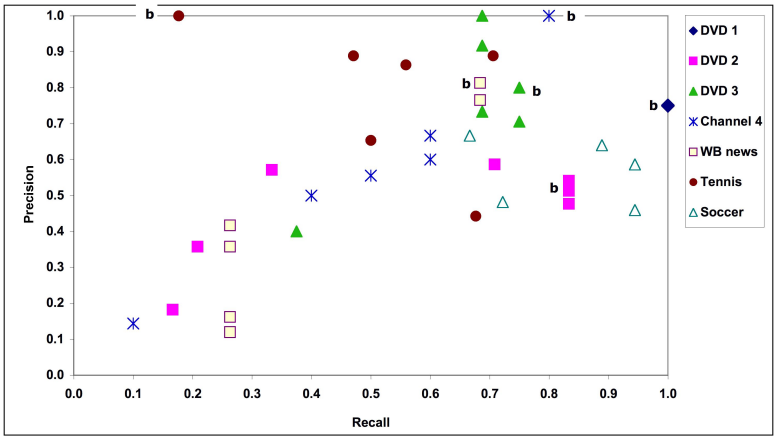


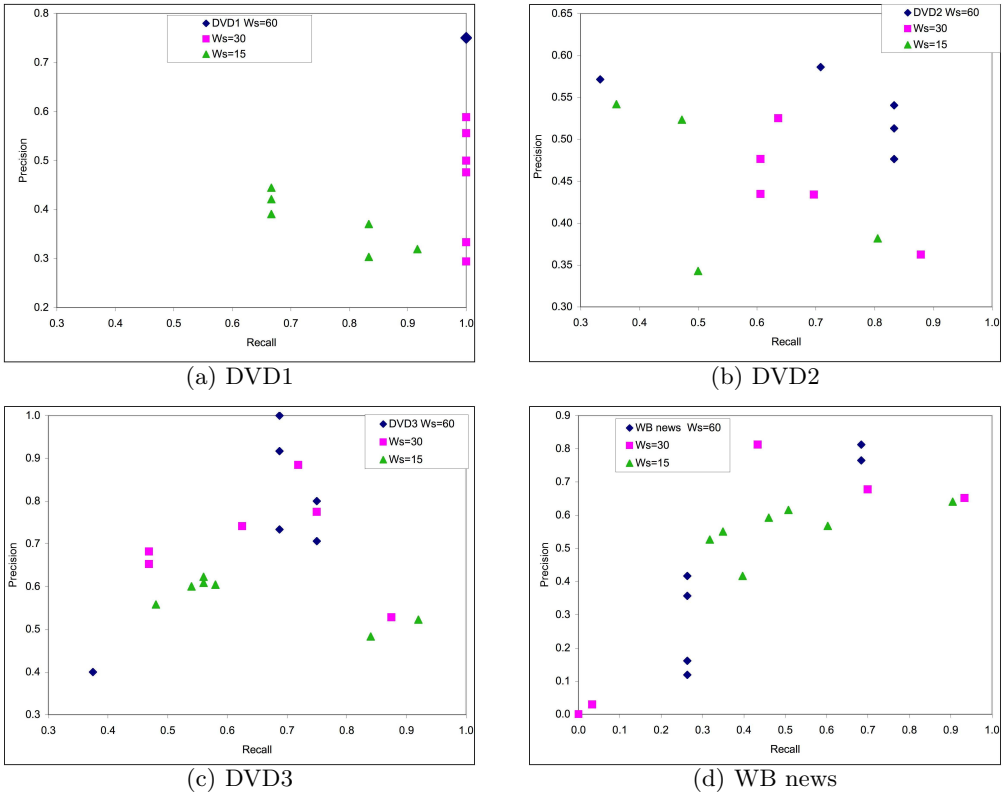
Figure 3. Precision Recall for Commercial Detection.

Data Separability

Finally, to find out if our data is actually separable, we examine the two groups of data points. Ideally, we want the *commercial* cluster(s) to be well-separated from the *program* clusters. We propose a cluster purity index to measure this separation and provide indications as to whether we can trust our clustering solution. While the Dunn's index can also be a measure for this separation, we find that the index's range varies from content to content, and thus making it difficult to define a goodness range.

To construct our index, we first compute a histogram based on the Mahalanobis distance from each data point to the centroid of the *commercial* group. The histogram counts the percentage of points that are within m number of standard deviations (σ_{cm}) from the centroid. Let H_0 denote the histogram for the *program* group and H_1 denote the *commercial* group. We plot the values of in H_1 against H_0 in Figure 2 for bins of one to ten or more standard deviations. The number of clusters formed is $K = 2$.

The plot shows curves for eight different videos and we notice that for the majority, the gradient of the curves is gentle except for the spike at the end. This trend happens when a large percentage of the *commercial* group



(a) DVD1 (b) DVD2

(c) DVD3 (d) WB news

Figure 4. Effect of varying W_S.

are within the first few σ_{cm} while only a small percentage of the *program* are enclosed. DVD2 and DVD3 are the only examples of content displaying a different behavior. For DVD3, we have roughly 75% of the *commercial* points while only less than 10% of the *program* at three σ_{cm}. For DVD2, the percentage of *program* points enclosed at three σ_{cm} is more than 40%. To measure the separability, we track the gradient of the curves and we compute the area under the curve up to the point when the gradient increases suddenly.

3. EMPIRICAL RESULTS

To evaluate the effectiveness of CM_Detect, we apply our scheme on a variety of video content. Since soccer highlights extraction can also be modeled as an "unusual" event detection problem, we will demonstrate that our scheme can also be used to detect highlights. The datasets used for our empirical studies are as follows:

- **DVD1** contains Japanese news video.
- **DVD2** is also a video of Japanese news but a fair amount of the regular program has background music.
- **DVD3** is animation titled "Princess Mononoke" in the Japanese language.
- **Channel 4** and **WB news** are American news content.
- **Tennis** and **Soccer** are sports video with commercials.

For each content, we extract the audio and visual features as described in Section 2.1. The window sizes W_L and W_S are set to $10mins$ and $1min$ for most of our experiments.

Our empirical studies seek to answer the following questions:

How effective is the Dunn's Index for selecting the optimal number of clusters?
What is the precision-recall performance?
What is the effect of varying the window size W_S?

For commercial detection, our objective is to filter out as many commercial segments as possible, without sacrificing any of the regular program. Thus, we want to have the highest possible precision, and some loss in recall performance can be tolerated.

3.1. Precision-Recall and Dunn's Index Evaluation

In Figure 3, we show the precision-recall plot for the seven programs listed above. For each program, we vary the number of clusters (K) from two to eight. The window sizes are $W_L = 10$ minutes and $W_S = 1$ minute. For values of K that maximize Dunn's Index, we placed a "**b**" marker next to the point on the plot. We observe that most of the points are clustered in the upper right quadrant, indicating that we can get decent precision and recall with our simple CM_Detect scheme. Moreover, we also notice that the values of K considered optimal by Dunn's Index also give us the best possible precision-recall for all the content except *tennis*. Thus, we can conclude that Dunn's Index is indeed a useful indicator for the optimal K value. We also notice that most of the points with low precision-recall performance happen to be from *DVD2* and *WB news*. From watching the video, we find that *DVD2* contains a lot of news broadcast with music in the background, causing the regular program to have high percentage of "music with speech" audio, and thus affecting our clustering results.

3.2. Effects of Varying W_S

Using a fixed window size fo W_L and W_S may not always be optimal. With $W_S = 1$ minute, we may not capture the precise location of commercial boundaries. For the global window, if we use a W_L that is too large, we may include a wide variety of the regular program, especially if the program content is also changing rapidly, and thus cause the notion of "usual" to be obscure. If "usual" is an ill-defined concept, our computation of departure in Section 2.2 may not be meaningful.

In this paper, we examine the effect of varying the local window size W_S, and Figure 4 precision-recall plots for four video contents with number of cluster K varying from two to eight. The global window size is 10 minutes. From the figure, we see that the W_S that gives the best precision-recall varies from content to content. For *DVD1* and *DVD2*, $W_S = 60$ seconds gives the best performance, while $W_S = 30$ seconds is more optimal for *DVD3* and *WB news*. In most of the content, it appears that $W_S = 15$ seconds is the worst. But we have to bear in mind that we are comparing a 15−second video chunk with a $W_L = 10$ minute global video chunk. At such small chunk size, we the fixed W_L may be too large.

Our studies shows that changing the window sizes does have an effect on our precision-recall performance. For future work, we plan to investigate ways in which we can adaptively choose an optimal window size for both W_L and W_S.

3.3. Soccer Highlights Extraction

To investigate the feasibility of applying our CM_Detect scheme to the more general problem of "unusual" event detection, we attempt to extract highlights from a soccer game. We tested our approach on two World Cup 2002 games, Brazil versus Turkey in the semi-final, and Brazil versus Germany in the final. In the first game, one goal was scored in the second-half of the game, and in the second game, two goals were scored after halftime. The ground truth for both games are generated manually by two individuals who had watched the games several times. Instead of using "music with speech" as one of our features, we use the "cheering" audio signal instead. Figure 5 shows the precision-recall results of using CM_Detect for highlights extraction. The performance is reasonable and we managed to detect all the goal-scoring moments in the games. Compared to a previous work,[8] our scheme achieves a higher recall while suffering a little loss in precision.

4. CONCLUSION

In this paper, we have proposed a simple scheme CM_Detect for detecting commercials. The scheme first extracts audio and visual features extracted from the video content, then it performs $k-$means clustering to separate the data points into a *commercial* and a *program* group. We make use of Dunn's Index to choose the optimal number of clusters for $k-$means, and we have also proposed a cluster purity index to determine if the data is separable or not. Finally, we choose the cluster(s) that possesses the largest standard deviation, and contains the highest

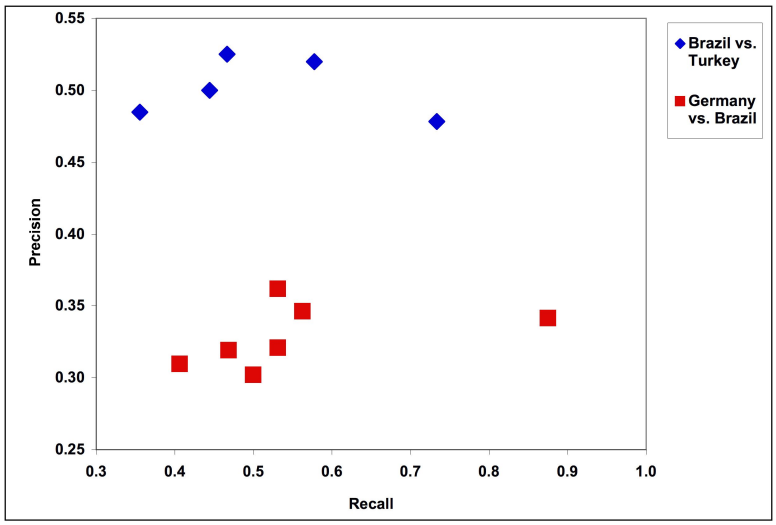


Figure 5. Precision Recall for Soccer Highlights Detection.

percentage of "music with speech" audio signal to be the *commercial* group. Through our extensive empirical studies, we have shown that we can achive more than 70% precision-recall for most of our content. We have also applied CM_Detect to the task of soccer highlights extraction and obtained good results.

For future work, we plan to investigate ways to adaptively choose the window sizes for computing departures, and we would also like to test our scheme on more "unusual" event detection problems such as video surveillance and generation of other sports highlights.

REFERENCES

1. Information and T. T. Center, "Comdetect http://www.ittc.ku.edu/jgauch/research/video/comdetect_overview.html," in *University of Kansas*, 2000.
2. R. Leinhart, C. Kuhmunch, and W. Effelsberg, "On the detection and recognition of television commercials," in *Proc. IEEE Conference on Multimedia Computing and Systems*, June 1997.
3. J. M. Sanchez, X. Binefa, P. Radeva, and J. Vitria, "Local color analysis for scene break detection applied to tv commercials recognition," in *Proc. 3rd. Intl. Conf. on Visual Information and Information Systems*, June 1999.
4. A. Hauptmann and M. Witbrock, "Story segmentation and detection of commercials in broadcast news video," in *Proceedings of Advances in Digital Libraries Conference*, April 1998.
5. A. Divakaran, K. Miyahara, K. A. Peker, R. Radhakrishnan, and Z. Xiong, "Video mining using combinations of unsupervied and supervised learning techniques," in *Technical Report, MERL*, 2002.
6. Z. Xiong, R. Radhakrishnan, A. Divakaran, and T. Huang, "Audio events extraction based highlights extraction from baseball, golf and soccer games in a unified framework," in *Proc. of ICASSP*, April 2003.
7. K. A. Peker, R. Cabasson, and A. Divakaran, "Rapid generation of sports highlights using the mpeg-7 motion activity descriptor," in *SPIE Conference on Storage and Retrieval from Media Databases*, January 2003.
8. Z. Xiong, R. Radhakrishnan, and A. Divakaran, "Generation of sports highlights using motion activity in combination with a common audio feature extraction framework," in *Proc. of ICIP*, September 2003.
9. J. C. Dunn, "A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters," in *Journal of Cybernetics*, **3**, 1973.